Speech Enhancement Using Kalman Filter-Based Algorithms

1. Project Members

- Enes Arda 2017401111

2. Objective:

For one reason or another, speech signals get corrupted by background noise, which significantly deteriorates the intelligibility of the speech signal. Clean speech signals are important not only for communication, but also for further processing, such as in speech coders and automatic speech recognition systems. Therefore, the objective of this project is to enhance the quality and the intelligibility of a corrupted speech signal by using Kalman Filter-Based Algorithms.

3. Introduction

Many different methods have been introduced for speech enhancement problem. One of the most important methods is stationary Wiener filtering method. As suggested in [1], due to the non-stationary nature of the speech signal, this method doesn't perform very well. Although non-stationary Wiener filtering method is introduced, where the Wiener filter is designed for short-time speech segments, this also doesn't utilize the knowledge about speech production process. Hence in [1] Kalman filtering method is introduced, which exploits the speech production model and accounts for the non-stationary nature of the speech. It is shown in [1] that Kalman filtering approaches perform better than Wiener filtering approaches. Moreover, in [2] Kalman filtering methods are compared with other approaches, such as spectral subtraction and short-time spectral amplitude estimator and it is concluded that Kalman filter-based algorithms performed better in most of the objective and subjective tests. Application of Kalman filter for speech enhancement has two parts:

1. Estimation of AR coefficients and noise parameters.
2. Applying Kalman filtering algorithms using the estimated parameters.

Different methods have been proposed for both steps. I will try to implement different parameter estimation techniques and different Kalman filter-based algorithms to find the best method to enhance speech signal.

4. Data

For clean speech signals LibriSpeech Audio Book Corpus [5] will be used. This corpus includes approximately 1000 hours of English speech recorded at 16kHz. For background noise the QUT-NOISE corpus [7] will be used. This dataset includes 20 noise sessions of at least 30 minute each. 5 common background noise scenarios are used: Cafe, Home, Street, Car and Reverb. The recordings are done at the sampling rate 16kHz and 48kHz. In addition, various non-speech sounds such as water sound, snare, door moving, footsteps can also be used as noise. The dataset in [6] can be used for this purpose. It includes 20 different non-speech sounds recorded at 16kHz.

5. Methods

5.1 Kalman Filter

Vector Kalman Filter assumes a state model of the form

$$\mathbf{S}(n) = \mathbf{A}(n)\mathbf{S}(n-1) + \mathbf{W}(n) \qquad (1)$$

and an observation model of the form

$$\mathbf{X}(n) = \mathbf{H}(n)\mathbf{S}(n) + \mathbf{v}(n) \qquad (2)$$

where

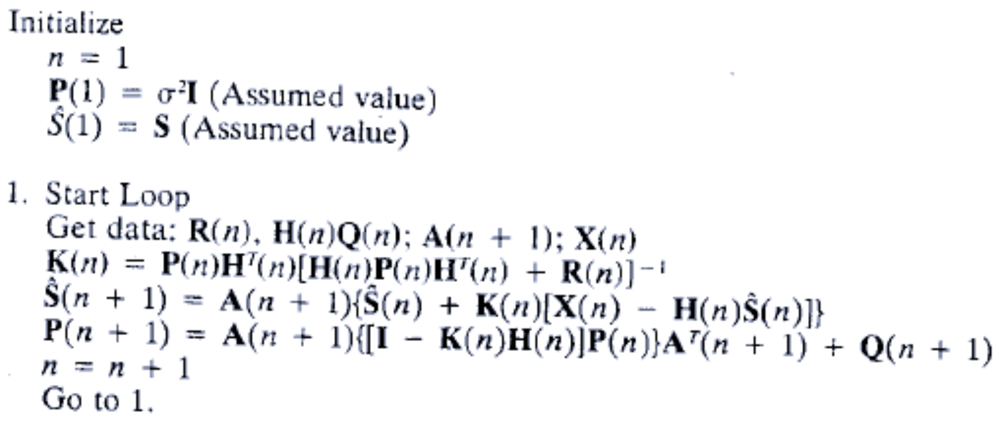
$\mathbf{S}(n)$ = signal or a state vector
$\mathbf{A}(n)$ = state matrix
$\mathbf{W}(n)$ = zero-mean Gaussian white noise uncorrelated with $\mathbf{S}(1)$ and $\mathbf{v}(n)$
$\mathbf{X}(n)$ = observation
$\mathbf{H}(n)$ = observation matrix
$\mathbf{v}(n)$ = observation noise; zero-mean Gaussian white noise uncorrelated with $\mathbf{S}(1)$ and $\mathbf{W}(n)$
$\mathbf{S}(1)$ = initial state; zero-mean Gaussian random variable with covariance matrix $\mathbf{P}(1)$

