Comparison of Wavelet and FFT Based Single Channel Speech Signal Noise Reduction Techniques

Ningping Fan, Radu Balan, Justinian Rosca
Siemens Corporate Research Inc.
{Ningping.Fan, Radu.Balan, Justinian.Rosca}@scr.siemens.com

ABSTRACT

This paper compares wavelet and short time Fourier transform based techniques for single channel speech signal noise reduction. Despite success of wavelet denoising of images, it has not yet been widely used for removal of noise in speech signals. We explored how to extend this technique to speech denoising, and discovered some problems in this endeavor. Experimental comparison with large amount test data has been performed. Our results have shown that although the Fourier domain methods still has the superiority, wavelet based alternatives can be very close, and enormous different configurations can still be tried out for possible better solutions.

Keywords: DWT, DWPT, wavelet, wavelet packet, FFT, noise control, speech enhancement, noise cancellation filter

1. INTRODUCTION

This paper compares techniques for single channel speech signal noise reduction based on different transformation techniques, namely discrete wavelet transform (DWT), discrete wavelet packet transform (DWPT), and short time Fourier transform (STFT). State of the art wavelet denoising techniques [1] have been very successfully applied to image noise reduction. However it has not yet been widely used to solve the speech signal noise reduction problem, as few publications in wavelet in comparison to enormous STFT papers. Because both Fourier and wavelet transforms are linear and noises are additive, the STFT solutions should applicable to the wavelet domain.

The motivation to use wavelet as a possible alternative for speech noise reduction is to explore new ways to reduce computational complexity and to achieve better noise reduction performance. Firstly, because the wavelet transform may not require overlapped windows due to the localization property, the same filter could process less data. Secondly, wavelet filter does not correspond to time domain convolution, so that shift-invariant is not preserved. However, the Fourier domain filters can still be extended to the wavelet domain, because they are derived according to the statistical properties of spectral components. The Martin minimum statistics noise estimator, the Wiener, the spectral subtraction, the Wolfe-Godsill, and the Ephraim-Malah filters can be extended in the wavelet domain as well. These filters are similar to the modern soft, hard, or shrinking threshold methods of wavelet denoising that both operate on spectral magnitude and retain the sign of wavelet transform coefficients (which equivalent to the phase in STFT). Thirdly, there are many different wavelets and various wavelet transform combinations. Therefore, possibilities for a better wavelet and a particular transform are great.

Our comparison uses the same algorithms but in different domains. A speech database containing male and female speakers mixed with common office noises, like fan, printer, and open window near street, are used for the test. Objective speech qualities are measured for comparison. DWT and DWPT based on seven different wavelets have been tried to compare with STFT. Generally speaking, DWPT is better then DWT and STFT is the best. We have not yet tried the un-decimated wavelet to preserving shift invariant [2], perceptual wavelet transforms [3] [4] [5], and various incomplete DWPT transforms [6]. The computational complexity of DWT is less than DWPT, and it varies depending on the wavelet and the levels used in the transform, as well as implementation efficiency. It can be much less or much more than STFT. For example, the Daubechies-4 DWT is 0.085 times of STFT, but the Battle-Lemarie DWT is 2.46 times, and the Battle-Lemarie DWPT is about 10.3 times.

2. SHORT-TIME FOURIER AND WAVELET TRANSFORMS

Short time Fourier transform (STFT) is the main approach used in current speech noise removal methods. The input signal $x(n)$ is first segmented into consecutive overlapping block sequences with zero padding $x(m,i)$, where *m* is a sample index within a block, and *i* is a block index. Each block is then transformed into frequency domain $X(k,i)$, as shown in figure 1.

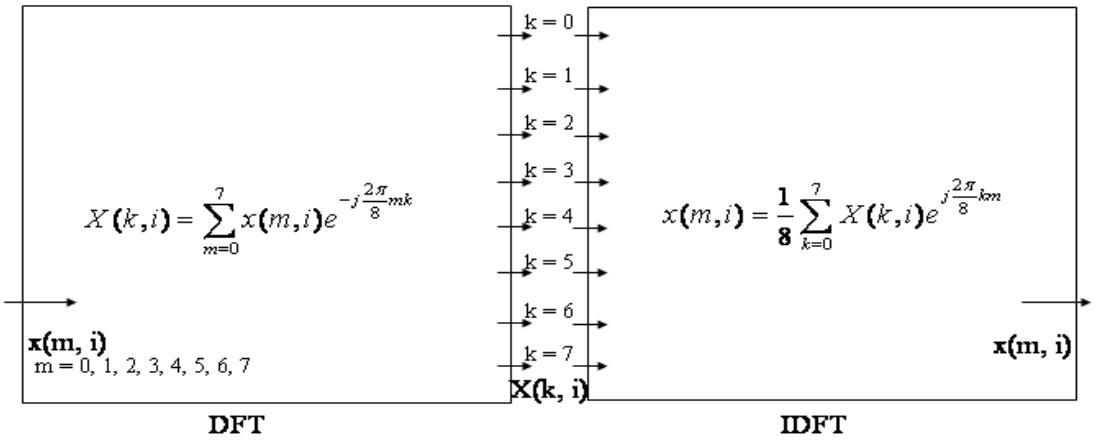


Figure 1 - Illustration of shot time DFT and IDFT

The first reason for success of STFT lies in the nature of the Fourier transform. It converts a time domain randomly fluctuating white noise signal into stationary frequency components, so that they can be detected and removed. The second reason is due to the block sequence mechanism and speech signal properties. In a speech signal, a vowel takes about .0.8 ~ 0.2 sec in average. Using the GSM mobile phone standard, one frame is about 0.02 sec, and there are about 10 ~ 40 frames per vowel. Even when a person continuously speaks, the speech signal is not continuous because there are unvoiced sections between vowels. This property has been exploited to estimate background noise, and to improve filter performance.

The wavelet transform is performed via a pair of filters *h* and *g*, which convolve input signal then decimate it into smooth half and detail half signals at the first level. The process then continuously operates on the smooth half until a final level reached, as shown in figure 2.

