IOSR Journal of Engineering (IOSRJEN)
e-ISSN: 2250-3021, **p**-ISSN: 2278-8719
Vol. 3, Issue 10 (October. 2013), ||V3|| PP 37-44

Comparison of Different Speech Enhancement Techniques

M.Balasubrahmanyam, G.Srinivasa Rao, R.Rahul
mbalu.422@gmail.com (srinu.rgm@gmail.com) (rahul_rayavarapu@yahoo.co.in)
(Assistant professor in SVIET, Nandamuru) (Assistant professor in BVSR), (KLU)

Abstract: - The speech enhancement is one of the important techniques used to improve the quality of a speech signal i.e. degraded by noise. Speech Enhancement using Kalman Filter require calculating the parameters of AR (auto-regressive) model, and performing a lot of matrix operations, which is non-adaptive. Speech Enhancement using Weiner filter very hard to find out the inverse matrix operations in the time domain but desired output is required. Adaptive Kalman Filter is constantly update the estimation of background noise, which is adaptive. AKF used to eliminate the matrix operations, reduces the calculating time and reduces the complexity. Perceptual Weighting filter is used to improve the performance of speech enhancement system. However the perceptual characteristics of the speech signal depends upon the perceptual characteristics of human ear. Compare the simulation results and different parameters(SNR, MSE, MMSE & CPU time), and also observe which one the better technique for speech enhancement.

Keywords: - *SpeechEnhancement, Kalman Filter, Adaptive Kalman Filter, Perceptual Weighting Filter, MSE, MMSE, SNR.*

I. INTRODUCTION

The background noise is a dominant source of errors in speech recognition systems. Noise reduction for speech signals has therefore application in entry procedures of those systems. The Kalman filter is known in signal processing for its efficient structure. There are many studies of using of Kalman filtering for noise reduction in speech signals. Speech signals are modeled as stationary AR process. Modeling and filtering noisy speech signals in the sub band domain. Since the power spectral densities (PSD's) of sub band speech signals are flatter than their full band signals, low-order AR models are satisfactory and only lower-order Kalman filters will be required. In the next focus on first-order modeling.Wiener filter is used to produce estimated pure signal from a given noise speech signal. Wiener filter is formulated to map an input signal to an output that is as close to a desired signal as possible. In the perceptual Weighting filter to improve quality of the speech signal based on the human auditory characteristics.

II. WEINER FILTER

Wiener filter is used to produce estimated pure signal from a given noise speech signal [3]. Wiener filter is formulated to map an input signal to an output that is as close to a desired signal as possible. It is a class of optimum linear filter, involves linear estimation of desired signal by adjusting the weights mean square error reduced between the desired signal X(f) and the filter output $\hat{X}$(f). We consider some applications of the Wiener filter in reducing broadband additive noise, in time-alignment of signals in multichannel or multisensory systems, and in channel equalization. In the frequency domain, the Wiener filter output Y(f) is the product of input signal X(f) and the filter frequency response W(f) .

$X(f)=Y(f).W(f)$ (2.1)

The estimation error signal $E(f)$ is defined as the difference between the desired signal $X(f)$ and the filter output $\hat{X}(f)$

$E(f)= X(f)-\hat{X}(f)$(2.2)

The mean square error at a frequency f is given by

$[|E(f)|^2]=E[|X(f)-\hat{X}(f)$(2.3)

