The University Of Sheffield.

EEE6081 (EEE421) Visual Information Engineering (VIE)

Topic 3: Signal Transforms

- Convolution as a matrix multiplication.
- What are the uses of transforms?
- Transform example.
- Orthogonal Transforms.
  - Perfect reconstruction.
  - Parseval’s Theorem.
- The Discrete Cosine Transform.
- Other Transforms.
- N-point transforms on signals.
- N-point transforms on images.

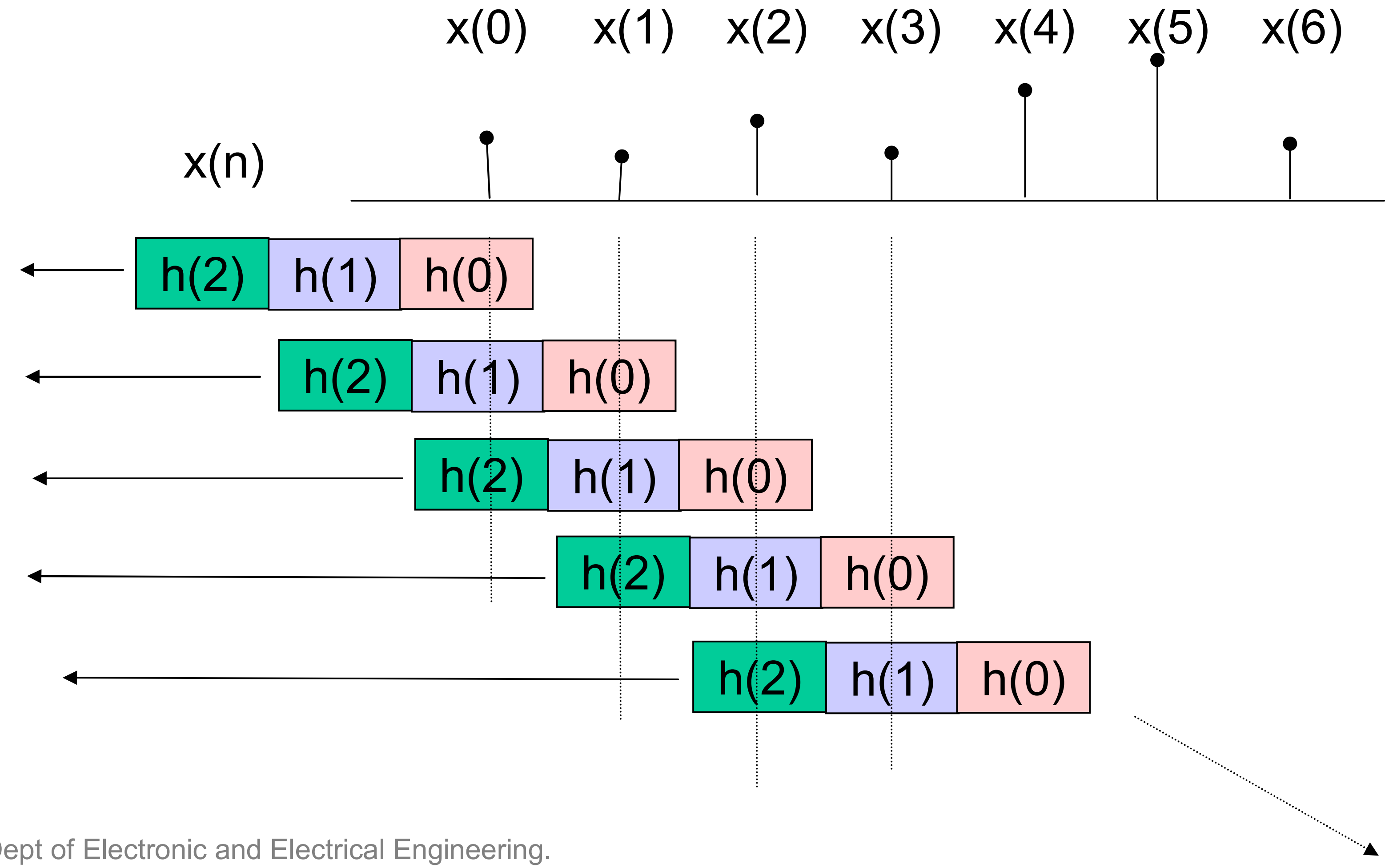
Dr Charith Abhayaratne
Email: c.abhayaratne@sheffield.ac.uk
Office: F176 Tel: 25893

Convolution as a matrix multiplication

$$y = x * h$$

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

h(k)= [h(0) h(1) h(2)]



Convolution as a matrix multiplication

What are the rest of the elements in the matrix?