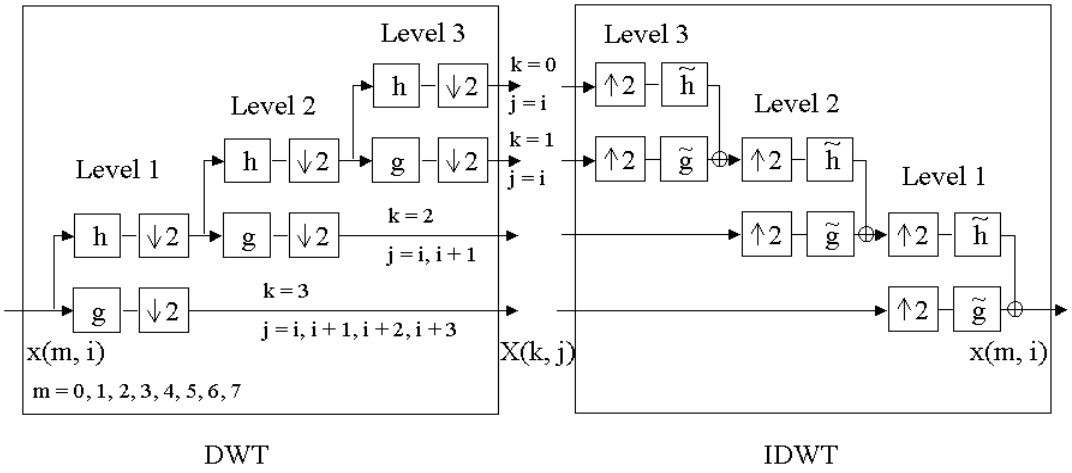


Figure 2 - Illustrations of DWT and IDWT in time-frequency presentation

For 8 input samples, it generates 4 frequency components (k=0,.., 3), with 4 samples in k=3, 2 samples in k=2, and 1 sample in k=1 and k=0 respectively, which is known as the time-frequency or time-scale presentation. Because it is difficult to visualize, a pseudo spectral presentation shown in figure 2 is used for the spectrum display. DWPT is shown in figure 4, which is same as DWT in the first level, but then it continuously operates on both smooth half and detail half until a final level reached. It has the full spectral components (k=0,…, 7), similar as FFT.

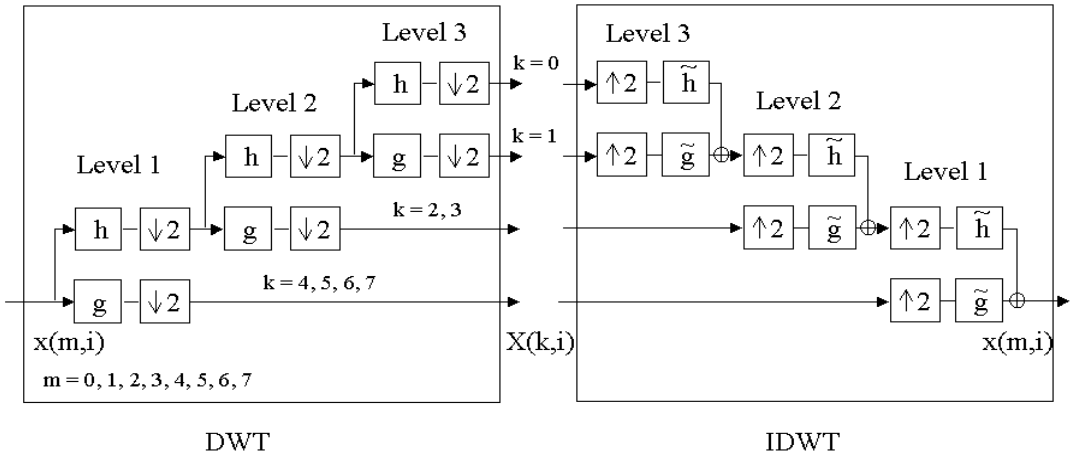


Figure 3 - Illustration of DWT and IDWT in pseudo frequency presentation

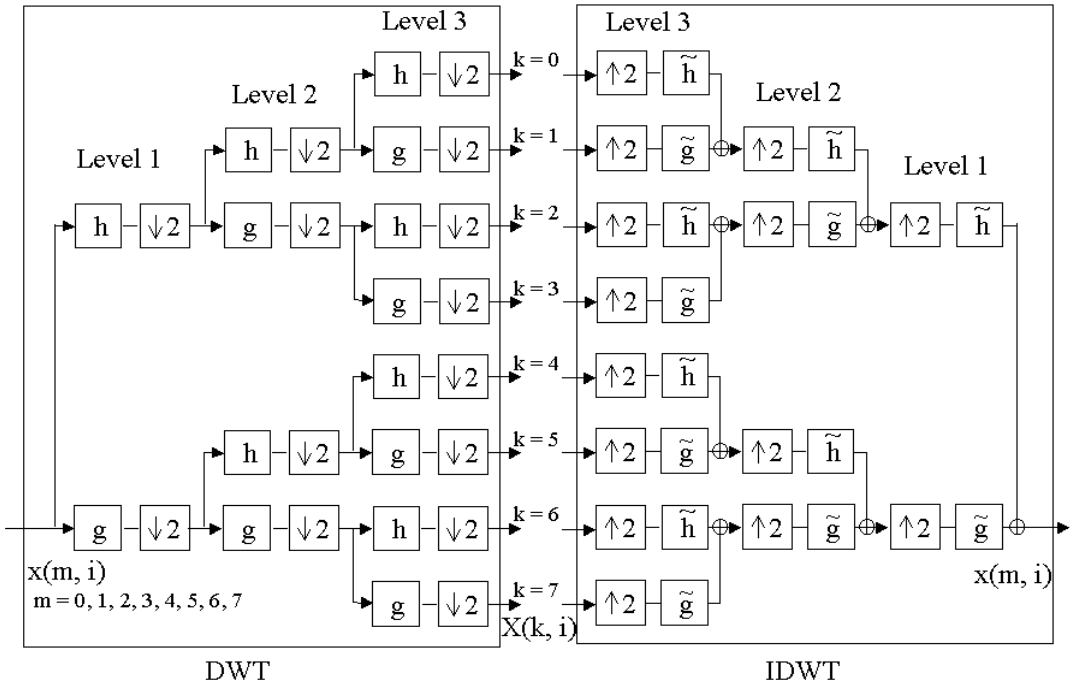
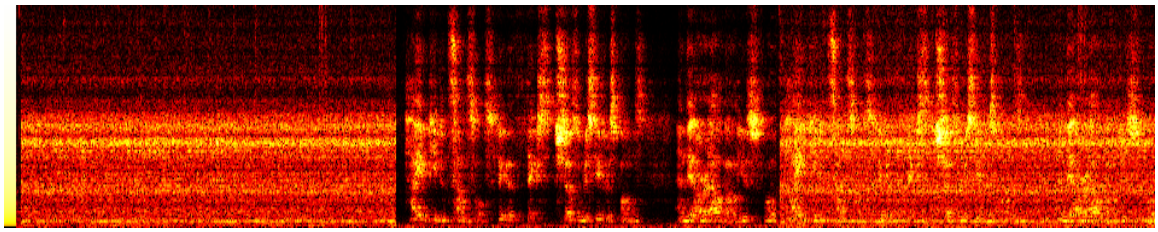


