Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 3159-0040
Vol. 2 Issue 6, June - 2015

Wavelet Analysis Applied In Image Denoising Using MATLAB

[1] **Brikena Xhaja**, (PhD Student)

Department of Mathematics
Faculty of Mathematics' and Physics' Engineering, Polytechnic University of Tirana, Albania
brikena_xhaja11@yahoo.com

[1]**Ligor Nikolla**, (Prof. As.)

Department of Mathematics
Faculty of Mathematics' and Physics' Engineering, Polytechnic University of Tirana, Albania
lnikolla@yahoo.com

***Abstract*—Wavelet analysis has drawn a great attention from mathematical sciences in various disciplines. The subject of wavelet is to create a common link between mathematicians, physicists, and electrical engineers. Wavelets are mathematical functions that cut up data into different frequency components, and then study each component with a resolution matched to its scale. They have advantages over traditional Fourier methods in analyzing physical situations where the signal contains discontinuities and sharp spikes. Wavelets were developed independently in the fields of mathematics, quantum physics, electrical engineering, and seismic geology. Interchanges between these fields have led to many new wavelet applications such as image compression, turbulence, human vision, radar, and earthquake prediction. This paper introduces wavelets to the interested technical person outside of the digital signal processing field. It is described the history of wavelets beginning with Fourier, compare wavelet transforms with Fourier transforms, state properties and other special aspects of wavelets, and finish with some interesting applications such as image compression, musical tones, and de-noising noisy data.**

Keywords—wavelets; signal processing; orthogonal basis functions; wavelet applications; denoising;

I. INTRODUCTION

The wavelet transform is an expansion that decomposes a given signal in a basis of orthogonal functions. In this sense, we can set a complete analogy with the Fourier Transform. While the Fourier Transform uses periodic, smooth and unlimited basis functions (i.e., sines and cosines), the wavelet transform uses non-periodic, non-smooth and finite support basis functions, allowing a much more meaningful representation through multi-resolution analysis.

Wavelets are functions that satisfy certain mathematical requirements and are used in representing data or other functions. Approximation using superposition of functions has existed since the early 1800's, when Joseph Fourier discovered that he could superpose sines and cosines to represent other functions. However, in wavelet analysis, the *scale* that we use to look at data plays a special role. Wavelet algorithms process data at different *scales* or *resolutions.* If we look at a signal with a large "window," we would notice gross features. Similarly, if we look at a signal with a small "window," we would notice small features. More clearly, we can say that the result in wavelet analysis is to see both the forest and the trees. This makes wavelets interesting and useful. For many decades, scientists have wanted more appropriate functions than the sines and cosines which comprise the bases of Fourier analysis, to approximate choppy signals. By their definition, these functions are non-local (and stretch out to infinity). They therefore do a very poor job in approximating sharp spikes. But with wavelet analysis, we can use approximating functions that are contained neatly in finite domains. Wavelets are well-suited for approximating data with sharp discontinuities. Other applied fields that are making use of wavelets include astronomy, acoustics, nuclear engineering, sub-band coding, signal and image processing, neurophysiology, music, magnetic resonance imaging, speech discrimination, optics, fractals, turbulence, earthquake-prediction, radar, human vision, and pure mathematics applications such as solving partial differential equations.

II. FOURIER ANALYSIS

A. *Fourier transforms*

Fourier's representation of functions as a superposition of sines and cosines has become ubiquitous for both the analytic and numerical solution of differential equations and for the analysis and treatment of communication signals. Fourier and wavelet analysis have some very strong links.

The Fourier transform's utility lies in its ability to analyze a signal in the time domain for its frequency content. The transform works by first translating a function in the time domain into a function in the frequency domain. The signal can then be analyzed for its frequency content because the Fourier coefficients of the transformed function represent the contribution of each sine and cosine function at each

frequency. An inverse Fourier transform does just what you'd expect; transform data from the frequency domain into the time domain.
The Fourier transform decomposes a signal in complex exponential functions at different frequencies. The equations used in the decomposition and reconstruction part will be given below:

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \qquad (1)$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{j\omega t} d\omega \qquad (2)$$

Where, t stands for time, $\omega = 2\pi f$ for frequency, x denotes the signal in the time domain and X denotes the signal in the frequency domain (also known as the spectrum of the original signal).
Projecting the signal on complex exponentials leads to good frequency analysis, but no time localization. The poor time localization is the main disadvantage of the Fourier transform, making it not suitable for all kind of applications.

