Report - Wavelets

There multiple methods of denoising a signal using wavelet. Depending on the need we can manually threshold and levels of decomposition required. If denoising is not happening in real time then we can manually choose threshold and levels of decomposition. However for real time noise removing application automatic selection of parameters of necessary. Both the cases are discussed that are as follow.

Wavelet Interval Dependent Denoising:

This type of wavelet denoising is used where different regions of the signal contains different levels of noise i.e. baseline wandering, variance in amplitude noise of detailed coefficients.

1. Denoising Using a Single Interval

Let’s say a signal SIG with length L contains some noise X. One way is to select threshold and parameters by using hit and trial method. Finally selecting the threshold that outputs minimum RMSE and SNR. In this case a signal interval equal to full length of the signal is selected with one threshold value. Denoising function is applied and then results are compared.

For selection of a single interval global threshold we can use two of the following methods:

**a) Ddencmp function from MATLAB library**

This function decompose a signal using Deubecies 1 wavelet. Then calculating threshold level of signal for removing noise by calculating median of the absolute of the signal and then dividing by a constant 0.674.

thr = (median(abs(signal)))/0.674

**b) Manually calculate threshold**

Without using the MATLAB multiple wavelet types can be tested for calculating global threshold of a signal and then selecting the one with minimum max SNR and minimum RMSE from the output

* 1. Calculate discrete wavelet transform using any wavelet type i.e. sym1, sym2, sym3, db1 db4.
  2. Calculate variance of the noise by taking median of absolute of detailed coefficient and then dividing the result by 0.674.
  3. Finally taking square root and log of the variance in step 2 to calculate global threshold value.

[Approximate, Detailed] = dwt(x,'sym4');

Noise level = median (abs (D))/0.674;

Global threshold = sqrt (2\*log(length(x)))\*noiselev;

So in this method a single interval with single threshold is used to denoise the signal. Whereas ‘wdencmp’ function is used for denoising the signal. The function decomposes the signal into its approximate and detailed coefficients. Thresholding is applied to detailed components and signal is reconstructed. Results are mentioned in table below. SNR of each wavelet/level is calculated. It can see that after level 5 with sym4 wavelet there is no significant decrease in SNR. So the best in case would be to select level 5 with sym4 for single interval wavelet denoising.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Wavelet/Level** | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **Haar** | -3.518 | -2.959 | -2.832 | -2.792 | -2.776 | -2.768 | -2.763 |
| **Sym1** | -3.518 | -2.959 | -2.832 | -2.792 | -2.776 | -2.768 | -2.763 |
| **Sym2** | -3.358 | -2.888 | -2.809 | -2.786 | -2.775 | -2.768 | -2.764 |
| **Sym3** | -3.269 | -2.864 | -2.803 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Sym4** | -3.235 | -2.853 | -2.801 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Sym5** | -3.189 | -2.849 | -2.801 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Sym6** | -3.173 | -2.847 | -2.800 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Sym7** | -3.136 | -2.845 | -2.800 | -2.785 | -2.775 | -2.769 | -2.763 |
| **Sym8** | -3.143 | -2.845 | -2.800 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Db1** | -3.518 | -2.959 | -2.832 | -2.792 | -2.776 | -2.768 | -2.763 |
| **Db2** | -3.358 | -2.888 | -2.809 | -2.786 | -2.775 | -2.768 | -2.764 |
| **Db3** | -3.269 | -2.864 | -2.803 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Db4** | -3.219 | -2.854 | -2.802 | -2.785 | -2.775 | -2.768 | -2.763 |
| **Db5** | -3.191 | -2.849 | -2.801 | -2.785 | -2.775 | -2.768 | -2.763 |
| **Db6** | -3.172 | -2.847 | -2.801 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Db7** | -3.154 | -2.845 | -2.800 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Db8** | -3.135 | -2.845 | -2.800 | -2.785 | -2.775 | -2.769 | -2.764 |

