

Speech and transform ii computer science



In this project the experimenter will seek to design and implement techniques in order to denoise a noisy audio signal using the MATLAB software and its functions, a literature review will be done and summarized to give details of the contribution to the area of study. Different techniques that have been used in the audio and speech processing procedure will be analyzed and studied. The implementation will be done using MATLAB version 7. 0.

Introduction

The Fourier analysis of a signal can be used as a very powerful tool; it can perform the functions of obtaining the frequency component and the amplitude component of signals. The Fourier analysis can be used to analyze components of stationary signals, these are signals that repeat, signals that are composed of sine and cosine components, but in terms of analyzing non stationary signals, these are signals that have no repetition in the region that is sampled, the Fourier transform is not very efficient. Wavelet transform on the other hand allows for these signals to be analyzed. The basic concept behind wavelets is that a signal can be analyzed by splicing it into different components and then these components are studied individually. In terms of their frequency and time, in terms of Fourier analysis the signal is analyzed in terms of its sine and cosine components but when a wavelet approach is adapted then the analysis is different, the wavelet algorithm employs a process and analyzed the data on different scales and resolution as compared to Fourier analysis. In using the wavelet analysis, a type of wavelet, referred to as being the mother wavelet is used as the main wavelet type for analysis; analysis is then performed from the mother wavelet that is

of higher frequency. From the Fourier analysis the frequency analysis of the signal is done with a simplified form of the mother wavelet, from the wavelet components that are achieved via this process further analysis can be done on these coefficients. Haar wavelet types are very compact and this is one of their defining features, its compact ability, as the interval gets so large it then starts to vanish, but the Haar wavelets have a major limiting factor they are not continuously differentiable. In the analysis of a given signal the time domain component can be used in the analysis of the frequency component of that signal, this concept is the Fourier transform, where a signal component is translated to the frequency domain from a time domain function, the analysis of the signal for its frequency component can now be done, and based on Fourier analysis this is possible because this analysis incorporates the cosine and sine of the frequency. Based on the Fourier transform a finite set of sampled points are analyzed this results in the discrete Fourier transforms, these sample points are typical to what the original signal looks like, to gather the approximate function of a sample, and the gathering of the integral, by the implementation of the discrete Fourier transforms. This is realized by the use of a matrix, the matrix contains an order of the total amount of points of sample, A the problem encountered worsens as the number of samples are increased. If there is uniform spacing between the samples then it is possible to factor in the Fourier matrix into the, multiplication of a few matrices, the results of this can be subjected to a vector of an order of the form $m \log m$ operations, the result of this know as the Fast Fourier Transform. Both Fourier transforms mentioned above are linear transforms. The transpose of the FFT and the DWT is what is referred to as the inverse transform matrix and they can be

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

cosine and sine, but in the wavelet domain more complex mother wavelet functions are formed. The domain of analysis in the Fourier transforms are the sine and cosine, but as it regards to wavelets there exist a more complex domain function called wavelets, mother wavelets are formed. The functions are localized functions, and are set in the frequency domain, can be seen in the power spectra. This proves useful in finding the frequency and power distribution. Based on the fact that wavelet transforms are transforms that are localized as compared to Fourier functions that are not, the Fourier function being mentioned are the sine and cosine, this feature of wavelet makes it a useful candidate in the purpose of this research, this feature of wavelets makes operations using wavelets transform ' sparse' and this is useful when used for noise removal. A major advantage of using wavelets is that the windows vary. A major application of this is to realize the portions and signals that are not continuous having short wavelet functions is a good practice to overcome this, but to obtain more in depth analysis having longer functions are best.

A practice that is utilized is having basis functions that are of short high frequency and basis functions that are of long low frequency (A. Graps, 1995-2004), point to note is that unlike Fourier analysis that have a limited basis function sine and cosine wavelets have unlimited set of basis functions . This is a very important feature as it allows wavelet to identify information from a signal that can be hidden by other time frequency methods, namely Fourier analysis.

Wavelets consist of different families within each family of wavelet there exist different subclasses that are differentiated based on the coefficients

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

that are decomposed and their levels of iteration, wavelets are mostly classified based on their number of coefficients, that is also referred to as their vanishing moments, a mathematical relationship relates both.