B. *Wavelet Transform*

Based in the limitations of the Fourier Transform (poor time localization) Grossman and Morlet gave in 1984 the formulation of the Continuous Wavelet Transform. Unlike, the Fourier transform that decomposes the signal into a basis of complex exponentials, the Wavelet Transform decomposes the signal over a set of dilated and translated wavelets [1].

This difference confers to the wavelet transform the advantage of performing a multiresolution analysis, meaning that it processes different frequencies in a different way. By using this technique, the time resolution is increased when we analyze a high frequency portion of the signal, and the frequency localization is increased when analyzing a low-frequency part of the same signal[2][3]. This type of analysis is suitable for signals that have both low-frequency components with long time duration and high-frequency components with short time duration, which is the case of most signals. If we consider a function (signal) $x \in L^2(R)$ and for analysis we use the mother wavelet $\psi(t)$, with its scaled and translated versions of

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right), \qquad (3)$$

Now, we can write the wavelet transform of $x(t)$ at time u and scale s as:

$$Wx(u,s) = \langle x, \psi_{u,s} \rangle = \int_{-\infty}^{+\infty} x(t) \frac{1}{\sqrt{s}} \psi^*\left(\frac{t-u}{s}\right) dt \qquad (4)$$

By looking at the equation above we can conclude that the Wavelet Transform can be seen as a convolution between the signal to be analyzed and the reverse function, $\overline{\psi}_s(t) = \frac{1}{\sqrt{s}} \psi * \left(-\frac{t}{s}\right)$ derived from the Mother Wavelet.

III. Comparison of Wavelet Transform with Fourier Transform

The fast Fourier transform (FFT) and the discrete wavelet transform (DWT) are both linear operations that generate a data structure that contains $\log_2 n$ segments of various lengths, usually filling and transforming it into a different data vector of length 2^n .

The mathematical properties of the matrices involved in the transforms are similar as well. The inverse transform matrix for both the FFT and the DWT is the transpose of the original. As a result, both transforms can be viewed as a rotation in function space to a different domain. For the FFT, this new domain contains basis functions that are sines and cosines. For the wavelet transform, this new domain contains more complicated basis functions called wavelets, mother wavelets, or analyzing wavelets.

Both transforms have another similarity. The basic functions are localized in frequency, making mathematical tools such as power spectra (how much power is contained in a frequency interval) and scalegrams useful at picking out frequencies and calculating power distributions.

The most interesting dissimilarity between these two kinds of transforms is that individual wavelet functions are *localized in space.* Fourier sine and cosine functions are not. This localization feature, along with wavelets' localization of frequency, makes many functions and operators using wavelets "sparse" when transformed into the wavelet domain. This sparseness, in turn, results in a number of useful applications such as data compression, detecting features in images, and removing noise from time series.

IV. Application in the Image Processing

In diverse fields from planetary science to molecular spectroscopy, scientists are faced with the problem of recovering a true signal from incomplete, indirect or noisy data. Wavelets can help to solve this problem, through a technique called *wavelet shrinkage and* *thresholding methods that* David Donoho has worked on for several years [5].

The technique works in the following way. When you decompose a data set using wavelets, you use filters that act as *averaging* filters and others that produce *details* [6]. Some of the resulting wavelet coefficients correspond to details in the data set. If the details are small, they might be omitted without substantially affecting the main features of the data set. The idea of *thresholding,* then, is to set to zero all coefficients that are less than a particular threshold. These coefficients are used in an inverse wavelet transformation to reconstruct the data set.

The basic wavelet denoising problem consists in, goven an input noisy image, dividing all its wavelet coefficients into relevant (if greater than a critical value) or irrelevant (if less than a critical value) and then process the coefficients from each one of these groups by certain specific rules.

Filtering and convolution are applied to achieve the signal decomposition in classical wavelet transform. In 1986, Meyer and Mallat found that the orthonormal wavelet decomposition and reconstruction can be implemented in the multi-resolution signal analysis framework. Multi-resolution analysis is now a standard method for constructing the orthonormal wavelet bases.

In this paper we will give an application of image denoising using wavelet analysis. This is done using "wavelet tool" in MATLAB. We used Simulink in MATLAB to transform an image with .jpg or .tiff format (in our case it is in the format .jpg) to a signal and then we can process over this signal.

