



Discrete Haar Wavelet Transforms

Patrick J. Van Fleet

Center for Applied Mathematics
University of St. Thomas
St. Paul, MN USA

PREP - Wavelet Workshop, 2006



UNIVERSITY of ST. THOMAS

Outline

Today's Schedule

Building the Haar Matrix

Putting Two Filters Together

Why the Word Wavelet?

Examples

Coding the Haar Transform

Implementing W_N

Implementing W_N^T

2D Haar Transform

Building the 2D Transform

Coding the 2D Transform

Iterating

In the Classroom

Teaching Ideas



UNIVERSITY of ST. THOMAS

Today's Schedule

9:00-10:15 **Lecture One:** Why Wavelets?

10:15-10:30 **Coffee Break** (OSS 235)

10:30-11:45 **Lecture Two:** Digital Images, Measures, and Huffman Codes

12:00-1:00 **Lunch** (Cafeteria)

1:30-2:45 **Lecture Three:** Fourier Series, Convolution and Filters

2:45-3:00 **Coffee Break** (OSS 235)

3:00-4:15 \Rightarrow **Lecture Four:** 1D and 2D Haar Transforms

5:30-6:30 **Dinner** (Cafeteria)





Building the Haar Matrix

Putting Two Filters Together

- ▶ Consider again the filter $\mathbf{h} = (h_0, h_1) = (\frac{1}{2}, \frac{1}{2})$.
- ▶ If we compute $\mathbf{y} = \mathbf{h} * \mathbf{x}$, we obtain the components

$$y_n = \frac{1}{2}x_n + \frac{1}{2}x_{n-1}$$

- ▶ We could write down the convolution matrix





Building the Haar Matrix

Putting Two Filters Together

- ▶ Consider again the filter $\mathbf{h} = (h_0, h_1) = (\frac{1}{2}, \frac{1}{2})$.
- ▶ If we compute $\mathbf{y} = \mathbf{h} * \mathbf{x}$, we obtain the components

$$y_n = \frac{1}{2}x_n + \frac{1}{2}x_{n-1}$$

- ▶ We could write down the convolution matrix





Building the Haar Matrix

Putting Two Filters Together

- ▶ Consider again the filter $\mathbf{h} = (h_0, h_1) = (\frac{1}{2}, \frac{1}{2})$.
- ▶ If we compute $\mathbf{y} = \mathbf{h} * \mathbf{x}$, we obtain the components

$$y_n = \frac{1}{2}x_n + \frac{1}{2}x_{n-1}$$

- ▶ We could write down the convolution matrix





Building the Haar Matrix

Putting Two Filters Together

- ▶ Consider again the filter $\mathbf{h} = (h_0, h_1) = (\frac{1}{2}, \frac{1}{2})$.
- ▶ If we compute $\mathbf{y} = \mathbf{h} * \mathbf{x}$, we obtain the components

$$y_n = \frac{1}{2}x_n + \frac{1}{2}x_{n-1}$$

- ▶ We could write down the convolution matrix





Putting Two Filters Together

Building the Haar Matrix

Putting Two Filters Together

$$H = \begin{bmatrix} \ddots & & & & & & & & & & & & & \\ & \ddots & & & & & & & & & & & & \\ & & \ddots & & & & & & & & & & & \\ & & & \ddots & & & & & & & & & & \\ & & & & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\ & & & & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & \\ \dots & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & \dots \\ & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & \\ & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & \\ & & & & & & & & \ddots & \ddots & & & \ddots \end{bmatrix}$$





Building the Haar Matrix

Putting Two Filters Together

- ▶ ... but we can't invert the process.
- ▶ What would we need to be able to invert the process?
- ▶ We have averages of consecutive numbers - if we had the directed distance between these averages and the consecutive numbers, then we could invert.
- ▶ The directed distance is exactly the sequence \mathbf{x} convolved with the filter $\mathbf{g} = (\frac{1}{2}, -\frac{1}{2})$.





Building the Haar Matrix

Putting Two Filters Together

- ▶ ... but we can't invert the process.
- ▶ What would we need to be able to invert the process?
- ▶ We have averages of consecutive numbers - if we had the directed distance between these averages and the consecutive numbers, then we could invert.
- ▶ The directed distance is exactly the sequence \mathbf{x} convolved with the filter $\mathbf{g} = (\frac{1}{2}, -\frac{1}{2})$.





Building the Haar Matrix

Putting Two Filters Together

- ▶ ... but we can't invert the process.
- ▶ What would we need to be able to invert the process?
- ▶ We have averages of consecutive numbers - if we had the directed distance between these averages and the consecutive numbers, then we could invert.
- ▶ The directed distance is exactly the sequence \mathbf{x} convolved with the filter $\mathbf{g} = (\frac{1}{2}, -\frac{1}{2})$.





Building the Haar Matrix

Putting Two Filters Together

- ▶ ... but we can't invert the process.
- ▶ What would we need to be able to invert the process?
- ▶ We have averages of consecutive numbers - if we had the directed distance between these averages and the consecutive numbers, then we could invert.
- ▶ The directed distance is exactly the sequence \mathbf{x} convolved with the filter $\mathbf{g} = (\frac{1}{2}, -\frac{1}{2})$.





