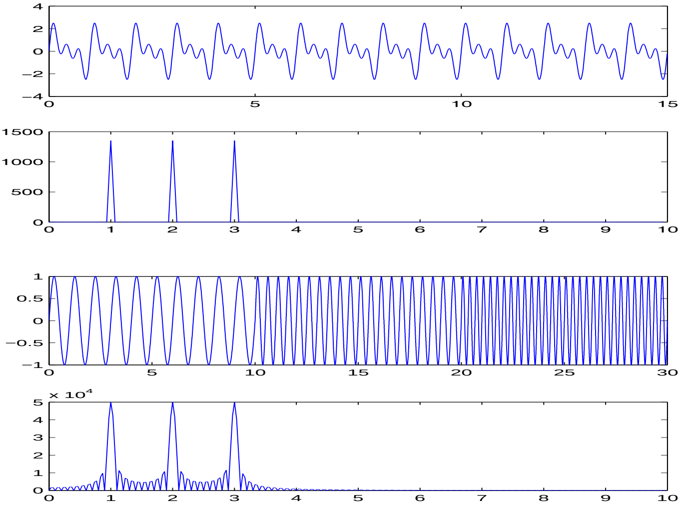
Image compression with wavelets and applications

Eugenio Hernández
UNIVERSIDAD AUTÓNOMA DE MADRID

Universidad de Valladolid
23 de febrero de 2017

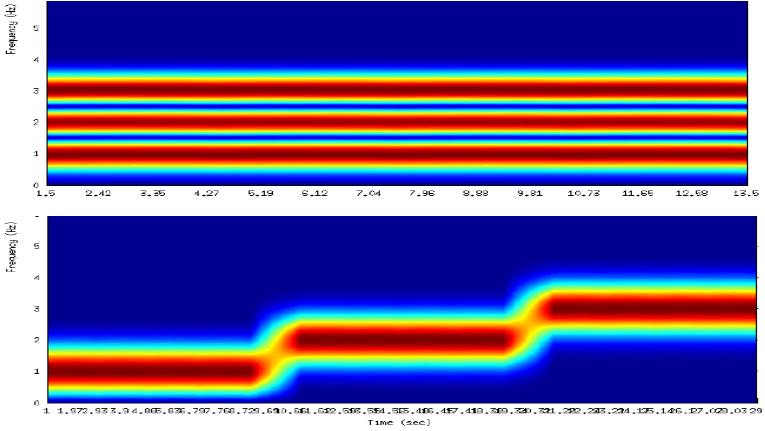
- The **Fourier transform**, $\mathcal{F}(f)(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx$, of two different signals may be similar.



1.2. Short time window transform.

- Given a window function g, such as $g(x) = \chi_{[0,1]}(x)$ or $g(x) = \frac{1}{\sqrt{10}} e^{-10x^2}$, the **Short time window transform** is

$$S_g(f)(t,\xi) = \int_{-\infty}^{\infty} f(x) g(x-t) e^{-2\pi i x\xi} dx.$$

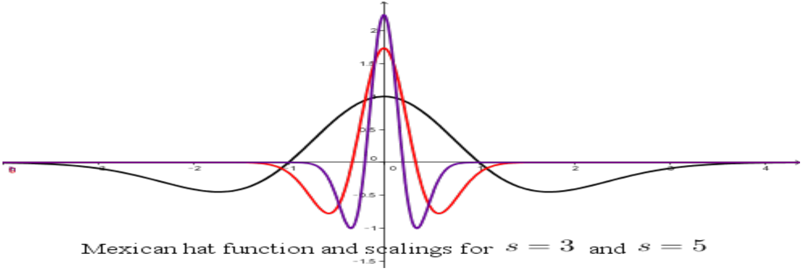


1.3. Continuous wavelet transform.

- Given a function ψ the **continuous wavelet transform** is

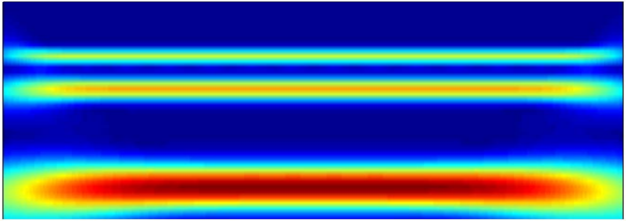
$$W_\psi(f)(t,s) = \int_{-\infty}^{\infty} f(x)\sqrt{s}\,\psi(sx-t)dx.$$

