

Image Compression Using the Haar Wavelet Transform

Peggy Morton and Arne Petersen
Math 45 — College of the Redwoods

December 19, 1997

Abstract

Here we are implementing a method of image compression that uses the Haar wavelet transform. First we will describe the process of the transformation and then show the results of this by displaying some images and their compressions.

1 Introduction

In many situations today we are presented with the task of conserving time and space: be it trying to get to work on time, or attempting to cram that one last book on our shelves. The situation we would like to address in this paper is that of conserving time and space when handling large images. This can be done by compressing the image. In doing this we will begin with an original image in which the color of each pixel is represented by a component of a matrix. The minimum of this matrix representing black, and the maximum representing white. We will then use a process called averaging and differencing to develop a new matrix representing the same image in a more concise manner. To then compress this image we will eliminate some of the unnecessary information, and arrive at an approximation of our original image.

1.1 Averaging and Differencing

For simplicity, to describe the averaging and differencing process we take only the first row of an 8x8 matrix. This row is shown below. Because our matrix is 8x8 the process will involve three steps ($2^3 = 8$).

[3 5 4 8 13 7 5 3]

- Step 1

For the first step we take the average of each pair of components in our original string and place the results in the first four positions of our new string. The remaining four numbers are the differences of the first element in each pair and its corresponding average (e.g. $3 - 4 = -1$, $4 - 6 = -2$). These numbers are called detail coefficients. Our result of the first step therefore contains four averages and four detail coefficients (**bold**) as shown below.

$$[4 \ 6 \ 10 \ 4 \ \mathbf{-1} \ \mathbf{-2} \ \mathbf{3} \ \mathbf{1}]$$

- Step 2

We then apply this same method to the first four components of our new string resulting in two new averages and their corresponding detail coefficients. The remaining four detail coefficients are simply carried directly down from our previous step. And the result for step two is as follows.

$$[5 \ 7 \ \mathbf{-1} \ \mathbf{3} \ -1 \ -2 \ 3 \ 1]$$

- Step 3

Performing the same averaging and differencing to the remaining pair of averages completes step three. The last six components have again been carried down from the previous step. We now have as our string, one row average in the first position followed by seven detail coefficients.

$$[6 \ \mathbf{-1} \ \mathbf{-1} \ 3 \ -1 \ -2 \ 3 \ 1]$$

1.2 Image Representation

We can now use the same averaging and differencing process and apply it to an entire matrix. The following matrix (A) represents the image in figure 1. Notice the larger components in the matrix represent the lighter shades of gray.

$$A = \begin{bmatrix} 64 & 2 & 3 & 61 & 60 & 6 & 7 & 57 \\ 9 & 55 & 54 & 12 & 13 & 51 & 50 & 16 \\ 17 & 47 & 46 & 20 & 21 & 43 & 42 & 24 \\ 40 & 26 & 27 & 37 & 36 & 30 & 31 & 33 \\ 32 & 34 & 35 & 29 & 28 & 38 & 39 & 25 \\ 41 & 23 & 22 & 44 & 45 & 19 & 18 & 48 \\ 49 & 15 & 14 & 52 & 53 & 11 & 10 & 56 \\ 8 & 58 & 59 & 5 & 4 & 62 & 63 & 1 \end{bmatrix}$$

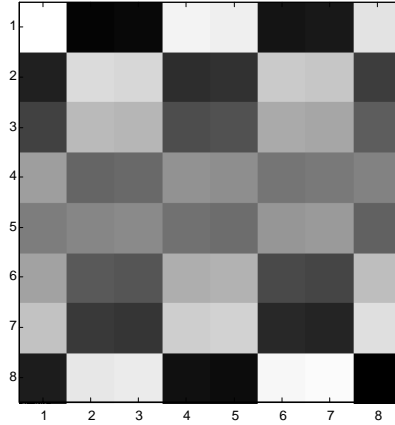


Figure 1.

If we apply the averaging and differencing to each of the rows the results are row averages in the first column and the remaining components being the detail coefficients of that row.

$$\begin{bmatrix} 32.5 & 0 & .5 & .5 & 31 & -29 & 27 & -25 \\ 32.5 & 0 & -.5 & -.5 & -23 & 21 & -19 & 17 \\ 32.5 & 0 & -.5 & -.5 & -15 & 13 & -11 & 9 \\ 32.5 & 0 & .5 & .5 & 7 & -5 & 3 & -1 \\ 32.5 & 0 & .5 & .5 & -1 & 3 & -5 & 7 \\ 32.5 & 0 & -.5 & -.5 & 9 & -11 & 13 & -15 \\ 32.5 & 0 & -.5 & -.5 & 17 & -19 & 21 & -23 \\ 32.5 & 0 & .5 & .5 & -25 & 27 & -29 & 31 \end{bmatrix}$$

If we now apply the averaging and differencing to the columns we get the following matrix.