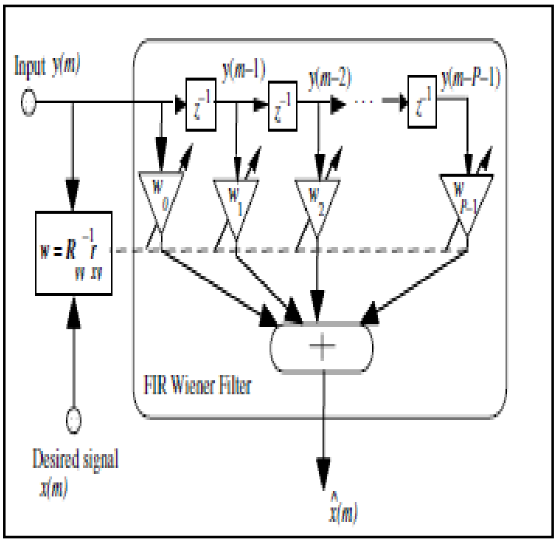


Fig. 2.1.1: Illustration of Wiener structure

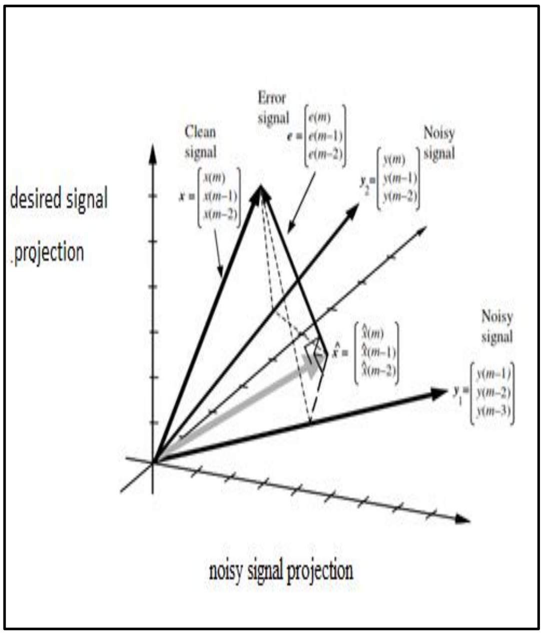


Fig. 2.1.2: The least square error projection of a desired signal vector x onto a plane containing the input signal vectors y_1 andy2

Clean speech signal is estimated through the Wiener filter. There are so many algorithms in the literature; it is extremely difficult if not impossible to find a universal analytical tool that can be applied to any speech enhancement algorithm. We choose Wiener filter as the basis since it is the most fundamental approach, and many algorithms are closely connected with this technique. Moreover, Wiener filter introduces less musical noise than spectral subtraction methods. Let the noisy signal can be expressed as:

$$y(n) = s(n) + d(n) \qquad (2.4)$$

Where $x(n)$ is the original clean speech signal and $d(n)$ is the additive random noise signal, uncorrelated with the original signal. Taking DFT to the observed signal gives

$$Y(m,k) = S(m,k) + D(m,k) \qquad (2.5)$$

Where $m = 1,2,\ldots,M$ is the frame index, $k = 1,2,\ldots.,K$ is the frequency bin index, M is the total number of frames and K is the frame length, $Y(m,k), S(m,k) and D(m,k)$ represent the short time spectral components of the $y(n), S(n) and (n)$, respectively. Basis speech enhancement methods involve estimating every frequency component of the clean speech $\hat{S}(m,k)$

$$\hat{S}(m,k) = H(m,k)Y(m,k) \qquad (2.6)$$

Where $H(m,k)$ is the noise suppression filter (denoising filter) chosen according to the a MMSE criterion .The error signal generated by this filter is

$$e(m,k) = \hat{S}(m,k) - S(m,k) = (H(m,k)-1)S(m,k) + H(m,k)D(m,k) \quad (2.7)$$

The first term in the equation (2.4) describes the speech distortion caused by the spectral weighting which can be minimized using $H(m,k) = 1$. The second term in the above equation is the residual noise distortion which can be minimized if the spectral weighting $H(m,k) = 0$ .Musical noise results from the pure tones present in the residual noise. In general noise suppression filter can be expressed as a function of the a posteriori *SNR* and a priori *SNR* given by

$$\gamma(m,k) = \frac{|Y(m,k)|^2}{\Gamma_d(m,k)} \qquad (2.8)$$

$$\xi(m,k) = \frac{\Gamma_s(m,k)}{\Gamma_d(m,k)} \qquad (2.9)$$

Where $\Gamma_d(m,k) = E\left\{|D(m,k)|^2\right\}$, by definition , is the noise power spectrum , an estimate of the which can be made easily during speech pauses and $\Gamma_s(m,k) = E\left\{|S(m,k)|^2\right\}$. An estimate of $\hat{\xi}(m,k)$ of $\xi(m,k)$ is given by the well-known decision directed approach and is expressed as

$$\hat{\xi}(m,k) = \alpha \frac{|H(m-1,k)Y(m-1,k)|^2}{\Gamma_d} + (1-\alpha)P'[V(m,k]. \qquad (2.10)$$

