Seth Goodman 2018-04-10

APSC 608 - Project 02

This project will explore some of the functionality of MATLAB's *Wavelet Toolbox* [1][2] by compressing a simple image.

Original image

Museum Island - Berlin, Germany

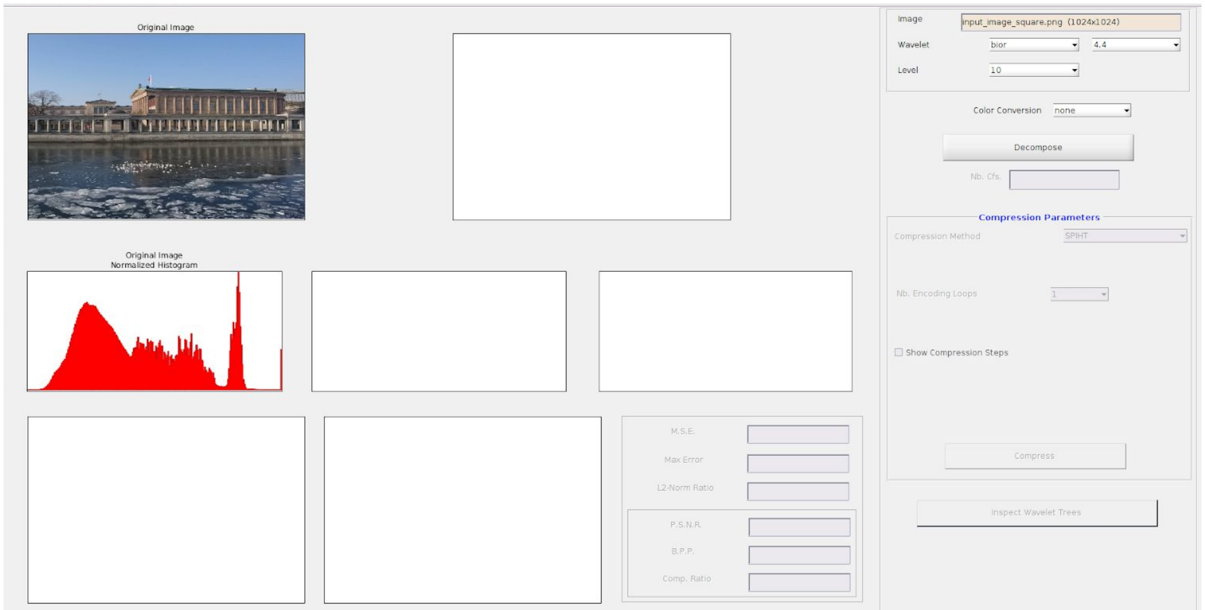
Open Wavelet Toolbox

```
waveletAnalyzer
```

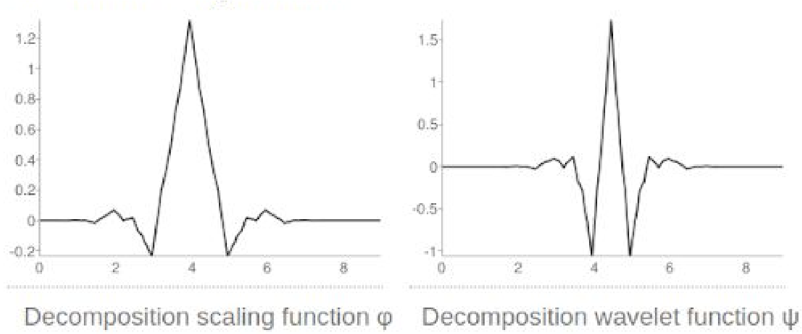
Select ***True Compression 2-D*** within the *Specialized Tools 2-D* section.

The image can be loaded by choosing ***File > Load Image > Matlab Support Formated*** from the menu, and selecting your image file. Using the original image will produce an error because the image must have dimensions (rows and columns of pixels) which are each a power of two. In order to use this image, it will be cropped to a 1024x1024 pixel section of the original image.