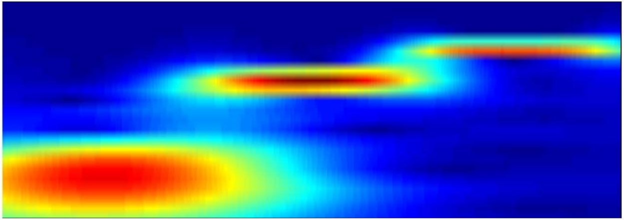
The function used by Morlet is a modulated gaussian $\psi(x) = \pi^{-1/4} e^{-iw_0 x} e^{-x^2/2}$. Another example is the Mexican hat function $\psi = \frac{2}{\sqrt{3}}\pi^{-1/4}(1-x^2)e^{-x^2/2}$.



Mexican hat function and scalings for $s = 3$ and $s = 5$

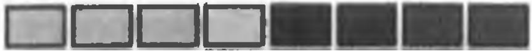
1.4. Continuous wavelet transform.





2. Discrete wavelet transform.
2.1. Haar wavelet transform in 1-D.

• Signal $\mathbf{x} = [200, 200, 200, 210, 40, 80, 100, 102]^T$ and its gray scale representation:



• First approximation: $[200, 205, 60, 101]^T$. This vector is not enough to determine $\mathbf{x}$.

• Directed differences: $[200, 205, 60, 101 | 0, 5, 20, 1]^T$.

• Second approximation: $[203, 81 | 2, 20 | 0, 5, 20, 1]^T$.

• Quantizing: $[203, 81 | 0, 20 | 0, 0, 20, 0]^T$.

• Huffman encoding ($0 \leftrightarrow 0; 20 \leftrightarrow 10; 81 \leftrightarrow 110; 203 \leftrightarrow 111$): 111110010001000 (15 bits instead of 64 bits: compression of 75.76 %)

• Signal recovered from quantize version:

$$[203, 203, 203, 203, 41, 81, 101, 101]^T.$$

2.2. Haar wavelet transform in 1-D.

- For a signal $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$, N even: for $k = 1, \ldots, N/2$,

$$\mathbf{x} \longrightarrow \left[\begin{array}{c} \mathbf{a} \\ \hline \mathbf{d} \end{array} \right], \quad a_k = \frac{x_{2k-1} + x_{2k}}{2}, \quad d_k = \frac{-x_{2k-1} + x_{2k}}{2}.$$

- Matrix interpretation for $\mathbf{x} = [x_1, x_2, x_3, x_4]^T$

$$\widetilde{W_4}\mathbf{x} = \begin{bmatrix} 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 \\ - & - & - & - \\ -1/2 & 1/2 & 0 & 0 \\ 0 & 0 & -1/2 & 1/2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ - \\ d_1 \\ d_2 \end{bmatrix} = \left[\begin{array}{c} \mathbf{a} \\ \hline \mathbf{d} \end{array} \right].$$

- Matrix interpretation for $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$:

$$\widetilde{W_N}\mathbf{x} = \left[\begin{array}{c} \widetilde{H}_{N/2} \\ \hline \widetilde{G}_{N/2} \end{array} \right] \mathbf{x} = \left[\begin{array}{c} \mathbf{a} \\ \hline \mathbf{d} \end{array} \right].$$

2.3. Inverse Haar wavelet transform in 1-D.

- Since $x_{2k-1} = a_k - d_k$ and $x_{2k} = a_k + d_k$ the matrix $\widetilde{W_N}$ can be inverted.
- For $N = 4$,

$$(\widetilde{W_4})^{-1} \begin{bmatrix} \mathbf{a} \\ \hline \mathbf{d} \end{bmatrix} = \left[\begin{array}{cc|cc} 1 & 0 & -1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 1 & 0 & 1 \end{array}\right] \begin{bmatrix} a_1 \\ a_2 \\ - \\ d_1 \\ d_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}.$$

