EBU6018
Advanced Transform Methods

Discrete Cosine Transform
DCT

Andy Watson

Discrete Fourier Transform DFT

Before looking in detail at the Discrete Cosine Transform, DCT, let's take a look back at the DFT.

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{-nk}$$

In general, x[n] is a sequence of complex numbers that are transformed into a sequence of complex numbers X[k]

Discrete Fourier Transform DFT

- Even if the input data is a sequence of real numbers, the output sequence will still be complex numbers.

- The input sequence is a series of samples of a continuous waveform that may be neither odd nor even, and with a sequence that is periodic

Discrete Cosine Transform DCT

- However, if the input data contains only real values from an even function, then the imaginary (sine) values of the DFT output are 0.
- We are then left with the real (cosine) output values of the DFT.
- So we now have a Discrete Cosine Transform DCT.

Discrete Cosine Transform DCT

- As with the DFT, we assume that the input sequence is periodic in order to obtain an accurate Fourier Transform.
- In addition, for the DCT the sequence is assumed to be even and periodic.
- We will consider the DCT Type II. This is the most common, and is used as a 2D transform to produce JPEG compressed images. (There are 8 versions altogether)

Discrete Cosine Transform DCT

The DCT can be used:

- To compress the data representing information in an image because of the characteristics of the human eye. The compressed data requires less memory for storage and can be transmitted more quickly.
- To process audio signals in the frequency domain to compensate for the characteristics of the human ear at different frequencies.

Discrete Cosine Transform DCT

Summary:

- Reversible (lossless) transform that represents a discrete signal as a set of cosine coefficients (DCT basis functions are orthogonal).
- Similar to DFT, but uses only cosines and therefore avoids any complex numbers.
- Real Input, real output
- We will consider only discrete with number of input= number of output (n=k).

1-Dimensional DCT

$$DCT[k] = c(k)\sum_{n=0}^{N-1} s[n]\cos\frac{\pi(2n+1)k}{2N} \qquad k = 0,1,2\ldots N-1$$

$$DCT[k] = \langle s, \psi_k \rangle$$

$$c(k) = \begin{cases} \sqrt{1/N} & k = 0 \\ \sqrt{2/N} & k \neq 0 \end{cases}$$

c(k) is the normalisation factor.

- Orthonormal

$$\langle \psi_m, \psi_n \rangle = \begin{cases} 1 & \text{if } m = n \\ 0 & \text{if } m \neq n \end{cases}$$

The Basis Functions Ψ_k are the cosine terms in the definition.
They are calculated for each value of k, with n = 0….N-1

1-Dimensional DCT

Example. N = 4 (input is a 4-point sequence)

For each value of k = 0….N-1, insert n = 0….N-1:

$$\psi_0 = (1,1,1,1)/2$$

$$\psi_1 = \sqrt{1/2}(\cos(\pi/8), \cos(3\pi/8), \cos(5\pi/8), \cos(7\pi/8))$$

$$\psi_2 = \sqrt{1/2}(\cos(\pi/4), \cos(3\pi/4), \cos(5\pi/4), \cos(7\pi/4))$$

$$\psi_3 = \sqrt{1/2}(\cos(3\pi/8), \cos(9\pi/8), \cos(15\pi/8), \cos(5\pi/8))$$

$$DCT[0] = \frac{1}{\sqrt{N}}\sum_{n=0}^{3} s[n]$$

$$DCT[1] = \sqrt{\frac{2}{N}}\sum_{n=0}^{3} s[n]\cos\frac{\pi(2n+1)}{8}$$

$$DCT[2] = \sqrt{\frac{2}{N}}\sum_{n=0}^{3} s[n]\cos\frac{\pi(2n+1)}{4}$$

$$DCT[3] = \sqrt{\frac{2}{N}}\sum_{n=0}^{3} s[n]\cos\frac{\pi(2n+1)3}{8}$$

Question…….

These 4 Basis Functions can be written in Matrix format.

- Calculate the elements of the 4x4 Basis Function Matrix.
- Then determine the output sequence if the input sequence is s[n] = [2, 3, 1, 4]

4x4 DCT Basis Matrix

$$\Psi = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ \frac{\cos\frac{\pi}{8}}{\sqrt{2}} & \frac{\cos\frac{3\pi}{8}}{\sqrt{2}} & \frac{\cos\frac{5\pi}{8}}{\sqrt{2}} & \frac{\cos\frac{7\pi}{8}}{\sqrt{2}} \\ \frac{\cos\frac{2\pi}{8}}{\sqrt{2}} & \frac{\cos\frac{6\pi}{8}}{\sqrt{2}} & \frac{\cos\frac{10\pi}{8}}{\sqrt{2}} & \frac{\cos\frac{14\pi}{8}}{\sqrt{2}} \\ \frac{\cos\frac{3\pi}{8}}{\sqrt{2}} & \frac{\cos\frac{9\pi}{8}}{\sqrt{2}} & \frac{\cos\frac{15\pi}{8}}{\sqrt{2}} & \frac{\cos\frac{21\pi}{8}}{\sqrt{2}} \end{bmatrix}$$

4x4 DCT Transform

$$\text{DCT}=\begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.50 & -0.50 & -0.50 & 0.50 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix}\begin{bmatrix} 2 \\ 3 \\ 1 \\ 4 \end{bmatrix}$$

1-Dimensional DCT

- For Image Compression using JPEG formats, an image is sub-divided into 8x8 blocks of data.
- The transform is then the dot-product of the basis function matrix with an 8-point input sequence to produce an 8-point output sequence.
- This is effectively correlation of the input data with a range of cosine waves of different frequency.

So an 8x8 Basis function is required……

The blocks in the following slide are the 8 samples of each cosine wave, for n = 0……7.

DCT Basis Functions

$s(0)$ $s(1)$ $s(2)$ $s(3)$ $s(4)$ $s(5)$ $s(6)$ $s(7)$

$$\frac{1}{2\sqrt{2}}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*0\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*1\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*2\pi)$$

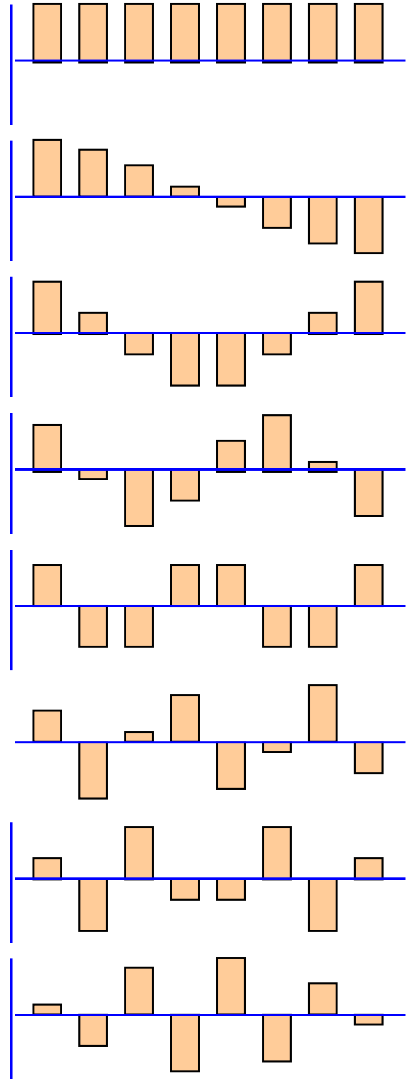
$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*3\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*4\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*5\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*6\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*7\pi)$$