Figure 4 - Illustration of DWPT and IDWPT in full frequency presentation

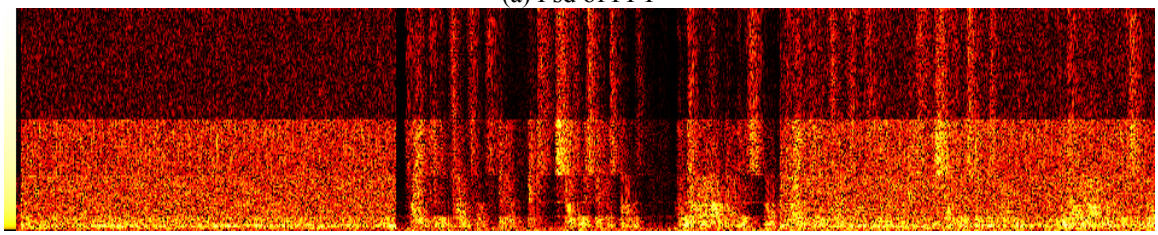
Because the localization property of wavelet, operating on entire signal once or block by block sequentially generates the same wavelet transform as long as the same final level is used. However, because the filtering algorithm utilizes the block sequence mechanism to match discontinuity of speech and to perform noise estimation, the same block size is used for both wavelet and FFT for easier comparison. Actually the same

notation such as $X(k,i)$ will be used to indicate both domains. A spectral component is complex in the Fourier domain and real in the Wavelet domain. A conjugate of a complex number is of the same real part plus the negative imaginary part, and a conjugate of a real number is the same number.

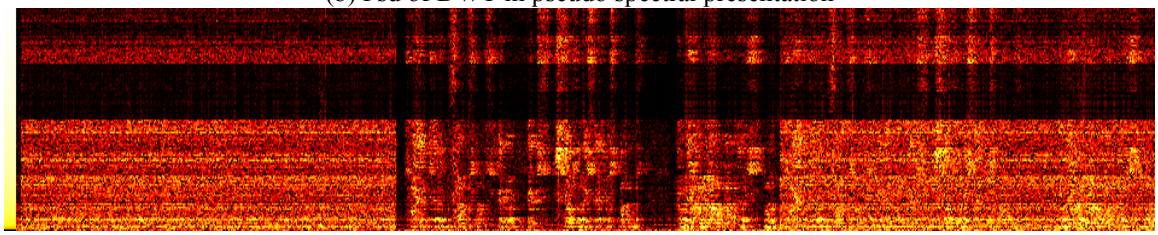
Figure 5 shows the power spectral density (Psd) of FFT, DWT, and DWPT for a same waveform, which starts with a fan noise section, then a clear speech section, and ended with a noisy speech section.



(a) Psd of FFT



(b) Psd of DWT in pseudo spectral presentation



(c) Psd of DWPT

Figure 5 - Power spectral densities of FFT, DWT, and DWPT for a signal starting with a noise section, then a clean speech section, and ended with a noisy speech section which is the summation of first two sections in time domain.

3. NOISE REDUCTION FILTERS

Using filters in transformed domains is the main approach in modern noise reduction techniques, because the noise magnitude can be estimated more accurately in those domains. As shown in figure 6, it is performed with a transfer function, $0 \le H(k,i) \le 1$, to produce the output $Y(k,i) = H(k,i) * X(k,i)$.

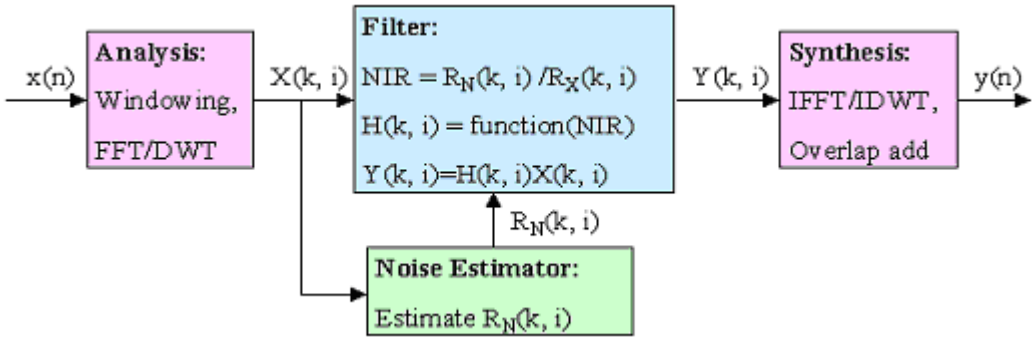


Figure 6 – The entire workflow of noise reduction operation

The $H(k,i)$ is a function of noise-to-input ratio (NIR). For low NIR, the *H* is large to maintain the signal component, while for high NIR, the *H* is small to reduce the noisy component. The phase in the Fourier domain or sign in the Wavelet domain of $X(k,i)$ is maintained intact. Following, we present rules widely used in the noise reduction algorithms, namely the Wiener, the spectral subtraction, the Wolfe-Godsill, and the Ephraim-Malah filters for both Fourier and wavelet domains.

3.1 The Wiener Filter

Because all the operations in the analysis stage are linear, $X(k,i)$ is consists of a signal component $S(k,i)$ plus a noise component $N(k,i)$.

$$X(k,i) = S(k,i) + N(k,i) \qquad 0 \le k < K \tag{1}$$

Where *i* is the block time index, *k* is transformed spectral components index, and *K* is the total number of transformed spectral components. We want to find a filter $H(k,i)$, so that the filtered output signal $Y(k,i) = H(k,i)X(k,i)$ will minimize the following objective function.

$$\begin{aligned} J(k,i) &= E\{(Y-S)\overline{(Y-S)}\} \\ &= E\{(HX-X+N)\overline{(HX-X+N)}\} \end{aligned} \tag{2}$$

To minimize *J*, we make its partial derivative with respect to *H*, and let the results equal to zero. The *k* and *i* indexes are dropped, because these derivation apply to all the *k* and *i* values.

$$\begin{aligned} \frac{\partial J}{\partial H} &= 2E\{(HX-X+N)\frac{\partial\overline{(HX-X+N)}}{\partial H}\} \\ &= 2E\{(HX-X+N)\overline{X}\} \\ &= 2(HE\{X\overline{X}\} - E\{X\overline{X}\} + E\{N\overline{X}\}) \\ &= 0 \end{aligned} \tag{3}$$