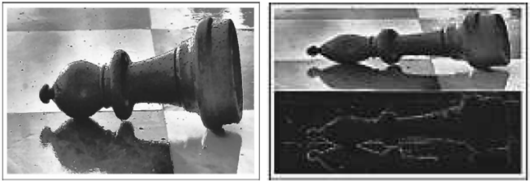
- The inverse Haar wavelet transform in 1-D is given by:

$$\mathbf{x} = 2\widetilde{W_N}^T \begin{bmatrix} \mathbf{a} \\ \hline \mathbf{d} \end{bmatrix} = 2 \left[\begin{array}{c|c} \widetilde{H}^T_{N/2} & \widetilde{G}^T_{N/2} \end{array}\right] \begin{bmatrix} \mathbf{a} \\ \hline \mathbf{d} \end{bmatrix}$$

- The matrix $W_N = \sqrt{2}\widetilde{W_N}$ is orthogonal

2.4. Haar wavelet transform in 2-D.

- A is an image of size $M \times N$, M and N both even.
- Apply W_M (1-D) to the columns of A: compute $W_M A$.



- Apply W_N (1-D) to the rows of $W_M A$: compute $W_M A W_N^T$.

- Block multiplication:

$$W_M A W_N^T = \left[\begin{array}{c} H_{M/2} \\ \hline G_{M/2} \end{array} \right] A \left[\begin{array}{c|c} H_{N/2}^T & G_{N/2}^T \end{array} \right]$$

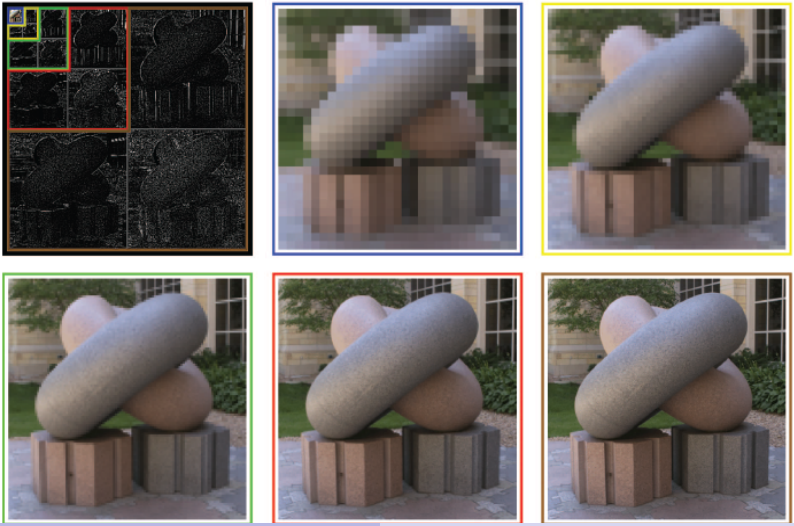
$$= \left[\begin{array}{c} H_{M/2}A \\ \hline G_{M/2}A \end{array} \right] \left[\begin{array}{c|c} H_{N/2}^T & G_{N/2}^T \end{array} \right]$$

$$= \left[\begin{array}{c|c} H_{M/2}AH_{N/2}^T & H_{M/2}AG_{N/2}^T \\ \hline G_{M/2}AH_{N/2}^T & G_{M/2}AG_{N/2}^T \end{array} \right] = \left[\begin{array}{c|c} \mathcal{B} & \mathcal{V} \\ \hline \mathcal{H} & \mathcal{D} \end{array} \right]$$

- This process can be iterated with the blur image.

2.4. Haar wavelet transform in 2-D.

- Decomposition and reconstruction of an image with 2-D Haar wavelet transform.



3.1. Convolution, filters, and Fourier series.

• For **h** and **x** bi-infinity sequences, their discrete convolution is a bi-infinite sequence $\mathbf{y} = \mathbf{h} * \mathbf{x}$ whose n-th component, $n \in \mathbb{Z}$, is

$$y_n = \sum_{k \in \mathbb{Z}} h_k x_{n-k}.$$

• The Haar Wavelet transform of **x** can be obtained by discrete convolution: take $x_n = 0$ if $n \leq 0$ or $n > N$.