$$\begin{bmatrix} 32.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & -4 & 4 & -4 \\ 0 & 0 & 0 & 0 & 4 & -4 & 4 & -4 \\ 0 & 0 & .5 & .5 & 27 & -25 & 23 & -21 \\ 0 & 0 & -.5 & -.5 & -11 & 9 & -7 & 5 \\ 0 & 0 & .5 & .5 & -5 & 7 & -9 & 11 \\ 0 & 0 & -.5 & -.5 & 21 & -23 & 25 & -27 \end{bmatrix}$$

We now have a matrix that represents our image in a more concise manner: with one overall average in the upper left-hand corner of the matrix. The remaining components are all detail coefficients that now represent the amount

of detail in that area of the image. Because we know this, we can eliminate some information from our matrix and still be capable of attaining a fairly good approximation of our original matrix. Doing this we choose some number (δ) and set equal to zero all elements with a magnitude less than δ .

Choosing $\delta = 5$ we eliminated eighteen of the detail coefficients (bold).

$$\begin{bmatrix} 32.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & 0 & 0 & 0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & 0 & \mathbf{0} & \mathbf{0} & 27 & -25 & 23 & -21 \\ 0 & 0 & \mathbf{0} & \mathbf{0} & -11 & 9 & -7 & \mathbf{0} \\ 0 & 0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & 7 & -9 & 11 \\ 0 & 0 & \mathbf{0} & \mathbf{0} & 21 & -23 & 25 & -27 \end{bmatrix}$$

To arrive at our approximation of the original matrix we now apply the inverse of the averaging and differencing operations.

$$\begin{bmatrix} 59.5 & 5.5 & 7.5 & 57.5 & 55.5 & 9.5 & 11.5 & 53.5 \\ 5.5 & 59.5 & 57.5 & 7.5 & 9.5 & 55.5 & 53.5 & 11.5 \\ 21.5 & 43.5 & 41.5 & 23.5 & 25.5 & 39.5 & 32.5 & 32.5 \\ 43.5 & 21.5 & 23.5 & 41.5 & 39.5 & 25.5 & 32.5 & 32.5 \\ 32.5 & 32.5 & 39.5 & 25.5 & 23.5 & 41.5 & 21.5 & 21.5 \\ 32.5 & 32.5 & 25.5 & 39.5 & 41.5 & 23.5 & 43.5 & 43.5 \\ 53.5 & 11.5 & 9.5 & 55.5 & 57.5 & 7.5 & 5.5 & 59.5 \\ 11.5 & 53.5 & 55.5 & 9.5 & 7.5 & 57.5 & 59.5 & 5.5 \end{bmatrix}$$

As shown in figure 2b we can get a good approximation of the original image by using this process. We have lost some of the detail in the image, but it is so minimal that the loss would not be noticeable in most cases.

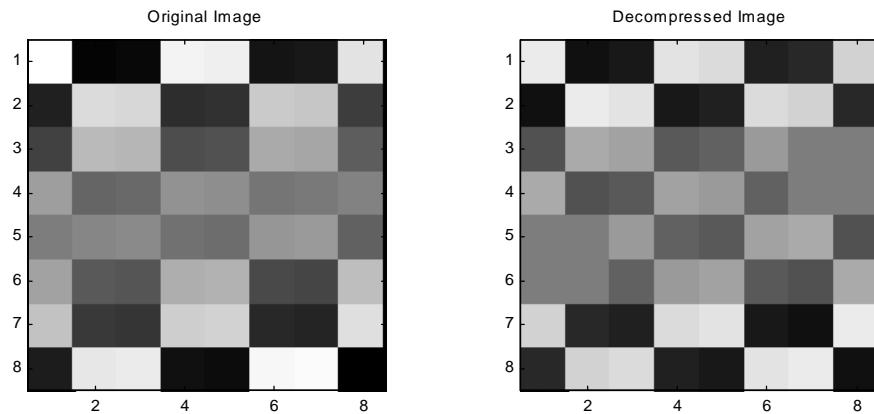


Figure 2

1.3 Using Linear Algebra

To apply the averaging and differencing using linear algebra we can use three matrices (A_1, A_2, A_3) that perform each of the three steps of the averaging and differencing process.