$S(0)$ DC

$$\left.\begin{array}{c} S(1) \\ S(2) \\ S(3) \\ S(4) \\ S(5) \\ S(6) \\ S(7) \end{array}\right\} \text{AC}$$

For each row of the basis function matrix, take k=0, 1, 2…….7, and for each value of k take x=0, 1, 2…….7

$$\Psi = \frac{1}{2}\begin{bmatrix} .71 & .71 & . & . & . & . & . & . \\ \cos\frac{\pi}{16} & \cos\frac{3\pi}{16} & . & . & . & . & . & \cos\frac{15\pi}{16} \\ \cos\frac{2\pi}{16} & \cos\frac{6\pi}{16} & . & . & . & . & . & \cos\frac{30\pi}{16} \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ \cos\frac{7\pi}{16} & \cos\frac{21\pi}{16} & . & . & . & . & . & \cos\frac{105\pi}{16} \end{bmatrix}$$

8x8 DCT Transform Matrix

The coefficients in each row of the transform matrix are the amplitudes of 8 samples of a cosine wave. The first row is a cosine wave of 0Hz (DC), then the frequencies of each cosine wave are increasing (AC).

$$\Psi = \frac{1}{2}\begin{bmatrix} .71 & .71 & .71 & .71 & .71 & .71 & .71 & .71 \\ .98 & .83 & .56 & .19 & -.19 & -.56 & -.83 & -.98 \\ .92 & .38 & -.38 & -.92 & -.92 & -.38 & .38 & .92 \\ .83 & -.19 & -.98 & -.56 & .56 & .98 & .19 & -.83 \\ .71 & -.71 & -.71 & .71 & .71 & -.71 & -.71 & .71 \\ .56 & -.98 & .19 & .83 & -.83 & -.19 & .98 & -.56 \\ .38 & -.92 & .92 & -.38 & -.38 & .92 & -.92 & .38 \\ .19 & -.56 & .83 & -.98 & .98 & -.83 & .56 & -.19 \end{bmatrix}$$

8x8 DCT – Example 1

Suppose we have an 8-point input sequence, this could be a row of pixel values:

S[n] = [2.1, 2.0, 2.11, 1.99, 1.98, 2.02, 2.08, 1.98]

The values in this sequence do not change much.

The DCT is:

8x8 DCT

Transposing back to a row, the output sequence is:

$S[k] = \frac{1}{2}[11.54, 0.10, 0.08, 0.02, -0.11, 0.17, 0.09, 0.13]$

- Only the first element is big, all the others are small.
- This shows that there is a high correlation between the input data and the first row of the transform matrix (the lowest frequency, 0Hz).
- That is, the input data has little variation.

DCT Basis Functions

$s(0)$
$s(1)$
$s(2)$
$s(3)$
$s(4)$
$s(5)$
$s(6)$
$s(7)$

$$\frac{1}{2\sqrt{2}}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*0\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*1\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*2\pi)$$

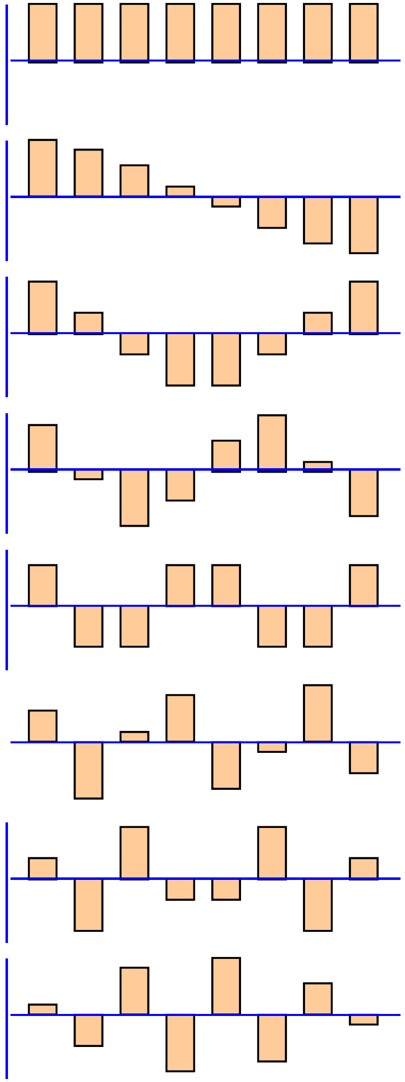
$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*3\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*4\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*5\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*6\pi)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos(\frac{2x+1}{16}*7\pi)$$



$S(0)$ DC

$$\left.\begin{array}{l} S(1) \\ S(2) \\ S(3) \\ S(4) \\ S(5) \\ S(6) \\ S(7) \end{array}\right\} \text{AC}$$

8x8 DCT – Example 2

Suppose we have another 8-point input sequence, this could be another row of pixel values:

S[n] = [2.1, 9.6, -11.2, 7.9, -10.1, 8.6, -6.7, 8.3]

The values in this sequence change a great deal from pixel to pixel.

The DCT is:

8x8 DCT

Transposing back to a row, the output sequence is:

S[k] = $\frac{1}{2}[6.04, -0.22, 13.68, 1.08, 5.61, -8.27, -0.27, -44.38]$

- Only the last element is very big, all the others are relatively small.
- This shows that there is a high correlation between the input data and the last row of the transform matrix (i.e. the highest frequency).
- That is, the input data has very large variation.

DCT Basis Functions

$s(0)$ $s(1)$ $s(2)$ $s(3)$ $s(4)$ $s(5)$ $s(6)$ $s(7)$

$$\frac{1}{2\sqrt{2}}\sum_{x=0}^{7} s(x)\cos\left(\frac{2x+1}{16}*0\pi\right)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos\left(\frac{2x+1}{16}*1\pi\right)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos\left(\frac{2x+1}{16}*2\pi\right)$$

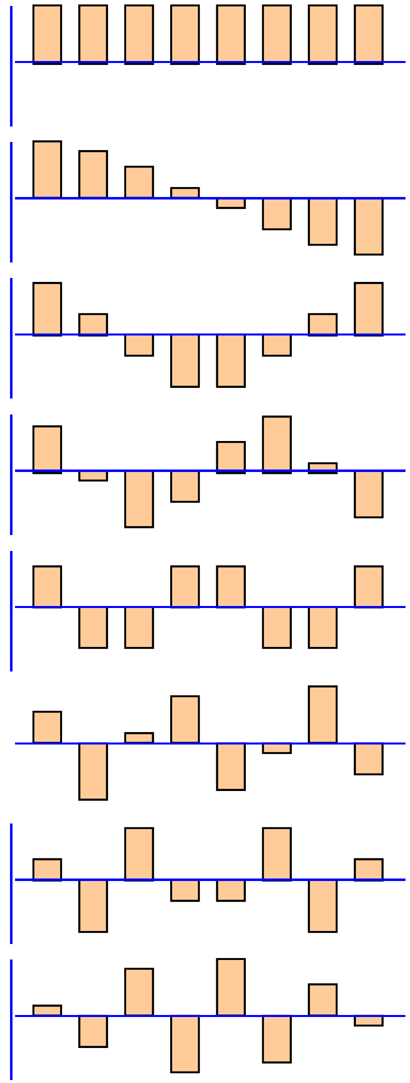
$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos\left(\frac{2x+1}{16}*3\pi\right)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos\left(\frac{2x+1}{16}*4\pi\right)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos\left(\frac{2x+1}{16}*5\pi\right)$$

$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos\left(\frac{2x+1}{16}*6\pi\right)$$