and it is assumed that the following covariance matrices are known:

$$E[\mathbf{W}(k)\mathbf{W}^{\mathbf{T}}(i)] = \begin{cases} \mathbf{Q}(k), & k = i \\ 0 & k \neq i \end{cases}$$

$$E[\mathbf{v}(k)\mathbf{v}^{\mathbf{T}}(i)] = \begin{cases} \mathbf{R}(k), & k = i \\ 0 & k \neq i \end{cases}$$

With these assumptions the optimum linear Kalman filter in the Minimum Mean Squared Error (MMSE) sense can be implemented iteratively in the following algorithm. Because it is in recursive form it also uses a minimum amount of storage.



```
Initialize
    $n = 1$
    $\mathbf{P}(1) = \sigma^2\mathbf{I}$ (Assumed value)
    $\hat{S}(1) = \mathbf{S}$ (Assumed value)

1. Start Loop
    Get data: $\mathbf{R}(n)$, $\mathbf{H}(n)\mathbf{Q}(n)$; $\mathbf{A}(n + 1)$; $\mathbf{X}(n)$
    $\mathbf{K}(n) = \mathbf{P}(n)\mathbf{H}^T(n)[\mathbf{H}(n)\mathbf{P}(n)\mathbf{H}^T(n) + \mathbf{R}(n)]^{-1}$
    $\hat{\mathbf{S}}(n + 1) = \mathbf{A}(n + 1)\{\hat{\mathbf{S}}(n) + \mathbf{K}(n)[\mathbf{X}(n) - \mathbf{H}(n)\hat{\mathbf{S}}(n)]\}$
    $\mathbf{P}(n + 1) = \mathbf{A}(n + 1)\{[\mathbf{I} - \mathbf{K}(n)\mathbf{H}(n)]\mathbf{P}(n)\}\mathbf{A}^T(n + 1) + \mathbf{Q}(n + 1)$
    $n = n + 1$
    Go to 1.
```

Fig1. Kalman Filtering Algorithm

In Fig1 $\hat{\mathbf{S}}(n)$ is an MMSE estimate of $\mathbf{S}(n)$ which can be expressed as
$\hat{\mathbf{S}}(n) = E[\mathbf{S}(n) \mid \mathbf{X}(n-1), \mathbf{X}(n-2) \ldots \mathbf{X}(1)]$

If our speech enhancement problem can be formulated as the state and the observation models in (1) and (2) Kalman Filtering Algorithm seen in Fig1 can be applied to the corrupted speech signal.

5.2 Proposed Model For Speech

Just like the basic idea in Linear Prediction Coding it can be assumed that the current speech sample can be closely approximated as a linear combination of past samples. Hence, speech can be represented by an autoregressive process such that the speech signal at the n^{th} time instant s(n) is given by:

$s(n) = \sum_{k=1}^{q} a(n,k)s(n-k) + e(n)$ (3), where e(n) is the excitation signal, which is an impulse train if the speech is voiced and zero mean white gaussian noise if the speech is unvoiced. In order for our model to accommodate both voiced and unvoiced speech the following model for the excitation signal is used:

$e(n) = b(n,p_n)e(n-p_n) + d(n)$ (4), where d(n) is generated by a zero-mean white Gaussian process with variance $\sigma^2_{d(n)}$, p_n is the instantaneous pitch period and $b(n,p_n)$ is the degree of voicing.