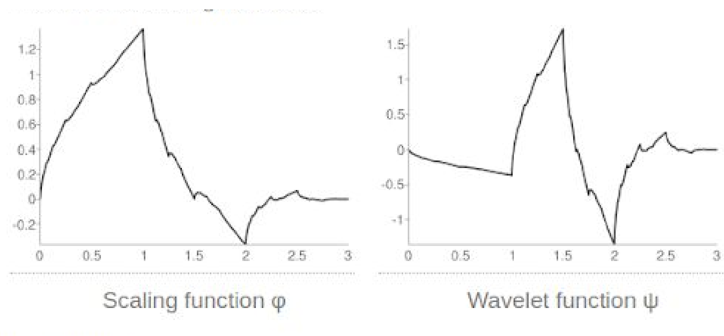
When the image is loaded, the tool will provide you with a normalized histogram of the original image.



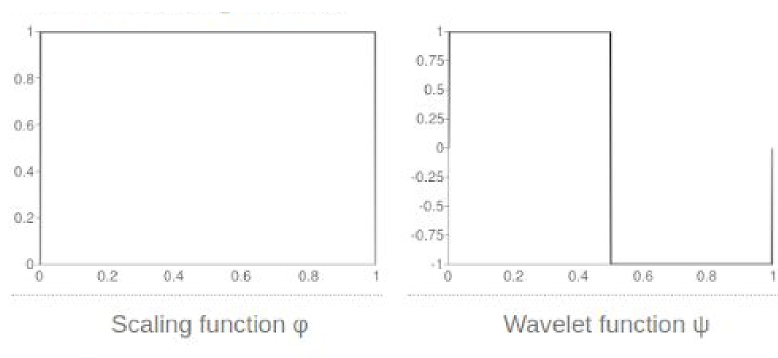
Next, you can select the wavelet along with the level of decomposition. For this project we will compare the Biorthogonal 4.4 wavelet (bior4.4), the Haar wavelet (haar), and the Daubechies 2 wavelet (db2). The scaling and wavelet functions for each of these wavelets can be seen below.



Biorthogonal 4.4 Wavelet [6]

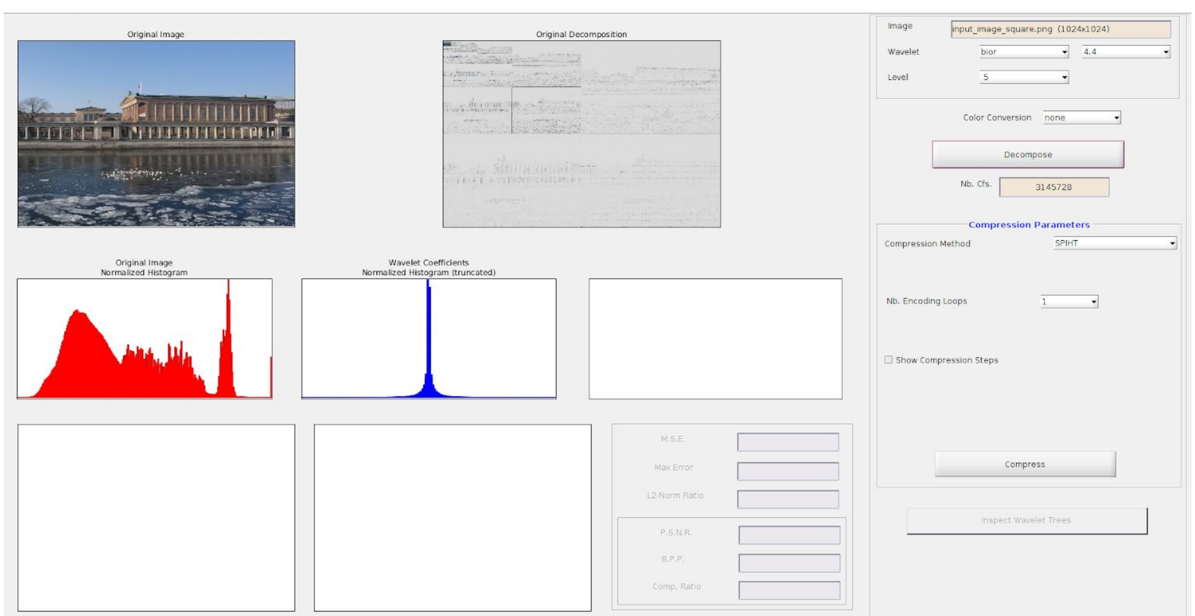


Daubechies 2 Wavelet [10]