Below we will give an example of an image which is saved in a folder in the .jpg format and then we loaded this image in MATLAB where now it is saved as a matrix. The better the quality of the image larger is the size of the matrix. Below is given the original image and then the matrixes that correspond for this image Fig. 1, [7][8].

Figure 1. Original image

```
>> [m,n,k]=size(A)
m = 2776
n = 1971
k =  3
        subplot(131); imagesc(A(:,:,1)); title('R');
        subplot(132); imagesc(A(:,:,2)); title('G');
        subplot(133); imagesc(A(:,:,3)); title('B');
```

The function rgb2gray(RGB) converts the true color image RGB (Red Blu Green) to the grayscale intensity image Figure 2 and Figure 3. The rgb2gray function converts RGB images to grayscale by eliminating the hue and saturation information while retaining the luminance.

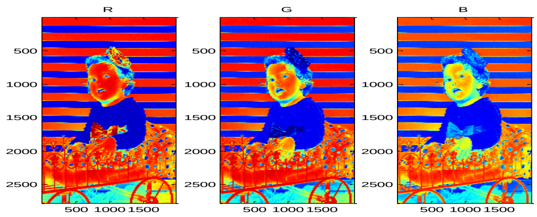

Figure 2. RBG image

```
>> colormap grey;
```

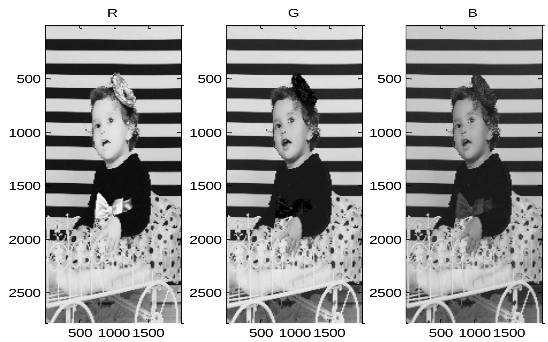

Figure 3. RGB image converted to greyscale image

```
B = double(A);
B(:,:,3) = 3*B(:,:,3);
B = uint8(B);
imshow(B);
```

Figure 4.

Wiener deconvolution can be useful when the point-spread function and noise level are known or can be estimated.

```
>> A=imread('amisi.jpg');
```

```
>> LEN=21;
>> THETA=11;
>> PSF=fspecial('motion',LEN,THETA);
>> blurred=imfilter(A,PSF,'conv','circular');
>> imshow(blurred)
```

Now, we will simulate a blurred image that might get from camera motion. This is done creating a point-spread function, PSF, corresponding to the linear motion across 31 pixels (LEN=31), at an angle of 11 degrees (THETA=11). To simulate the blur, convolve the filter with the image using *imfilter*.

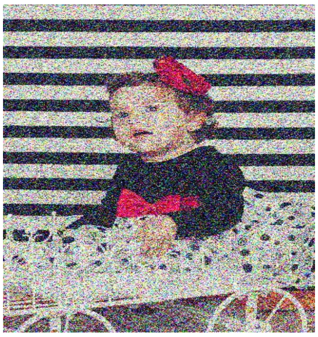

Figure 5. a) Original Image; b) Simulated blur image

Now, we will have to restore the Blurred Image.

The simplest syntax for deconvwnr is deconvwnr(A, PSF, NSR), where A is the blurred image, PSF is the point-spread function, and NSR is the noise-power-to-signal-power ratio. The blurred image formed in the Figure 5. b) has no noise, so we'll use *0* for NSR.

```
>> wnr1=deconvwnr(blurred, PSF,0);

>> imshow(wnr1)
```

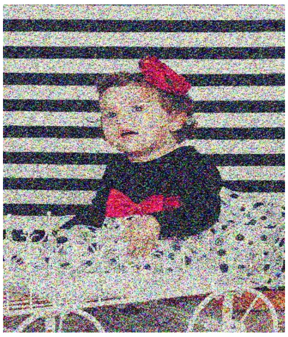

Figure 6. Blurred image without noise

```
noise_mean = 0;
noise_var = 0.0001;
blurred_noisy = imnoise(blurred, 'gaussian', ...
                noise_mean, noise_var);
imshow(blurred_noisy)
```