Fig above showing examples of wavelets (N. Rao 2001)

One of the most helpful and defining features of using wavelets is that the experimenter has control over the wavelet coefficients for a wavelet type. Families of wavelets were developed that proved to be very efficient in the representation of polynomial behavior' the simplest of these is the Haar wavelet. The coefficients can be thought of as being filters; these are then placed in a transformation matrix and applied to a raw data vector. The different coefficients are ordered with patterns that work as a smoothing filter and another pattern whose function is to realize the detail information of the data (D. Aerts and I. Daubechies 1979). The coefficient matrix for the wavelet analysis is then applied in a hierarchical algorithm, based on its arrangement odd rows contain the different coefficients, the coefficients will be acting as filters that perform smoothing and the rows that are even will have the coefficients of the wavelets that contains the details from the analysis, it is to the full length data the matrix is first applied, it is then smoothed and disseminated by half after this process the step is repeated with the matrix., where more smoothing takes place and the different coefficients are halved, this process is repeated several times until the data that remains is smoothed, what this process actually does is to bring out the highest resolutions from that data source and data smoothing is also performed. In the removal of noise from data wavelet applications have proved very efficient and successful, as can be seen in work done by David <https://assignbuster.com/speech-and-transform-ii-computer-science/>

Donoho, the process of noise removal is called wavelet shrinkage and thresholding. When data is decomposed using wavelets, actually filters are used as averaging filters while the other produce details, some of the coefficients will relate to some details of the data set and if a given detailed is small, it can then be removed from the data set without affecting any major feature as it relates to the data. The basic idea of thresholding is setting coefficients that are at a particular threshold or less than a particular threshold to zero, these coefficients are then later used in an inverse wavelet transform to reconstruct the data set (S. Cai and K. Li, 2010)

Literature Review

The work done by Student Nikhil Rao (2001) was reviewed, according to the work that was done a completely new algorithm was developed that focused on the compression of speech signals, based on techniques for discrete wavelet transforms. The MATLAB software version 6 was used in order to simulate and implement the codes. The steps that were taken to achieve the compression are listed below;

Choose wavelet function

Select decomposition level

Input speech signal

Divide speech signal into frames

Decompose each frame

Calculate thresholds

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

Truncate coefficients

Encode zero-valued coefficients

Quantize and bit encode

Transmit data frame

Parts of extract above taken from said work by Nikhil Rao (2001). Based on the experiment that was conducted the Haar and Daubechies wavelets were utilized in the speech coding and synthesis the functions that were used that are a function of the MATLAB suite are as follows; dwt, wavedec, waverec, and idwt, they were used in computing the wavelet transforms Nikhil Rao (2001). The wavedec function performs the task of signal decomposition, and the waverec function reconstructs the signal from its coefficients. The idwt function functions in the capacity of the inverse transform on the signal of interest and all these functions can be found in the MATLAB software. The speech file that was analyzed was divided up into frames of 20 ms, which is 160 samples per frame and then each frame was decomposed and compressed, the file format utilized was . OD files, because of the length of the files there were able to be decomposed without being divided up into frames. The global and by-level thresholding was used in the experiment, the main aim of the global thresholding is the maintenance of the coefficients that are the largest, this not being dependent on the size of the decomposition tree for the wavelet transform. Using the level thresholding the approximate coefficients are kept at the decomposition level, during the process two bytes are used to encode the zero values. The function of the

very first byte is the specification of the starting points of zeros and the other byte tracks successive zeros.

The work done by Qiang Fu and Eric A. Wan (2003) was also reviewed; their work was the enhancement of speech based on wavelet de-noising framework. In their approach to their objective, the noisy speech signal was first processed using a spectral subtraction method; the aim of this involves the removal of noise from the signal of study before the application of the wavelet transform. The traditional approach was then done where the wavelet transforms are utilized in the decomposition of the speech into different levels, thresholding estimation is then on the different levels, however in this project a modified version on the Ephraim/Malah suppression rule was utilized for the threshold estimates. To finally enhance the speech signal the inverse wavelet transform was utilized. It was shown the pre processing of the speech signal removed small levels of noise but at the same time the distortion of the original speech signal was minimized, a generalized spectral subtraction algorithm was used to accomplish the task above this algorithm was proposed by Bai and Wan.

