Queen Mary
University of London

EBU6018
Advanced Transform Methods

Week 3.4 – Transform Matrices

Dr Yixuan Zou

Lecture Outline

1. Transform Matrices with Examples
    1. Discrete Fourier Transform
    2. Discrete Cosine Transform
    3. Discrete Wavelet Transform
2. Comparing DFT, DCT, DWT

Discrete Fourier Transform

- The Continuous Fourier Transform is defined as:

$$X(\omega) = \int_{t=-\infty}^{t=\infty} e^{-j\omega t} \cdot x(t)dt$$

However ANY continuous transform is not practical (infinite number of values across infinite time) so transforms need to be implemented discretely.

- The Discrete Fourier Transform (DFT) is defined as:

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j\frac{2\pi}{N}nk} = X(\omega)\Big|_{\omega=\frac{2\pi}{N}k}$$

for $k = 0, 1, \ldots, N$-1

Discrete Fourier Transform

- This can be written as:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}nk} = \sum_{n=0}^{N-1} x[n]\, W_N^{nk}$$

where $W_N = e^{-\frac{j2\pi}{N}}$

- The matrix formula for an N-point DFT can be written as $X \; = F_N x$
  - where x is the N-point input sequence of samples of a continuous signal
  - F_N is the N-by-N DFT matrix
  - X is the DFT of the signal

Queen Mary University of London

Discrete Fourier Transform - Matrix

- We can produce an N-by-N Fourier Matrix where n are input samples and k are output frequencies (n=k):

n

k

$$\boxed{F_N[k, n] = {W_N}^{nk}}$$

DFT:

- $X[k] = \sum_{n=0}^{N-1} x[n]\, W_N^{nk}$
- $F_N[k, n] = {W_N}^{nk}$
- $W_N = e^{-\frac{j2\pi}{N}}$
- $X = F_N x$

Queen Mary University of London

Discrete Fourier Transform - Matrix

*Count from 0

- The N-by-N Fourier Matrix (N = 0…..(n-1)):

$$F_N = \begin{bmatrix} 1 & 1 & 1 \ldots & 1 \\ 1 & \omega & \omega^2 \ldots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 \ldots & \omega^{2(N-1)} \\ \cdots & \cdots & \cdots & \cdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} \ \ldots & \omega^{(N-1)^2} \end{bmatrix}$$

DFT:

- $W_N = e^{-\frac{j2\pi}{N}} = \omega$
- $F_N[k,n] = W_N{}^{nk} = \omega^{nk}$

Discrete Fourier Transform – 2x2 Matrix

- Now, consider $\omega = e^{-\frac{j2\pi}{N}} = [\cos\left(\frac{2\pi}{N}\right) - \mathrm{j}\sin(\frac{2\pi}{N})]$ $so\ \omega^{N} = e^{-j2\pi} = 1$

- All the entries of F_N are on the unit circle in the complex plane and raising each one to the N^{th} power gives 1.

- For example, if N = 2, $e^{-j2\pi/2} = \cos(\pi) - \mathrm{j}\sin(\pi)$ = -1

- Then $F_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$, normalised.

Queen Mary University of London

Discrete Fourier Transform – 4x4 Matrix

- For N = 4 (N = 0…..3), $\omega = e^{-j2\pi/4} = [\cos\left(\frac{\pi}{2}\right) - i\sin(\frac{\pi}{2})] = -\mathrm{i}$
- So the normalised 4x4 Fourier Matrix is:
- $F_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & (-i)^2 & (-i)^3 \\ 1 & (-i)^2 & (-i)^4 & (-i)^6 \\ 1 & (-i)^3 & (-i)^6 & (-i)^9 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix}$
- Each row corresponds to an increasing frequency.

Discrete Fourier Transform – Example

- Calculate the DFT of the input sequence $s[n] = [2, 3, 1, 4]$

- Answer:

Discrete Fourier Transform – Example

- Calculate the DFT of the input sequence s[n] = [2, 3, 1, 4]

- Answer:

$$\text{DFT} = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{bmatrix}\begin{bmatrix} 2 \\ 3 \\ 1 \\ 4 \end{bmatrix} = \begin{bmatrix} 5 \\ 0.5 + 0.5i \\ -2 \\ 0.5 - 0.5i \end{bmatrix}$$

Discrete Fourier Transform – Summary

- Each row of the Fourier Matrix corresponds to a cosine wave and a sine wave of the same frequency.
- Even if the input data is a sequence of real numbers, the output sequence will be complex numbers
- Multiplying an input sequence by the Fourier Matrix gives an output which is the correlation between the input data and a series of cosine and sine waves of increasing frequency.
- This is equivalent to what a Fourier Series does for a periodic input waveform.
- The 2x2 and 4x4 Fourier Matrices are relatively trivial. The 8x8 is less trivial and will not be considered here.