Building the Haar Matrix

Putting Two Filters Together

- ▶ ... but we can't invert the process.
- ▶ What would we need to be able to invert the process?
- ▶ We have averages of consecutive numbers - if we had the directed distance between these averages and the consecutive numbers, then we could invert.
- ▶ The directed distance is exactly the sequence \mathbf{x} convolved with the filter $\mathbf{g} = (\frac{1}{2}, -\frac{1}{2})$.





Building the Haar Matrix

Putting Two Filters Together

Indeed if

$$y_n = \frac{1}{2}x_n + \frac{1}{2}x_{n-1} \quad \text{and} \quad z_n = \frac{1}{2}x_n - \frac{1}{2}x_{n-1}$$

then

$$x_n = y_n + z_n \quad \text{and} \quad x_{n-1} = y_n - z_n$$





Building the Haar Matrix

Putting Two Filters Together

Perhaps we could invert the process if we used both filters. We know that G is





Putting Two Filters Together

Building the Haar Matrix

Putting Two Filters Together

$$G = \begin{bmatrix} \ddots & & & & & & & & & & & & & \\ & \ddots & & & & & & & & & & & & \\ & & \ddots & & & & & & & & & & & \\ & & & \ddots & & & & & & & & & & \\ & & & & \ddots & & & & & & & & & \\ & & & & & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ \dots & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ & & & & & & & & & \ddots & \ddots & & \ddots \end{bmatrix}$$





Putting Two Filters Together

Building the Haar Matrix

Putting Two Filters Together

So that

$$\begin{bmatrix} H \\ - \\ G \end{bmatrix} \cdot \mathbf{x} = \begin{bmatrix} \mathbf{y} \\ - \\ \mathbf{z} \end{bmatrix}$$





Building the Haar Matrix

Putting Two Filters Together

If we think about inverting, we can write down:





Putting Two Filters Together

Building the Haar Matrix

Putting Two Filters Together

$$\begin{array}{rcl}
 \vdots & & \vdots \\
 y_1 - z_1 & = & \frac{x_1 + x_0}{2} - \frac{x_1 - x_0}{2} = x_0 \\
 y_1 + z_1 & = & \frac{x_1 + x_0}{2} + \frac{x_1 - x_0}{2} = x_1 \\
 y_2 - z_2 & = & \frac{x_2 + x_1}{2} - \frac{x_2 - x_1}{2} = x_1 \\
 y_2 + z_2 & = & \frac{x_2 + x_1}{2} + \frac{x_2 - x_1}{2} = x_2 \\
 y_3 - z_3 & = & \frac{x_3 + x_2}{2} - \frac{x_3 - x_2}{2} = x_2 \\
 y_3 + z_3 & = & \frac{x_3 + x_2}{2} + \frac{x_3 - x_2}{2} = x_3 \\
 \vdots & & \vdots
 \end{array}$$





Building the Haar Matrix

Putting Two Filters Together

But there is some redundancy here - we do not need all the values of y_n , z_n to recover x_n :





Putting Two Filters Together

Building the Haar Matrix

Putting Two Filters Together

$$\begin{array}{rcl}
 \vdots & & \vdots \\
 y_1 - z_1 & = & \frac{x_1 + x_0}{2} - \frac{x_1 - x_0}{2} = x_0 \\
 y_1 + z_1 & = & \frac{x_1 + x_0}{2} + \frac{x_1 - x_0}{2} = x_1 \\
 y_2 - z_2 & = & \frac{x_2 + x_1}{2} - \frac{x_2 - x_1}{2} = x_1 \\
 y_2 + z_2 & = & \frac{x_2 + x_1}{2} + \frac{x_2 - x_1}{2} = x_2 \\
 y_3 - z_3 & = & \frac{x_3 + x_2}{2} - \frac{x_3 - x_2}{2} = x_2 \\
 y_3 + z_3 & = & \frac{x_3 + x_2}{2} + \frac{x_3 - x_2}{2} = x_3 \\
 \vdots & & \vdots
 \end{array}$$





Building the Haar Matrix

Putting Two Filters Together

- ▶ So we can omit every other row in H , G and still produce enough to be able to recover \mathbf{x}
- ▶ This is called *downsampling*.
- ▶ We are also now in a position to **truncate** our matrix. Indeed, if $\mathbf{x} = (x_0, \dots, x_N)$, then it is natural to truncate the matrix and write:





Building the Haar Matrix

Putting Two Filters Together

- ▶ So we can omit every other row in H , G and still produce enough to be able to recover \mathbf{x}
- ▶ This is called *downsampling*.
- ▶ We are also now in a position to **truncate** our matrix. Indeed, if $\mathbf{x} = (x_0, \dots, x_N)$, then it is natural to truncate the matrix and write:





Building the Haar Matrix

Putting Two Filters Together

- ▶ So we can omit every other row in H , G and still produce enough to be able to recover \mathbf{x}
- ▶ This is called *downsampling*.
- ▶ We are also now in a position to **truncate** our matrix. Indeed, if $\mathbf{x} = (x_0, \dots, x_N)$, then it is natural to truncate the matrix and write:





Building the Haar Matrix

Putting Two Filters Together

- ▶ So we can omit every other row in H , G and still produce enough to be able to recover \mathbf{x}
- ▶ This is called *downsampling*.
- ▶ We are also now in a position to **truncate** our matrix. Indeed, if $\mathbf{x} = (x_0, \dots, x_N)$, then it is natural to truncate the matrix and write:





Putting Two Filters Together

Building the Haar Matrix

Putting Two Filters Together

$$\tilde{W}_N = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & \dots & 0 & 0 \\ \vdots & & & & \ddots & & & & \vdots \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \hline -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & & 0 & 0 \\ 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & \dots & 0 & 0 \\ \vdots & & & & \ddots & & & & \vdots \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & -\frac{1}{2} & \frac{1}{2} \end{bmatrix}$$





Building the Haar Matrix

Putting Two Filters Together

This matrix is easy to invert if we remember the formulas:

$$x_n = y_n + z_n \quad \text{and} \quad x_{n-1} = y_n - z_n$$

We have:





Putting Two Filters Together

Building the Haar Matrix

Putting Two Filters Together

$$\tilde{W}_N^{-1} = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ \vdots & & \ddots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array} \right]$$





Building the Haar Matrix

Putting Two Filters Together

- ▶ Note we are very close to having \tilde{W}_N an orthogonal matrix.
- ▶ We have $\tilde{W}_N^T = \frac{1}{2} \tilde{W}_N^{-1}$.
- ▶ If we multiply \tilde{W}_N by $\sqrt{2}$, we will obtain an orthogonal matrix.
- ▶ We have:





Building the Haar Matrix

Putting Two Filters Together

- ▶ Note we are very close to having \tilde{W}_N an orthogonal matrix.
- ▶ We have $\tilde{W}_N^T = \frac{1}{2} \tilde{W}_N^{-1}$.
- ▶ If we multiply \tilde{W}_N by $\sqrt{2}$, we will obtain an orthogonal matrix.
- ▶ We have:





Building the Haar Matrix

Putting Two Filters Together

- ▶ Note we are very close to having \tilde{W}_N an orthogonal matrix.
- ▶ We have $\tilde{W}_N^T = \frac{1}{2} \tilde{W}_N^{-1}$.
- ▶ If we multiply \tilde{W}_N by $\sqrt{2}$, we will obtain an orthogonal matrix.
- ▶ We have:





Building the Haar Matrix

Putting Two Filters Together

- ▶ Note we are very close to having \tilde{W}_N an orthogonal matrix.
- ▶ We have $\tilde{W}_N^T = \frac{1}{2} \tilde{W}_N^{-1}$.
- ▶ If we multiply \tilde{W}_N by $\sqrt{2}$, we will obtain an orthogonal matrix.
- ▶ We have:





Building the Haar Matrix

Putting Two Filters Together

- ▶ Note we are very close to having \tilde{W}_N an orthogonal matrix.
- ▶ We have $\tilde{W}_N^T = \frac{1}{2} \tilde{W}_N^{-1}$.
- ▶ If we multiply \tilde{W}_N by $\sqrt{2}$, we will obtain an orthogonal matrix.
- ▶ We have:





Putting Two Filters Together

Building the Haar Matrix

Putting Two Filters Together

$$W_N = \begin{bmatrix} H \\ - \\ G \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 & 0 & & 0 & 0 \\ 0 & 0 & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & & 0 & 0 \\ \vdots & & & & \ddots & & \vdots \\ 0 & 0 & 0 & 0 & \dots & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \hline -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 & 0 & & 0 & 0 \\ 0 & 0 & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & & 0 & 0 \\ \vdots & & & & \ddots & & \vdots \\ 0 & 0 & 0 & 0 & \dots & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \end{bmatrix}$$





Building the Haar Matrix

Putting Two Filters Together

- ▶ W_N is called the **Discrete Haar Wavelet Transform**
- ▶ The filter

$$\mathbf{h} = \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2} \right)$$

is called the **Haar filter**.

- ▶ Note that $H(\omega) = \frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2} e^{j\omega}$ satisfies $H(\pi) = 0$, but $H(0) = \frac{\sqrt{2}}{2}$. We will still consider this to be a lowpass filter—the $\frac{\sqrt{2}}{2}$ resulted when we made the transform orthogonal.





Building the Haar Matrix

Putting Two Filters Together

- ▶ W_N is called the **Discrete Haar Wavelet Transform**
- ▶ The filter

$$\mathbf{h} = \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2} \right)$$

is called the **Haar filter**.

- ▶ Note that $H(\omega) = \frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2} e^{j\omega}$ satisfies $H(\pi) = 0$, but $H(0) = \frac{\sqrt{2}}{2}$. We will still consider this to be a lowpass filter—the $\frac{\sqrt{2}}{2}$ resulted when we made the transform orthogonal.





Building the Haar Matrix

Putting Two Filters Together

- ▶ W_N is called the **Discrete Haar Wavelet Transform**
- ▶ The filter

$$\mathbf{h} = \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2} \right)$$

is called the **Haar filter**.

- ▶ Note that $H(\omega) = \frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2} e^{j\omega}$ satisfies $H(\pi) = 0$, but $H(0) = \frac{\sqrt{2}}{2}$. We will still consider this to be a lowpass filter—the $\frac{\sqrt{2}}{2}$ resulted when we made the transform orthogonal.





Building the Haar Matrix

Putting Two Filters Together

- ▶ W_N is called the **Discrete Haar Wavelet Transform**
- ▶ The filter

$$\mathbf{h} = \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2} \right)$$

is called the **Haar filter**.