In order to represent the unvoiced speech $b(n,p_n)$ is set to 0 and therefore e(n) = d(n). Therefore, e(n) is a white Gaussian noise. On the other hand, in order to represent the voiced speech p_n is set to the pitch period of the voiced speech, $b(n,p_n)$ is set close to 1 and $\sigma^2_{d(n)}$ is set close to 0 so that $e(n) \approx e(n-p_n)$. Therefore, the excitation signal is periodic with the pitch period. If the periodicity of the voiced speech is less, $b(n,p_n)$ is smaller than 1 and $\sigma^2_{d(n)}$ is larger than 0. Hence the degree of periodicity of the voiced speech can be tuned. To represent silence both $b(n, p_n)$ and $\sigma^2_{d(n)}$ are set to 0 so $e(n) = 0$.

The goal is to develop an algorithm for computing the MMSE estimate of $s(n)$, which can be expressed as $\hat{s}(n) = E[s(n) \mid y(n+\tau), \ldots, y(n), \ldots, y(1)]$, where τ is the number of future samples of the noisy speech to be used. This can be done with the Kalman filter algorithm in Fig1. To formulate the speech model equations in a form required by the Kalman filter (1), (2) the following matrices and vectors are defined.

$$\mathbf{s}_n = \begin{bmatrix} s(n) \\ s(n-1) \\ s(n-2) \\ \vdots \\ s(n-r+1) \end{bmatrix} \qquad \text{(r x 1)} \qquad \text{where } r = max(q, \tau + 1)$$

$$\mathbf{\Gamma_1} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad \text{(r x 1)}$$

$$\mathbf{A}_n = \begin{bmatrix} a(n,1) & \ldots & a(n,q) & \ldots & 0 \\ 1 & 0 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \qquad \text{(r x r)}$$

Equation (3) is equivalent to the following state-space equation:

$\mathbf{s}_n = \mathbf{A}_n \mathbf{s}_{n-1} + \mathbf{\Gamma}_1 e_n$, where $e_n = e(n)$ (5)

In order to reformulate the equation (4) into state-space form it can be written as:

$e(n) = \sum_{l=1}^{p} b(n, l) e(n-l) + d(n)$ (6), where p is constant and is equal to the maximum possible pitch period of human speech. $b(n, l) = 0$ for all $l \neq p_n$ where p_n is the pitch period.

Equation (6) is equivalent to the following state-space equation:

$\mathbf{e}_n = \mathbf{B}_n \mathbf{e}_{n-1} + \mathbf{\Gamma}_2 d_n$ (7)

where,

$$\mathbf{e}_n = \begin{bmatrix} e(n) \\ e(n-1) \\ e(n-2) \\ \vdots \\ e(n-p+1) \end{bmatrix} \qquad \text{(p x 1)}$$

$$\boldsymbol{\Gamma}_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad \text{(p x 1)}$$

$$\mathbf{B}_n = \begin{bmatrix} b(n,1) & b(n,2) & \dots & \dots & b(n,p) \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} \qquad \text{(p x p)}$$

$d_n = d(n)$

Equation (5) and (7) can be combined into a single state-space equation:

$\mathbf{x}_{n+1} = \mathbf{F}_n \mathbf{x}_n + \boldsymbol{\Gamma}_3 d_{n+1}$ (8)

where,

$$\mathbf{x}_n = \begin{bmatrix} \mathbf{s}_{n-1} \\ \mathbf{e}_n \end{bmatrix} \qquad \text{(r+p x 1)}$$

$$\boldsymbol{\Gamma}_3 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix} \text{(1 is at the } (r+1)^{th} \text{ position)} \qquad \text{(r+p x 1)}$$

$$\mathbf{F}_n = \begin{bmatrix} \mathbf{A}_n & \mathbf{\Gamma}_1\mathbf{\Gamma}_2^T \\ \mathbf{O} & \mathbf{B}_{n+1} \end{bmatrix} \qquad \text{(r+p x r+p)}$$

If the observed signal is corrupted by the noise it can be represented as follows:

$y(n) = s(n) + w(n)$ (9), where w(n) is zero-mean white Gaussian noise (the case of colored noise will be handled later)

Equation (9) is equivalent to the following state-space equation:

$y(n) = \mathbf{\Gamma}_4\mathbf{x}_{n+1} + w(n)$ (10)

where,

$$\mathbf{\Gamma}_4 = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad \text{(r+p x 1)}$$

The state equation (8) and the observation equation (10) are in exactly the same form as (1) and (2). Therefore, Kalman filter algorithm in Fig1 can be applied in the following manner.