✠ Compute $\mathbf{u} = \mathbf{h} * \mathbf{x}$ for $h_0 = \sqrt{2}/2 = h_1$.

✠ Compute $\mathbf{v} = \mathbf{g} * \mathbf{x}$ for $g_0 = \sqrt{2}/2$, $g_1 = -\sqrt{2}/2$.

✠ Downsample **u** and **v** by keeping only the even components of each vector.

✠ Truncate the downsample vectors **u** and **v** to obtain **a** and **d**.

• Vectors **h** and **g** are called FILTERS

3.2. Convolution, filters, and Fourier series.

• For a finite length filter **f** its Fourier series is the trigonometric polynomial

$$F(\xi) = \sum_k f_k e^{ik\xi}.$$

• For $\mathbf{h} = [h_0, h_1]^T = [\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}]^T$,

$$H(\xi) = \frac{\sqrt{2}}{2} + \frac{\sqrt{2}}{2} e^{i\xi} = \sqrt{2} e^{i\xi/2} \cos(\xi/2)$$

is a LOWPASS FILTER.

• For $\mathbf{g} = [g_0, g_1]^T = [\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}]^T$,

$$G(\xi) = \frac{\sqrt{2}}{2} - \frac{\sqrt{2}}{2} e^{i\xi} = -\sqrt{2} i e^{i\xi/2} \sin(\xi/2)$$

is a HIGHPASS FILTER.

3.3. Convolution, filters, and Fourier series.

• The lowpass and highpass filters have the following properties:

(1)

$$i)\ |H(0)| = \sqrt{2} \qquad ii)\ |H(\pi)| = 0\,.$$

(2)

$$|H(\xi)|^2 + |H(\xi+\pi)|^2 = 2.$$

(3)

$$i)\ |G(0)| = 0 \qquad ii)\ |G(\pi)| = \sqrt{2}\,.$$

(4)

$$|G(\xi)|^2 + |G(\xi+\pi)|^2 = 2.$$

(5)

$$H(\xi)\overline{G(\xi)} + H(\xi+\pi)\overline{G(\xi+\pi)} = 0.$$

4.1. Daubechies orthogonal wavelet transform.

- When applying Haar Wavelet Transform to $\mathbf{v} = [2, 2, 200, 200]^T$ one gets $\sqrt{2}[0, 200|0, 0]^T$, and the singularity between 2 and 200 is not detected. This is because the Haar filter is too short.

- I. Daubechies idea is to use longer finite filters $\mathbf{h} = [h_0, h_1, h_2, h_3]^T$, $\quad \mathbf{g} = [g_0, g_1, g_2, g_3]^T$ with the following matrix orthogonal:

$$W_8 = \begin{bmatrix} h_3 & h_2 & h_1 & h_0 & 0 & 0 & 0 & 0 \\ 0 & 0 & h_3 & h_2 & h_1 & h_0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_3 & h_2 & h_1 & h_0 \\ h_1 & h_0 & 0 & 0 & 0 & 0 & h_3 & h_2 \\ - & - & - & - & - & - & - & - \\ g_3 & g_2 & g_1 & g_0 & 0 & 0 & 0 & 0 \\ 0 & 0 & g_3 & g_2 & g_1 & g_0 & 0 & 0 \\ 0 & 0 & 0 & 0 & g_3 & g_2 & g_1 & g_0 \\ g_1 & g_0 & 0 & 0 & 0 & 0 & g_3 & g_2 \end{bmatrix}$$

4.2. Daubechies orthogonal wavelet transform.

- Since W_8 has to be orthogonal:

$$(6)\ h_0^2 + h_1^2 + h_2^2 + h_3^2 = 1\,, \quad (7)\ h_0 h_2 + h_1 h_3 = 0\,.$$

- From $H(\pi) = 0$, (8) $h_0 - h_1 + h_2 - h_3 = 0\,.$

- The system of equations (6), (7), and (8) have infinitely many solutions.