- Step 1

When multiplying the string by the first matrix the first four columns are taking the average of each pair, and the last four columns take the corresponding differences.

$$\begin{aligned}
 & \left[\begin{array}{cccccccc} 3 & 5 & 4 & 8 & 13 & 7 & 5 & 3 \end{array} \right] \begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 0 & -\frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & -\frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & -\frac{1}{2} \end{bmatrix} \\
 &= \left[\begin{array}{cccccccc} 4 & 6 & 10 & 4 & -1 & -2 & 3 & 1 \end{array} \right]
 \end{aligned}$$

- Step 2

The second matrix works in much the same way; the first four columns now perform the averaging and differencing to the remaining pairs, and the identity matrix in the last four columns carry down the detail coefficients from step1.

$$\begin{aligned}
 & \left[\begin{array}{cccccccc} 4 & 6 & 10 & 4 & -1 & -2 & 3 & 1 \end{array} \right] \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & -\frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \left[\begin{array}{cccccccc} 5 & 7 & -1 & 3 & -1 & -2 & 3 & 1 \end{array} \right]
 \end{aligned}$$

- Step 3

Similarly in the final step, the averaging and differencing is done by the first two columns of the matrix, and the identity matrix carries down the detail

coefficients from step 2.

$$\begin{aligned}
 & \begin{bmatrix} 5 & 7 & -1 & 3 & -1 & -2 & 3 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \\
 & = \begin{bmatrix} 6 & -1 & -1 & 3 & -1 & -2 & 3 & 1 \end{bmatrix}
 \end{aligned}$$

To simplify this process we can multiply these three matrices together to obtain a single transform matrix ($W = A_1A_2A_3$). We can now multiply our original string by just one transform matrix to go directly from the original string to the final results of step 3.

$$\begin{aligned}
 & \begin{bmatrix} 3 & 5 & 4 & 8 & 13 & 7 & 5 & 3 \end{bmatrix} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{4} & 0 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & -\frac{1}{2} & 0 & 0 & -\frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{4} & 0 & 0 & -\frac{1}{2} & 0 & 0 \\ 0 & 0 & -\frac{1}{4} & 0 & 0 & -\frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{4} & 0 & 0 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 & -\frac{1}{4} & 0 & 0 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 & -\frac{1}{4} & 0 & 0 & 0 & -\frac{1}{2} \\ 0 & 0 & 0 & -\frac{1}{4} & 0 & 0 & 0 & -\frac{1}{2} \end{bmatrix} \\
 & = \begin{bmatrix} 6 & -1 & -1 & 3 & -1 & -2 & 3 & 1 \end{bmatrix}
 \end{aligned}$$

In the following equations we simplify this process of matrix multiplication; first the averaging and differencing and second the inverse of those operations.

1. $T = ((AW)^T W)^T$

$$\begin{aligned}
 T &= (W^T A^T W)^T \\
 T &= W^T (A^T)^T (W^T)^T \\
 T &= W^T A W
 \end{aligned}$$
2. $(W')^{-1} T W^{-1} = A$

$$(W^{-1})' T W^{-1} = A$$

1.4 Image Compressions

To visualize what this transform does on a larger scale we've compressed these images to different degrees; the first being the original image, followed by successively better approximations.

Figure 3b was compressed to the greatest extent, that is after doing the averaging and differencing steps we choose a rather large value for delta. This eliminated quite a few components from the transformed matrix. This transformed matrix now has only 631 non-zero entries as opposed to the 65,536 of the original matrix, giving a compression ratio of approximately 103 to 1. Utilizing only the non-zero numbers and their positions in the matrix, you can see that storing and transmitting a matrix with less non-zero entries is going to take up much less space and therefore transmit faster. We have now demonstrated the amount of space that can be saved by compressing an image, but as shown in figure 3b a great deal of detail has been lost and its usefulness is questionable.

To try and reach a happy medium in figure 3c we have chosen a lower value for delta, giving a transformed matrix with 1411 non-zero entries and a compression ratio of about 46 to 1, the detail achieved in this image was reasonable and as noted above, we still have achieved a fairly high compression ratio.

The final image was compressed the least. The transformed matrix here has 9,060 non-zero entries whereas the original matrix again had 65,536. This gives us a compression ratio of approximately 7 to 1. Here we have an even better approximation to the original image



Figure 3a. Original Image



Figure 3b



Figure 3c



Figure 3d

The above figures show just a few of the possibilities for one picture. Each of these approximations would be appropriate in different situations. For example figure 3b could be transmitted over the internet much faster than the original, giving the user the opportunity to decide if he or she wanted to wait for the entire image or simply move on. As the approximations get better they become more and more useful. Figure 3d, for example, is very close to the original picture and therefore would be nearly as useful as the original.

Throughout this paper we've discussed how image compression gives a transformed matrix with far less nonzero entries than the original. And that by using this form of representing the image we can save a considerable amount of space and time when storing and transmitting the image. As you can see, image compression using the Haar wavelet transform can render some very nice approximations to large images that would otherwise become a problem to store.

1.5 References

- The m-files used to do the compressions of these images were obtained from Dr. Colm Mulcahy of Spelman College @ <http://www.spelman.edu/~colm/>
- The images used were obtained from Waterloo Fractal Compression Project @ <http://links.uwaterloo.ca/>
- Mulcahy, Colm Ph.D, "Image compression using the Haar wavelet transform," *Spelman Science and Math Journal*, 22 - 31.
- Mulcahy, Colm, "Plotting and Scheming with Wavelets" *Mathematics Magazine* **69**, 5,(1996),323-343.