1) Initialization:
$a(0,1) = \ldots = a(0,q) = s(0) = \ldots = s(1-q) = e(0) = \ldots = e(-p) = y(0) = \sigma^2_{w(0)} = 0$
$\mathbf{P}_0 = \mathbf{O}_{(r+p)x(r+p)}, \hat{\mathbf{x}}_0 = \mathbf{O}_{(r+p)x1}$

2) Recursion: For n = 1, 2, …,
$\mathbf{Q}_n = \mathbf{F}_{n-1}\mathbf{P}_{n-1}\mathbf{F}_{n-1}^T + \sigma^2_{d(n)}\mathbf{\Gamma}_3\mathbf{\Gamma}_3^T$
$\mathbf{G}_n = \mathbf{Q}_n\mathbf{\Gamma}_4(\mathbf{\Gamma}_4^T\mathbf{Q}_n\mathbf{\Gamma}_4 + \sigma^2_{w(n-1)})^{-1}$
$\mathbf{P}_n = (I - \mathbf{G}_n\mathbf{\Gamma}_4^T)\mathbf{Q}_n$
$\hat{\mathbf{x}}_n = \mathbf{F}_{n-1}\hat{\mathbf{x}}_{n-1} + \mathbf{G}_n(y(n-1) - \mathbf{\Gamma}_4^T\mathbf{F}_{n-1}\hat{\mathbf{x}}_{n-1})$

3) Output: For n = 1, 2, …,

$\hat{s}(n) = [0,0,...,1,...,0]\hat{\mathbf{x}}_{n+\tau+1}$, where there are τ zeros before 1.

Because the maximum pitch period of human speech can be very high this algorithm contains multiplication, summation and inverses of very large matrices. Therefore, the computation cost of the algorithm is very high. However, this computational cost can be reduced by exploiting the fact that most of the matrices are sparse.

5.2 Parameter Estimation

The algorithm proposed in the previous section requires knowledge about the parameters of the observation model (9) and those of the speech model (3) and (4).

For the observation model the only parameter that has to be known is σ_w^2, the variance of the noise. To estimate the variance of the noise I used a voice activity detector and simply take the first 100ms of the speech signal as the noise only segment and achieved similar results for the zero-mean white Gaussian noise case. The voice activity detector first finds the heart of signal via conservative energy threshold, refines the beginning and ending using a tighter threshold in energy and finally checks outside the regions using zero crossing threshold.

For the speech model there are 4 time-varying and 3 constant parameters. Time varying parameters are:

1) $a(n,k)$ filter coefficients
2) p_n pitch period
3) $b(n,p_n)$ the degree of periodicity
4) $\sigma_{d(n)}^2$ variance of the signal in (6)

The constant parameters are:

1) τ, the number of future samples of the noisy speech to be used to estimate the current sample.
2) q, LPC order
3) p, the maximum pitch period of human speech

Firstly, time varying parameters are estimated from the noisy speech and filtering is done. Then the filtered signal is used to estimate the time varying filters and the new parameters are used to filter the signal. In this iterative way the clean speech and the parameters are estimated alternatively and the iteration is stopped when the difference between consecutive estimations is small enough.