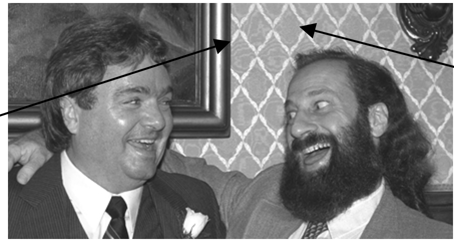
$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos\left(\frac{2x+1}{16}*7\pi\right)$$



$S(0)$ DC

$\left.\begin{array}{l} S(1) \\ S(2) \\ S(3) \\ S(4) \\ S(5) \\ S(6) \\ S(7) \end{array}\right\}$ AC

Original



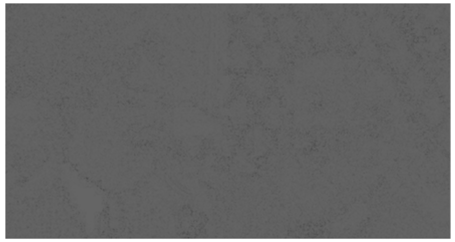
Large variation

Small variation

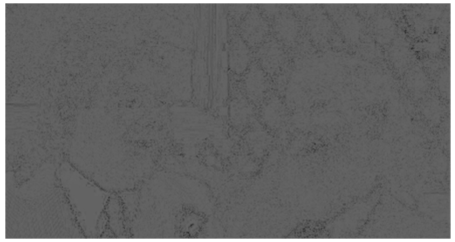
Compressed

Error

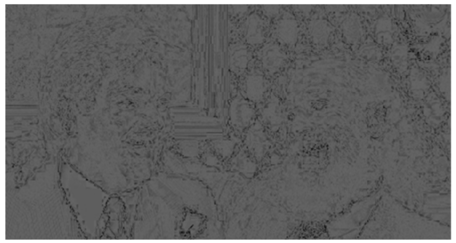
75%



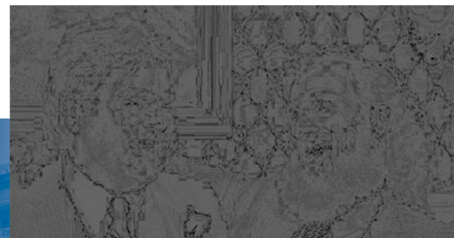
20%



5%



3%



Inverse DCT – to go back to the image domain

- DCT $$DCT[k] = c(k)\sum_{n=0}^{N-1} s[n]\cos\frac{\pi(2n+1)k}{2N}$$

- Inverse DCT $$s[n] = \sum_{k=0}^{N-1} c(k)DCT[k]\cos\frac{\pi(2n+1)k}{2N}$$

2D & nD Transforms

- Many transforms are applied in 1 Dimension.
- But some can also be applied in 2D or *n*D. E.g. Fourier:

1-D: $S(\omega) = \int_{-\infty}^{\infty} s(t) e^{-j\omega t} dt$

2-D: $S(w,v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} s(x,y) e^{-j(wx+vy)} dxdy$

$$= \int_{-\infty}^{\infty}\left(\int_{-\infty}^{\infty} s(x,y) e^{-jwx} dx\right) e^{-jvy} dy$$

n-D: $S(\mathbf{w}) = \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty} s(\mathbf{x}) e^{-j(\mathbf{w}\cdot\mathbf{x})} d\mathbf{x}$

Dot Product

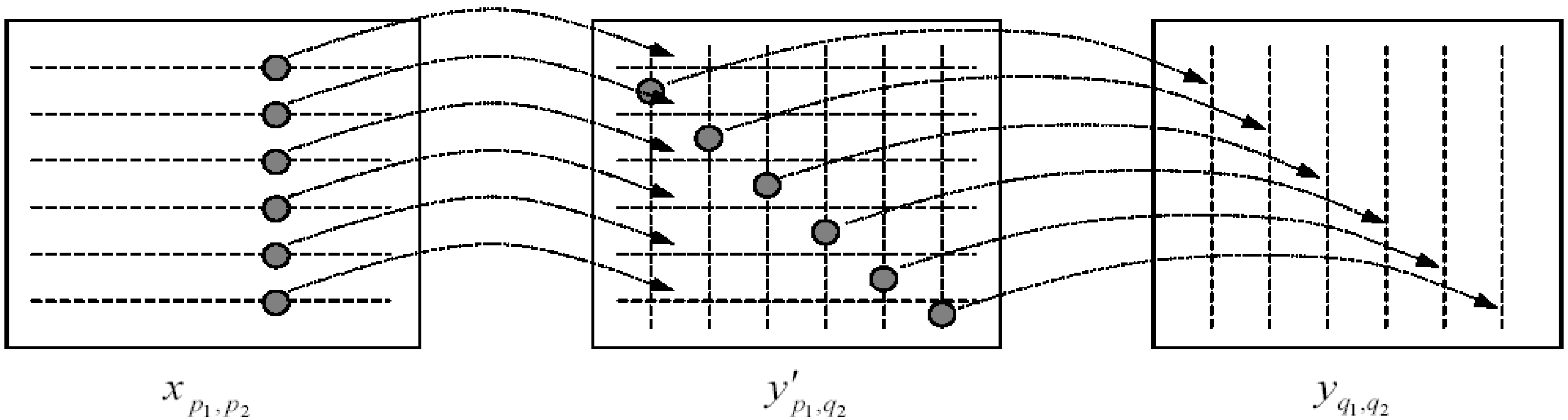
where $\mathbf{w} = (w_1, w_2, \ldots, w_n)$ and $\mathbf{x} = (x_1, x_2, \ldots, x_n)$.

An n-D transform is *separable* if we can apply sequence of 1-D transforms (see 2-D case above).

Separable DCT applied to Image Compression

Separable Transforms

May be implemented by applying the one dimensional transform first to the rows of the image and then to its columns (note that changing the application order does not change the result).



2-dimensional DCT

- Defined as:

$$DCT_{2d}[i,j] = c(i,j)\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} s[m,n]\cos\frac{\pi(2m+1)i}{2N}\cos\frac{\pi(2n+1)j}{2N}$$

$$c(i,j) = \begin{cases} 1/N & i=0, j=0 \\ \sqrt{2}/N & i\neq 0, j=0 \\ \sqrt{2}/N & i=0, j\neq 0 \\ 2/N & i\neq 0, j\neq 0 \end{cases}$$