- ▶ Note that $H(\omega) = \frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2} e^{j\omega}$ satisfies $H(\pi) = 0$, but $H(0) = \frac{\sqrt{2}}{2}$. We will still consider this to be a lowpass filter—the $\frac{\sqrt{2}}{2}$ resulted when we made the transform orthogonal.





Building the Haar Matrix

Why the Word Wavelet?

- ▶ The word **wavelet** comes from the more classical treatment of the topic. Here, we work in $L^2(\mathbb{R})$ and *downsampling* is basically a way to move between nested subspaces V_j that are generated by the translates and dilates of a single **scaling function**.
- ▶ If V_0 is the space of piecewise constants with possible breaks at \mathbb{Z} , then the characteristic function $\phi(t) = \chi_{[0,1)}(t)$ and its translates form an orthonormal basis for V_0 .





Building the Haar Matrix

Why the Word Wavelet?

- ▶ The word **wavelet** comes from the more classical treatment of the topic. Here, we work in $L^2(\mathbb{R})$ and *downsampling* is basically a way to move between nested subspaces V_j that are generated by the translates and dilates of a single **scaling function**.
- ▶ If V_0 is the space of piecewise constants with possible breaks at \mathbb{Z} , then the characteristic function $\phi(t) = \chi_{[0,1)}(t)$ and its translates form an orthonormal basis for V_0 .





Building the Haar Matrix

Why the Word Wavelet?

- ▶ The word **wavelet** comes from the more classical treatment of the topic. Here, we work in $L^2(\mathbb{R})$ and *downsampling* is basically a way to move between nested subspaces V_j that are generated by the translates and dilates of a single **scaling function**.
- ▶ If V_0 is the space of piecewise constants with possible breaks at \mathbb{Z} , then the characteristic function $\phi(t) = \chi_{[0,1)}(t)$ and its translates form an orthonormal basis for V_0 .





Why the Word Wavelet?

Building the Haar Matrix

Why the Word Wavelet?



UNIVERSITY of ST. THOMAS



Wednesday, 7 June, 2006

Lecture 4

Discrete Haar Wavelet Transforms



Building the Haar Matrix

Why the Word Wavelet?

If V_1 is the space of piecewise constants with possible breakpoints at $\frac{1}{2}\mathbb{Z}$, then $V_0 \subset V_1$, and the functions $\sqrt{2}\phi(2t - k)$ form an orthonormal basis for V_1 .





Building the Haar Matrix

Why the Word Wavelet?

Note that

$$\phi(t) = \sqrt{2} \left(\frac{\sqrt{2}}{2} \phi(2t) + \frac{\sqrt{2}}{2} \phi(2t - 1) \right)$$

is called a **dilation equation**.





Building the Haar Matrix

Why the Word Wavelet?

- ▶ We can get the Haar filter coefficients from the dilation equation.
- ▶ The word **wavelet** refers to the function $\psi(t)$ that generates a basis for the orthogonal complement of V_0 in V_1 .
- ▶ In this case, the wavelet function is

$$\psi(t) = \begin{cases} 1 & 0 \leq t < \frac{1}{2} \\ -1 & \frac{1}{2} \leq t < 1 \end{cases}$$





Building the Haar Matrix

Why the Word Wavelet?

- ▶ We can get the Haar filter coefficients from the dilation equation.
- ▶ The word **wavelet** refers to the function $\psi(t)$ that generates a basis for the orthogonal complement of V_0 in V_1 .
- ▶ In this case, the wavelet function is

$$\psi(t) = \begin{cases} 1 & 0 \leq t < \frac{1}{2} \\ -1 & \frac{1}{2} \leq t < 1 \end{cases}$$





Why the Word Wavelet?

Building the Haar Matrix

Why the Word Wavelet?

- ▶ We can get the Haar filter coefficients from the dilation equation.
- ▶ The word **wavelet** refers to the function $\psi(t)$ that generates a basis for the orthogonal complement of V_0 in V_1 .
- ▶ In this case, the wavelet function is

$$\psi(t) = \begin{cases} 1 & 0 \leq t < \frac{1}{2} \\ -1 & \frac{1}{2} \leq t < 1 \end{cases}$$





Building the Haar Matrix

Why the Word Wavelet?

- ▶ We can get the Haar filter coefficients from the dilation equation.
- ▶ The word **wavelet** refers to the function $\psi(t)$ that generates a basis for the orthogonal complement of V_0 in V_1 .
- ▶ In this case, the wavelet function is

$$\psi(t) = \begin{cases} 1 & 0 \leq t < \frac{1}{2} \\ -1 & \frac{1}{2} \leq t < 1 \end{cases}$$





Building the Haar Matrix

Why the Word Wavelet?

Note that $\psi(t) \in V_1$ and

$$\psi(t) = \phi(t) = \sqrt{2} \left(\frac{\sqrt{2}}{2} \phi(2t) - \frac{\sqrt{2}}{2} \phi(2t-1) \right)$$

so that the highpass filter $\mathbf{g} = \left(\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2} \right)$ can be read from this dilation equation.





Why the Word Wavelet?

Building the Haar Matrix

Why the Word Wavelet?

- ▶ I opted to stay away from the classical approach to wavelets.
- ▶ It is beautiful theory, but too much for sophomores and juniors.
- ▶ I believe it's better to give them a practical introduction to Fourier series and convolution, and then derive the the discrete wavelet transform by using a lowpass/highpass filter pair and downsampling.