- I. Daubechies idea was to make H more flat at π by imposing $H'(\pi) = 0$. The new equation is:

$$(9)\ h_1 - 2h_2 + 3h_3 = 0.$$

- The four equations (6), (7), (8), and (9) have two solutions. One of them is:

$$h_0 = \frac{1+\sqrt{3}}{4\sqrt{2}}, \quad h_1 = \frac{3+\sqrt{3}}{4\sqrt{2}}, \quad h_2 = \frac{3-\sqrt{3}}{4\sqrt{2}}, \quad h_3 = \frac{1-\sqrt{3}}{4\sqrt{2}}\,.$$

- Clever choice: $\mathbf{g} = [h_3, -h_2, h_1, -h_0]^T$.

4.3. Daubechies orthogonal wavelet transform.

• Construction of longer even-length filters $\mathbf{h} = [h_0, \ldots, h_L]^T$, L odd, that produce W_N orthogonal matrix.

I. DAUBECHIES

• Let $H(\xi) = \sum_k h_k e^{ik\xi}$ and $G(\xi) = \sum_k g_k e^{ik\xi}$. Then,

$$(A)\ |H(\xi)|^2 + |H(\xi+\pi)|^2 = 2 \quad \Leftrightarrow \quad \sum_k h_k h_{k-2n} = \delta_{0,n}\,, \ \ n \in \mathbb{Z}.$$

• Also (B)

$$H(\xi)\overline{G(\xi)} + H(\xi+\pi)\overline{G(\xi+\pi)} = 0 \quad \Leftrightarrow \quad \sum_k h_k g_{k-2n} = 0\,, \ \ n \in \mathbb{Z}.$$

4.4. Daubechies orthogonal wavelet transform.

S. Mallat

• Suppose $H(\xi)$ satisfies (A). Then

$$G(\xi) = -e^{iL\xi}\overline{H(\xi+\pi)}$$

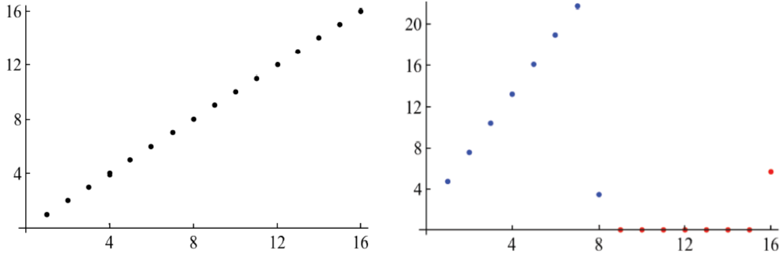
satisfies (A).
• Moreover $H(\xi)$ and $G(\xi)$ satisfy (B).

I. Daubechies

• Assume $H^{(m)}(\pi) = 0, m = 0, 1, \dots, \dfrac{L-1}{2}$.
• There are $M = 2^{\lfloor \frac{L+2}{4} \rfloor}$ real valued lowpass filters that produce orthogonal wavelet transforms.

4.5. Unwanted effects.

• The wrapping of filters in orthonormal wavelet transforms produces unwanted effects. To see this, apply Daubechies orthogonal transform W_{16} to the vector $\mathbf{x}$, where $x_k = k, \ k = 1, 2, \ldots, 16$, with the lowpass filter or length 4 constructed above.



• The last two terms of $\mathbf{x}$ are combined with the first two to produce the last weighted average and difference.

4.6. Biorthogonal filters for JPEG 2000.

• The condition $W_N^{-1} = W_N^T$ is hard to fulfill for filters of length larger than 4 and produces unwanted effects at borders.

• Construct two set of filters **h** and $\widetilde{\mathbf{h}}$ poducing transforms W_N and $\widetilde{W}_N$ such that $\widetilde{W}^{-1} = W^T$.

• These will be called biorthogonal filters and they must satisfy:

$$\widetilde{W_M} W_N^T == \left[\begin{array}{c} \widetilde{H_{N/2}} \\ \hline \widetilde{G_{N/2}} \end{array} \right] \left[\begin{array}{c|c} H_{N/2}^T & G_{N/2}^T \end{array} \right]$$

$$= \left[\begin{array}{c|c} \widetilde{H_{N/2}} H_{N/2}^T & \widetilde{H_{N/2}} G_{N/2}^T \\ \hline \widetilde{G_{N/2}} H_{N/2}^T & \widetilde{G_{N/2}} G_{N/2}^T \end{array} \right] = \left[\begin{array}{c|c} I_{N/2} & 0_{N/2} \\ \hline 0_{N/2} & I_{N/2} \end{array} \right] \qquad (10)$$

4.6. Biorthogonal filters for JPEG 2000.

• Let $H(\xi)$ and $\widetilde{H}(\xi)$ be the Fourier series of a pair of biorthogonal filters $\mathbf{h}$ and $\widetilde{\mathbf{h}}$.