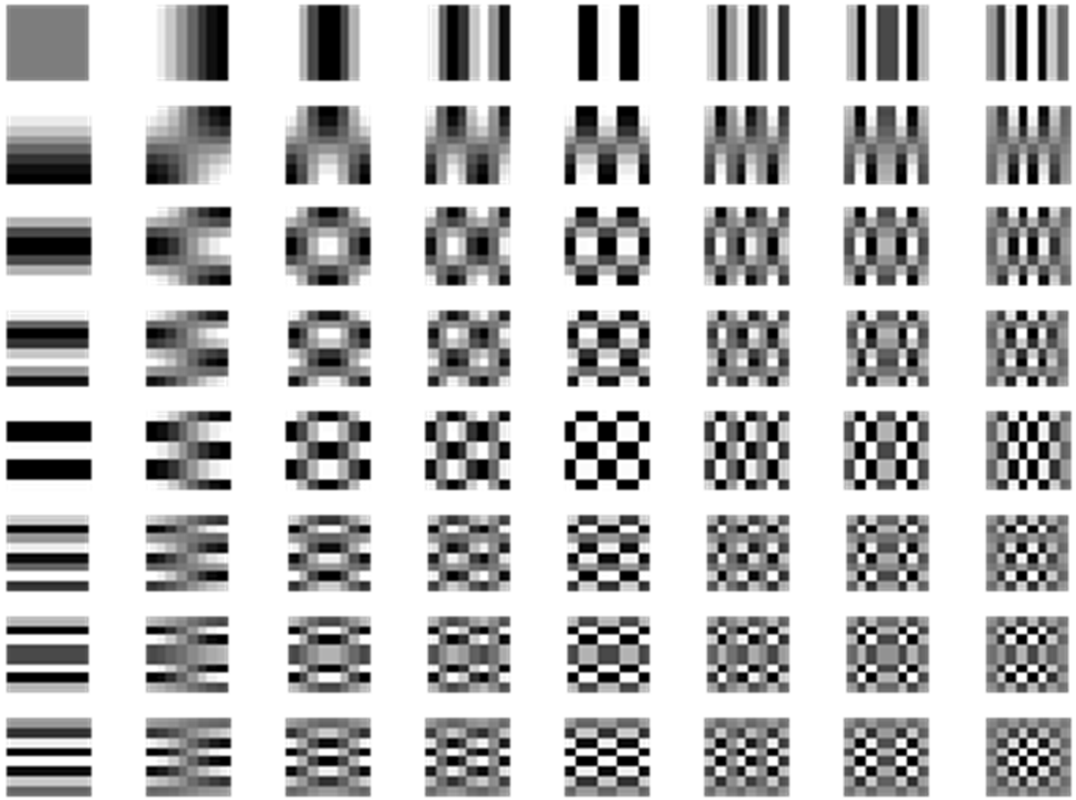
- Compare with:

$$DCT[k] = c(k)\sum_{n=0}^{N-1} s[n]\cos\frac{\pi(2n+1)k}{2N}$$

- Separable:

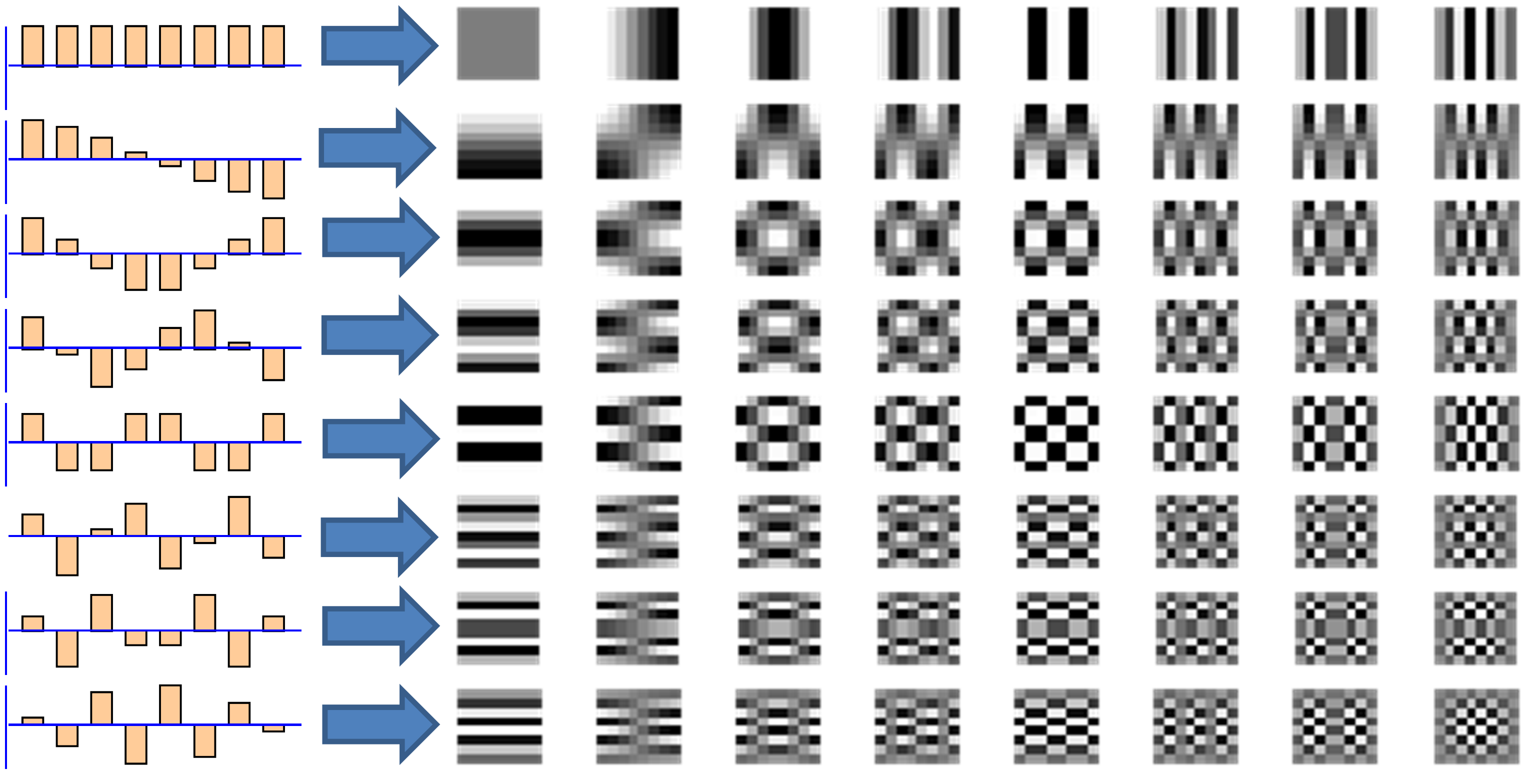
$$DCT_{2d}[i,j] = c(i)\sum_{m=0}^{N-1}\cos\frac{\pi(2m+1)i}{2N}\left[c(j)\sum_{n=0}^{N-1} s[m,n]\cos\frac{\pi(2n+1)j}{2N}\right]$$