For each n the estimates for $a(n,k)$'s for k=1, 2, …, q are obtained from the "smoothed" version of y(n) with the Autocorrelation method using Durbin-Levinson algorithm. In [3] it is suggested that if the "smoothed" version of y(n) weren't used the estimates for $a(n,k)$'s would vary so drastically that the undesirable "musical" noise would become apparent. By smoothing operation this musical noise is weakened. To perform the smoothing first the short-time magnitude spectrum and the phase spectrum of the segment near the sample (32ms) is computed (hamming window is used for windowing). Second, the short-time magnitude spectra of four neighboring segments, which overlap with the original segment by 12ms and 24ms are computed. Third, smooth magnitude spectrum is generated by taking the minimum of the five magnitude spectra for each frequency. This operation reduces the undesirable "spikes" that appear in the frequency domain and because $a(n,k)$'s are closely related to the spectral envelope these "spikes" would cause $a(n,k)$'s to vary drastically. Fourth, the time domain signal is obtained by taking the inverse FFT of the "smooth" magnitude spectrum combined with the phase spectrum obtained in the first step. Finally, $a(n,k)$'s are estimated using Durbin-Levinson algorithm.

The pitch period p_n is estimated from the autocorrelation function of the speech segment in 32ms neighborhood of the sample y(n), with 40% center clipping. The peak index of the autocorrelation function between 3ms-12.5ms is found and defined as the pitch period.

The periodicity $b(n,p_n)$ is estimated using the ratio $\frac{R(p_n)}{R(0)}$. If the ratio is greater than 0.5 the speech segment is considered periodic and $b(n,p_n)$ is set to $\frac{R(p_n)}{R(0)}$. If the ratio is smaller than 0.5 $b(n,p_n)$ is set to 0.

d(n) is computed from (3) and (4) and $\sigma^2_{d(n)}$ is estimated as the variance of the segment in 8 ms neighborhood centered at d(n).

A larger τ means that more noisy samples are used and therefore more information is exploited. However, this comes at a cost of increased complexity as τ affects the size of the matrices in the state and the observation models. For our purposes τ is chosen as 100 to give a compromise between two scenarios.

The value of q, which is the order of the speech model, is chosen so that the spectral envelope of the speech can be adequately represented. The rule of thumb is that there should be $\frac{F_s}{1000}$ poles for vocal tract, 2-4 poles for radiation and 2 for glottal pulse. Since in our case $F_s = 16000$ q is chosen to be 20.

Human pitch period rarely exceeds 12.5ms. Therefore $12.5ms \times F_s = 200$ is used as the maximum pitch period.

The summary of the enhancement model can be seen in Fig2.

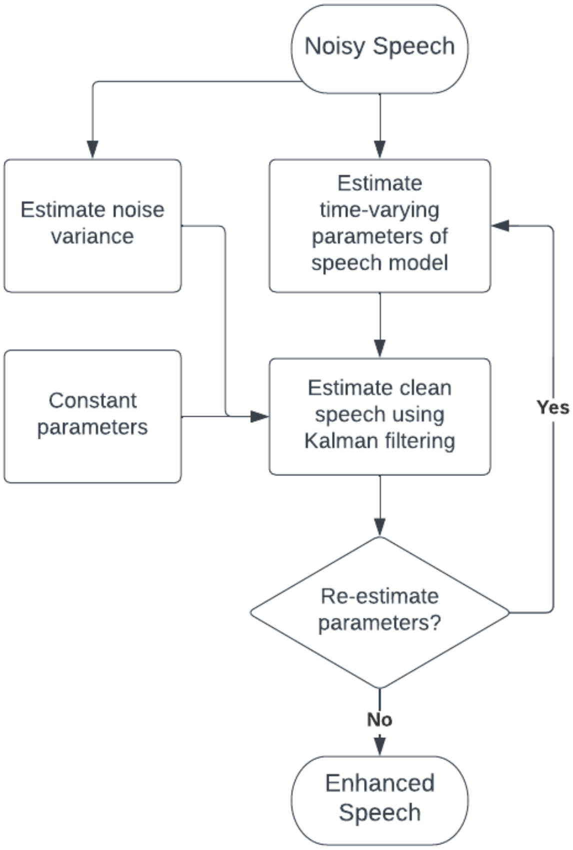


Fig2. Block Diagram of the Speech Enhancement Method

5.3 Colored Noise

The previous speech enhancement method is based on the zero-mean white Gaussian noise assumption; however, the noise doesn't have to be white. In fact, in real life scenarios it is usually colored. Therefore, colored noise should also be handled in our speech enhancement model.

