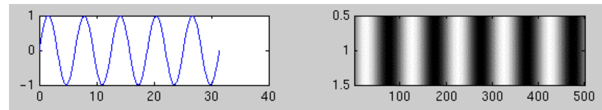
More on image transforms

C306
Fall 2001
Martin Jagersand

Sine function

```
t = 0:pi/50:10*pi
I = sin(t)
subplot(221)
plot(t,I)
```

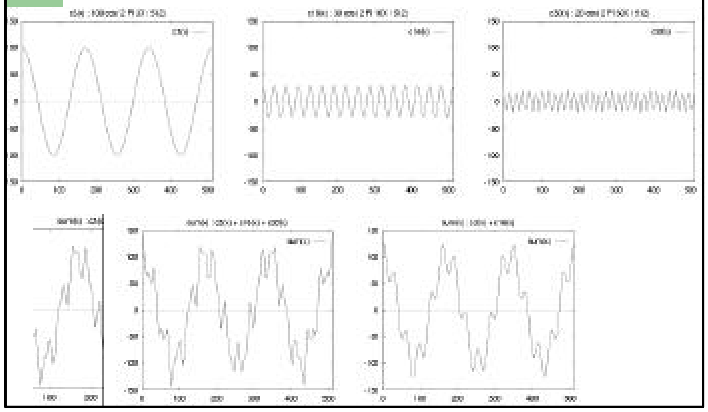
- `Image(I) ???`
- `subplot(222)`
  `image(64*(I+1)/2)`
  `colormap gray`



Adding sines

Adding sines

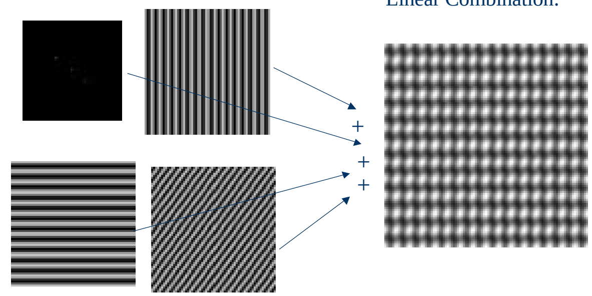
Adding sines



Constructing an image

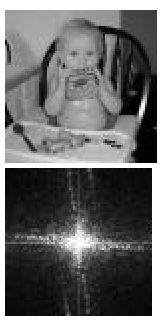
Basis vectors

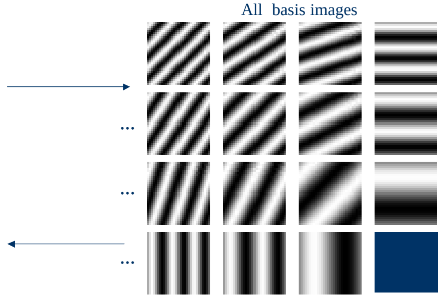
Linear Combination:



Analyzing an image

All basis images





intensity ~ that frequency's coefficient

Cosine Transforms:

- Used in jpeg, mpeg compression.
- These are the most common compression techniques.
- Example: Jpeg compression:

1. Image -> 8x8 blocks
2. Cos trans each block
3. Quantize the transform coefficients
4. Code non-zero coefficients

Definition, Gonzalez
Discrete Cosine Transform, DCT

- Forward Transform:

$$C(u) = \alpha(u) \sum_{x=0}^{N-1} f(x) \cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

- Inverse:

$$f(x) = \sum_{u=0}^{N-1} \alpha(u) C(u) \cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

- Where: $\alpha(u) = \begin{cases} 1/\sqrt{N}; & \text{for } u = 0 \\ 2/\sqrt{2N}; & \text{for } u \geq 1 \end{cases}$

Definition, matrix form:

- Let C=Af
- f=A'C ('= transpose), where:

$$A = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 1 & \sqrt{2}\cos(3\pi/2N) & \cdots & \sqrt{2}\cos((2N-1)\pi/2N) \\ \vdots & & \ddots & \vdots \\ 1 & \sqrt{2}\cos(3(N-1)\pi/2N) & \cdots & \sqrt{2}\cos((2N^2-3N+1)\pi/2N) \end{pmatrix}$$

Example
4x4 Cosine transform

- Cosine basis Test "image"

$$\begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Transformed Im

- Why [2, 0, 0, 0]' ???

Why??
Transforms graphically

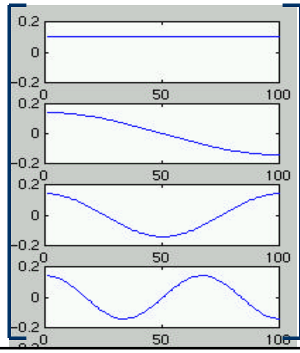
- Image – Transform pair 1

$$\begin{bmatrix} 1 \\ 0.5 \\ -0.5 \\ -1 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 \\ 1.58 \\ 0 \\ -0.11 \end{bmatrix}$$

- Image – Transform pair 2

$$\begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \Rightarrow \begin{bmatrix} 5 \\ -2.23 \\ 0 \\ -0.16 \end{bmatrix}$$