The wavelets transform for this approach utilized using wavelet packet decomposition, for this process a six stage tree structure decomposition approach was taken this was done using a 16-tap FIR filter, this is derived from the Daubechies wavelet, for a speech signal of 8khz the decomposition that was achieved resulted in 18 levels. The estimation method that was used to calculate the threshold levels were of a new type, the experiments took into account the noise deviation for the different levels, and each different time frame. An altered version of the Ephraim/Malah rule for <https://assignbuster.com/speech-and-transform-ii-computer-science/>

suppression was used to achieve soft thresholding. The re-synthesis of the signal was done using the inverse perceptual wavelet transform and this is the very last stage.

Work done by S. Manikandan, entitled (2006) focused on the reduction of noise that is present in a wireless signal that is received using special adaptive techniques. The signal of interest in the study was corrupted by white noise. The time frequency dependent threshold approach was taken to estimate the threshold level, in this project both the hard and soft thresholding techniques were utilized in the de-noising process. As with the hard thresholding coefficient below a certain values are scaled, in the project a universal threshold was used for the Gaussian noise that was added the error criterion that was used was under 3 mean squared, based on the experiments that were done it was found out that this approximation is not very efficient when it comes to speech, this is mainly because of poor relations amongst the quality and the existence to the correlated noise. A new thresholding technique was implemented in this technique the standard deviation of the noise was first estimated of the different levels and time frames. For a signal the threshold is calculated and is also calculated for the different sub-band and their related time frame. The soft thresholding was also implemented, with a modified Ephraim/Malah suppression rule, as seen before in the other works that were done in this area. Based on their results obtained, there was an unnatural voice pattern and to overcome this, a new technique based on modification from Ephraim and Mala is implemented.

Procedure

The procedure that undertaken involved doing several voice recording and reading the file using the 'wavread' function because the file was done in a .wav format

The length to be analyzed was decided, for the my project the entire length of the signal was analyzed

The uncorrupted signal power and signal to noise ratio (SNR) was calculated using different MATLAB functions

Additive White Gaussian Noise (AWGN) was then added to the original recorded, making the uncorrupted signal now corrupted

The average power of the signal corrupted by noise and also the signal to noise ratio (SNR) was then calculated

Signal analysis then followed, the procedure involved in the signal analysis included:

The wavedec function in MATLAB was used in the decomposition of the signal.

The detail coefficients and approximated coefficients were then extracted and plots made to show the different levels of decomposition

The different levels of coefficient were then analyzed and compared, making detailed analysis that the decomposition resulted in

After decomposition of the different levels de-nosing took place this was done with the ddencomp function in MATLAB,

The actual de-noising process was then undertaken using `wdencomp` function in MATLAB, plot comparison was made to compare the noise corrupted signal and the de-noised signal

The average power and SNR of the de-noised signal was done and comparison made between it and the original and the de-noised signal.

Implementation/Discussion

The first part of the project consisted of doing a recording in MATLAB, a recording was done of my own voice and the default sample rate was used were $F_s = 11025$, codes were used to do recordings in MATLAB and different variables were altered and specified based on the codes used, the m file that is submitted with this project gives all the codes that were utilized for the project, the recordings were done for 9 seconds the `wavplay` function was then used to replay the recording that was done until a desired recording was obtained after the recording was done a `wavwrite` function was then used to store the data that was previously recorded into a wav file. The data that was written into a wav file was originally stored in variable `y` and then given the name `recording1`. A plot was then made to show the wave format of the speech file recorded.

Fig 1

Fig1 Plot above showing original recording without any noise corruption

According to fig1 the maximum amplitude of the signal is +0.5 and the minimum amplitude being -0.3 from observation with the naked eye it can

be seen that most of the information in the speech signal is confined between the amplitude $+0.15 - -0.15$.

The power of the speech signal was then calculated in MATLAB using a periodogram spectrum this produces an estimate of the spectral density of the signal and is computed from the finite length digital sequence using the Fast Fourier Transform (The MathWorks 1984-2010) the window parameter that was used was the Hamming window, the window function is some function that is zero outside some chosen interval. The hamming window is a typical window function and is applied typically by a point by point multiplication to the input of the fast fourier transform, this controls the adjacent levels of spectral artifacts which would appear in the magnitude of the fast fourier transform results, for a case where the input frequencies do not correspond with the bin center. Convolution that occurs within the frequency domain can be considered as windowing this is basically the same as performing multiplication within the time domain, the result of this multiplication is that any samples outside a frequency will affect the overall amplitude of that frequency.