Discrete Cosine Transform

- If the input sequence applied to a DFT 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.
- As with the DFT, we assume that the input sequence is periodic in order to obtain an accurate Fourier Transform.
- For the DCT, the sequence is assumed to be even and periodic.

Discrete Cosine Transform

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

$$DCT[k] = \langle s, \psi_k \rangle \qquad c(k) = \begin{cases} \sqrt{1/N} & k = 0 \\ \sqrt{2/N} & k \neq 0 \end{cases}$$

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

Discrete Cosine Transform

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

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

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

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

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

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

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

$$DCT[2] = \sqrt{\frac{2}{N}}\sum_{n=0}^{3} s[n]\cos\frac{\pi(2n+1)}{4} \qquad DCT[3] = \sqrt{\frac{2}{N}}\sum_{n=0}^{3} s[n]\cos\frac{\pi(2n+1)3}{8}$$

Discrete Cosine Transform – Example

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 $s[n] = [2, 3, 1, 4]$

Discrete Cosine Transform – Example Solution

- Construct the 4x4 DCT matrix:

$$\Psi = \begin{bmatrix} 1/2 & & & \\ \frac{cos(\frac{\pi}{8})}{\sqrt{2}} & & & \\ & & & \\ & & & \end{bmatrix}$$

Discrete Cosine Transform – Example Solution

- Construct the 4x4 DCT matrix:

$$\Psi = \begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.50 & -0.50 & -0.50 & 0.50 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix}$$

Discrete Cosine Transform – Example Solution

- Multiply the DCT matrix with the input sequence:

$$\text{DCT}=\begin{bmatrix}0.50 & 0.50 & 0.50 & 0.50\\ 0.65 & 0.27 & -0.27 & -0.65\\ 0.50 & -0.50 & -0.50 & 0.50\\ 0.27 & -0.65 & 0.65 & -0.27\end{bmatrix}\begin{bmatrix}2\\3\\1\\4\end{bmatrix}=\begin{bmatrix}5.00\\-2.10\\1.00\\-1.84\end{bmatrix}$$

Discrete Cosine Transform – DCT application

- The DCT is used to perform image compression to produce jpeg format. Note: the DCT is lossless, it is the output that is compressed.
- For this format, 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.

Discrete Cosine Transform – Basis Functions

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

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

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

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

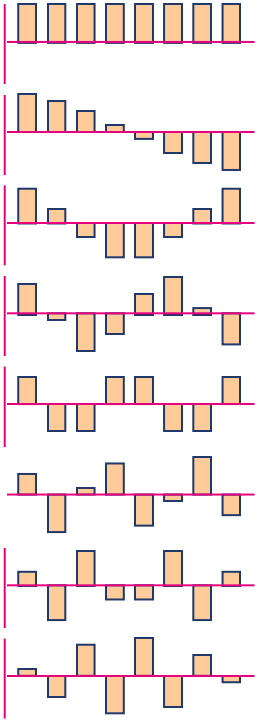
$$\frac{1}{2}\sum_{x=0}^{7} s(x)\cos\left(\frac{2x+1}{16}*3\pi\right)$$

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

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

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

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



$S(0)$ DC

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

Discrete Cosine Transform – 8x8 DCT Matrix

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

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

Discrete Cosine Transform – 8x8 DCT 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).

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

Discrete Cosine Transform – 8x8 DCT Matrix 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:

Discrete Cosine Transform – 8x8 DCT Matrix Example 1

$$\text{DCT} = \frac{1}{2}\underbrace{\begin{bmatrix} .71 & .71 & .71 & .71 & .71 & .71 & .71 & .71 \\ .98 & .83 & .56 & .19 & -.19 & -.56 & -.83 & -.98 \\ .92 & .38 & -.38 & -.92 & -.92 & -.38 & .38 & .92 \\ .83 & -.19 & -.98 & -.56 & .56 & .98 & .19 & -.83 \\ .71 & -.71 & -.71 & .71 & .71 & -.71 & -.71 & .71 \\ .56 & -.98 & .19 & .83 & -.83 & -.19 & .98 & -.56 \\ .38 & -.92 & .92 & -.38 & -.38 & .92 & -.92 & .38 \\ .19 & -.56 & .83 & -.98 & .98 & -.83 & .56 & -.19 \end{bmatrix}}_{\text{8x8 DCT matrix}} \underbrace{\begin{bmatrix} 2.1 \\ 2.0 \\ 2.11 \\ 1.99 \\ 1.98 \\ 2.02 \\ 2.08 \\ 1.98 \end{bmatrix}}_{\text{8x1 input sequence}}$$

Queen Mary University of London

Discrete Cosine Transform – 8x8 DCT Matrix Example 1

Transposing back to a row, the output sequence is:

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

- Interpret the output:
  - 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.