Where $V(m,k) = \gamma(m,k) - 1,\ P[x] = x$ if $x \geq 0$ and $P[x] = 0$ otherwise.

The noise suppression gain function is given as

$$H(m,k) = \frac{\xi(m,k)}{1+\xi(m,k)} \qquad (2.11)$$

III. LIMITATIONS OF WIENER FILTER

Apart from the performance being limited by the accuracy of noise estimation, which additionally is limited by the performance of speech/pause detectors, the main problem with Wiener filtering is the processing distortions caused by random variations of the noise spectrum. The three sources have been attributed to the distortion of the instantaneous estimate of the magnitude:

a) the finite variance of the instantaneous noise power spectrum.

b) the cross-product terms from above equation, and

Irrespective of the methods used for estimating the noise statistics, the true short spectrum of the noise for specific segment being processed, will always have a finite variance and thus the noise estimate will always be over or under the estimate of the true noise level. Therefore, wherever the noisy signal level is near the level of the estimated noise spectrum, spectral subtraction results in some randomly located negative values for the estimated clean speech magnitude. The non-linear mapping of the negative, or small valued spectral estimates, results in the estimated magnitude spectrum to consist of a succession of randomly spaced spectral peaks. This leads to an annoying residual noise, also called musical noisedue to their narrow band spectrum and presence of tone-like characteristics. This noise although very different from the original noise, can sometimes be very disturbing.

A poorly designed Wiener filter, can sometime results in a signal that is of a lower perceived quality and lower information content, than the original noisy signal. Most of the research in past decade been focused in ways to combat the problem of musical noise. It is literally impossible to minimize musical noise without affecting the speech, and hence as mentioned earlier, there is a tradeoff between the amount of noise reduction and speech distortion. It is due to this fact that several perceptual based approaches, wherein instead of completely eliminating the musical noise (and introducing distortion), the noise is masked taking advantage of the masking properties of the auditory system.

Another artifact is phase distortion, caused by the assumption that the ear is insensitive to the phase. As mentioned earlier, the phase is taken from the noisy signal. Experiments with “ideal” spectral subtraction (where the magnitude of each frame is taken from the clean signal and the phase from the noisy signal) show that this becomes significant as the SNR decreases, resulting in a “hoarse” or “rough” sounding voice. However as mentioned earlier, it is very difficult to estimate both magnitude and phase, and using the noisy phase is an acceptable trade off in algorithms based on short-term magnitude estimation.

IV. KALMAN FILTER

Kalman filter is a mathematical method named after Rudolf E. Kalman 1960, through Peter Swerling actually developed a similar algorithm earlier. Kalman(1960) and Kalman and buky (1961). It was developed as a recursive solution to the discrete-data linear filtering problem.

A Kalman filter is simply an optimal recursive data processing algorithm. There are many ways of defining optimal, dependent upon the criteria chosen to evaluate performance.

The Kalman filter is optimal with respect to virtually any criterion that makes sense. One aspect of this optimality is that the Kalman filter incorporates all information that can provided to it. The different methods were proposed by [2]-[6]. The many of themethods needs to estimates the parameters of *AR* model at first, and then perform the noise reduction using Kalman filter algorithm. In this process, the calculation of LPC (linear prediction coding) coefficient and inverse matrix increase the complexity of the filtering algorithm. [3] and [4] have been given a simple Kalman filtering algorithm without calculating LPC coefficient in the AR model, but the algorithm still contains a large number of redundant data and matrix inverse operations. This algorithm is non-adaptive algorithm.