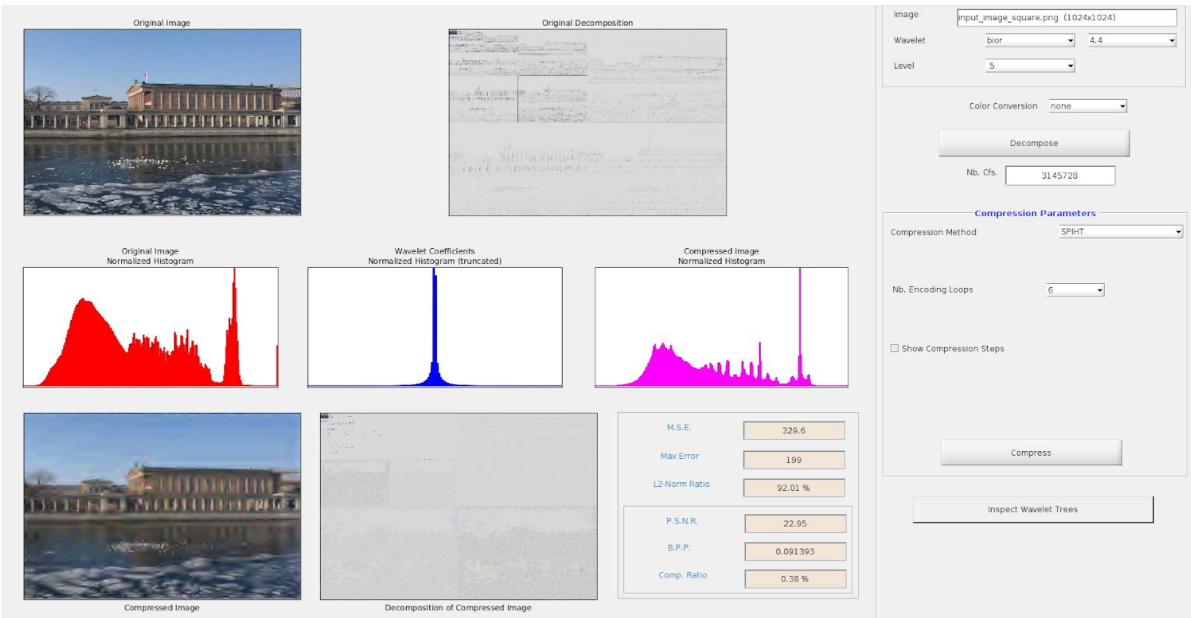


Haar Wavelet [7]

Once the wavelet and level are selected, use the **decompose** button to decompose the image and generate a truncated normalized histogram of the wavelet coefficients.



Having decomposed the image using the selected wavelet and decomposition level, we can now choose the method of compression and number of loops to use. More loops will improve the quality of the compressed image at the cost of computational time and compression ratio. Once these have been selected, clicking **Compress** will run the compression process.



Once the compression has been completed, additional figures and stats will appear in the toolbox window, as seen above. This includes the normalized histogram of the compressed image, the actual compressed image, the decomposition of the compressed image, and compression statistics which are defined below[3].

Quantitative and Perceptual Quality Measures

The following quantitative measurements and measures of perceptual quality are useful for analyzing wavelet signals and images.

- **M S E** — Mean square error (MSE) is the squared norm of the difference between the data and the signal or image approximation divided by the number of elements. The MSE is defined by:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m=1}\sum_{j=0}^{n=1}|X(i,j) - X_c(i,j)|^2$$

- **Max Error** — Maximum error is the maximum absolute squared deviation in the signal or image approximation.
- **L2-Norm Ratio** — L2-norm ratio is the ratio of the squared L2-norm of the signal or image approximation to the input signal or image. For images, the image is reshaped as a column vector before taking the L2-norm
- **P S N R** — Peak signal-to-noise ratio (PSNR) is a measure of the peak error in decibels. PSNR is meaningful only for data encoded in terms of bits per sample or bits per pixel. The higher the PSNR, the better the quality of the compressed or reconstructed image. Typical values for lossy compression of an image are between 30 and 50 dB. When the PSNR is greater than 40 dB, then the two images are indistinguishable. The PSNR is defined by:

$$PSNR = 10 \cdot \log_{10}\left(\frac{255^2}{MSE}\right)$$

- **B P P** — Bits per pixel ratio (BPP) is the number of bits required to store one pixel of the image. The BPP is the compression ratio multiplied by 8, assuming one byte per pixel (8 bits).
- **Comp Ratio** — Compression ratio is ratio of the number of elements in the compressed image divided by the number of elements in the original image, expressed as a percentage.