Then, we have the Wiener filter H to be

$$\begin{aligned} H_{Wiener} &= 1 - \frac{E\{N(\overline{S}+\overline{N})\}}{E\{X\overline{X}\}} \\ &= 1 - \frac{E\{N\overline{S}\} + E\{N\overline{N}\}}{E\{X\overline{X}\}} \\ &= 1 - \frac{E\{N\overline{N}\}}{E\{X\overline{X}\}} \\ &= 1 - \frac{R_N}{R_X} \end{aligned} \tag{4}$$

Where $E\{N\overline{S}\} = 0$ because signal and noise components are uncorrelated. The quantity R_N / R_X is called as noise to input ratio (NIR) in this paper, which satisfies the following constraints.

$$0 \le \frac{R_N}{R_X} = \frac{E\{N\overline{N}\}}{E\{(\overline{S}+\overline{N})(\overline{S}+\overline{N})\}} = \frac{E\{N\overline{N}\}}{E\{S\overline{S}\}+E\{N\overline{N}\}} \le 1 \tag{5}$$

Therefore, purely as a result of optimization, $0 \le H_{Wiener} \le 1$.

Because the speech signal is pseudo stationary, the R_X is estimated via a first order linear predictor along the time index i. Because the noise is generally assumed to be stationary, R_N is also being improved via a first order linear predictor along the time index over the original noise estimator's output $\hat{R}_N$.

$$\begin{aligned} \widetilde{R}_X(k,i) &= \alpha \widetilde{R}_X(k,i-1) + X(k,i)\overline{X(k,i)} \\ \widetilde{R}_N(k,i) &= \beta \widetilde{R}_N(k,i-1) + \hat{R}_N(k,i) \end{aligned} \tag{6}$$

Because the errors in R_X and R_N estimation, the ideal Wiener's rule (4) will cause a music tone and speech distortions. To overcome those problems, a parametric Wiener filter is generally applied

$$H_{PWiener}(k,i) = \max\left(1 - \gamma(k,i)\frac{\widetilde{R}_N(k,i)}{\widetilde{R}_X(k,i)},\ h(k,i)\right) \tag{7}$$

where γ is a scaling factor for NIR, and *h* is a floating floor for the transfer function. The γ and *h* can be constants [7], or even better be adaptive to the noise estimation as follows.

$$\begin{aligned} \gamma(k,i) &= \gamma_{\min} + \gamma_0 \log((K+1)\widetilde{R}_N(k,i)+1) \\ h(k,i) &= \max\left(h_{\max} - h_0 \log((K+1)\widetilde{R}_N(k,i)+1),\ h_{\min}\right) \end{aligned} \tag{8}$$

Intuitively (8) reflects the following ideas. When noise level is low, the γ will decrease and *h* will increase, so as to reduce the filtering operation and resulted speech distortion that cannot masked by the low noise. When noise is high, the reverse will happen, so as to increase filtering operation to reduce more noise while the speech distortion can be masked by the high noise [8].

3.2 The Spectral Subtraction Filter

The most widely alternative to the Wiener filter is the spectral subtraction filter. It is not derived from optimization of some cost function, but from an intuitive idea - to remove the noise magnitude [9].

$$\begin{aligned} \sqrt{E\{X\overline{X}\}} - \sqrt{E\{N\overline{N}\}} &= \sqrt{E\{Y\overline{Y}\}} \\ &= H\sqrt{E\{X\overline{X}\}} \end{aligned} \tag{9}$$

Thus the spectral subtraction filter H, is given by

$$\begin{aligned} H_{SS} &= 1 - \frac{\sqrt{E\{N\overline{N}\}}}{\sqrt{E\{X\overline{X}\}}} \\ &= 1 - \sqrt{\frac{R_N}{R_X}} \end{aligned} \tag{10}$$

Similarly, we have $0 \le H_{SS} \le 1$.

To overcome the distortion problem, its adaptive parametric form is actually applied.

$$H_{PSS}(k,i) = \max\left(1 - \sqrt{\gamma(k,i)\frac{\widetilde{R}_N(k,i)}{\widetilde{R}_X(k,i)}},\ h(k,i)\right) \tag{11}$$

The spectral subtraction is widely used in multiple microphone settings, where one microphone to pickup noisy speech signals and other microphones to pickup noise only sources. After proper delay and attenuation, the spectral noise magnitude is subtracted from the input spectral magnitude to produce a clean speech output, which is typically used in helicopters.

3.3 The Wolfe-Godsill Filter

Assuming Gaussian distribution of both speech and noise spectral components as follows:

$$S(k,i) \sim \mathrm{N}(0, \lambda_S(k,i)I), \quad N(k,i) \sim \mathrm{N}(0, \lambda_N(k,i)I) \tag{12}$$

The Wolfe-Godsill filter is the maximum a posteriori (MAP) estimation of jointly spectral amplitude and phase [10]. The filter is given as

$$\begin{aligned} H_{WG,MAP} &= \frac{\xi(k,i) + \sqrt{\xi^2(k,i) + 2(1+\xi(k,i))\frac{\xi(k,i)}{\gamma(k,i)}}}{2(1+\xi(k,i))} \\ \xi(k,i) &= \frac{\lambda_S(k,i)}{\lambda_N(k,i)} = \frac{R_X(k,i) - R_N(k,i)}{R_N(k,i)} = \gamma(k,i) - 1 \\ \gamma(k,i) &= \frac{E\{X(k,i)\overline{X(k,i)}\}}{\lambda_N(k,i)} = \frac{R_X(k,i)}{R_N(k,i)} \end{aligned} \tag{13}$$

Replacing ξ with γ, we can have

$$\begin{aligned} H_{WG,MAP} &= \frac{1 - \frac{1}{\gamma(k,i)} + \sqrt{\left(1 - \frac{1}{\gamma(k,i)}\right)^2 + 2\left(\frac{1}{\gamma(k,i)} - \frac{1}{\gamma^2(k,i)}\right)}}{2} \\ &= \frac{1 - \frac{R_N(k,i)}{R_X(k,i)} + \sqrt{\left(1 - \frac{R_N(k,i)}{R_X(k,i)}\right)^2 + 2\left(\frac{R_N(k,i)}{R_X(k,i)} - \left(\frac{R_N(k,i)}{R_X(k,i)}\right)^2\right)}}{2} \end{aligned} \tag{14}$$

Similarly, the transfer function satisfies $0 \le H_{WG,MAP} \le 1$.