Kalman Filter Drawbacks:

1. Lot of matrix operations are used, which usually non-adaptive.
2. Complexity.
3. Calculating time is more.

3.1Kalman Filtering Algorithm

A clean speech signal $s(n)$ can be defined as a p-th (AR) autoregressive process and n- th of the noisy speech signal $y(n)$ is expressed as

$s(n) = \sum_{i=1}^{p} a_i\,(n)s(n-i) + w(n)$(3.1.1)

In (3.1.1), a_i is the i-th AR coefficient, $w(n)$ is the white Gaussian noise which the mean is zero and the variance is known.

$y(n)=s(n)+v(n)$ (3.1.2)

In (3.1.2) $v(n)$ is the additive observation noise, its mean is zero and variance is known.

In this paper, it is assumed that the variance is known, but in practice we need to estimate it by the initial segment included in the $y(n)$.

(3.1.1) and (3.1.2) can be expressed as the state equation and the observation equation which are given by

State equation is

$x(n)=F(n)x(n-1)+Gw(n)$ (3.1..3)

Observation equation is

$y(n)=Hx(n)+v(n)$ (3.1.4)

$F(n)$ is the p by p transition matrix expressed as

$$F(n) = \begin{bmatrix} 0 & 1 & 0 \dots \dots . & 0 \\ \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} (3.1.5)$$

Where G is the input vector and H is the observation vector.

By using the *LPC* coefficient in the conventional Kalman filter is to estimate the observations of the speech signal, this process is easy. This part spends half the time of the total algorithm.

The transition matrix F and the observation matrix H are modified. They has defined as

$$F=H=\begin{bmatrix} 0 & 0 & 0 \dots \dots . & & 0 \\ \\ 1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 \dots . & 1 & \cdots & 0 \end{bmatrix} (3.1.6)$$

It is also defined as the $p\times 1$ state vector $Z(n)=[\ s(n) \dots . s(n-p+1)\ \ s(n-p+2)]$, the $p\times 1$ input vector $Q(n)=[s(n)\ \ 0\ \dots .\ \ 0]$, and the $1\times p$ observation vector $R(n)=[1, v(n) \dots . v(n-p+2)]$.

Finally, (3.1.3) and (3.1.4) can be written into the matrix operations by

State equation is

$X(n)=F\times X(n-1)+Q(n)$ (3.1.7)

Observation equation is

$Y(n)=H\times X(n)+R(n)$ (3.1.8)

State equation consisted of the speech signal, and an observation equation consisted of the speech signal and additive noise [3].

The purpose of each iteration of a Kalman filter is to update the estimation of the state vector of a system (and the covariance of that vector) based upon the information in a new observation.

The recursive estimation of Kalman filtering algorithm is shown below

$X(0|0) = 0, P(0|0) = I$(3.1.9)

$R_V\,(n) = \delta_v^2$ (3.1.10)

$R_S(n)[i,j] = \begin{cases} E\big(Y(n) * Y(n)\big) - \delta_v^2 \text{ (I, j = 1)} \\ 0 \quad\quad \text{otherwise} \end{cases}$(3.1.11)

[iteration]

$P(n|n\text{-}1)=F*P(n\text{-}1|n\text{-}1)*F^T+G*R_S(n)*G^T$ (3.1.12)

$K(n)=P(n|n\text{-}1)*G^T|G * P(n|n-1) * G^T + R_V\,(n)$ (3.1.13)

$X(n|n\text{-}1)=F*X(n\text{-}1|n\text{-}1)$ (3.1.14) $X(n|n)=X(n|n\text{-}1)+K*(y(n)\text{-}G*X(n|n\text{-}1))$ (3.1.15)

$P(n|n)=(I\text{-}K(n)*G)*P(n|n\text{-}1)$ (3.1.16)

$S(n)=K(n)*y(n)$ (3.1.17)

In the above case the noise variance δ_v^2 is known. This algorithm abrogates the computation of the AR coefficient.