• They must be lowpass filters:

$$(11) \quad \widetilde{H}(0) = H(0) = \sqrt{2}\,, \qquad \widetilde{H}(\pi) = H(\pi) = 0.$$

A. COHEN, I. DAUBEHIES, J.C. FEAUVEAU

• A pair of biorthogonal filters must satisfy

$$(12) \quad \widetilde{H}(\xi)\overline{H(\xi)} + \widetilde{H}(\xi+\pi)\overline{H(\xi+\pi)} = 2 \;\Leftrightarrow\; \sum_{k\in\mathbb{Z}} \widetilde{h}_k h_{k-2n} = \delta_{0,n}\,.$$

• Taking $\widetilde{G}(\xi) = -e^{i\xi}\overline{H(\xi+\pi)} \;\Leftrightarrow\; \widetilde{g}_k = (-1)^k h_{1-k}$ and $G(\xi) = -e^{i\xi}\overline{\widetilde{H}(\xi+\pi)} \;\Leftrightarrow\; g_k = (-1)^k \widetilde{h}_{1-k}$, equality (10) holds.

4.6. Biorthogonal filters for JPEG 2000.

- Start with $\widetilde{h} = [h_{-1}, h_0, h_1]^T = \frac{\sqrt{2}}{4}[1, 2, 1]^T$, symmetric of length 3.
- $\widetilde{h}$ satisfies $\widetilde{H}(\xi) = \frac{\sqrt{2}}{4}(e^{-i\xi} + 2 + e^{i\xi}) = \frac{\sqrt{2}}{2}(1 + \cos\xi)$. Thus,

$$\widetilde{H}(0) = \sqrt{2} \qquad \widetilde{H}(\pi) = 0.$$

- Find $\mathbf{h} = [h_2, h_1, h_0, h_1, h_2]$ (symmetric of length 5) such that (11) and (12) hold.

4.6. Biorthogonal filters for JPEG 2000.

- This is equivalent to the following linear system:

$$\left\{ \begin{array}{ll} h_0 - 2h_1 + 2h_2 = 0 \\ h_0 + h_1 \qquad\qquad = \sqrt{2} \\ \qquad h_1 + 2h_2 = 0 \end{array} \right\}.$$

- Solution: $h_2 = -\frac{\sqrt{2}}{8}, \quad h_1 = \frac{\sqrt{2}}{4}, \quad h_0 = \frac{3\sqrt{2}}{4}.$
- This is CDF(5,3) filter similar to the ones used in JPEG2000.

JPEG2000

A modify version of CDF(5,3) that takes integers to integers is used for lowless compression and a CDF(9,7) is used for lossy compression in JPEG2000.

4.7. CDF (5,3) biorthogonal wavelet transform

$$W_8 = \frac{\sqrt{2}}{8}\begin{bmatrix} 3 & 2 & -1 & 0 & 0 & 0 & -1 & 2 \\ -1 & 2 & 3 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & 3 & 2 & -1 & 0 \\ -1 & 0 & 0 & 0 & -1 & 2 & 3 & 2 \\ 2 & -4 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -4 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -4 & 2 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 2 & -4 \end{bmatrix}$$

$$\widetilde{W}_8 = \frac{\sqrt{2}}{8}\begin{bmatrix} 4 & 2 & 0 & 0 & 0 & 0 & 0 & 2 \\ 0 & 2 & 4 & 2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 4 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 4 & 2 \\ 2 & -3 & 2 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 2 & -3 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & -3 & 2 & 1 \\ 2 & 1 & 0 & 0 & 0 & 1 & 2 & -3 \end{bmatrix}$$

5. Bibliography