`Table 1 : Denoising Using Wdencmp Single Interval

1. **Level Independent Threshold Stationary Wavelet Transform (SWT)**

Testing SWT for the purpose of removing noise is because standard wavelet transform can’t observe changing frequency (first increasing to singularity point and then decreasing) whereas SWT can observe changing frequency i.e. chirp signal. Variance in detailed components is readily available in accelerometer data so SWT can be tested for good results. Following are the results. Maximum SNR is achieved at Level 5 using db1 wavelet.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Wavelet/Level** | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **Haar** | -3.087 | -2.826 | -2.791 | -2.778 | -2.770 | -2.765 | -2.761 |
| **Sym1** | -3.087 | -2.826 | -2.791 | -2.778 | -2.770 | -2.765 | -2.761 |
| **Sym2** | -3.032 | -2.829 | -2.795 | -2.781 | -2.772 | -2.766 | -2.763 |
| **Sym3** | -3.014 | -2.831 | -2.796 | -2.782 | -2.773 | -2.767 | -2.763 |
| **Sym4** | -3.007 | -2.833 | -2.797 | -2.783 | -2.773 | -2.767 | -2.763 |
| **Sym5** | -3.005 | -2.835 | -2.797 | -2.783 | -2.774 | -2.767 | -2.763 |
| **Sym6** | -3.004 | -2.835 | -2.797 | -2.783 | -2.774 | -2.768 | -2.763 |
| **Sym7** | -3.005 | -2.836 | -2.797 | -2.784 | -2.774 | -2.768 | -2.763 |
| **Sym8** | -3.005 | -2.837 | -2.798 | -2.784 | -2.774 | -2.768 | -2.763 |
| **Db1** | -3.087 | -2.826 | -2.791 | -2.778 | -2.770 | -2.765 | -2.761 |
| **Db2** | -3.032 | -2.829 | -2.795 | -2.781 | -2.772 | -2.766 | -2.763 |
| **Db3** | -3.014 | -2.831 | -2.796 | -2.782 | -2.773 | -2.767 | -2.763 |
| **Db4** | -3.007 | -2.833 | -2.797 | -2.783 | -2.773 | -2.767 | -2.763 |
| **Db5** | -3.005 | -2.835 | -2.797 | -2.783 | -2.774 | -2.767 | -2.763 |
| **Db6** | -3.004 | -2.835 | -2.797 | -2.783 | -2.774 | -2.768 | -2.763 |
| **Db7** | -3.005 | -2.836 | -2.797 | -2.784 | -2.774 | -2.768 | -2.763 |
| **Db8** | -3.005 | -2.837 | -2.798 | -2.784 | -2.774 | -2.768 | -2.763 |

Table 2 Single Interval Stationary Wavelet Transform

1. **Advanced Automatic Interval Dependent Denoising**

In this technique, threshold and intervals in each level are computed automatically based on the intensity and variance of the noise in each detailed coefficient of the signal.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Wavelet/Level** | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **Haar** | -3.518 | -2.959 | -2.832 | -2.792 | -2.776 | -2.768 | -2.763 |
| **Sym1** | -3.518 | -2.959 | -2.832 | -2.792 | -2.776 | -2.768 | -2.763 |
| **Sym2** | -3.358 | -2.888 | -2.809 | -2.786 | -2.775 | -2.768 | -2.764 |
| **Sym3** | -3.269 | -2.864 | -2.803 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Sym4** | -3.235 | -2.853 | -2.801 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Sym5** | -3.189 | -2.849 | -2.801 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Sym6** | -3.173 | -2.847 | -2.800 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Sym7** | -3.136 | -2.845 | -2.800 | -2.785 | -2.775 | -2.769 | -2.763 |
| **Sym8** | -3.143 | -2.845 | -2.800 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Db1** | -3.518 | -2.959 | -2.832 | -2.792 | -2.776 | -2.768 | -2.763 |
| **Db2** | -3.358 | -2.888 | -2.809 | -2.786 | -2.775 | -2.768 | -2.764 |
| **Db3** | -3.269 | -2.864 | -2.803 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Db4** | -3.219 | -2.854 | -2.802 | -2.785 | -2.775 | -2.768 | -2.763 |
| **Db5** | -3.191 | -2.849 | -2.801 | -2.785 | -2.775 | -2.768 | -2.763 |
| **Db6** | -3.172 | -2.847 | -2.801 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Db7** | -3.154 | -2.845 | -2.800 | -2.785 | -2.775 | -2.768 | -2.764 |
| **Db8** | -3.135 | -2.845 | -2.800 | -2.785 | -2.775 | -2.769 | -2.764 |