This diagram represents the cosine basis functions for a 2D-DCT

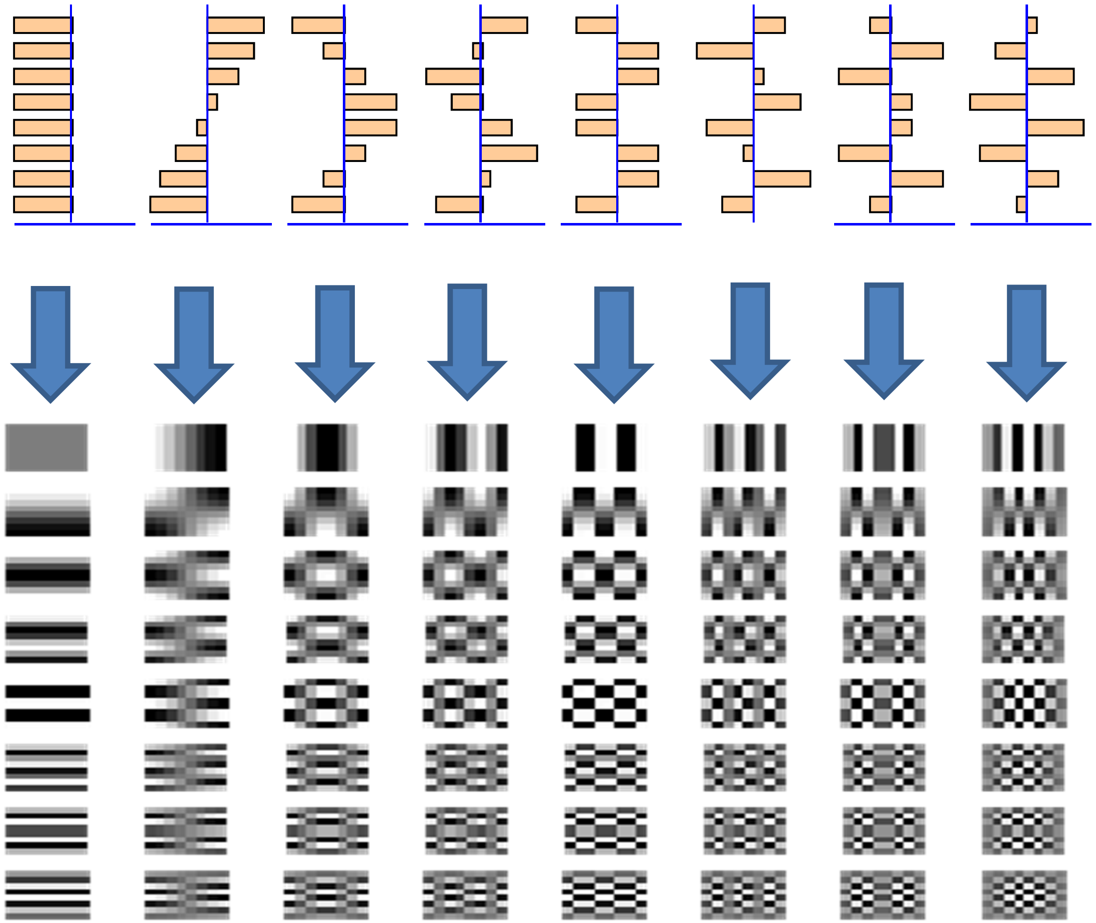


After the 2D transform is carried out, there will be a number in each of these 64 locations. The value of the number represents the amount of change in the input data at that location.

Compare the first column of this diagram with the basis cosine waves of an 8-point DCT:



Compare the first row of this diagram with the basis cosine waves of an 8-point DCT:



We know from the examples in the lecture that there is a correlation between the variation in the input data and the cosine functions:

Ex 1. S[n] = [2.1, 2.0, 2.11, 1.99, 1.98, 2.02, 2.08, 1.98] gives the DCT output [little change in input values]

S[k] = $\frac{1}{2}$[11.54, 0.10, 0.08, 0.02, -0.11, 0.17, 0.09, 0.13]

Ex 2. S[n] = [2.1, 9.6, -11.2, 7.9, -10.1, 8.6, -6.7, 8.3] gives the DCT output [large change in input values]

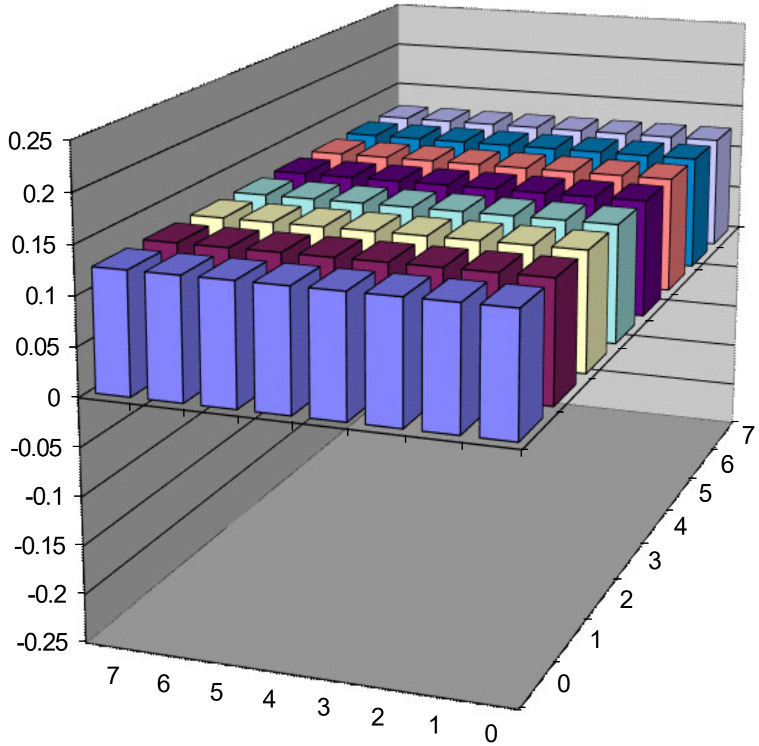
S[k] =

$$\frac{1}{2}[6.04, -0.22, 13.68, 1.08, 5.61, -8.27, -0.27, -44.38]$$

So if there are a large number in a location in the DCT output then we know something about the variation in the input data

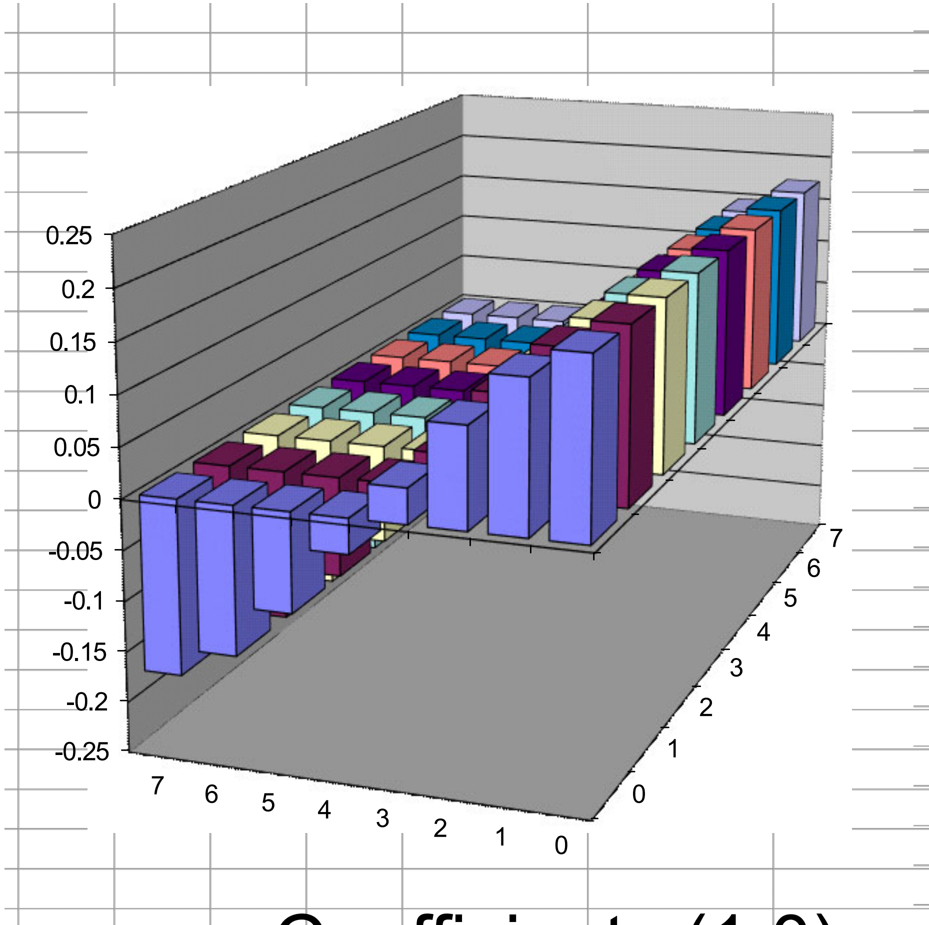
- When we perform the 2D DCT, we will have a number in each of the 64 locations
- Big numbers mean there is a high correlation between the variation in the input data and the frequency of the relevant cosine wave
- If the input data is from an image, then we know if there is little change in that 8x8 part of the image or large changes in that part