Because of the pseudo stationary property, the first order linear prediction along the time index *i* is used to improve the accuracy as follows.

$$\begin{aligned} \widetilde{\xi}(k,i) &= \kappa H_{WG,MAP}(k,i-1)X(k,i-1) + (1-\kappa)(\widetilde{\gamma}(k,i) - 1) \\ \widetilde{\gamma}(k,i) &= \frac{\widetilde{R}_X(k,i)}{\widetilde{R}_N(k,i)} \end{aligned} \tag{15}$$

To overcome the distortion problem, a floating floor is used in its parametric form.

$$H_{PWG,MAP}(k,i) = \max\left(\frac{\tilde{\xi}(k,i) + \sqrt{\tilde{\xi}^2(k,i) + 2(1+\tilde{\xi}(k,i))\frac{\tilde{\xi}(k,i)}{\tilde{\gamma}(k,i)}}}{2(1+\tilde{\xi}(k,i))},\ h(k,i) \right) \tag{16}$$

3.3 The Ephraim-Malah Filter

One popular filter is the Ephraim-Malah filter [11], which is the minimum mean-square error spectral amplitude estimator, assuming the same Gaussian distribution as in (12). The filter is given by

$$\begin{aligned}
H_{EM,MMSE} &= \frac{\sqrt{\pi\upsilon(k,i)}}{2\gamma(k,i)}\left[(1+\upsilon(k,i))\mathrm{I}_0\left(\frac{\upsilon(k,i)}{2}\right) + \upsilon(k,i)\mathrm{I}_1\left(\frac{\upsilon(k,i)}{2}\right)\right]\exp\left(\frac{-\upsilon(k,i)}{2}\right) \\
\upsilon(k,i) &= \frac{\xi(k,i)}{1+\xi(k,i)}\gamma(k,i) = \gamma(k,i) - 1 \\
\xi(k,i) &= \frac{\lambda_S(k,i)}{\lambda_N(k,i)} = \frac{R_X(k,i) - R_N(k,i)}{R_N(k,i)} = \gamma(k,i) - 1 \\
\gamma(k,i) &= \frac{E\{X(k,i)\overline{X(k,i)}\}}{\lambda_N(k,i)} = \frac{R_X(k,i)}{R_N(k,i)}
\end{aligned} \tag{17}$$

Because (18) requires to calculate the exponential and modified Bessel functions I_0 and I_1, it is difficult to implement the filter accurately unless using Matlab. The C code implementation of the Bessel functions such as the Numerical Recipes has a very limited working range, but $0 \le \upsilon \le \infty$. Since the Wolfe-Godsill filter has the similar transfer function as shown in figure 6, we decided to evaluate that filter instead.

To summarize, four filter transfer functions are plotted as variables of the noise-to-input ratio using Matlab.

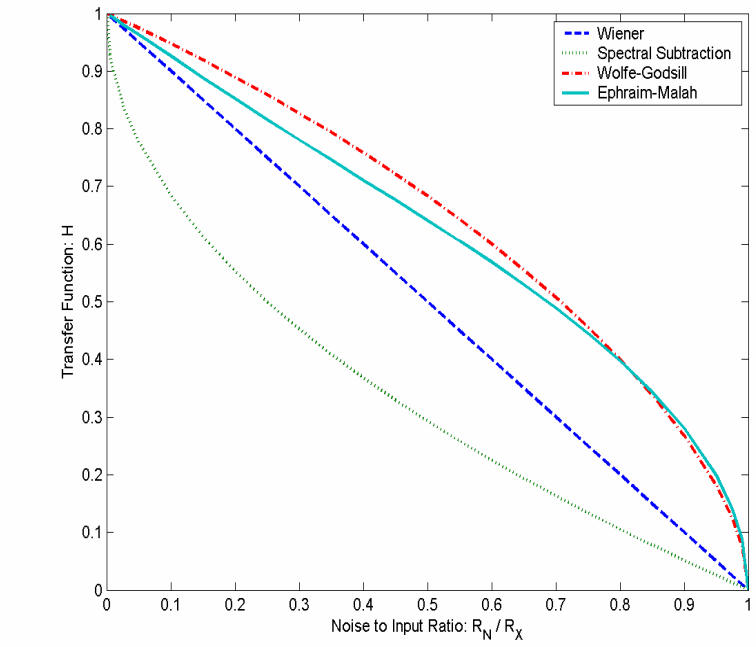


Figure 7 – The transfer functions of the Wiener, Spectral Subtraction, Wolfe-Godsill, and Ephraim-Malah Filters

The figure 6 shows that the spectral subtraction has the most noise reduction power, and then the Wiener filter, and the least is the Wolfe-Godsill's rule. However in practice the over noise reduction of former two filters will cause speech distortion due to inaccuracy in noise estimation, the scaling parameter is used to control the degree of noise reduction without sever distortions. The advantage in the Wolfe-Godsill's rule is to drop a scaling parameter, so that only one floor parameter needs to be optimized.

3.4 The Martin Noise Estimator

The power of a spectral component magnitude (PS) tracked along the time index *i* is known as the periodogram. In a periodogram, the noise behaves like a slow fluctuating background with a few grouped sparks due to the speech, as the speech is not continuous but noise is. This observation has been exploited to estimate the noise spectral magnitude. One particular method is based on the minimum statistics, and known as the Martin noise estimator [12].

After adaptively smoothing the periodogram, the method tracks the minimum within a moving window of fixed length. Then the minimum is statistically bias corrected, and is taken as the noise magnitude power estimation. Essentially the method is to track a slow varying baseline due to noise and cut through a few sparks due to speech. Figure 4 shows examples of noise tracking in both STFT and DWT periodograms of a sample wave file. The first half is clean speech and last half is noise corrupted.