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ . \end{bmatrix} = \begin{bmatrix} h(0) & & & & \\ h(1) & h(0) & & & \\ h(2) & h(1) & h(0) & & \\ & h(2) & h(1) & h(0) & \\ & & . & . & . \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ . \end{bmatrix}$$

output

Convolution matrix
(a block diagonal matrix)
Let’s call it H (The transform matrix)

input

Invertibility property: If $H^{-1}H = I$ (The Identity matrix)

For most filters: Either H is not invertible or there exists an H^{-1}, but is not stable.
Therefore, filters are usually lossy transforms.

Transforms

A transform of a signal is a new representation of that signal.

Consider the following system of equations:

$y(0)= x(0) + x(1)$

$y(1)= x(0) - x(1)$

$y(2)= x(2) + x(3)$

$y(3)= x(2) - x(3)$

This transforms x into y using a 4-point transform.

In other words the signal is represented in y-domain using the linear combinations of signal components in the x domain.

What is the inverse transform for the above transform?

Transforms

$y(0)= x(0) + x(1)$

$y(1)= x(0) - x(1)$

$y(2)= x(2) + x(3)$

$y(3)= x(2) - x(3)$

1. Write down this transform in matrix representation:
2. Repeat the same for the inverse transformation
3. Check the Invertibility condition

Transforms

$$y(0)= x(0) + x(1)$$
$$y(1)= x(0) - x(1)$$
$$y(2)= x(2) + x(3)$$
$$y(3)= x(2) - x(3)$$

1. Why do we use transforms?
   (We will discuss the solution throughout this topic
So, write them down here when you have learned them)

   1.
   2.
   3.

An example: Consider $x(0)= 12$ $x(1)=10$ $x(2)= -9$ $x(3)= -10$ #Plot X

Compute: $y(0)$ $y(1)$ $y(2)$ $y(3)$ #Plot Y

What can you learn about this data from the y-domain representation?
How do you interpret the transform domain values.

Now set $y(1)=y(3)=0$ and compute the new x values. #Plot new X

What have you learned about transform domain processing?

Transforms

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}}_{H} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix} \qquad \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix} = \underbrace{\frac{1}{2}\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}}_{H^{-1}} \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix}$$

Is H the transpose of H^{-1} ?

We can split the factor ½ into (1/sqrt(2) x 1sqrt(2)) and use as the normalistaion constant for both H and H^{-1}.

Now, the inverse is the transpose of the original matrix.
This is true only when the transform is an **orthogonal transform**:

Compute the sum of squares of the output (y) and show that $||x||^2 = ||y||^2$.
In this case we call the transform is **unitary**.

Orthogonal Transforms

- Discrete transforms "map" data from one domain into another.

$$x \longrightarrow \boxed{\text{Transform}} \longrightarrow c$$

- x is input data on time or space domain.
- c is the transform coefficient domain (For the Fourier transform it is frequency domain).

- 1D transforms have the form:

(also called an N-point transform)