Fig2

Fig2 plot showing periodogram spectral analysis of original recording

From the spectral analysis it was calculated that the power of the signal is 0.0011 watt

After the signal was analyzed noise was added to the signal, the noise that was added was additive gaussian white noise (AWGN), and this is a random

signal that contains a flat power spectral density (Wikipedia, 2010). At a given center frequency additional white noise will contain equal power at a fixed bandwidth; the term white is used to mean that the frequency spectrum is continuous and is also uniform for the entire frequency band. In the project additive is used to simply mean that this impairment to the original signal is corrupting the speech; The MATLAB code that was used to add the noise to the recording can be seen in the m file.

For the very first recording the power in the signal was set to 1 watt and the SNR set to 80, the applied code was set to signal z, which is a copy of the original recording y, below is the plot showing the analysis of the noise corrupted recording.

Fig3

Fig3 plot showing the original recording corrupted by noise

Based on observation of the plot above it can be estimated that information in the original recording is masked by the additive white noise to the signal, this would have a negative effect as the clean information would be masked out by the noise, a process known as aliasing. Because the amplitude of the additive noise is greater than the amplitude of the recording it causes distortion observation of the graph shows the amplitude of the corrupted signal is greater than the original recording. The noise power of the corrupted signal was calculated by the division of the signal power and the signal to noise ratio, the noise power calculated from the first recording is $1.37e-005$. The noise power of the corrupted signal is $1.37e-005$; the spectrum periodogram was then used to calculate the average power of the

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

corrupted signal , based on the MATLAB calculations the power was calculated to be 0. 0033 watt

Fig4

Fig4 plot showing periodogram spectral analysis of corrupted signal

From analysis of the plot above it can be seen that the frequency of the corrupted signal spans a wider band, the original recording spectral frequency analysis showed a value of -20Hz as compared to the corrupted signal showed a value of 30Hz this increase in the corrupted signal is attributed to the noise added and this masked out the original recording again as before the process of aliasing.

It was seen that the average power of the corrupted was greater than the original signal, the increase in power can be attributed to the additive noise added to the signal this caused the increase in power of the signal.

The signal to noise ratio (SNR) of the corrupted signal was calculate from the formula corrupted power/noise power , and the corrupted SNR was found to be 240 as compared to 472. 72 of the de-noised, the decrease in signal to noise ratio can be attributed to the additive noise this resulted in the level of noise to the level of clean recording to be greater this is the basis for the decreased SNR in the corrupted signal, the increase in the SNR in the clean signal will be discussed further in the discussion.

The reason there was a reduce in the SNR in the corrupted signal is because the level of noise to clean signal is greater and this is basis of signal to noise comparison, it is used to measure how much a signal is corrupted by noise

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

and the lower this ratio is, the more corrupted a signal will be. The calculation method that was used to calculate this ratio is

Where the different signal and noise power were calculated from MATLAB as seen above

The analysis of the signal then commenced a .wav file was then created for the corrupted signal using the MATLAB command `wavwrite`, with F_s being the sample frequency, N being the corrupted file and the name being noise recording, a file `x1` that was going to be analysed was created using the MATLAB command `wavread`.

Wavelet multilevel decomposition was then performed on the signal `x1` using the MATLAB command `wavedec`, this function performs the wavelet decomposition of the signal, the decomposition is a multilevel one dimensional decomposition, and discrete wavelet transform (DWT) is using pyramid algorithms, during the decomposition the signal is passed through a high pass and a low pass filter. The output of the low pass is further passed through a high pass and a low pass filter and this process continues (The MathWorks 1994-2010) based on the specification of the programmer, a linear time invariant filter, this being a filter that passes high frequencies and attenuates frequency that are below a threshold called the cut off frequency, the rate of attenuation is specified by the designer. While on the other hand the opposite to the high pass filter, is the low pass filter this filter will only pass low frequency signals but attenuates signal that contain a higher frequency than the cut off. Based on the decomposition procedure above the process was done 8 times, and at each level of decomposition the actual

signal is down sampled by a factor of 2. The high pass output at each stage represents the actual wavelet transformed data; these are called the detailed coefficients (The MathWorks 1994-2010).