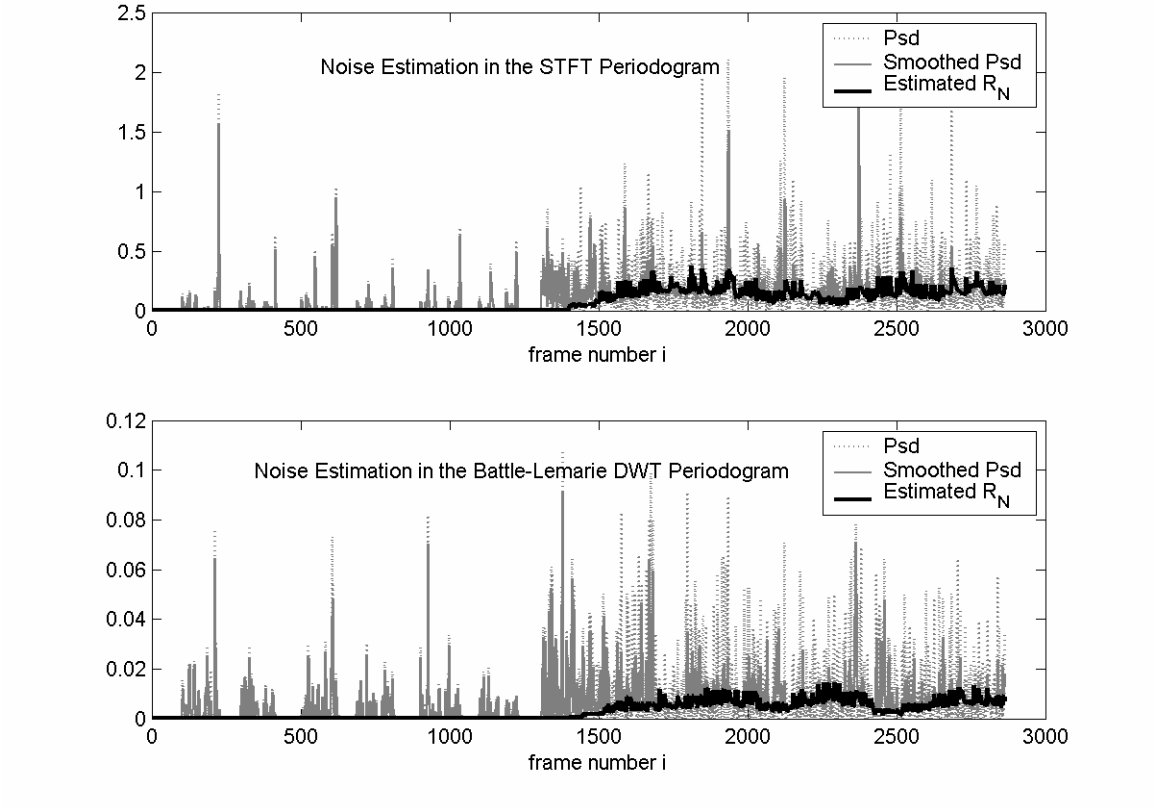


Figure 8 - Examples of noise magnitude tracking in periodograms of STFT and DWT

4. EXPERIMENTATION

Experiment data were taken in an ordinary office room using a modified Siemens Optipoint500 phone unit. Four clean speech signals, three in conference mode and one in handset mode, were recorded first. Seven noise signals from an air-conditioned room background, table fan, laser printer and opening window near street were then recorded separately. Speech and noise sources were positioned at different locations in the

room. The clean speech signals were then mixed with the noise signals at four different ratios. The noisy signals were then processed using three filtering algorithms, the spectral subtraction, the Wiener filter, and the Wolfe-Godsill filter. The Martin's noise estimator was used in all cases.

All testing signals are sampled at 16 KHz and stored as 16 bits per sample. Spectral analysis and synthesis modules in the Fourier domain are implemented according to the GSM mobile phone standard. 160 samples of input data block are prefixed with last 40 samples of previous block, and multiplied by a cosine windowing function and padded 56 zeros at end for 256 samples FFT. After spectral filtering, IFFT is performed and then overlap-added to produce 160 samples of output data.

Table 1 - Average objective quality scores for comparison of different noise reduction filters at FFT and various DWT domains