$$c_n = \sum_{i=0}^{N-1} f_{ni} x_i \qquad for\ n = 0,\ldots,N-1.$$

N coefficients

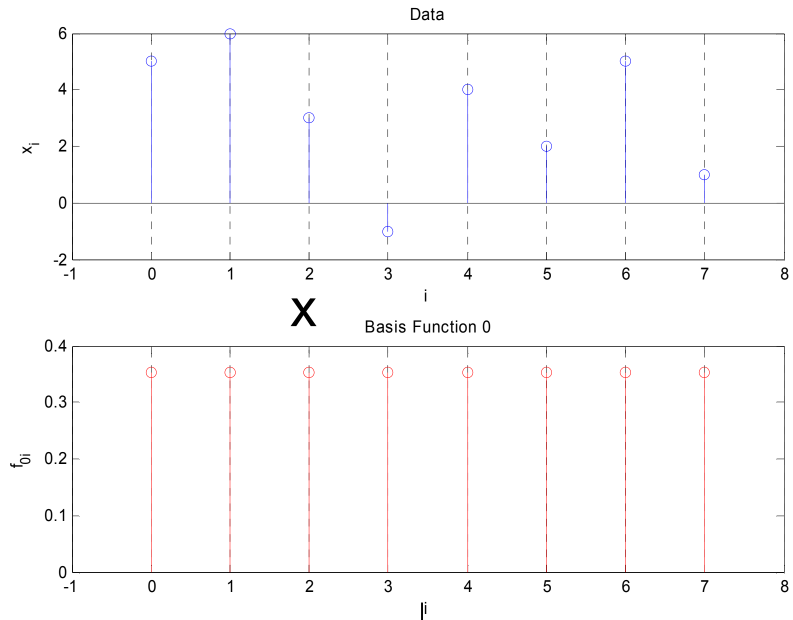
N basis functions

N data values

- The corresponding matrix notation: C=FX ,
- Rows of F represent corresponding **basis functions** of the transform.

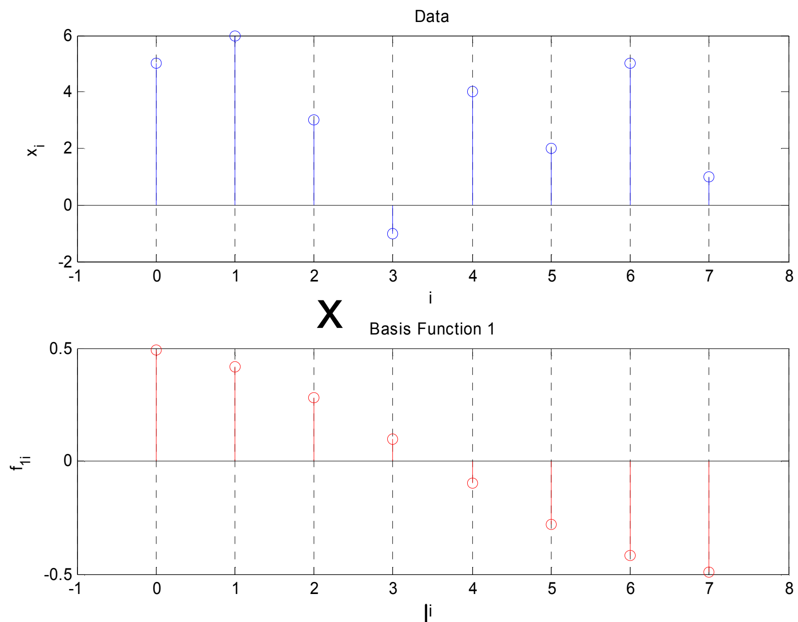
Orthogonal Transforms

x



f_0

x



f_1

c_0= sum of all products

$$=\sum f_{0i} x_i$$

c_1= sum of all products

$$=\sum f_{1i} x_i$$

How do you find the n^{th} coefficient c_n ?

Orthogonal Transforms

- Inverse transform reconstructs data.

$$x_j = \sum_{n=0}^{N-1} g_{jn} c_n \qquad for\ j = 0,\ldots,N-1.$$

- We need perfect reconstruction.
- Let's expand the inverse transform:

$$x_j = \sum_{n=0}^{N-1} g_{jn} \sum_{i=0}^{N-1} f_{ni} x_i$$
$$= \sum_{i=0}^{N-1} x_i \sum_{n=0}^{N-1} g_{jn} f_{ni}$$

- We will get perfect reconstruction if

$$\sum_{n=0}^{N-1} g_{jn} f_{ni} = 1 \quad \text{when } i = j$$
$$= 0 \quad \text{when } i \neq j$$

- i.e., the Identity matrix.
- For Orthogonal Transforms -------------------------- $g_{jn} = f_{jn}$ (transpose)
- The orthogonality condition: $\boxed{\sum_{n=0}^{N-1} f_{jn} f_{ni} = \delta_{ji}}$

Orthogonal Transforms

- Consider the total power of the data:

$$P = \sum_j (x_j)^2 = \sum_j \left(\sum_n f_{jn} c_n \right)^2$$

- When you multiply this out, you get the sum of all possible pair products.

$$\begin{aligned} P &= \sum_j \sum_m \sum_n f_{jn} c_n f_{jm} c_m \\ &= \sum_n \sum_m c_n c_m \sum_j f_{nj} f_{jm} \\ &= \sum_n \sum_m c_n c_m \delta_{nm} \\ &= \sum_n c_n{}^2 \end{aligned}$$

Homework:
Prove the same using matrix representation.