Discrete Cosine Transform – 8x8 DCT Matrix 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:

Discrete Cosine Transform – 8x8 DCT Matrix Example 2

$$\text{DCT} = \frac{1}{2}\begin{bmatrix} .71 & .71 & .71 & .71 & .71 & .71 & .71 & .71 \\ .98 & .83 & .56 & .19 & -.19 & -.56 & -.83 & -.98 \\ .92 & .38 & -.38 & -.92 & -.92 & -.38 & .38 & .92 \\ .83 & -.19 & -.98 & -.56 & .56 & .98 & .19 & -.83 \\ .71 & -.71 & -.71 & .71 & .71 & -.71 & -.71 & .71 \\ .56 & -.98 & .19 & .83 & -.83 & -.19 & .98 & -.56 \\ .38 & -.92 & .92 & -.38 & -.38 & .92 & -.92 & .38 \\ .19 & -.56 & .83 & -.98 & .98 & -.83 & .56 & -.19 \end{bmatrix}\begin{bmatrix} 2.1 \\ 9.6 \\ -11.2 \\ 7.9 \\ -10.1 \\ 8.6 \\ -6.7 \\ 8.3 \end{bmatrix}$$

Discrete Cosine Transform – 8x8 DCT Matrix Example 2

Transposing back to a row, the output sequence is:

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

- To interpret the output:
  - 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.

Discrete Cosine Transform – Summary

- The Discrete Cosine Transform correlates an input sequence of real samples of an input signal with a set of cosine waveforms of increasing frequency.

- The output is a set of real values.

- When applied to images, it transforms from the spatial domain to a rate-of-change domain.

- It is used in the production of jpeg image compression format.

Discrete Wavelet Transform

- Wavelets are a class of functions which are of short duration and are oscillatory.
- They are used for perform time-frequency analysis.
- They are of the form:

$$\psi(t) \to \psi\left(\frac{t-b}{a}\right)$$

They are used as basis functions by translation (b) and scaling (a)

$$CWT(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} s(t)\psi^{*}\left(\frac{t-b}{a}\right)dt$$

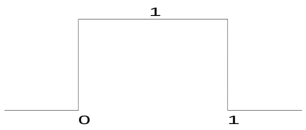
$$= \int_{-\infty}^{\infty} s(t)\psi_{a,b}{}^{*}(t)dt = < s, \psi_{a,b} >$$

As with any transform,
they are implemented discretely.

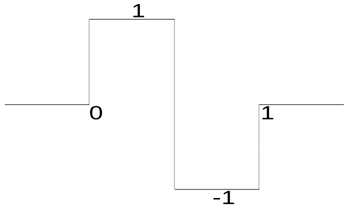
Discrete Wavelet Transform – Haar Function

- The simple Haar Functions will be used to illustrate a discrete implementation of Wavelet Transforms.
- Haar Functions are a set of scaled and translated Haar Scaling Functions and Haar Wavelet Functions:

Scaling function



Wavelet function



$$\varphi_{00} = \begin{bmatrix} 1 & 1 \end{bmatrix}$$

$$\psi_{00} = \begin{bmatrix} 1 & -1 \end{bmatrix}$$

These are orthogonal $$< \varphi_{00}, \psi_{00} > = \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = 0$$

Discrete Wavelet Transform – Haar Matrix

These two functions can be written in matrix form:

$$H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

The normalised Haar Matrix is:

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

Discrete Wavelet Transform – Matrix Normalization

Replace each row of the matrix by its normalised values.

For example $H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$

Row 1 norm $\sqrt{4}$, row 2 norm $\sqrt{4}$, row 3 norm $\sqrt{2}$, row 4 norm $\sqrt{2}$

Normalised $H_4 = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & -0.5 & -0.5 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$