qm	gSNR (dB)				sSNR (dB)				fwsSNR (dB)				isD				WSS			
org	-0.1	2.31	5.9	10.31	-3.12	-0.71	2.95	7.44	1.51	4.85	9.64	15.18	0.46	0.32	0.2	0.12	43.3	34	24.56	16.03
spectral subtraction																				
fft	1.44	4.63	7.96	11.01	-0.7	2.16	5.24	8.2	3.9	6.75	9.63	12.4	0.41	0.29	0.2	0.13	41.64	32.31	23.63	15.96
wp0	1.03	4.05	6.61	9.52	-1.51	1.37	3.9	6.93	3.12	6.83	8.04	10.62	0.47	0.32	0.25	0.14	41.56	32.46	24.61	16.68
wp1	0.84	3.79	7.64	11.29	-1.76	1.04	4.78	8.3	2.83	6.43	10.25	13.39	0.51	0.34	0.2	0.11	42.39	33.27	23.65	15.84
wp2	1.01	4	7.68	11.54	-1.54	1.31	4.91	8.63	3.09	6.77	11.18	15	0.48	0.32	0.19	0.11	41.67	32.63	23.81	15.91
wp3	1.02	4.04	6.29	9.25	-1.5	1.36	3.56	6.69	3.12	6.82	7.91	10.6	0.48	0.32	0.27	0.15	41.42	32.44	25.53	17.43
wp4	0.84	3.77	7.35	11.01	-1.8	1.01	4.46	8.03	2.79	6.38	9.89	13.06	0.56	0.37	0.21	0.12	42.4	33.43	24.28	16.29
wp5	0.96	3.93	7.43	11.31	-1.6	1.22	4.62	8.39	3	6.69	10.82	14.7	0.49	0.33	0.2	0.11	41.51	32.59	24.3	16.24
wp6	0.94	3.9	6.56	9.47	-1.71	1.13	3.85	6.88	2.86	6.63	8.03	10.61	0.46	0.31	0.25	0.14	42.22	33.05	24.75	16.76
wt0	0.81	3.71	7.23	10.86	-1.82	0.96	4.37	7.91	2.74	6.28	9.76	12.89	0.5	0.34	0.21	0.12	42.35	33.41	24.38	16.3
wt1	0.67	3.51	7.05	10.74	-2	0.74	4.18	7.78	2.55	6.06	9.64	12.83	0.52	0.35	0.22	0.12	42.44	33.57	24.58	16.49
wt2	0.78	3.66	7.19	10.85	-1.87	0.91	4.34	7.9	2.72	6.29	9.8	12.93	0.5	0.34	0.21	0.12	42.32	33.37	24.38	16.29
wt3	0.79	3.71	7.27	10.93	-1.84	0.97	4.41	7.98	2.76	6.37	9.88	13.04	0.5	0.34	0.21	0.12	42.4	33.4	24.35	16.24
wt4	0.65	3.51	7.06	10.76	-2.02	0.74	4.19	7.79	2.51	6.02	9.66	12.88	0.57	0.37	0.23	0.13	42.68	33.75	24.72	16.61
wt5	0.79	3.65	7.12	10.72	-1.83	0.91	4.28	7.79	2.73	6.28	9.75	12.84	0.51	0.35	0.22	0.12	42.21	33.43	24.51	16.51
wt6	0.61	3.39	6.92	10.6	-2.11	0.61	4.05	7.65	2.34	5.82	9.44	12.61	0.49	0.33	0.21	0.12	42.95	34.04	25.05	16.93
Wiener filter																				
fft	1.6	4.78	8.17	11.47	-0.35	2.47	5.55	8.69	4.12	7.17	10.41	13.74	0.43	0.3	0.2	0.13	42.03	32.82	24.03	16.19
wp0	1.09	4.09	7.59	11.24	-1.37	1.47	4.73	8.25	3.17	6.98	10.19	13.32	0.47	0.31	0.2	0.12	42.27	32.84	23.76	15.89
wp1	0.9	3.84	7.64	11.49	-1.62	1.17	4.86	8.59	2.87	6.62	11.14	14.98	0.52	0.33	0.19	0.11	42.81	33.45	23.88	15.94
wp2	1.07	4.04	6.58	9.47	-1.4	1.42	3.88	6.89	3.14	6.93	8.03	10.58	0.48	0.32	0.25	0.14	42.25	32.93	24.55	16.63
wp3	1.08	4.07	7.62	11.25	-1.36	1.47	4.77	8.27	3.18	7	10.21	13.31	0.48	0.31	0.2	0.12	42.03	32.78	23.61	15.81
wp4	0.9	3.83	7.67	11.52	-1.66	1.13	4.9	8.61	2.83	6.55	11.18	14.98	0.57	0.37	0.19	0.11	42.81	33.61	23.75	15.87
wp5	1.02	3.96	6.33	9.28	-1.46	1.32	3.58	6.69	3.05	6.82	8.12	10.78	0.48	0.32	0.31	0.18	41.96	32.83	25.96	17.9
wp6	0.98	3.9	7.31	10.94	-1.59	1.2	4.42	7.96	2.88	6.66	9.91	13.05	0.45	0.3	0.22	0.13	42.49	33.1	24.48	16.5
wt0	0.87	3.76	7.32	11.16	-1.66	1.1	4.54	8.27	2.8	6.44	10.61	14.43	0.5	0.33	0.2	0.11	42.22	33.36	24.31	16.24
wt1	0.73	3.58	7.15	11.06	-1.84	0.89	4.36	8.16	2.6	6.21	10.44	14.33	0.54	0.35	0.21	0.12	42.34	33.51	24.5	16.4
wt2	0.84	3.73	7.28	11.15	-1.69	1.07	4.51	8.27	2.78	6.46	10.67	14.51	0.51	0.33	0.2	0.12	42.15	33.29	24.31	16.23
wt3	0.86	3.79	7.36	11.23	-1.65	1.14	4.6	8.36	2.84	6.57	10.77	14.63	0.51	0.33	0.2	0.11	42.25	33.34	24.26	16.16
wt4	0.73	3.61	7.18	11.07	-1.85	0.92	4.38	8.16	2.57	6.19	10.4	14.3	0.6	0.38	0.22	0.12	42.67	33.77	24.67	16.53
wt5	0.85	3.69	7.19	11.02	-1.68	1.05	4.44	8.15	2.79	6.44	10.61	14.42	0.52	0.34	0.21	0.12	42.05	33.38	24.48	16.45
wt6	0.59	3.37	6.96	10.88	-2.04	0.67	4.17	7.99	2.27	5.82	10.16	14.1	0.51	0.33	0.2	0.11	42.96	34.05	25.02	16.82
Wolfe-Godsill																				
fft	1.5	4.67	7.8	10.66	-0.48	2.31	5.18	7.96	3.78	6.31	9.07	11.86	0.44	0.32	0.23	0.15	42.22	32.98	24.39	16.62
wp0	0.79	3.55	7.39	11.23	-1.69	0.9	4.58	8.31	2.77	5.35	10.74	14.59	0.59	0.41	0.21	0.12	41.61	33.2	24.49	16.45
wp1	0.59	3.24	6.47	9.3	-2.01	0.52	3.73	6.7	2.49	5.1	7.92	10.43	0.65	0.44	0.25	0.15	43	34.4	24.98	17.23
wp2	0.77	3.51	7.49	11.11	-1.74	0.83	4.63	8.13	2.73	5.32	10.14	13.24	0.59	0.41	0.2	0.12	41.91	33.47	23.85	16.07
wp3	0.79	3.55	7.52	11.35	-1.7	0.89	4.74	8.45	2.78	5.35	11.04	14.85	0.59	0.41	0.2	0.11	41.6	33.2	23.95	16.09
wp4	0.65	3.3	6.57	9.59	-2.02	0.53	3.71	6.83	2.56	5.25	8.06	10.66	0.74	0.51	0.23	0.13	42.9	34.71	25.59	17.71
wp5	0.72	3.45	7.49	11.21	-1.82	0.74	4.59	8.2	2.63	5.23	10.25	13.42	0.6	0.41	0.2	0.11	41.76	33.5	24.12	16.21
wp6	0.78	3.46	7.5	11.41	-1.9	0.63	4.69	8.49	2.51	5.24	11.05	14.95	0.52	0.36	0.19	0.11	43.36	34.51	24.1	16.16
wt0	0.58	3.17	6.16	9.07	-2.13	0.41	3.46	6.59	2.51	5.13	7.94	10.59	0.61	0.43	0.26	0.15	43.1	34.21	25.27	17.13
wt1	0.45	2.98	6.02	9.04	-2.29	0.21	3.31	6.55	2.39	5.06	8.03	10.82	0.64	0.44	0.27	0.16	43.22	34.42	25.53	17.48
wt2	0.56	3.15	6.15	9.1	-2.16	0.38	3.46	6.61	2.52	5.17	8.01	10.68	0.61	0.43	0.27	0.15	43.04	34.11	25.2	17.1
wt3	0.54	3.17	6.21	9.18	-2.17	0.41	3.51	6.67	2.51	5.18	8.04	10.74	0.62	0.43	0.26	0.15	43.11	34.17	25.23	17.02
wt4	0.43	3	6.09	9.16	-2.33	0.21	3.37	6.63	2.38	5.17	8.26	11.1	0.72	0.5	0.31	0.18	43.44	34.87	25.96	17.79
wt5	0.57	3.11	6	8.79	-2.12	0.37	3.34	6.34	2.49	5.11	7.84	10.41	0.63	0.44	0.27	0.16	42.99	34.27	25.51	17.65
wt6	0.47	2.91	5.91	8.94	-2.26	0.16	3.21	6.43	2.36	5.03	7.98	10.7	0.57	0.39	0.25	0.14	43.64	34.85	26.12	18.07

For the wavelet domain, 160 samples of input data block are prefixed with last 96 samples of previous block for 256 samples DWT (wt) and DWPT (wp). The levels of DWT are to the highest possible value 8. After filtering in the wavelet domain, the inverse wavelet transform is performed and the last 160 samples are taken to resemble an output data stream. Seven different wavelet bases were tested. They are Battle-Lemarie (0), Burt-Adelson (1), Coiflet-6 (2), Daubechies-D20 (3), Haar (4), Pseudo-coiflet-4 (5), and Spline-3-7 (6).