Fig 5

Fig 5 above levels decomposition (The MathWorks 1994-2010)

Block C above contains the decomposition vectors and Block L contains the bookkeeping vector, based on the representation above a signal X of a specific length is decomposed into coefficients, the first part of the decomposition produces 2 sets of coefficients the approximate coefficient $cA1$ and the detailed coefficient $cD1$, to get the approximate coefficient the signal x is convolved with low pass filter and to get the detailed coefficient signal x is convolved with a high pass filter. The second stage is similar only this time the signal that will be sampled is $cA1$ as compared to x before with the signal further being sampled through high and low pass filter again to produce approximate and detailed coefficients respectively hence the signal is down sampled and the factor of down sampling is two

The algorithm above (The MathWorks 1994-2010) represents the first level decomposition that was done in MATLAB, the original signal $x(t)$ is decomposed into approximate and detailed coefficient, the algorithm above represents the signal being passed through a low pass filter where the detail coefficients are extracted to give $D2(t)+D1(t)$ this analysis is passed through a single stage filter bank further analysis through the filter bank will produce greater stages of detailed coefficients as can be seen with the algorithm below (The MathWorks 1994-2010).

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

The coefficients, $A_c^m(k)$ and $A_d^m(k)$ for $m = 1, 2, 3$ can be calculated by iterating or cascading the single stage filter bank to obtain a multiple stage filter bank (The MathWorks 1994-2010).

Fig6

Fig6 showing graphical representation of multilevel decomposition (The MathWorks 1994-2010)

At each level it is observed the signal is down sampled and the sampling factor is 2. At d8 observation shows that the signal is down sampled by 2^8 i. e. $60,000/2^8$. All this is done for better frequency resolution. Lower frequencies are present at all time; I am mostly concerned with higher frequencies which contains the actual data.

I have used Daubechies wavelet type 4 (db4), the Daubechies wavelet are defined by computing the running averages and differences via scalar products with scaling signals and wavelets (M. I. Mahmoud, M. I. M. Dessouky, S. Deyab, and F. H. Elfouly, 2007) For this type of wavelet there exist a balance frequency response but the phase response is non linear. The Daubechies wavelet types uses windows that overlap in order to ensure that the coefficients of higher frequencies will show any changes in their high frequency, based on these properties the Daubechies wavelet types proves to be an efficient tool in the de-noising and compression of audio signals. A For the Daubechies D4 transform, this transform has 4 wavelet types and scaling coefficient functions, these coefficient functions are shown below

The different steps that are involved in the wavelet transforms, will utilize different scaling functions, to the signal of interest if the data being analyzed contains a value of N , the scaling function that will be applied will be applied to calculate $N/2$ smoothed values. The smoothed values are stored in the lower half of the N element input vector for the ordered wavelet transform. The wavelet function coefficient values are $g_{0A} = h_3$ $g_{1A} = -h_2$ $g_{2A} = h_1$ $g_{3A} = -h_0$

The different scaling function and wavelet function are calculated using the inner product of the coefficients and the four different data values. The equations are shown below (Ian Kaplan, July 2001);

The repetition of the of the steps of the wavelet transforms was then used in the calculation of the function value of the wavelet and the scaling function value, for each repetition there is an increase by two in the index and when this occurs a different wavelet and scaling function is produced.

Fig 7

Diagram above showing the steps involved in forward transform

(The MathWorks 1994-2010)

The diagram above illustrates steps in the forward transform, based on observation of the diagram it can be seen that the data is divided up into different elements, these separate elements are even and the first elements are stored to the even array and the second half of the elements are stored in the odd array. In reality this is folded into a single function even though

the diagram above goes against this, the diagrams shows two normalized steps.

The input signal in the algorithm above (Ian Kaplan, July 2001) is then broken down into what are called wavelets. One of the most significant benefits of use of wavelet transforms is the fact that it contains a window that varies, to identify signal not continuous having base functions that are short is most desirable. But in order to obtain detailed frequency analysis it is better to have long basis function. A good way to achieve this compromise is having a short high frequency functions and also long low frequency ones (Swathi Nibhanupudi, 2003)

Wavelet analysis contains an infinite basis functions, this allows wavelet transforms and analysis with the ability realize cases that can not be easily realized by other time frequency methods, namely Fourier transforms.

MATLAB codes are then used to extract the detailed coefficients, the m file shows these codes, the detailed coefficients that are Daubechies orthogonal type wavelets D2-D20 are often used. The numbers of coefficients are represented by the index number, for the different wavelets they contain vanishing moments that are identical to the halve of the coefficients. This can be seen using the orthogonal types where D2 contain only one moment and D4 two moments and so on, the vanishing moment of the wavelets refers to its ability to represent the information in a signal or the polynomial behavior. The D2 type that contains only one moment will encode polynomial of one coefficient easily that are of constant signal component. The D4 type

will encode polynomial of two coefficients, the D6 will encode coefficient of three polynomial and so on.