Effectively compressing an image is a balance between how well the image will be preserved and the amount of data stored. Wavelet based compression can provide a highly effective methods of data compression, and has been notably implemented by the FBI for storing fingerprints [8].

To achieve good compression, MATLAB offers a number of compression methods, seen in the table below [3], which implement different compression algorithms. For this project we will compare SPIHT and WDR methods, as they will provide good representations of two distinct methods of wavelet compression [9]. Most of the other methods offered are a variant or very similar to these two methods (spiht: ezw, stw, spiht_3d; wdr: aswdr). MATLAB also provides three versions of MMC compression (Morphing Match Chain), a relatively newer compression method [4][5], which we will not be including in this project.

MATLAB Name	Compression Method Name
'ezw'	Embedded Zerotree Wavelet
'spiht'	Set Partitioning In Hierarchical Trees
'stw'	Spatial-orientation Tree Wavelet
'wdr'	Wavelet Difference Reduction
'aswdr'	Adaptively Scanned Wavelet Difference Reduction
'spiht_3d'	Set Partitioning In Hierarchical Trees 3D for truecolor images

Using 5 as our baseline level of composition, we will test the wavelet and compression method combinations with 6 and 10 loops.

Table 1

Wavelet	Level	Compression	Loops	Bits Per Pixel	Compression Ratio (%)	MSE
bio4.4	5	spiht	6	0.091393	0.38	329.6
bio4.4	5	spiht	10	1.1974	4.99	17.6
bio4.4	5	wdr	6	0.17436	0.73	156
bio4.4	5	wdr	10	3.5147	14.64	5.727
db2	5	spiht	6	0.096542	0.40	321.8
db2	5	spiht	10	1.4017	5.84	18
db2	5	wdr	6	0.1944	0.81	158.7
db2	5	wdr	10	4.1631	17.35	5.894
haar	5	spiht	6	0.097397	0.41	369.6
haar	5	spiht	10	1.6714	6.96	21.89
haar	5	wdr	6	0.20642	0.86	177.5
haar	5	wdr	10	4.8882	20.37	7.939

If we simply look at bits per pixel (BPP) and the compression ratio (CR), it would appear that bior4.4 with level 5 decomp and 6 loops of Spiht compression gives the best result.

Wavelet	**Level**	**Compression**	**Loops**	**Bits Per Pixel**	**Compression Ratio (%)**
bio4.4	5	spiht	6	0.091393	0.38

However, when you look at the actual compressed image, the quality is clearly quite poor.

The reason for this can be seen if we go back and look at the mean square error (MSE), which it turns out is very high.

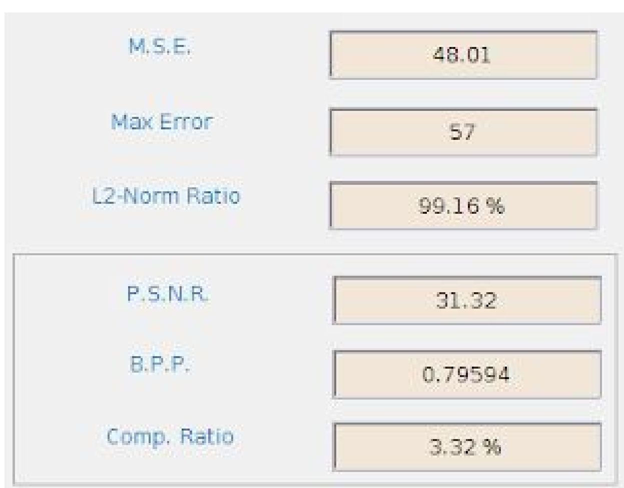
Wavelet	Level	Compression	Loops	Bits Per Pixel	Compression Ratio (%)	MSE
bio4.4	5	spiht	6	0.091393	0.38	329.6

This illustrates the importance of considering all factors in compression, such as MSE, and not just the BPP and CR. With that in mind, a review of the output from Table 1 indicates that none of our compression tests performed well, overall. Good BP and CR tests had large MSE, while lower MSE tests had poor BPP and CR. To further explore how we could achieve better compression performance, we can adjust the level of decomposition used. The Haar wavelet was chosen for this portion of the tests, with the results below in Table 2.