Building the Haar Matrix

Why the Word Wavelet?

- ▶ I opted to stay away from the classical approach to wavelets.
- ▶ It is beautiful theory, but too much for sophomores and juniors.
- ▶ I believe it's better to give them a practical introduction to Fourier series and convolution, and then derive the the discrete wavelet transform by using a lowpass/highpass filter pair and downsampling.





Building the Haar Matrix

Why the Word Wavelet?

- ▶ I opted to stay away from the classical approach to wavelets.
- ▶ It is beautiful theory, but too much for sophomores and juniors.
- ▶ I believe it's better to give them a practical introduction to Fourier series and convolution, and then derive the the discrete wavelet transform by using a lowpass/highpass filter pair and downsampling.





Building the Haar Matrix

Why the Word Wavelet?

- ▶ I opted to stay away from the classical approach to wavelets.
- ▶ It is beautiful theory, but too much for sophomores and juniors.
- ▶ I believe it's better to give them a practical introduction to Fourier series and convolution, and then derive the the discrete wavelet transform by using a lowpass/highpass filter pair and downsampling.





Building the Haar Matrix

Examples

Let's have a look at the Mathematica notebook

`HaarTransforms1D.nb`

for a bit more on Haar Transforms.





Coding the Haar Transform

Implementing W_N

- ▶ The natural inclination when coding the DHWT is to simply write a loop and compute the lowpass portion and the highpass portion in the same loop.
- ▶ This bogs down in Mathematica and is also difficult to generalize when we consider longer filters.
- ▶ If we look at the lowpass portion of the transform, $H\mathbf{v}$, we can see a better way to code things.



- ▶ The natural inclination when coding the DHWT is to simply write a loop and compute the lowpass portion and the highpass portion in the same loop.
- ▶ This bogs down in Mathematica and is also difficult to generalize when we consider longer filters.
- ▶ If we look at the lowpass portion of the transform, $H\mathbf{v}$, we can see a better way to code things.



- ▶ The natural inclination when coding the DHWT is to simply write a loop and compute the lowpass portion and the highpass portion in the same loop.
- ▶ This bogs down in Mathematica and is also difficult to generalize when we consider longer filters.
- ▶ If we look at the lowpass portion of the transform, $H\mathbf{v}$, we can see a better way to code things.





Coding the Haar Transform

Implementing W_N

Consider $H\mathbf{v}$ when $\mathbf{v} \in \mathbb{R}^8$. We have

$$H\mathbf{v} = \frac{\sqrt{2}}{2} \begin{bmatrix} v_1 + v_2 \\ v_3 + v_4 \\ v_5 + v_6 \\ v_7 + v_8 \end{bmatrix}$$





Coding the Haar Transform

Implementing W_N

If we rewrite this, we have

$$H\mathbf{v} = \frac{\sqrt{2}}{2} \begin{bmatrix} v_1 + v_2 \\ v_3 + v_4 \\ v_5 + v_6 \\ v_7 + v_8 \end{bmatrix} = \begin{bmatrix} v_1 & v_2 \\ v_3 & v_4 \\ v_5 & v_6 \\ v_7 & v_8 \end{bmatrix} \cdot \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix} = V\mathbf{h}$$





Coding the Haar Transform

Implementing W_N

In a similar way we see that

$$G\mathbf{v} = V\mathbf{g}$$

So all we need to do to compute $W_n\mathbf{v}$ is to create V , multiply it with \mathbf{h} and \mathbf{g} , and join to the two blocks together!





Coding the Haar Transform

Implementing W_N

Here is some Mathematica code to do it:

```
DHWT[v_] := Module[{V, lp, hp, y},
  V = Partition[v, 2, 2];
  lp = V.{1, 1};
  hp = V.{1, -1};
  y = Join[lp, hp];
  Return[Sqrt[2]*y/2];
];
```





Coding the Haar Transform

Implementing W_N^T

- ▶ Writing the code for the inverse transform is a bit trickier.
- ▶ Now the computation is

$$W_N^T \mathbf{v} = \begin{bmatrix} H^T & G^T \end{bmatrix} \mathbf{v}$$

- ▶ Let's again look at a vector $\mathbf{v} \in \mathbb{R}^8$ and consider the product $W_8^T \mathbf{v}$:



Coding the Haar Transform

Implementing W_N^T

- ▶ Writing the code for the inverse transform is a bit trickier.
- ▶ Now the computation is

$$W_N^T \mathbf{v} = \left[H^T \mid G^T \right] \mathbf{v}$$

- ▶ Let's again look at a vector $\mathbf{v} \in \mathbb{R}^8$ and consider the product $W_8^T \mathbf{v}$:









Coding the Haar Transform

Implementing W_N^T

$$W_8^T \mathbf{v} = \frac{\sqrt{2}}{2} \begin{bmatrix} 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \end{bmatrix}$$





Coding the Haar Transform

Implementing W_N^T

or

$$W_8^T \mathbf{v} = \frac{\sqrt{2}}{2} \begin{bmatrix} v_1 - v_5 \\ v_1 + v_5 \\ v_2 - v_6 \\ v_2 + v_6 \\ v_3 - v_7 \\ v_3 + v_7 \\ v_4 - v_8 \\ v_4 + v_8 \end{bmatrix}$$





Coding the Haar Transform

Implementing W_N^T

- ▶ The matrix V takes a bit different shape this time.
- ▶ Now V is

$$V = \begin{bmatrix} v_1 & v_5 \\ v_2 & v_6 \\ v_3 & v_7 \\ v_4 & v_8 \end{bmatrix}$$

- ▶ We need to dot V with both \mathbf{h} and \mathbf{g} but then intertwine the results.