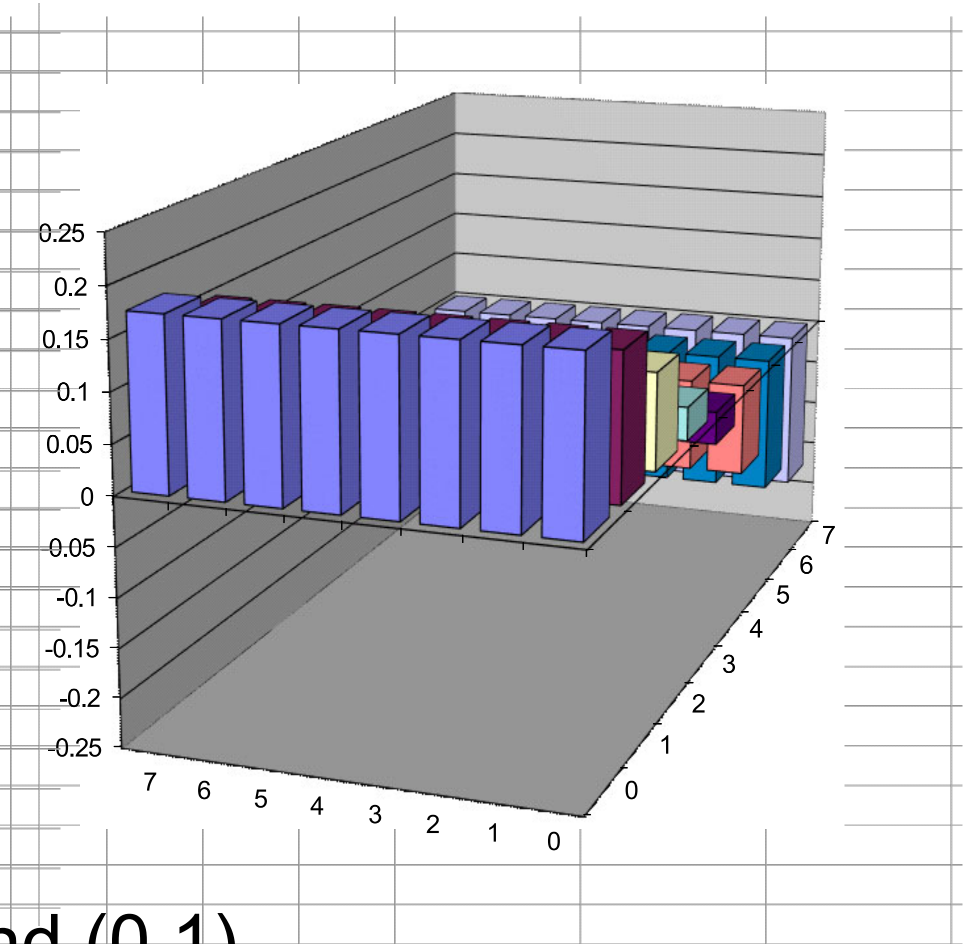
Inverse DCT of Selected Coefficients



– Coefficient (0, 0)
  - i.e., F(0, 0) = 1, all others = 0
– Characterizes overall average

Inverse DCT of Selected Coefficients

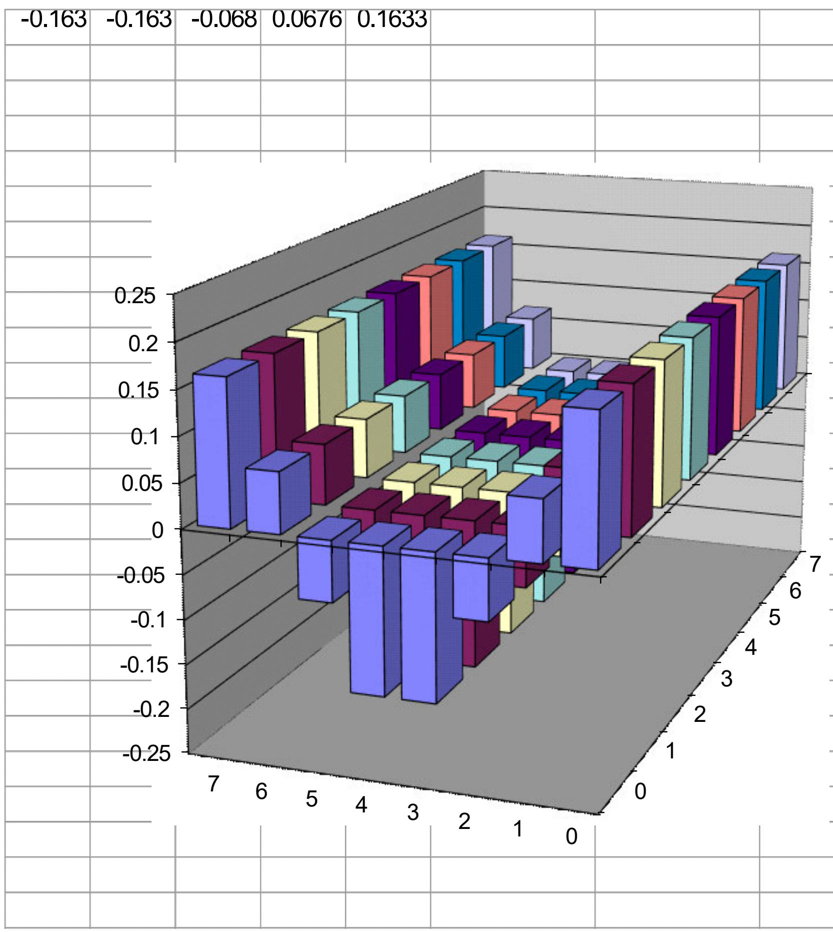




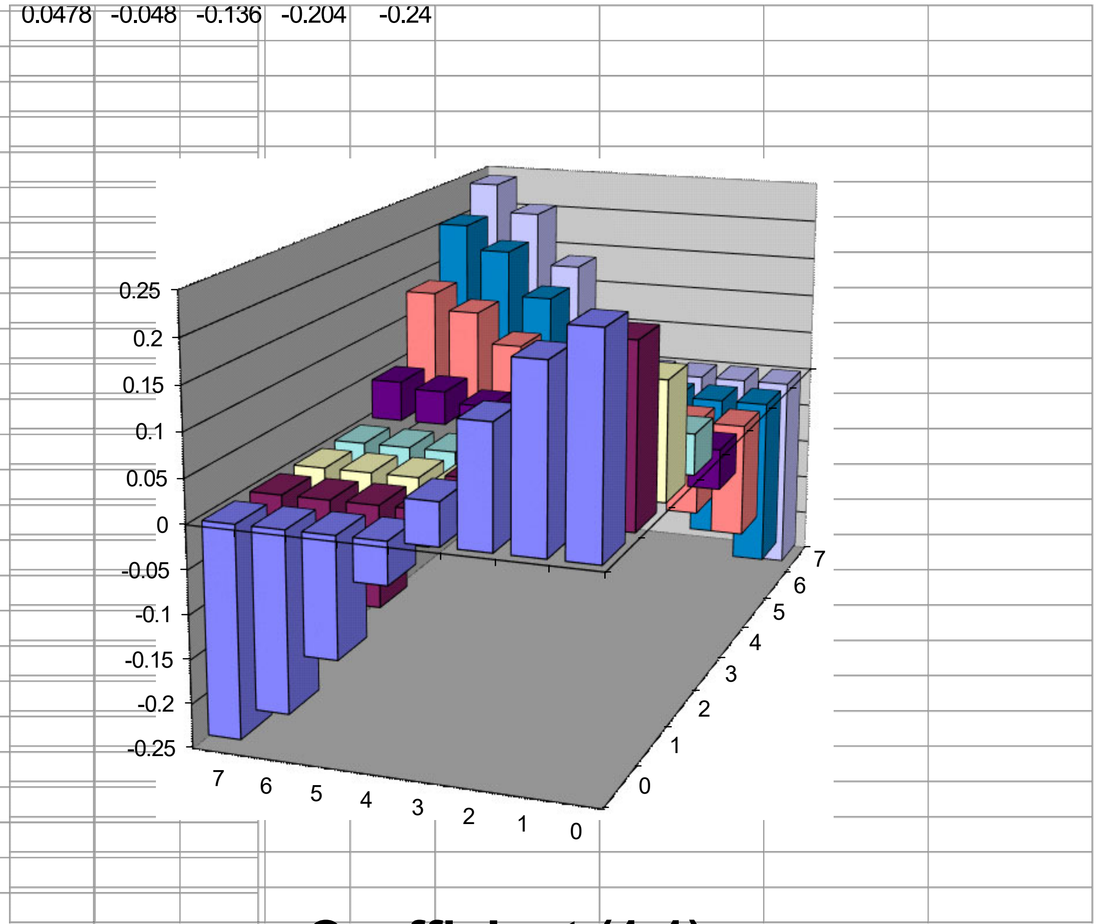
– Coefficients (1,0) and (0,1)