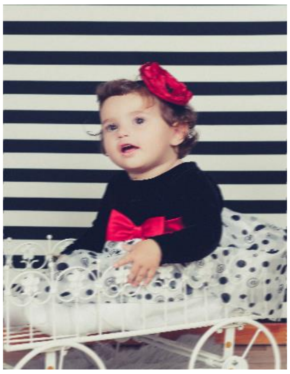

Figure 7. Blurred noisy image

In our first restoration attempt, we'll tell deconvwnr that there is no noise (NSR = 0). When NSR = 0, the Wiener restoration filter is equivalent to an ideal inverse filter. The ideal inverse filter can be extremely sensitive to noise in the input image, as the next image shows:

Figure 8. Restored blurred noisy image. 1st attempt

The noise was amplified by the inverse filter to such a degree that only the barest hint of the girl's shape is visible.

In our second attempt we supply an estimate of the noise-power-to-signal-power ratio.

```
signal_var = var(I(:));
wnr3 = deconvwnr(blurred_noisy, PSF, noise_var /
signal_var);
imshow(wnr3)
```

If we pass a uint8 image to imfilter, it will quantize the output in order to return another uint8 image.

```
>>blurred_quantized = imfilter(I, PSF, 'conv', 'circular');
>>class(blurred_quantized)
```

Again, we'll try first telling deconvwnr that there is no noise.

```
>>wnr4 = deconvwnr(blurred_quantized, PSF, 0);
>>imshow(wnr4)
>>title('Restoration of blurred, quantized image using NSR = 0');
```

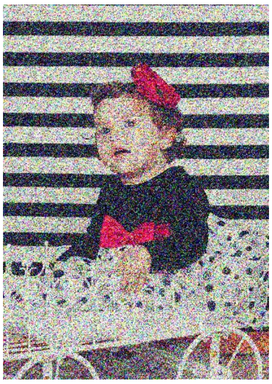

Figure 9. Restored the blurred, quantized image

Next, we supply an NSR estimate to deconvwnr.

Figure 10. Restored blurred image

```
>>uniform_quantization_var = (1/256)^2 / 12;
>>signal_var = var(im2double(I(:)));
>>wnr5 = deconvwnr(blurred_quantized, PSF, ...
  uniform_quantization_var / signal_var);
>>imshow(wnr5)
>>title('Restoration of Blurred, Quantized Image Using Computed
NSR');
```

CONCLUSIONS

Image denoising is a required preprocessing step in several applications in image processing and pattern recognition. Therefore, estimating a signal that is degraded by noise has been of interests to a wide community of researchers. The goal of image denoising is to remove the noise as much as possible, while retaining important features, such as edges and fine details. Traditional image denoising have been based on linear filtering, where most usual choices are Winner filtering.

Application with both simulated and real image data provided good results. The obtained results indicated a significant improvement in the denoising performance, showing the effectiveness of wavelet transform used in this application.

Future works may include the use and investigation of more wavelet decomposition levels, other kinds of wavelet transforms, as well as the filtering of other kinds of noise.

REFERENCES

[1] Meyer, Y.; *Wavelets: Algorithms and Applications,* Society for Industrial and Applied Mathematics, Philadelphia, 1993.

[2] Kaiser, G., *A Friendly Guide to Wavelets,* Birkhauser, Boston, 1994.

[3] Daubechies, I. "Orthonormal Bases of Compactly Supported Wavelets," *Comm. Pure Appl. Math.,* Vol 41, 1988.

[4] Wickerhauser, V., *Adapted Wavelet Analysis from Theory to Software,* AK Peters, Boston, 1994, pp. 213-214, 237, 273-274, 387.

[5] Donoho, D., "Nonlinear Wavelet Methods for Recovery of Signals, Densities, and Spectra from Indirect and Noisy Data," *Different Perspectives on Wavelets, Proceeding of Symposia in Applied Mathematics,* Vol 47, I. Daubechies ed. Amer. Math. Soc., Providence, R.I., 1993, pp. 173-205.

[6] Vidakovic, B. , Muller, P., "Wavelets for Kids," 1994, unpublished.

[7] Jansen, A., Noise reduction by wavelet thresholding, Springer – Verlag, 2001.

[8] Nasri, M., Nezamabady – pour, H. Image denoising in the wavelet domain using a new adaptive thresholding function, 2009.