Coding the Haar Transform

Implementing W_N^T

- ▶ The matrix V takes a bit different shape this time.
- ▶ Now V is

$$V = \begin{bmatrix} v_1 & v_5 \\ v_2 & v_6 \\ v_3 & v_7 \\ v_4 & v_8 \end{bmatrix}$$

- ▶ We need to dot V with both \mathbf{h} and \mathbf{g} but then intertwine the results.





Coding the Haar Transform

Implementing W_N^T

- ▶ The matrix V takes a bit different shape this time.
- ▶ Now V is

$$V = \begin{bmatrix} v_1 & v_5 \\ v_2 & v_6 \\ v_3 & v_7 \\ v_4 & v_8 \end{bmatrix}$$

- ▶ We need to dot V with both \mathbf{h} and \mathbf{g} but then intertwine the results.





Coding the Haar Transform

Implementing W_N^T

- ▶ The matrix V takes a bit different shape this time.
- ▶ Now V is

$$V = \begin{bmatrix} v_1 & v_5 \\ v_2 & v_6 \\ v_3 & v_7 \\ v_4 & v_8 \end{bmatrix}$$

- ▶ We need to dot V with both \mathbf{h} and \mathbf{g} but then intertwine the results.





Coding the Haar Transform

Implementing W_N^T

Let's return to the Mathematica notebook

HaarTransforms1D.nb

to see how to code the inverse.



2D Haar Transform

Building the 2D Transform

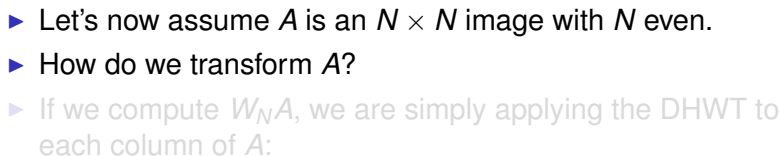
- ▶ Let's now assume A is an $N \times N$ image with N even.
- ▶ How do we transform A ?
- ▶ If we compute $W_N A$, we are simply applying the DHWT to each column of A :



- ▶ Let's now assume A is an $N \times N$ image with N even.
- ▶ How do we transform A ?
- ▶ If we compute $W_N A$, we are simply applying the DHWT to each column of A :



Building the 2D Transform





2D Haar Transform

Building the 2D Transform

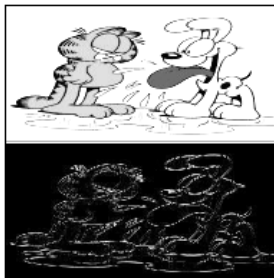
- ▶ Let's now assume A is an $N \times N$ image with N even.
- ▶ How do we transform A ?
- ▶ If we compute $W_N A$, we are simply applying the DHWT to each column of A :



Building the 2D Transform



A


$$W_N A$$
 UNIVERSITY of ST. THOMAS

Building the 2D Transform



Building the 2D Transform

- ▶ We've processed the columns of A - what should we do to process the rows of A as well?
- ▶ Answer: Compute $W_N A W_N^T$.



Building the 2D Transform

- ▶ We've processed the columns of A - what should we do to process the rows of A as well?
- ▶ **Answer:** Compute $W_N A W_N^T$.



Building the 2D Transform

2D Haar Transform

Building the 2D Transform



A



$$W_N A W_N^T$$





2D Haar Transform

Building the 2D Transform

- ▶ If we look at $W_N A W_N^T$ in block format, we can get a better idea what's going on.
- ▶

$$\begin{aligned}
 W_N A W_N^T &= \begin{bmatrix} H \\ G \end{bmatrix} A \begin{bmatrix} H \\ G \end{bmatrix}^T = \begin{bmatrix} HA \\ GA \end{bmatrix} \begin{bmatrix} H^T & | & G^T \end{bmatrix} \\
 &= \begin{bmatrix} HA H^T & | & HA G^T \\ \hline GA H^T & | & GA G^T \end{bmatrix} \\
 &= \begin{bmatrix} \mathcal{B} & | & \mathcal{V} \\ \hline \mathcal{H} & | & \mathcal{D} \end{bmatrix}
 \end{aligned}$$





2D Haar Transform

Building the 2D Transform

- ▶ If we look at $W_N A W_N^T$ in block format, we can get a better idea what's going on.



$$\begin{aligned}
 W_N A W_N^T &= \begin{bmatrix} H \\ G \end{bmatrix} A \begin{bmatrix} H \\ G \end{bmatrix}^T = \begin{bmatrix} HA \\ GA \end{bmatrix} \begin{bmatrix} H^T & G^T \end{bmatrix} \\
 &= \begin{bmatrix} HA H^T & HA G^T \\ GA H^T & GA G^T \end{bmatrix} \\
 &= \begin{bmatrix} \mathcal{B} & \mathcal{V} \\ \mathcal{H} & \mathcal{D} \end{bmatrix}
 \end{aligned}$$