- Parseval's Theorem: $\boxed{\sum_i x_i{}^2 = \sum_i c_i{}^2, \quad \text{provided} \quad \sum_{j=0}^{N-1} f_{nj} f_{mj} = \delta_{nm},}$ i.e., the orthogonality condition.

The Discrete Cosine Transform (DCT)

- Uses Cosines as basis functions:
- The N-point DCT

$$c_n = \sqrt{\frac{e_n}{N}} \sum_{i=0}^{N-1} \left[\cos\left(\frac{(2i+1)n\pi}{2N} \right) \right] x_i$$

$$e_n = \begin{cases} 1 & \text{when } n = 0 \\ 2 & else \end{cases}$$

Please bring your results to the next lecture

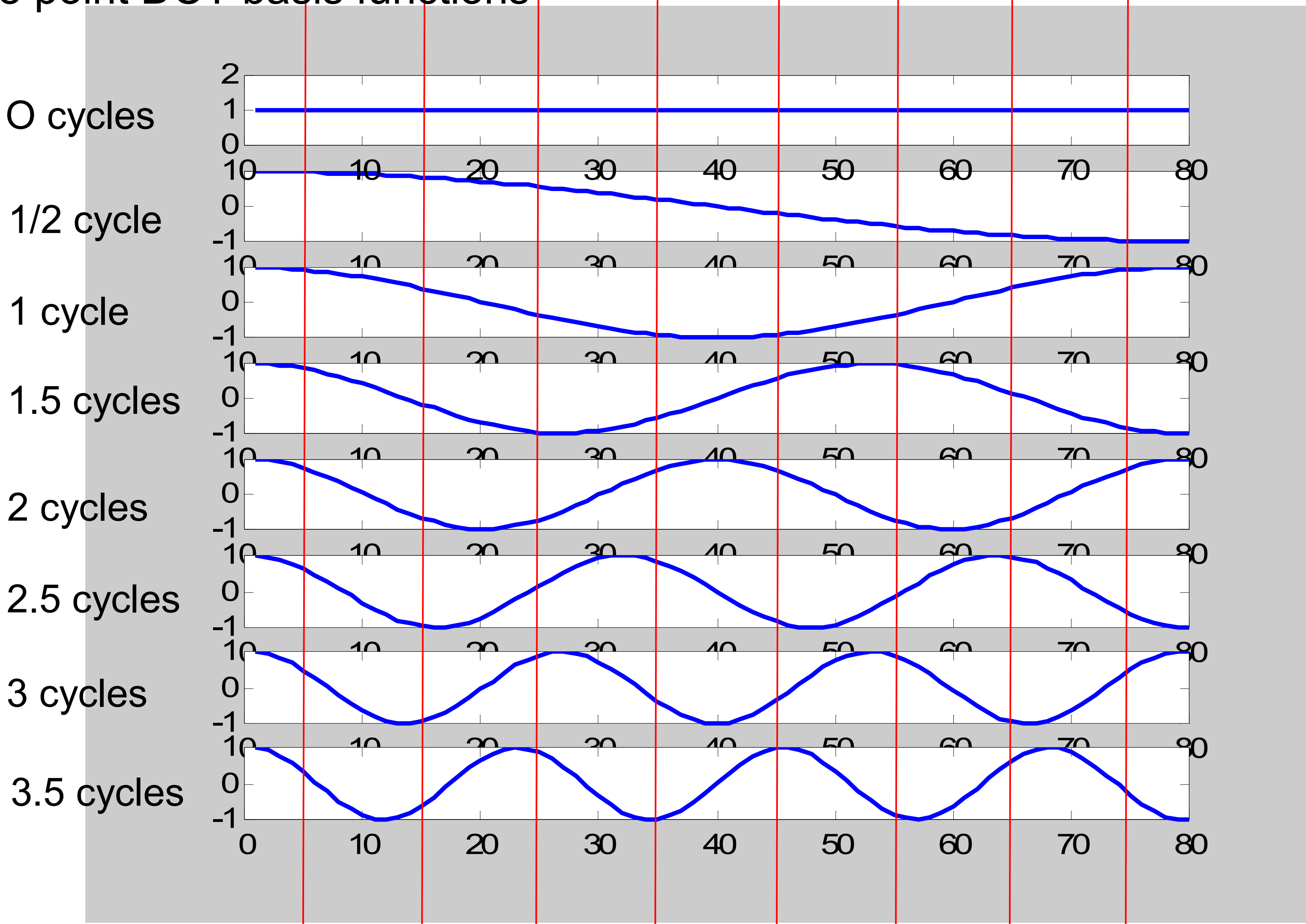
Homework:

Using MATLAB

1. Find out the N-point DCT transforms matrix for N=2, 4 & 8.
   Hint: >lookfor dct %to find out command for computing dct in Matlab
   We Know Y=HX for transforms in matrix notation
   What is Y when X=I, where I is the Identity matrix of NxN elements.
2. Plot them using the “stem” command.
   Hint: >help stem
3. Verify that these DCTs are orthogonal
4. Compute the Inverse of H for all DCTs and derive an expression.

The Discrete Cosine Transform (DCT)

8 point DCT basis functions



1. The Coefficients are real.
2. Has half as well as full period cosines.
3. Symmetry can be either odd or even.
4. Can compute using the FFT

The Discrete Cosine Transform (DCT)

- Consider the input data:
  X= [5 6 3 4 3 4 2 3]
- H= 8-point DCT transform
- Y=HX gives
  Y= [10.6066 2.4635 0.6533 0.6539 0 -1.0878 -0.2706 -1.8222]
- Is X or Y more correlated?
- Y_{new}=[11 2 1 1 0 -1 0 -2] by rounding.
- X_{new}=[5 6 3 4 3 5 2 3] by inverse DCT.
- Sum of x^2=124 in Y, 112.5 out of 124 is coming from a single coefficient.

Transforms

- Why do we need transforms?

1. To analyse data or signals. (Different features can be identified in different representations)
2. To decorrelate data.
3. To compact power of data into a fewer coefficients.
4. To use above 2 & 3 to compress data.

- Other transforms

1 Discrete Sine Transform (DST) – Sine waves as bases

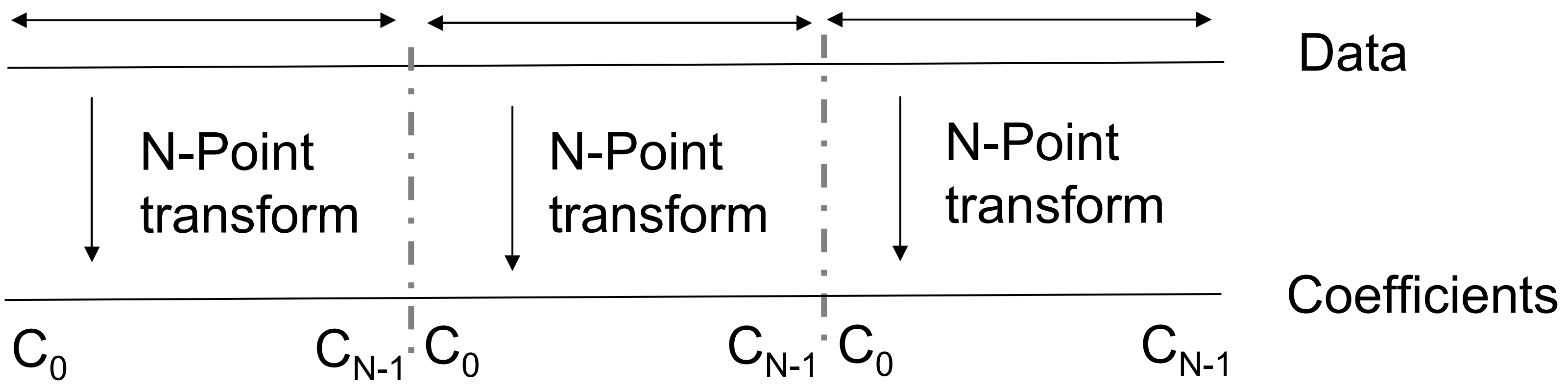
2 Walsh Hadamard Transform – Square waves as bases

3 Wavelet Transforms - Short localised waves as bases

Transforms

- An N-point transform
  - Contains N basis functions
  - When applied on N data points, results in N coefficients.