Table 3 Automatic Computations of Parameter Wavelets

1. **Wavelet Packet Filtering**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Wavelet/Level** | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| **Haar** | -3.518 | -2.959 | -2.959 | -2.959 | -2.776 | -2.776 | -2.763 |
| **Sym1** | -3.518 | -2.959 | -2.959 | -2.959 | -2.776 | -2.776 | -2.763 |
| **Sym2** | -3.358 | -2.888 | -2.809 | -2.786 | -2.775 | -2.775 | -2.764 |
| **Sym3** | -3.269 | -2.864 | -2.803 | -2.785 | -2.785 | -2.775 | -2.764 |
| **Sym4** | -3.235 | -2.853 | -2.801 | -2.785 | -2.775 | -2.775 | -2.764 |
| **Sym5** | -3.189 | -2.849 | -2.801 | -2.785 | -2.775 | -2.775 | -2.764 |
| **Sym6** | -3.173 | -2.847 | -2.800 | -2.785 | -2.775 | -2.775 | -2.764 |
| **Sym7** | -3.136 | -2.845 | -2.800 | -2.785 | -2.775 | -2.775 | -2.763 |
| **Sym8** | -3.143 | -2.845 | -2.800 | -2.785 | -2.775 | -2.775 | -2.764 |
| **Db1** | -3.518 | -2.959 | -2.959 | -2.959 | -2.776 | -2.776 | -2.763 |
| **Db2** | -3.358 | -2.888 | -2.809 | -2.786 | -2.775 | -2.775 | -2.764 |
| **Db3** | -3.269 | -2.864 | -2.803 | -2.785 | -2.785 | -2.775 | -2.764 |
| **Db4** | -3.219 | -2.854 | -2.802 | -2.785 | -2.785 | -2.775 | -2.763 |
| **Db5** | -3.191 | -2.849 | -2.801 | -2.785 | -2.785 | -2.775 | -2.763 |
| **Db6** | -3.172 | -2.847 | -2.801 | -2.785 | -2.785 | -2.775 | -2.764 |
| **Db7** | -3.154 | -2.845 | -2.800 | -2.785 | -2.785 | -2.775 | -2.764 |
| **Db8** | -3.135 | -2.845 | -2.800 | -2.785 | -2.785 | -2.775 | -2.764 |

After carefully observing calculated SNR, it can be seen that there is no significant SNR decrease than -2.770 db. So this in the case the preferred method from wavelet is to use the one having minimum mathematical computations with respect to hardware requirements.

Questions/Answers

Q # 1: why is the constant 0.674 used? Can this be calculated?

Answer: This constant is taken and used from different references to calculate variance of the noise

<https://www.egr.msu.edu/~aviyente/ecelecture8.pdf> --> Slide 13

<http://www.pe.org.pl/articles/2013/5/13.pdf> --> Page 2

Questions 2: Did you investigate only one channel or are your findings consistent with all channels?

Answer: Yes! Code has been tested for the channels.

Questions 3: Did you read/study any papers for this, or is it only based on the Documentation of Matlab? Can you provide links to the papers?