V. ADAPTIVE KALMAN FILTERING ALGORITHM

Due to the noise changes with the surrounding environment, it is necessary to constantly update the estimation of noise. So we can get a more accurate expression of noise. An adaptive Kalman filtering algorithm for speech enhancement can adapt to any changes in environmental noise, and also it can constantly update the estimation of background noise.

Everyone known Kalman filtering algorithm is very well. Adaptive kalman filtering algorithm can estimate system process noise and measurement noise on-line according to the measured value and filtered value, tracking changes of noise in real time to amend the filter parameters, and improve the filtering effect.

In this adaptive kalman filter, we can set a reasonable threshold, it is used to determine whether the current speech frame is noise or not. It consists of mainly two steps: one is updating the variance of the environmental noise R_v(n), and the second one is updating the threshold *U*.

1) Updating the variance of the environmental noise by

$R_v(n)=(1\text{-}d)\times R_v(n)+d\times R_u(n)$ (4.1)

In above equation d is the loss factor that can limit the length of the filtering memory, and enhance the role of new observations under the current estimates. Making new data play a major role in the estimation, and leaving the old data forgotten gradually. According to the [7]……..? its formula is

$d=1\text{-}b/1\text{-}b^{t+1}$(4.2)

b is the forgetting factor(0<b<1), usually ranged from 0.95 to 0.99. In this paper the value of b is considered 0.99.

Before implementation of (18), we will compare between the variance of the current speech frame $R_u(n)$ and threshold *U* which has been updated in the previous iteration. If $R_u(n)$ is less than or equal to *U* the current speech frame can be considered as noise, and then the algorithm will re-estimate the noise variance.

In this paper,$R_u(n)$ can't replace $R_V(n)$ directly. In order to reduce the error, we used.

2) Updating and threshold by

$U=(1\text{-}d)\ U+\ d\ \ R_u(n)$ (4.3)

In (17) , d is used again to reduce the error. However, there will be a large error when the noise is large, because the updating threshold *U* is not restricted by the limitation $R_u(n)<U$. It is only affected by $R_u(n)$. So we must add another limitation before implementation of (20). In order to rule out of speech frames which their *SNR* (Signal-to-noise rate) is high enough, it is defined that δ_r^2 is the variance of pure speech signals, δ_x^2 is the variance of the input noise speech signals, and δ_v^2 is the variance of background noise.we calculate two *SNRs* and compare between them.According to [6], one for the current speech frame is

$SNR_1(n)=10\times\ log_{10}\ (\frac{\delta_r^2(n)-\ \delta_v^2(n)}{\delta_v^2(n)})$ (4.4)

Another for the whole speech signal is

$SNR_0(n)=10\times log_{10}(\frac{\delta_r^2-\ \delta_v^2(n)}{\delta_v^2(n)})$ (4.5)

In (4.4) and (4.5), n is the number of speech frames, and δ_v^2 has been updated I order to achieve a higher accuracy. The speech frame is noise when $SNR_1(n)$ is less than or equal to $SNR_0(n)$, or $SNR_0(n)$ is less than zero and then these frames will be follow the second limitation ($R_u(n)\leq U$). However, if $SNR_1(n)$ is larger than $SNR_0(n)$, the noise estimation will be attenuated to avoid damaging the speech signals.

The recursive estimation of AdaptiveKalman filtering algorithm is shown below

```
[Intialization]
$S(0)=0, Rv(1)=\delta_v^2(1)$   (variance of the first speech frame)   (4.6)
[iteration]
 If$SNR1(n) <= SNR0(n)$   ||
$SNR0(n) < 0$   then   (4.7)
If  $Ru(n) \le \boldsymbol{U}$then
$Rv(n)$=variance of the environmental noise
$Ru(n)$=variance of the current speech frame
 1.$R_v(n)=(1-d)\times R_v(n)+d\times R_u(n)$     (4.8)
          End
2.$U=(1-d)U+dR_u(n)$ (4.9)
          Else
3.$Rv(n)=Rv(n)/1.2$(4.10)
            End
4.$Rs(n)=Rs(n)=E\big(Y(n)*Y(n)\big)-Rv(n)$(4.11)
    5.$K(n)=Rs(n)=Rs(n)/(Rs(n)+Rv(n)$ (4.12)
6.$S(n)=K(n)*y(n)$            (4.13)
```

VI. PERCEPTUAL WEIGHTING FILTER ALGORITHM

Weighting filters are widely used in the measurement of electrical noise on telephone circuits, and in the assessment of noise as perceived through the acoustic response of different types of instruments.