Discrete Wavelet Transform – 4x4 Haar Matrix Example

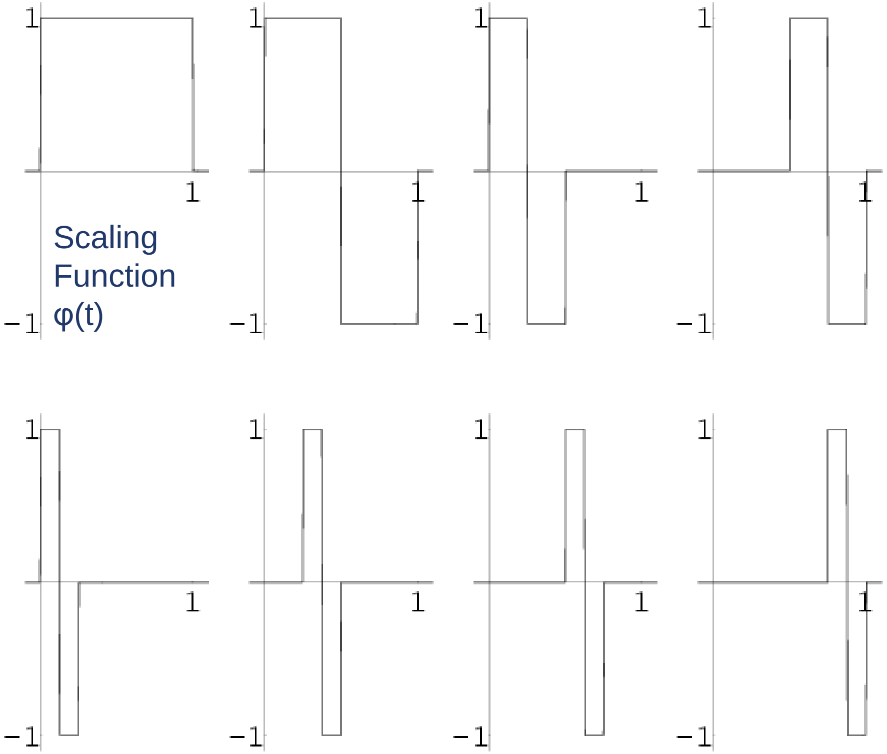
Apply the Haar Transform to the 4-point input sequence:

$$S[n] = [2, 5 -3, 7]$$

Discrete Wavelet Transform – 4x4 Haar Matrix Example

$$\frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}\begin{bmatrix} 2 \\ 5 \\ -3 \\ 7 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 11 \\ 3 \\ (2-5)\sqrt{2} \\ (-3-7)\sqrt{2} \end{bmatrix} = \begin{bmatrix} \frac{11}{2} \\ \frac{3}{2} \\ \frac{-3}{\sqrt{2}} \\ \frac{-10}{\sqrt{2}} \end{bmatrix}$$

Discrete Wavelet Transform – Haar Functions



Wavelet Function:

$$\psi(x) \equiv \begin{cases} 1 & 0 \le x \le \frac{1}{2} \\ -1 & \frac{1}{2} \le x \le 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\psi_{jk}(x) \equiv \psi(2^j x - k),$$

$\phi_{00} = \phi\,(x)$

$\psi_{00} = \psi\,(x)$

$\psi_{10} = \psi\,(2x)$

$\psi_{11} = \psi\,(2x\text{-}1)$

$\psi_{20} = \psi\,(4x)$

$\psi_{21} = \psi\,(4x\text{-}1)$

$\psi_{22} = \psi(4x\text{-}2)$

$\psi_{23} = \psi(4x\text{-}3)$

Discrete Wavelet Transform – 8x8 Haar Matrix

The un-normalised 8x8 Haar Matrix can be used to show how a Haar Matrix is derived:

$$H_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \begin{array}{c} \varphi_0(t) \\ \psi_0(t) \\ \psi_{1,0}(t) \\ \psi_{1,1}(t) \\ \psi_{2,0}(t) \\ \psi_{2,1}(t) \\ \psi_{2,2}(t) \\ \psi_{2,3}(t) \end{array}$$

The first row gives the smoothed value of the input sequence.
Then subsequent rows correspond to increasing frequencies (similar to DCT).
Translation of each frequency allows time information to be determined also.

The matrix would need to be normalised before it could be applied directly to a transform.

Discrete Wavelet Transform – 8x8 Haar Matrix

Replace each row by its normalised value:

$$H_8 = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix}$$

Discrete Wavelet Transform – Summary

- We have seen that a Haar Matrix can be constructed to perform Haar Transforms directly.
- The Haar Transform is fast because the matrix contains many zero terms and is real.
- It can be used to identify frequency components in the non-stationary input signal to be analysed and the times at which they occur (feature extraction of short-duration artifacts).
- It can be used to identify trends in the input data.
- It can be used for compression by reducing or eliminating the coefficients corresponding to high frequencies in the signal and then inverting the transform.

Transform Methods - Application

Transforms are used for a variety of purposes. For example:

- **Fourier Transforms** are used to obtain the frequency spectrum of a function of time.
- **Cosine Transforms** are used to identify the rate-of-change of the input data.
- **Wavelet Transforms** are used to identify trends in the input data (“approximations”) and to identify short-duration features or artifacts in the input data.

Queen Mary
University of London