The scaling and wavelet function have to be normalized and this normalization factor is a factor A . The coefficients for the wavelet are derived by the reverse of the order of the scaling function coefficients and then by reversing the sign of the second one (D4 wavelet = $\{-0.1830125, -0.3169874, 1.1830128, -0.6830128\}$) mathematically, this looks like

where k is the coefficient index, b is a wavelet coefficient and c a scaling function coefficient. N is the wavelet index, ie 4 for D4 (M.

Bahoura, J. Bouat. 2009)

Fig 7

Plot of fig 7 showing approximated coefficient of the level 8 decomposition

Fig 8

Plot of fig 8 showing detailed coefficient of the level 1 decomposition

Fig 9

Plot of fig 9 showing approximated coefficient of the level 3 decomposition

Fig 10

Plot of fig 10 showing approximated coefficient of the level 5 decomposition

Fig 11

Plot of fig 11, showing comparison of the different levels of decomposition

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

Fig12

Plot fig12 showing the details of all the levels of the coefficients;

The next step in the de-noising process is the actual removal of the noise after the coefficients have been realized and calculated the MATLAB functions that are used in the de-noising functions are the `ddencmp` and the `wdencomp` function

This process actually removes noise by a process called thresholding, De-noising, the task of removing or suppressing uninformative noise from signals is an important part of many signal or image processing applications. Wavelets are common tools in the field of signal processing. The popularity of wavelets in de-noising is largely due to the computationally efficient algorithms as well as to the sparsity

of the wavelet representation of data. By sparsity I mean that majority of the wavelet coefficients have very small magnitudes whereas only a small subset of coefficients have large magnitudes. I may informally state that this small subset contains the interesting informative part of the signal, whereas the rest of the coefficients describe noise and can be discarded to give a noise-free reconstruction.

The best known wavelet de-noising methods are thresholding approaches, see e. g. In hard thresholding all the coefficients with greater magnitudes as compared to the threshold are retained unmodified this is because they comprise the informative part of data, while the rest of the coefficients are considered to represent noise and set to zero. However, it is reasonable to

assume that coefficients are not purely either noise or informative but mixtures of those. To cope with this soft thresholding approaches have been proposed, in the process of soft thresholding coefficients that are smaller than the threshold are made zero, however the coefficients that are kept are made smaller towards zero by an amount of the threshold value in order to decrease the effect of noise assumed to corrupt all the wavelet coefficients. In my project I have chosen to do a eight level decomposition before applying the de-noising algorithm, the decomposition levels of the different eight levels are obtained, because the signal of interest is filtered through eight different stages of low pass filters. Using the approximate sequence is therefore un-necessary, for the de-noising process. An assumption that is taken is that the Gaussian noise that is added is already known. The best techniques that I came across for the de-noising process is the use of soft thresholding, this techniques provides better results as compared to its counterpart hard thresholding, the soft thresholding techniques results in a smoother approximation. For the de-noising of audio the soft thresholding techniques provides the best results. The actual thresholding value was calculate using MATLAB and was found to be 0. 0825 For denosing some parameters have to be set. Like some threshold value. Those elements in the signal whose value is lower than the threshold is made zero, this is hard thresholding. In soft thresholding the same procedure is applied but to remove the discontinuities the nonzero values are shrunken towards zero. So, the function ' ddencomp' gives the default threshold level and by default soft-thresholding is used . Also here ' den' means you are calculating default

values for signal de-noising (The MathWorks 1994-2010) , above shows the default threshold value that was set from the MATAB code above.

Seeing that the speech file that was analyzed is a one dimensional signal then the MATLAB function that performs the de-nosingof this signal is the ‘ wdencomp’ function, this function performs the de-nosing process using wavelets , the actual MATLAB code used for the de-noising process is given in the m file presented with this report. In the code presented the clean represents the final de-noised signal, but before this procedure can be done the actual thresholding value must be calculated, that is where the code, the function ‘ ddencomp’ gives the default values for the de-nosing process that will be needed later, in the case of my project for de-nosing the variable IN1 is set to ‘ den’ for the de-nosing purpose, the variable ‘ wv’ wavelet and ‘ x’ being the signal. The variable THR represents thresholding, SORH represents soft thresholding and the variable KEEPAPP allows the approximate coefficients to be kept, all the values I obtained from MATLAB is shown Threshold value being 0. 0825, ‘ sorh’ being ‘ s’ and the ‘ keepapp’ value being 1.