Usually, the perceptual weighting procedure often Results in speech coder performance. A commonly used weighting filter is based on the linear prediction coefficients that represent the short-term correlation in the speech signal [8]. A representative perceptual weighting filter $W(z)=\frac{A(z)}{A(\frac{z}{\gamma})}=\frac{1-\sum_{i=1}^{p}a_iZ^{-i}}{1-\sum_{i=1}^{p}a_i\gamma^iZ^{-i}}$ is given by Where

$A(z)$ represents the pth-order LP analysis filters and a_i is the LP coefficient. To compute the filter coefficients for this filter, linear predictive analysis is used in [8]. Also, γ is a perceptually weighting factor which does not alter the center formant frequency, but just broadens the bandwidth of the formants. Specifically, frequency broadening δ_f given by $\delta f=(fs/\pi)\ln\gamma$. Where fs is the sampling frequency in hertz.

For that reason, the weighting filter deemphasizes the formant structure while emphasizing the formant valleys of the speech signal. This results in a larger matching error in the region of the formants, where spectral masking makes the auditory systems less sensitive to quantization error. The most suitable value of γ is subjectively selected by listening tests, and for 8KHZ sampling, γ is adopted as 0.9here.

Table 8.1.1: Different filtering methods comparision of *SNR* for male and female speech signal

SNR$_{in}$ [dB]	**Weiner Filter**	**Kalman filter SNR$_{out}$ [dB]**	**Adaptive KF SNR$_{out}$ [dB]**	**Perceptual weighting FilterSNR$_{out}$ [dB]**
Male=3.40	**4.78**	**4.90**	**5.05**	**9.014**
Female=6.70	**7.23**	**7.90**	**9.56**	**13.05**

Table 8.1.2: Different filtering methods comparision of *MSE* for male and female speech signal

Speech Signal	**Weiner Filter**	**Kalman Filter**	**Adaptive Klman Filter**	**Perceptual Weighting Filter**
Male	**0.423**	**0.432**	**0.0451**	**0.002**
Female	**0.124**	**0.324**	**0.0321**	**0.001**

Table 8.1.3: Different filtering methods comparion of CPU time for male and female speech signal

Speech Signal	**Weiner Filter**	**Kalman Filter**	**Adaptive Klman Filter**	**Perceptual Weighting Filter**
Male	**9.723 sec**	**9.603 sec**	**5.773 sec**	**3.801 sec**
Female	**8.645 sec**	**8.560 sec**	**3.456 sec**	**2.956 sec**