Answer: Yes! I took some guesses from few papers. MATLAB documentation was very helpful for comparing results and methods. Some of reference links are as follow

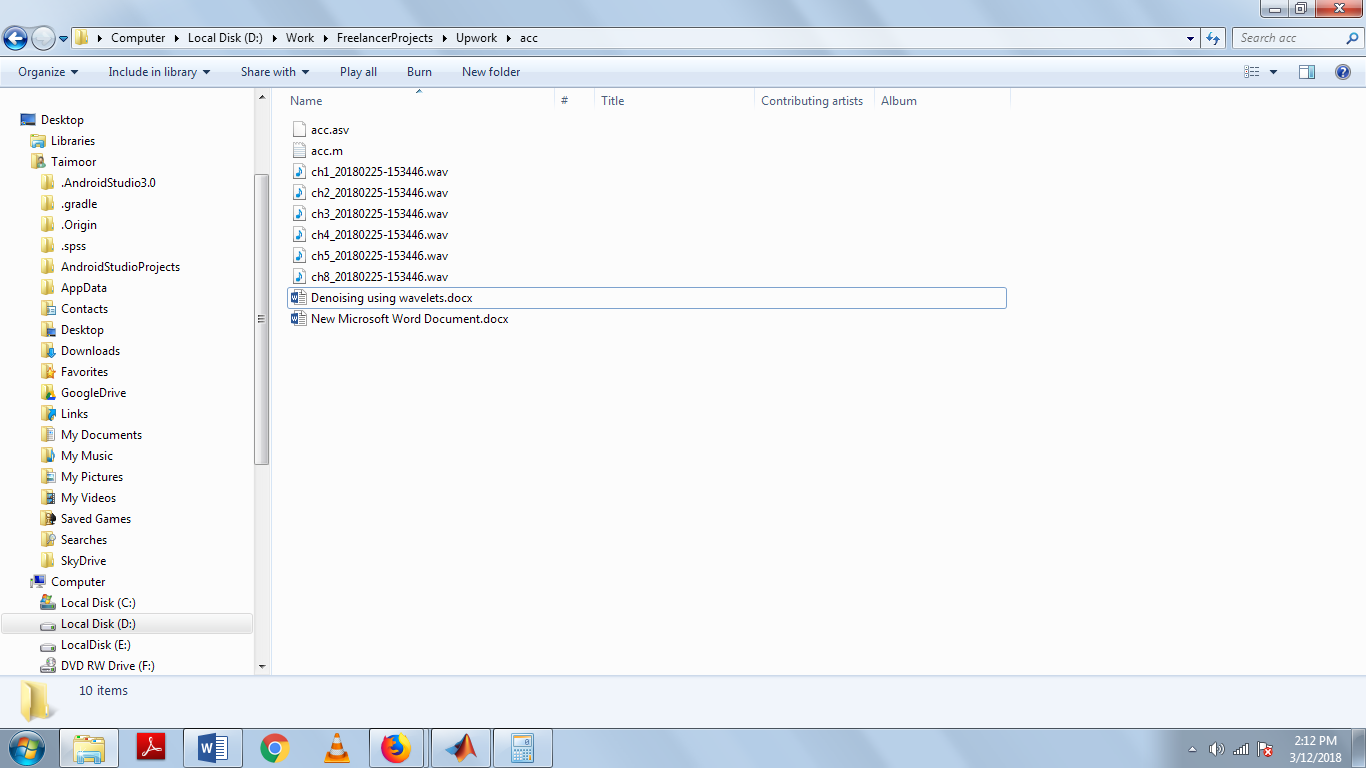
<http://www.pe.org.pl/articles/2013/5/13.pdf>