Fig 13

Plot of fig 3 showing de-noised signal after wavelet thresholding and denosing algorithm was performed.

Fig 14

Plot of Fig 14 showing comparison of the original signal vs de-noised signal

Fig15

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

Plot of fig 15 showing the comparison of the de-noised signal and noisy signal

Observation of fig. 13-fig. 15 shows significant difference between the noisy signal as compared to the de-noised and original, based on observation of the de-noised signal it can be seen the level of noise is lot less as compared to the noisy signal based on pure observation to quantify these observations the signal to noise ration calculated for both can be observed the clean signal to noise ratio was calculated to be 472. 7 and the corrupted signal to noise ratio was calculated to be 240. It can be seen that the signal to noise ratio of the de-noised signal is greater than that of the corrupted signal, based on these results the de-noised signal the level of noise in the signal is less as a greater signal to noise ratio indicates less corruption in the signal. The results indicate there is a greater level of signal power than noise power in the de-noised signal, the de-nosing algorithm showed a vast improvement in the signal to noise ratio.

Several similar recordings where done with different noise power and noise SNR used and in the de-noising process several different decompositions levels were also used as well as the Daubechies filter type used, the other procedures that were used are similar to the original procedure main difference is the variables that were chosen, below shows graphs and calculations for several different de-noised recordings.

Fig16

Plot above showing comparison of the 3rd recording and de-noising process that was done

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

The graph above shows the results obtained after the de-noising procedure using wavelet transform was applied to the noise corrupted signal. The procedures used to generate the signal and the noise is similar to the procedures above, in the de-noising process Daubechies wavelet type 8 was used and only 5 levels of decomposition was done, the thresholding value that was used in this de-noising process was 0.0580, based on the calculations done in MATLAB the corrupted SNR was calculated to be 227.1 and the clean SNR calculated to be 361.3 based on these calculations it can be seen that there was some cleansing of the noise corrupted signal after the wavelet thresholding procedure was performed. Based on observation of the clean signal it is evident that de-noising process was successful.

A fourth recording was done with similar properties to the recordings above, however the Daubechies type 10 filter was used and a similar 5 level decomposition was used because this signal was a different recording the signal power and SNR of the corrupted signal was different, when the de-noising procedure was applied a threshold value of 0.1000 was used and this threshold was used as a variable to calculate the clean signal, based on calculations in MATLAB the SNR of the corrupted and clean signals were compared, the SNR was found to be 1.2212×10^3 and 3.2692×10^3 respectively the improvement in the SNR after the de-noising procedure showed the improvement of the signal quality, below is a graph showing comparison of the signals from the fourth recording.

Fig 17

Plot above showing comparison of the 4th recording and de-noising process that was done

The MATLAB function `wden` was also used in the de-noising process to produce a separate de-noised signal, `de-noised2` the MATLAB code that was used in this de-noising is given in the 'm-file' presented with the report.

The `wden` function is a one dimensional de-noising function, it performs an automatic de-noising process (The MathWorks 1994-2010). Based on the MATLAB code used, a de-noised signal `XD` is formed, the input signal that results in this de-noised signal is a signal `X`, that is realized by the thresholding of the wavelet coefficients. The `TPTR` function that was used in my program was `minimaxi` this function uses a fix threshold to yield minimax performance. The minimax function is designed to realize the minimum of the maximum mean square error obtained for the worst function in a given set (The MathWorks 1994-2010). The variable `SHOR` that I choose to use is 's' for soft thresholding as it yields better results as compared to hard thresholding where it was evident that the signal still contained audible noise in the output signal. The variable for the function `SCAL` that I used in my program was 'sln' this function uses an estimation property for rescaling, this is realized based on the noise level and the first level coefficients.

Fig18

Plot above showing comparison of the 5th recording and de-noising process that was done

Based on the plot above it can be seen that the signal resembles the original denoised signal, but based on empirical information the signal that is reproduced by this procedure yields better results, the noise level is less as can be proven by the signal to noise ratio of the denoised signal, the SNR was calculated to be 160. 2, which is seen to be greater than the original denoised signals, the results indicate there is a greater level of signal power than noise power in the de-noised2 signal, as compared to the original denoised signal, based on observation of listening to both de-noised signals it could be heard that there was vast improvement in the SNR and the quality of the signal as noise was removed from the signal through wavelet thresholding and de-noising.