– Capture horizontal or vertical gradient

Inverse DCT of Selected Coefficients

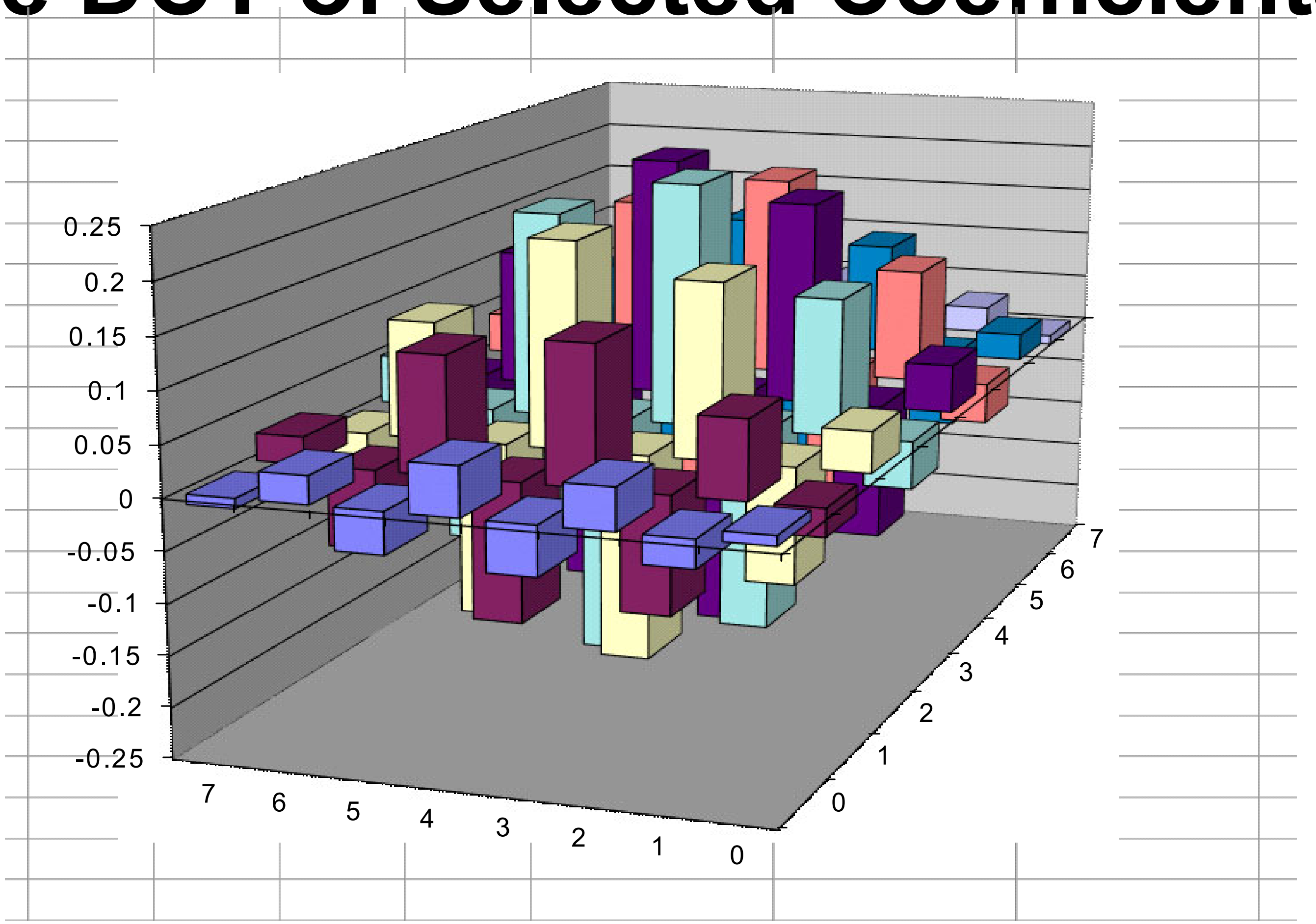


- Coefficient (2,0)
- Captures vertical banding



- Coefficient (1,1)
- Captures diagonal variation

Inverse DCT of Selected Coefficients



– **Coefficient (7,7)**
– **Captures high spatial variations**

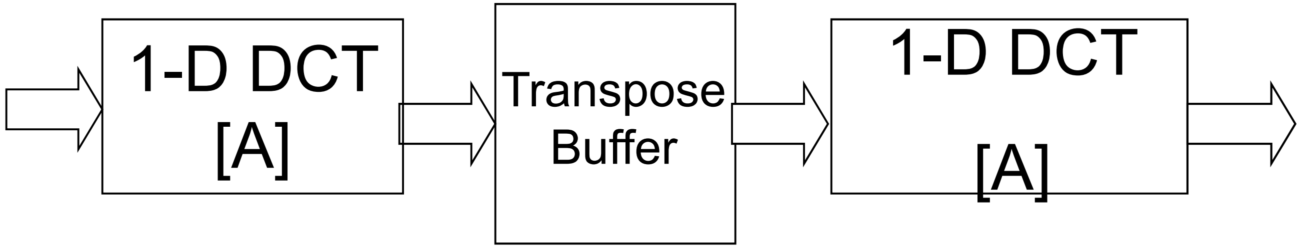
DCT and Image Compression

- DCT is relatively easy to implement, and is lossless
- DCT energy compaction allows lossy image compression
  - Just a few of the transform coefficients represent the majority of the **energy** in the sequence
- DCT used in
  - JPEG image compression format
  - MPEG video compression formats.
- Fast algorithms exist for computation.
  - Fixed point integer arithmetic
- Good perceptual properties.
  - Losing higher frequency results in a bit of blurring.