2D Haar Transform

Building the 2D Transform

- ▶ If we look at $W_N A W_N^T$ in block format, we can get a better idea what's going on.
- ▶

$$\begin{aligned}
 W_N A W_N^T &= \begin{bmatrix} H \\ G \end{bmatrix} A \begin{bmatrix} H \\ G \end{bmatrix}^T = \begin{bmatrix} HA \\ GA \end{bmatrix} \begin{bmatrix} H^T & G^T \end{bmatrix} \\
 &= \begin{bmatrix} HA H^T & HA G^T \\ GA H^T & GA G^T \end{bmatrix} \\
 &= \begin{bmatrix} B & V \\ \mathcal{H} & \mathcal{D} \end{bmatrix}
 \end{aligned}$$





- ▶ HAH^T averages along the columns of A and then along the rows of HA . This will produce an approximation (or blur) \mathcal{B} of A .
- ▶ HAG^T averages along the columns of A and then differences along the rows of HA . This will produce vertical differences \mathcal{V} between \mathcal{B} and A .



- ▶ HAH^T averages along the columns of A and then along the rows of HA . This will produce an approximation (or blur) \mathcal{B} of A .
- ▶ HAG^T averages along the columns of A and then differences along the rows of HA . This will produce **vertical differences** \mathcal{V} between \mathcal{B} and A .





2D Haar Transform

Building the 2D Transform

- ▶ GAH^T differences along the columns of A and then averages along the rows of GA . This will produce **horizontal differences** \mathcal{H} between \mathcal{B} and A .
- ▶ GAG^T differences along the columns of A and then differences along the rows of GA . This will produce **diagonal differences** \mathcal{V} between \mathcal{B} and A .



Building the 2D Transform

Building the 2D Transform

2D Haar Transform

Building the 2D Transform

To better understand these block forms, let's look at the
Mathematica notebook

`HaarTransforms2D.nb`



2D Haar Transform

Coding the 2D Transform

- ▶ Coding the 2D Haar transform is easy - we already have a routine that will apply the DHWT to each column of A ,
- ▶ so we can easily write a routine to compute $C = W_N A$. Let's call this routine \bar{W} .
- ▶ Our goal is to compute $B = W_N A W_N^T = C W_N^T$.
- ▶ It turns out that writing code for $C W_N^T$ is a bit tedious, but if we use some linear algebra ...



2D Haar Transform

Coding the 2D Transform

- ▶ Coding the 2D Haar transform is easy - we already have a routine that will apply the DHWT to each column of A ,
- ▶ so we can easily write a routine to compute $C = W_N A$.
Let's call this routine \bar{w} .
- ▶ Our goal is to compute $B = W_N A W_N^T = C W_N^T$.
- ▶ It turns out that writing code for $C W_N^T$ is a bit tedious, but if we use some linear algebra ...



2D Haar Transform

Coding the 2D Transform

- ▶ Coding the 2D Haar transform is easy - we already have a routine that will apply the DHWT to each column of A ,
- ▶ so we can easily write a routine to compute $C = W_N A$.
Let's call this routine \bar{W} .
- ▶ Our goal is to compute $B = W_N A W_N^T = C W_N^T$.
- ▶ It turns out that writing code for $C W_N^T$ is a bit tedious, but if we use some linear algebra ...





2D Haar Transform

Coding the 2D Transform

- ▶ Coding the 2D Haar transform is easy - we already have a routine that will apply the DHWT to each column of A ,
- ▶ so we can easily write a routine to compute $C = W_N A$.
Let's call this routine \bar{W} .
- ▶ Our goal is to compute $B = W_N A W_N^T = C W_N^T$.
- ▶ It turns out that writing code for $C W_N^T$ is a bit tedious, but if we use some linear algebra ...





2D Haar Transform

Coding the 2D Transform

- ▶ Coding the 2D Haar transform is easy - we already have a routine that will apply the DHWT to each column of A ,
- ▶ so we can easily write a routine to compute $C = W_N A$.
Let's call this routine \mathbb{W} .
- ▶ Our goal is to compute $B = W_N A W_N^T = C W_N^T$.
- ▶ It turns out that writing code for $C W_N^T$ is a bit tedious, but if we use some linear algebra ...



- $$B^T = W_N C^T$$



Coding the 2D Transform

- If we transpose both sides of $B = CW_N^T$, we have

$$B^T = W_N C^T$$



- ▶ So we can simply apply \mathbb{W} to C^T and transpose the result.
- ▶ One student wasn't so sure about this ...
- ▶ Let's return to `HaarTransforms2D.nb` to write some code for the 2D Haar Wavelet Transform.



- ▶ So we can simply apply \mathbb{W} to C^T and transpose the result.
- ▶ One student wasn't so sure about this ...
- ▶ Let's return to `HaarTransforms2D.nb` to write some code for the 2D Haar Wavelet Transform.



- ▶ So we can simply apply \mathbb{W} to C^T and transpose the result.
- ▶ One student wasn't so sure about this ...
- ▶ Let's return to `HaarTransforms2D.nb` to write some code for the 2D Haar Wavelet Transform.





2D Haar Transform

Iterating

- ▶ It's time to explain the `NumIterations` directive you have seen in the Mathematica notebooks.
- ▶ We can motivate the idea by looking at the cumulative energy of an image A and its wavelet transform.