Wavelet packet transform was also analyzed and used in the de-noising process as oppose to wavelet transforms, the main difference between the wavelet transform and the wavelet packet transform is that in the packet transform the signal is passed through more filters as compared to its DWT counterpart, in the wavelet packet transform the level s are calculated by the decomposition of both detailed and approximated coefficients. With the wavelet packet transform the decomposition produces $2nA$ sets of different coefficients, however with the sampling process the coefficients remains the same as compared to the DWT. The same sampling rate of 11025 was used as before and the speech file almost identical. The same matlab codes were used in simulating this, illustrated in the ' m-file'

White noise was then added of exactly the same properties as before and the MATLAB code used to simulate this is identical to procedure above.

However the decomposition function that was used was different from that

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

used in the wavelet transform the function used was 'wpdec', this MATLAB function performs wavelet packet decomposition as illustrated by the figure below

Fig. 16

Plot above of fig 16 showing WP decomposition.

The code for the WP tree is illustrated in the 'm-file' WP tree that corresponds to the WP decomposition at 8 levels to the noise corrupted signal N. The coefficients were then realized using the function 'wpccoef' this function gives the coefficients that are associated with the input signal N, of the WP tree shown above. The MATLAB function that was used to realize the de-noising variables and the threshold value is the 'ddencmp' function the thresholding value that was used was 0.3500, this code is almost identical to the codes used in wavelet transform above however it acts on WP instead. Based on the values obtained the clean signal was realized using the MATLAB function 'wpdencmp' this function is a wavelet packet de-noising function, this function has similar properties to the 'wdencmp' function but acts on wavelet packets instead, again the code used is almost identical to that used above. Below are diagrams showing original vs. noisy vs. clean recordings

Fig 17

Fig 17 above showing comparison of the original, noisy and denoised signal as a result of WP transform

Based on the signal to noise ratio that was calculated in MATLAB it was proved that the WP analysis

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

actually performed the de-noising function it was intended to do the SNR of the de-noised signal being higher than the SNR of the noise corrupted signal, the SNR of the corrupted signal being 89.4 and the SNR of the clean signal being 278.9 the increase in the SNR of the de-noised signal actually proved that it contains more information than corrupted noise the SNR was calculated based on the signal powers as calculated above.

Conclusion

Based on the work that was done in this project the method and procedures that were implemented resulted in a successful implementation on the wavelet transform to a speech signal in order to result in the de-noising of that speech signal. The aim of the project was to enhance speech signal by using wavelet thresholding, the software that was used in the simulation was MATLAB v7.0, MATLAB is a very powerful mathematical software and was efficient in the performing the objectives of the project, on comparison of both corrupted signal, corrupted with white noise and the de-noised signal based on the results and calculation done it is seen that there is much improvement in the speech files that were analyzed and improvement in the Signal to noise ratio of the signal.

The procedures that were employed in this project were sufficient enough to provide efficient de-noising of the signal, further research that was done revealed that if wavelet packets were used there could have been greater clarity and improvement in the output signal. The procedure that was done to remove the noise and improve the SNR is outlined below;

Choose wavelet function

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

Select decomposition level

Input speech signal

Divide speech signal into frames

Decompose each frame

Calculate thresholds

Perform de-noising

Part of the procedure take from work done by J. Karam 2008, the daubechies 4 wavelet type was used and compared to the other types gave the better results in terms of output signal quality and SNR.

Based on the experiments done it could be proven that the wavelet tree is built recursively, with this process the decomposition and approximation only to the mother wavelet. In the experiments different thresholding values where used and compared and the value that gave the best denoised output based on the graphs and calculation of the SNR was choosed

Based on the experiments conducted a comparison was done and its was proved that using the wavelet packet thresholding and analysis as compared to the wavelet thresholding the WPT actually provided better signal de-noising as compared to the WT, the fact that using the WPT the signal is passed through more filters this act to provide better smoothing and de-noising of the signal the improvement in SNR also proved this observation. Using wavelet packets filters are applied to both the approximated and decomposition filter output.

<https://assignbuster.com/speech-and-transform-ii-computer-science/>

Wavelets are a powerful tool to analyze signals and by splitting up the signals into sub-bands and then analyzing each individual signals independently based on their time and frequency component useful information can be gathered and this proved efficient and helpful in the de-noising process.