The enhanced results were compared with the original clean speech signals to obtain objective quality measurements. The global SNR (gSNR), segmental SNR (sSNR), frequency-weighted segmental SNR (fwsSNR) were used to measure the improvements, while the Itakura-Saito distance (isD), and weighted spectral slope (WSS) were used to measure the distortions. The average scores for various algorithms using the Fourier and wavelet transforms are shown in Table 1.

The results show that the Fourier transform (fft) is still the best in terms of noise reduction quality. The next is the wavelet packet transform (wp), and last is the wavelet transform (wt). Among the wavelet bases, the Battle-Lemarie (0) and Daubechies_20 (3) are better than the others at low SNR signals. All the methods have achieved noise reduction in terms of improving three SNR indexes, except few cases at highest input SNR. Many wavelet packets (wp) have less distortion and better SNR than the FFT at high SNR signals. However the wavelet transforms (wt) are worse, and even worse than the original noisy input in many cases.

Table 2 shows a CPU time comparison for the transforms with respect to the STFT as one unit.

Table 2 - Computer CPU times for various transforms in reference to STFT time

Abr.	transforms	Implementation	CPU Time (time of STFT)
fft	Shot time Fourier transform	Custom implementation of FFT	1
wp0	Battle-Lemarie wavelet packet	UBC Imager Wavelet Package [13]	10.304
wp1	Burt-Adelson wavelet packet	UBC Imager Wavelet Package	3.016
wp2	Coiflet-6 wavelet packet	UBC Imager Wavelet Package	7.779
wp3	Daubechies-D20 wavelet packet	UBC Imager Wavelet Package	8.608
wp4	Haar wavelet packet	UBC Imager Wavelet Package	0.949
wp5	Pseudo-coiflet-4 wavelet packet	UBC Imager Wavelet Package	4.745
wp6	Spline-3-7 wavelet packet	UBC Imager Wavelet Package	4.356
wt0	Battle-Lemarie wavelet transform	UBC Imager Wavelet Package	2.458
wt1	Burt-Adelson wavelet transform	UBC Imager Wavelet Package	0.882
wt2	Coiflet-6 wavelet transform	UBC Imager Wavelet Package	1.898
wt3	Daubechies-D20 wavelet transform	UBC Imager Wavelet Package	2.084
wt4	Haar wavelet transform	UBC Imager Wavelet Package	0.390
wt5	Pseudo-coiflet-4 wavelet transform	UBC Imager Wavelet Package	1.255
wt6	Spline-3-7 wavelet transform	UBC Imager Wavelet Package	1.153
wt7	Haar wavelet transform	Custom implementation	0.067
wt8	Daubechies-D4 wavelet transform	Custom implementation	0.085

The UBC image wavelet package is used for comparison of wavelet bases (0 - 6). Because it is not optimized for speed, wt7 and wt8 of custom implementations are added for a better computer time reference.

CONCLUSION

Noise reduction filters are formulated for both Fourier and wavelet domains. Experiments using real-world noise speech data have shown that the Fourier transform, the wavelet packet transform, and the wavelet transform are the best, second, and last respectively in general SNR sense. The wavelet packet transform can achieve less distortion and is better for high SNR signals. There are still many incomplete wavelet and wavelet packet transforms not being tested. Therefore, a better solution in wavelet domain is still possible.

REFERENCES

1. D.L. Donoho, *De-noising by Soft-thresholding*, IEEE Trans. Inform. Theory, 41(3):613-627, May 1995
2. M. Lang, H. Guo, J.E. Odegard, C.S. Burrus, and R.O. Wells, *Noise Reduction Using an Undecimated Discrete Wavelet Transform*, IEEE,Signal Processing Letters vol. 67, pp. 1586–1604, Dec. 1995.
3. Q. Fu and E. Wan, *Perceptual Wavelet Adaptive Denoising of Speech*, In EUROSPEECH2003, pp. 577-580, 2003
4. E. Jafer and A.E. Mahdi, *Wavelete-based Perceptual Speech Enhancement Using Adaptive Threshold Estimation*, In EUROSPEECH2003, pp. 569-572, 2003
5. C.T. Lu, C.M. College, *Speech Enhancement Using Robust Weighting Factors for Critical-Band-Wavelet-Packet Transform*, In ICASSP2004, Montreal, Canada, pp. 721-724, 2004
6. G.H. Ju and L.S. Lee, *Speech Enhancement and Improved Recognition Accuracy by Integrating Wavelet Transform and Spectral Subtraction Algorithm*, In EUROSPEECH2003, pp. 1377-1380, 2003
7. M. Berouti, R. Schwartz, and J. Makhoul, *Enhancement of Speech Corrupted by Acoustic Noise*, In. ICASSP1979, Washington, DC, Apr. 1979, pp. 208–211.
8. N. Fan, *Low Distortion Speech Denoising Using an Adaptive parametric Wiener Filter*, In ICASSP2004, Montreal, Canada, pp. 309-312, 2004
9. N. Virag, *Single Channel Speech Enhancement Based on Masking Properties of the Human Auditory System*, IEEE Trans. Speech and Audio Processing, vol. 7, pp. 126–137, Mar. 1999
10. P. J. Wolfe, and S. J. Godsill, *Efficient Alternatives to the Ephraim and Malah Suppression Rule for Audio Signal Enhancement*, EURASIP Journal on Applied Signal Processing, EURASIP JASP 2003: 10, pp. 1043-1051, 2003
11. Y. Empraim, and D. Malah, *Speech Enhancement Using a Minimum Mean-Square Error Short-Time Spectral Amplitude Estimator*, IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-32, pp. 1109–1121, Dec. 1984.
12. R. Martin, *Noise Power Spectral Density Estimation Based on Optimal Smoothing and Minimum Statistics*, IEEE Trans. Speech and Audio Processing, vol. 9, no. 5, pp. 504-512, July 2001.
13. B. Lewis, *Wvlt – the Imager Wavelet Library*, http://www.cs.ubc.ca/nest/imager/contributions/bobl/wvlt/top.html, August 1995.