- If the length of data (L) is larger than N,
  - First the data is partitioned into segments with N data points
  - and then each segment is transformed using the N-point transform.



The University Of Sheffield.

Transforms

- N-point transforms on images:

Consider an image (x) with NxN pixels:

The 2D transform coefficients c_{mn} of image pixel values x_{ij} are obtained by realising the 2D transform as a **separable transform**.

i.e., first transforming the rows (or columns) using the 1D transform (F) followed by transforming the columns (or rows). The order of operation does not matter. (Thus called separable).

$$y_{ni} = \sum_{j=0}^{N-1} f_{nj} x_{ji} \qquad for\ n = 0,\ldots,N-1.$$

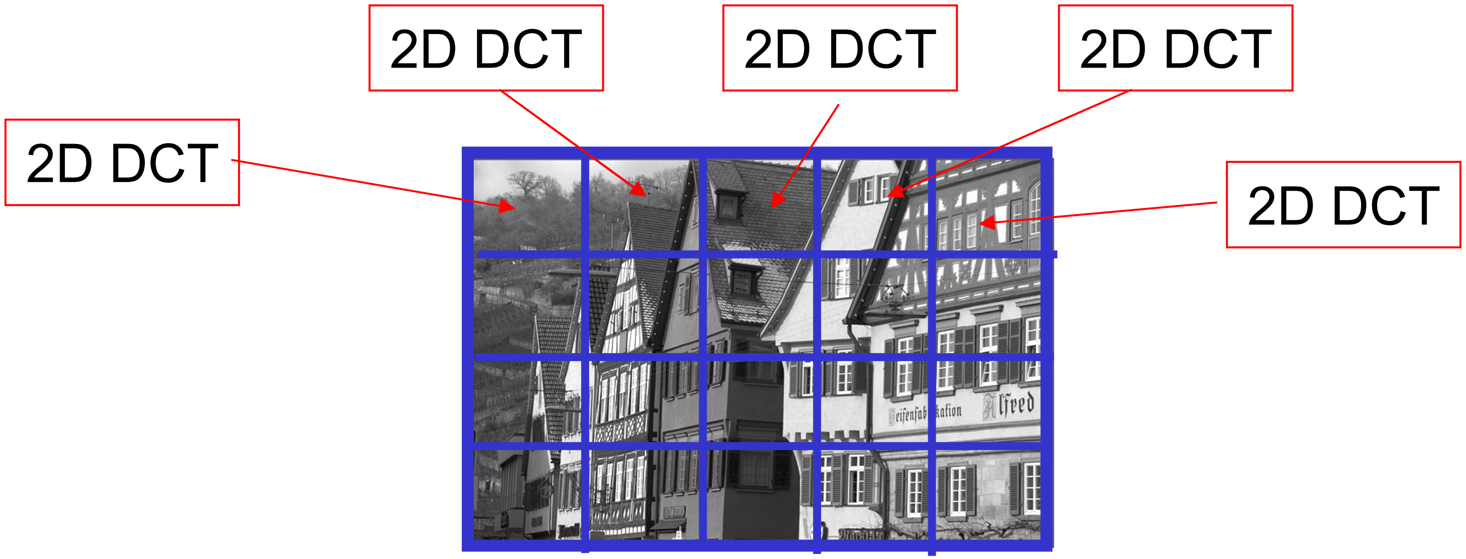
$$c_{mn} = \sum_{i=0}^{N-1} f_{mi} y_{in} \qquad for\ m = 0,\ldots,N-1.$$

What is the corresponding matrix-based representation?

If F is an orthogonal transform, the resulting 2D transform is also orthogonal and satisfies the Parseval's theorem. (Try to prove this)

Transforms

- When the image size is larger than NxN pixels:
  - First partition the image into blocks of NxN size.
  - Then for each block apply the 2D separable transform.
- Example:
  - Usually in image compression 8-point DCT is used. Therefore, 8x8 blocks and 8-point 2D DCT has been used in the JPEG standard.



Transforms - Summary

- What is a transform?
- The uses of transforms?
- The terminology
- Invertible transforms
- Orthogonal transforms
- The Parseval's Theorem and its usage
- How to compute transform coefficients?
- Signal reconstruction using transform coefficients
- The DCT
- The use of transforms on images

- Now you can attempt
  - Q1. of problem set 1
  - Q2 of 2007/08 exam