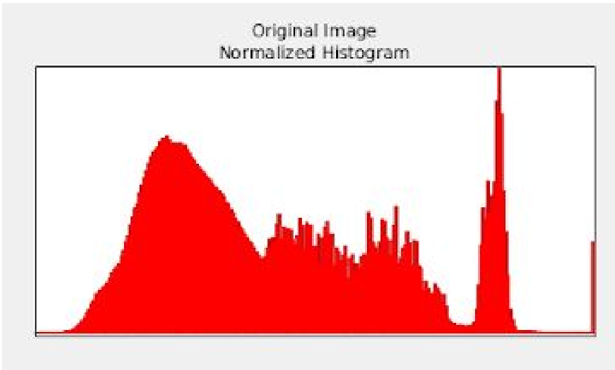
Table 2

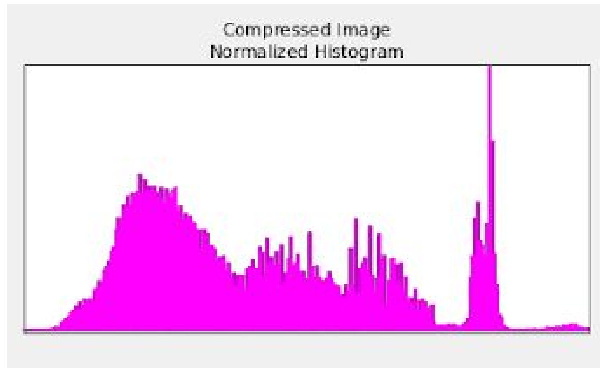
Wavelet	Level	Compression	Loops	Bits Per Pixel	Compression Ratio (%)	MSE
haar	5	spiht	6	0.097397	0.41	369.6
haar	5	spiht	10	1.6714	6.96	21.89
haar	8	spiht	10	0.14017	0.58	193.8
haar	8	spiht	12	0.79594	3.32	48.01
haar	10	spiht	10	0.040955	0.17	359.1
haar	10	spiht	12	0.34562	1.44	101.9

As you can see, the Haar Wavelet with level 8 decomposition using Spiht Compression (12 loops) has a good balance of bits per pixel, compression ratio, and MSE.



Looking at the results in detail we see that the compressed image had a very similar histogram to the original image.





And a perceptual check of the resulting image confirms it is still decent quality.

The full set of results, compressed outputs, and original input images used in this project can be found on GitHub (https://github.com/sgoodm/apsc608/tree/master/project_02).

References:

[1]
https://www.mathworks.com/help/wavelet/ref/waveletanalyzer-app.html

[2]
https://www.mathworks.com/help/wavelet/ug/two-dimensional-true-compression.html#f4-110904
9

[3]
https://www.mathworks.com/help/wavelet/ug/wavelet-compression-for-images.html

[4]
http://fastcompression.blogspot.com/p/mmc-morphing-match-chain.html

[5]
http://cbloomrants.blogspot.com/2011/09/09-23-11-morphing-matching-chain.html

[6]
http://wavelets.pybytes.com/wavelet/bior4.4/

[7]
http://wavelets.pybytes.com/wavelet/haar/

[8] M Sifuzzaman, M.R. Islam, M.Z. Ali. Application of Wavelet Transform and its Advantages Compared to Fourier Transform. Journal of Physical Sciences, Vol. 13, 2009, 121-134
https://pdfs.semanticscholar.org/63f4/22a5095dd2e4e190ce7f589cab75232402f1.pdf

[9]
http://shodhganga.inflibnet.ac.in/jspui/bitstream/10603/143312/9/09_chapter%203.pdf

[10]
http://wavelets.pybytes.com/wavelet/db2/