2D Haar Transform

Iterating

- ▶ It's time to explain the `NumIterations` directive you have seen in the Mathematica notebooks.
- ▶ We can motivate the idea by looking at the cumulative energy of an image A and its wavelet transform.





2D Haar Transform

Iterating

- ▶ It's time to explain the `NumIterations` directive you have seen in the Mathematica notebooks.
- ▶ We can motivate the idea by looking at the cumulative energy of an image A and its wavelet transform.



2D Haar Transform

Iterating

Here is a 200×200 image and its transform:



A



UNIVERSITY of ST. THOMAS



2D Haar Transform

Iterating

- ▶ To give you an idea, the largest 10000 elements in A make up about 36.5% of the energy in A while the first 10000 elements in the transform comprise about 99.5% of the energy in the transform.
- ▶ The wavelet transform is totally invertible, so if we were to Huffman encode the transform, the bit stream should be markedly smaller.





2D Haar Transform

Iterating

- ▶ To give you an idea, the largest 10000 elements in A make up about 36.5% of the energy in A while the first 10000 elements in the transform comprise about 99.5% of the energy in the transform.
- ▶ The wavelet transform is totally invertible, so if we were to Huffman encode the transform, the bit stream should be markedly smaller.





2D Haar Transform

Iterating

- ▶ To give you an idea, the largest 10000 elements in A make up about 36.5% of the energy in A while the first 10000 elements in the transform comprise about 99.5% of the energy in the transform.
- ▶ The wavelet transform is totally invertible, so if we were to Huffman encode the transform, the bit stream should be markedly smaller.





2D Haar Transform

Iterating

- ▶ We can get even more concentration of the energy if we **iterate** the wavelet transform. That is, after computing the wavelet transform of A , we extract the blur and compute a wavelet transform of it.
- ▶ We could repeat this process p times if the dimensions of A are divisible by 2^p .





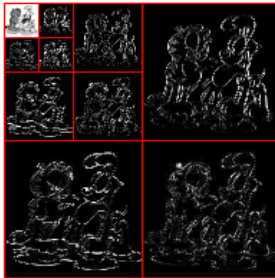
2D Haar Transform

Iterating

- ▶ We can get even more concentration of the energy if we **iterate** the wavelet transform. That is, after computing the wavelet transform of A , we extract the blur and compute a wavelet transform of it.
- ▶ We could repeat this process p times if the dimensions of A are divisible by 2^p .



or 3 times:



In the Classroom

Teaching Ideas

- ▶ The students really enjoy the material in this chapter. It is quite straightforward and ties together everything new we've done to date.
- ▶ I have them look at the entropy of particular vectors when processed by the Haar transform. This gives them some idea of the potential for wavelet-based compression.





In the Classroom

Teaching Ideas

- ▶ The students really enjoy the material in this chapter. It is quite straightforward and ties together everything new we've done to date.
- ▶ I have them look at the entropy of particular vectors when processed by the Haar transform. This gives them some idea of the potential for wavelet-based compression.





In the Classroom

Teaching Ideas

- ▶ The students really enjoy the material in this chapter. It is quite straightforward and ties together everything new we've done to date.
- ▶ I have them look at the entropy of particular vectors when processed by the Haar transform. This gives them some idea of the potential for wavelet-based compression.





In the Classroom

Computer Usage

- ▶ As you might imagine, we do lots of coding in this chapter.
- ▶ I let the students work in pairs and they write code for the Haar transform and its inverse (1D and 2D) as well as iterated versions.
- ▶ They can get pretty frustrated with Mathematica at this point - it is good to show them some simple debugging techniques.



- ▶ As you might imagine, we do lots of coding in this chapter.
- ▶ I let the students work in pairs and they write code for the Haar transform and its inverse (1D and 2D) as well as iterated versions.
- ▶ They can get pretty frustrated with Mathematica at this point - it is good to show them some simple debugging techniques.





In the Classroom

Computer Usage

- ▶ As you might imagine, we do lots of coding in this chapter.
- ▶ I let the students work in pairs and they write code for the Haar transform and its inverse (1D and 2D) as well as iterated versions.
- ▶ They can get pretty frustrated with Mathematica at this point - it is good to show them some simple debugging techniques.





In the Classroom

Computer Usage

- ▶ As you might imagine, we do lots of coding in this chapter.
- ▶ I let the students work in pairs and they write code for the Haar transform and its inverse (1D and 2D) as well as iterated versions.
- ▶ They can get pretty frustrated with Mathematica at this point - it is good to show them some simple debugging techniques.





In the Classroom

Computer Usage

- ▶ I let them use their own images/audio files (sometimes dangerous).
- ▶ To test their iterated 1D inverse, they must download an audio clip from my website that has been transformed p times, guess at p , and then apply their inverse to guess the movie clip.





In the Classroom

Computer Usage

- ▶ I let them use their own images/audio files (sometimes dangerous).
- ▶ To test their iterated 1D inverse, they must download an audio clip from my website that has been transformed p times, guess at p , and then apply their inverse to guess the movie clip.





In the Classroom

Computer Usage

- ▶ I let them use their own images/audio files (sometimes dangerous).
- ▶ To test their iterated 1D inverse, they must download an audio clip from my website that has been transformed p times, guess at p , and then apply their inverse to guess the movie clip.