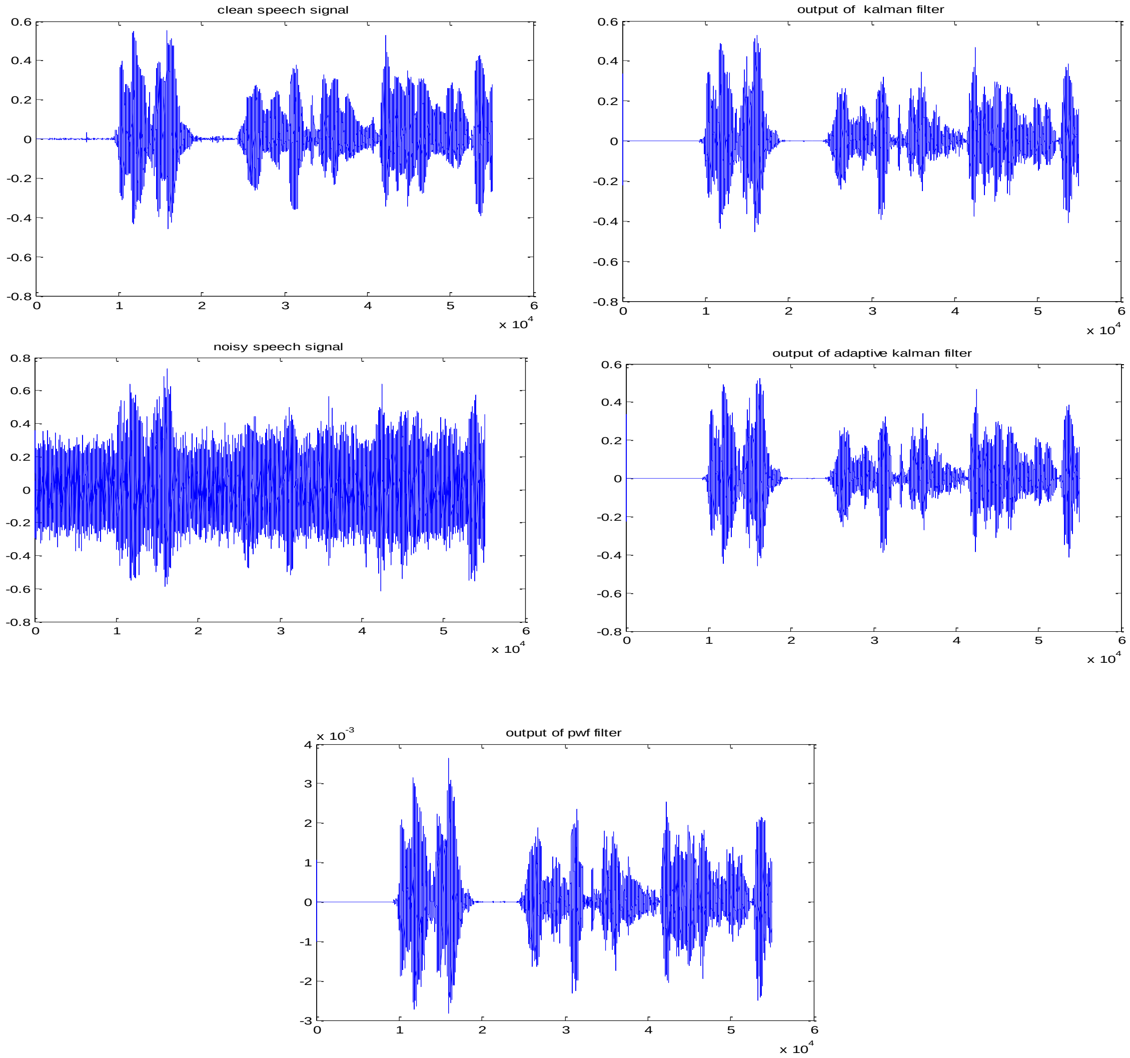
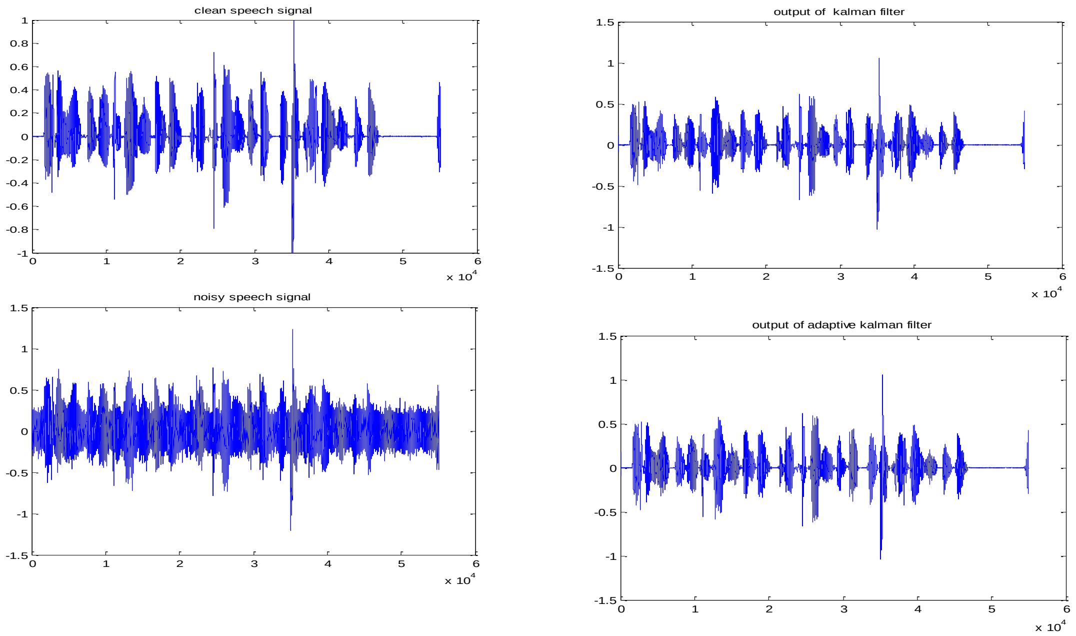