*iopscience.iop.org/article/10.1088/0957-0233/17/4/019*

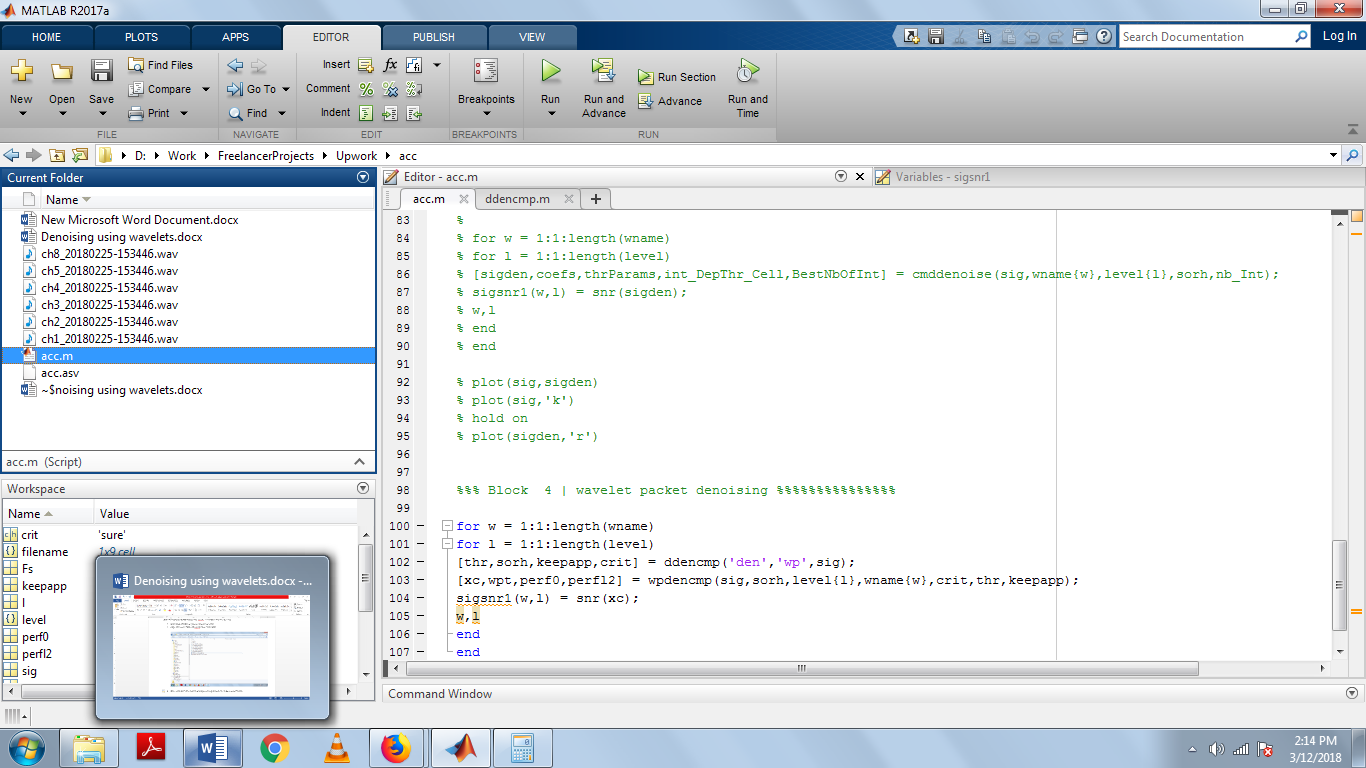
Question 4: Can you provide short instructions on how to run your code in Matlab? It would help me to validate the code quicker.

Answer: There is a single Matlab file “acc.m”. It contains 4 blocks of code

1. First make a folder anywhere in your hard disk
2. Copy all the wav files and acc.m file in that folder



1. Now open MATLAB, look at workspace and give path to newly created folder.



1. As mentioned there are four blocks of code in the m file with different wavelet techniques implement. If you want run block # 1 simply uncomment that (block # 1) part of the code and press run as shown in screenshots.

I hope all of your questions are answered elaborately.