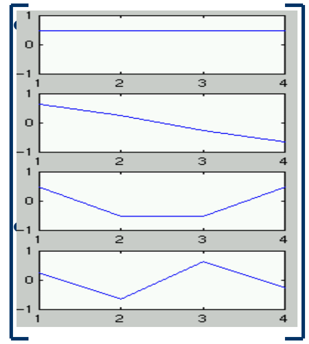
Why??
Transforms graphically



$$\begin{bmatrix} 1 \\ 0.5 \\ -0.5 \\ -1 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 \\ 1.58 \\ 0 \\ -0.11 \end{bmatrix}$$

$$\begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \Rightarrow \begin{bmatrix} 5 \\ -2.23 \\ 0 \\ -0.16 \end{bmatrix}$$

Why??
Transforms graphically

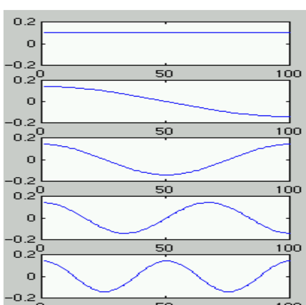


$$\begin{bmatrix} 1 \\ 0.5 \\ -0.5 \\ -1 \end{bmatrix} \Rightarrow \begin{bmatrix} 0 \\ 1.58 \\ 0 \\ -0.11 \end{bmatrix}$$

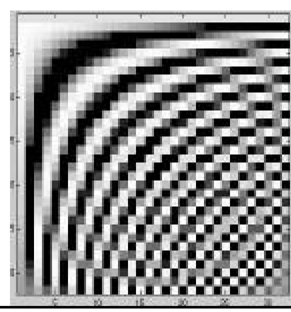
$$\begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \Rightarrow \begin{bmatrix} 5 \\ -2.23 \\ 0 \\ -0.16 \end{bmatrix}$$

Cosine bases

- First 5 vectors:



- Image of full 32x32:



Properties of the cosine transform

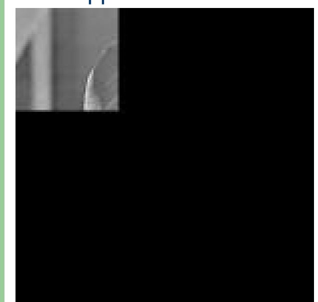
- Real and orthogonal, ie $\mathbf{A}^{-1} = \mathbf{A}^T$
- Not the real part of the DFT, but related (try type dct in matlab)
- Can be computed in O(NlogN) time
- “Compacts” the energy in images into a few coefficients. (Compression)

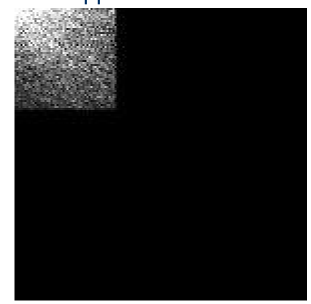
Examle:
Energy compaction

- Original Lena image
- 2D DCT

Example:
Cropping in image and DCT:

- Cropped Lena
- Cropped DCT



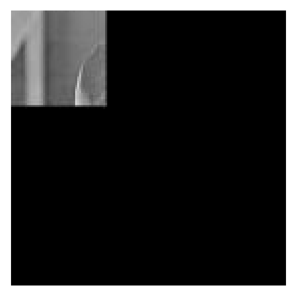


Comparison
Image recovery from cropped DCT

- Original image
- Recovered image

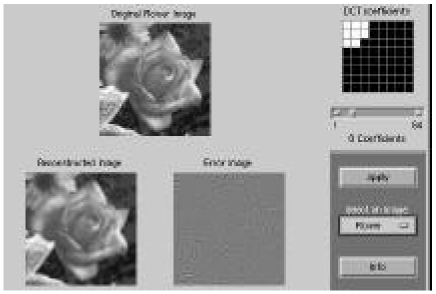
Example
Image recovery from cropped DCT

- Cropped image
- Recovered image

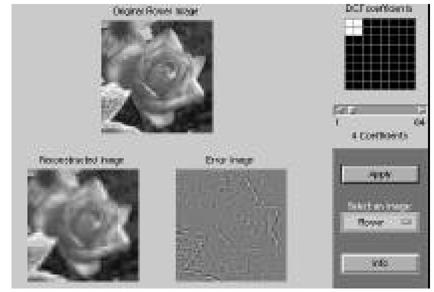


Block coding
Example, 8 coefficients

- Block coded image (8x8 pixel blocks)



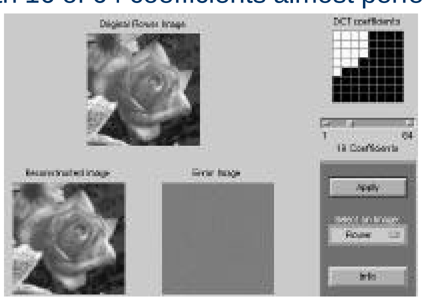
Example 2, using only 4 coeff



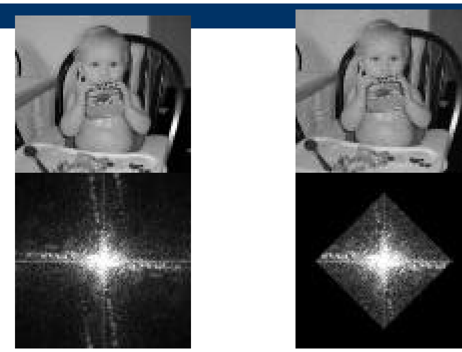
- Notice “grainy” result from using only low frequency coeff

Example 3

- With 16 of 64 coefficients almost perfect:



Cropping in Fourier space



Data Reduction: only use *some* of the existing frequencies