Fig.1. the filtering results for the male speech with noise.



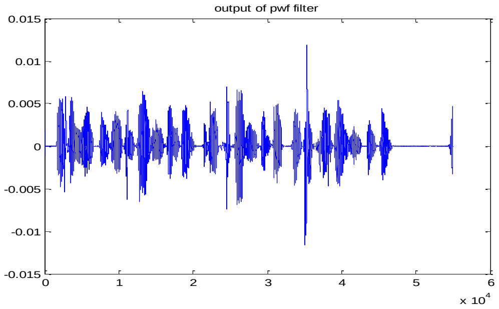


Fig.2. the filtering results for the female speech with noise.

VII. CONCLUSION

Comparison of the simulation results Adaptive KalmanFilter and Perceptual Weighting Filter Algorithms were better than the Weiner Filter and Kalman Filter. In the Weiner Filter calculation of the inverse matrix operations are hard in time domain, and in Kalman Filter lot of matrix operations are required, calculating time is more, and more complexity. In the perceptual Weighting Filter provide human auditory characteristics.

REFERENCES

[1] Quanshen Mai, Dongzhi He, YibinHou, Zhangqin Huang, “Speech Enhancement Using AdapativeKalmanFilterining Algorithm”,pp.327-332, August 2011

[2] ZHANG Xiu-zhen, FU Xu-hui, WANG Xia, Improved kalman filter method for speech Improvendenhancement. Computer Applications, Vol.28, enhancement.Computer pp.363-365, Dec 2008.

[3] Naritanabe, Toshiniro Furukawa, Shigeo Ts,ujji. Fast noise Suppression algorithm with Kalman filter Theory. Second International Symposium on Universal Communication, pp.411-415, 2008.

[4] Nri Tanabe, ToshiniroFurukawa,Hideaki Matsue and Shigeo Tsujji. Kalman Filter for Robust NoiseSuppression in White and Colored Noise.IEEE International SymposiumonCirciuts and Systems 2008.1172-1175.

[5] WU Chung-ling,HAN Chong-zhao. Square-root Quadrature Filter.Acta Electronica Sinics,Vol.37, No.5, pp.987-992, May.2009.

[6] SU Wan-Xin, HUANG Chun-mei LIU Peiwei, MA Ming-long.Application of adaptive Kalman filter in intial alignment of inertial navigation system. Journal of Chinese Inertial Technology Vol.18, No.1, pp. 44-47,Feb.2010.

[7] GAO Yu,ZHANG Jain-qui. Kalman filter with Wavelet-Based Unknown Measurement NoiseEstimation and its Application for information Fusion. Acta Electronica Sinica, Vol.35, No.1, pp.108-111,Jan.2007.

[8] XIE Hua.Adaptive Speech Enhancement Base on Discrere Cosine Transformation in High Noise Environment. Harbin Engineering University,2006.

[9] Ioon-Hyuk Chang, perceptual Weighting Filter for robust speech modification, pp.1090-1094.