2D DCT

- Break the image up into 8 x 8 (64-pixel) blocks and transform them independently.
- Simplifies computation and memory requirements
- DCT is separable, so 2D DCT can be computed by applying 1D transforms separately to the rows and columns
- Rows and columns of blocks are 8 pixels each
- So: need only design a DCT for 8 x 8 transforms
- Result – transform:
  block of 64 intensity values into 64 coefficients

2D DCT Implementation



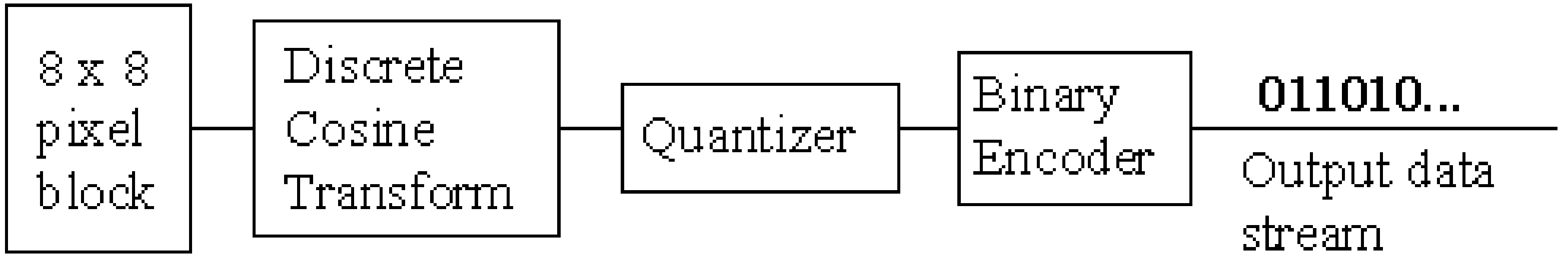
The 8-point DCT can be written as a matrix transform
$Y=AX$

Simplification due to symmetry of A.

Rearranging is possible yielding fast algorithms to compute the DCT (c.f. Fast Fourier algorithms).

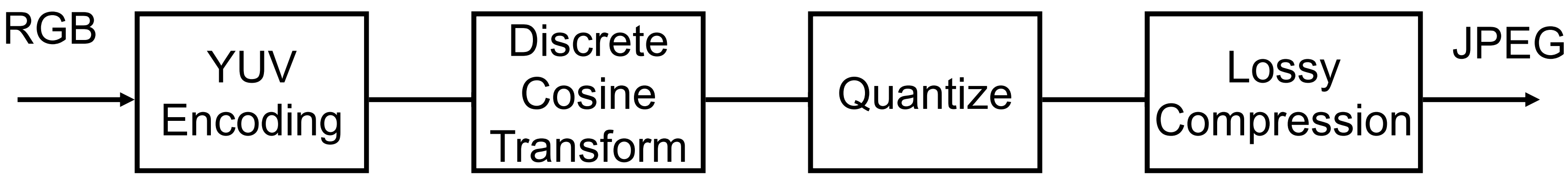
For instance, the 8 point DCT may be computed in just 11 multiplications.

JPEG Compression - Macroblocks



- YUV encoded image (like RGB)
- Y is cut into 8x8 tiled pixel regions.
- U and V cut into 8x8 tiled pixel regions.
- Macroblock defined as 4 Y tiles that form a 16x16 pixel region and associated U and V tiles.
- Macroblocks organized in row order fashion from top to bottom.

JPEG Encoding Steps



- Encoding
  - Convert to different colour representation
  - Typically get 2:1 compression
- Discrete Cosine Transform (DCT) (Lossless)
  - Transform 8 X 8 pixel blocks
- Quantize
  - Reduce precision of DCT coefficients
  - Lossy step
- Lossy Compression
  - Express image in highly compressed form

YUV Encoding

- Computation
  - RGB numbers between 0 and 255
  - *Luminance* Y encodes grayscale intensity between 0 and 255
  - *Chrominance* U, V encode color information between –128 and +127
    - Similar to Color (Hue) and Tint (Saturation) controls on color TV
- Conversion

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

  - Values saturate at ends of ranges
- Color Subsampling
  - Average U,V values over 2 X 2 blocks of pixels
  - Human eye less sensitive to variations in color than in brightness

DCT Compression

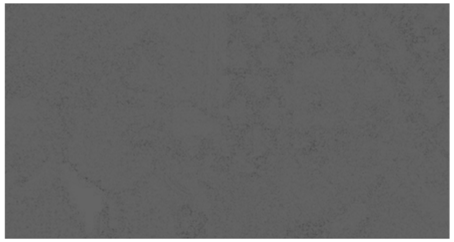
- Each tile (aka block) in a macroblock is transformed with a 2D DCT.
  - 1d: 8 pixel values are transformed into 8 DCT coefficients.
  - 2d: apply 1d transform to all of the rows and then apply 1d transform to all of the columns.
- Each block is now 64 coefficients instead of 64 pixel values.
- Each coefficient quantized independently.
  - Allows larger quantization factors to be used with higher frequency coefficients.
- Quantization is controlled by Quantization table

Original

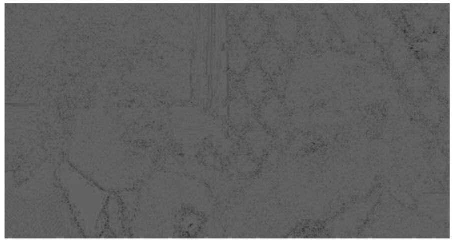
Compressed

Error

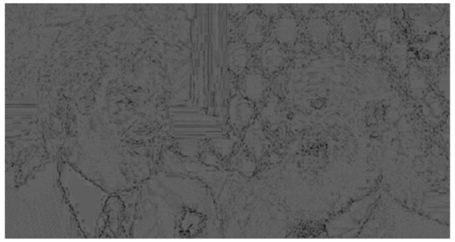
75%



20%

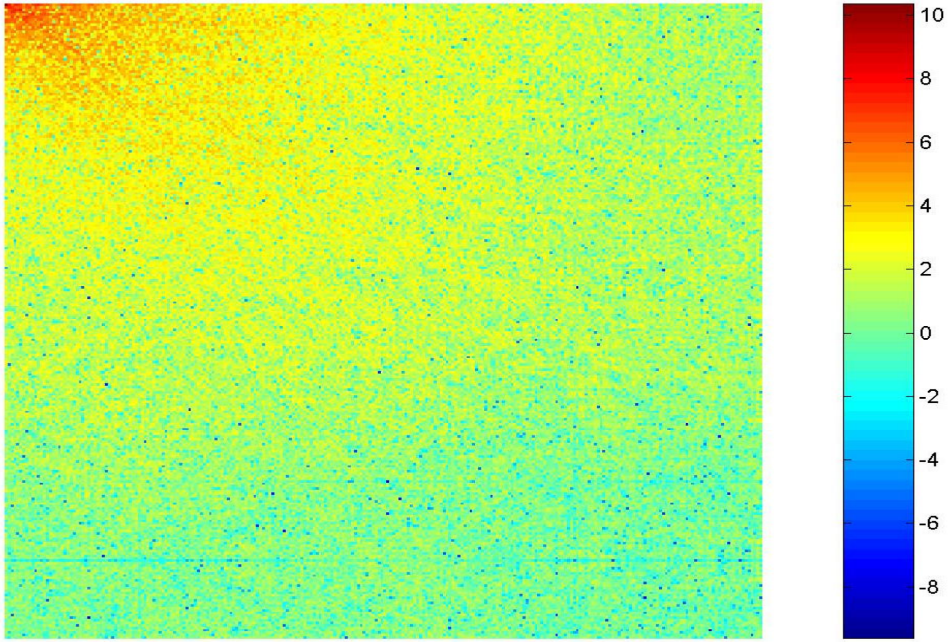


5%



3%

The logarithmic distribution of DCT coefficients



DCT compression via thresholding: 10:1

DCT coefficients are thresholded to zero