A colored noise $w(n)$ can be modeled as an Autoregressive process.

$w(n) = \sum_{i=1}^{u} c(n,i)w(n-i) + v(n)$, where $v(n)$ is zero-mean white Gaussian process. In [8] they reported that such modeling is appropriate as long as the filter order is large enough. Then $y(n)$ is filtered using $c(n,i)$'s as the a coefficients.

$$y_f(n) = y(n) - \sum_{i=1}^{u} c(n,i)y(n-i)$$

The resulting filtered signal can be expressed as:

$y_f(n) = s_f(n) + v(n)$, where $s_f(n) = s(n) - \sum_{i=1}^{u} c(n,i)s(n-i)$ denotes a filtered speech signal. In [2] it is shown that $s_f(n)$ also fits the proposed speech model (3). The problem of estimating $s_f(n)$ from $y_f(n)$ is exactly the same as the previous problem as the noise here is again white Gaussian process. Therefore, enhancement method in Fig2 can be used to estimate $s_f(n)$ and estimate of $s(n)$ can be obtained by inverse-filtering.

$\tilde{s}(n) = \sum_{i=1}^{u} c(n,i)\tilde{s}(n-i) + \tilde{s}_f(n)$, where $\tilde{s}_f(n)$ is the estimated filtered signal and $\tilde{s}(n)$ is the desired estimate.

First, one has to identify the frames which only contain noise. This is done using the voice activity detector proposed in section 5.2. The $c(n,i)$'s are computed from the noise frames using Durbin-Levinson algorithm. 16 is chosen as the filter order as it is suggested in [3]. $c(n,i)$'s are used to compute $y_f(n)$ and the speech enhancement method is applied on this filtered noisy signal. Then the MMSE estimate of the filtered speech signal $\tilde{s}_f(n)$ is found and finally, the desired enhanced speech signal is calculated by inverse-filtering. The overall block diagram can be seen in Fig3.

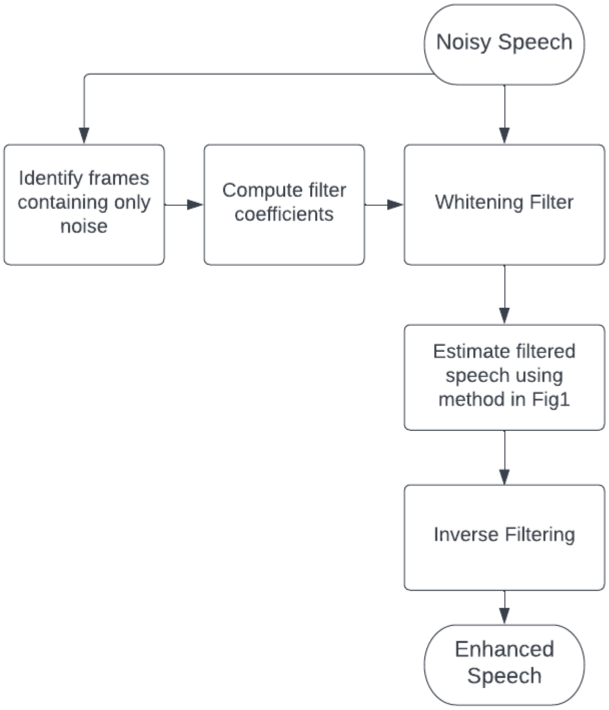

Noisy Speech
Identify frames containing only noise
Compute filter coefficients
Whitening Filter
Estimate filtered speech using method in Fig1
Inverse Filtering
Enhanced Speech

Fig3. Block Diagram of the Speech Enhancement Method When the Noise is Colored

5.4 Sub-band Kalman Filtering

In [9] and [10] it is suggested to decompose a signal into different frequency bands by using Discrete Wavelet Transform and then perform Kalman Filtering on each of the sub-bands. Then the desired enhanced signal is reconstructed using filtered sub-bands by performing IDWT. This approach can be summarized in Fig4.

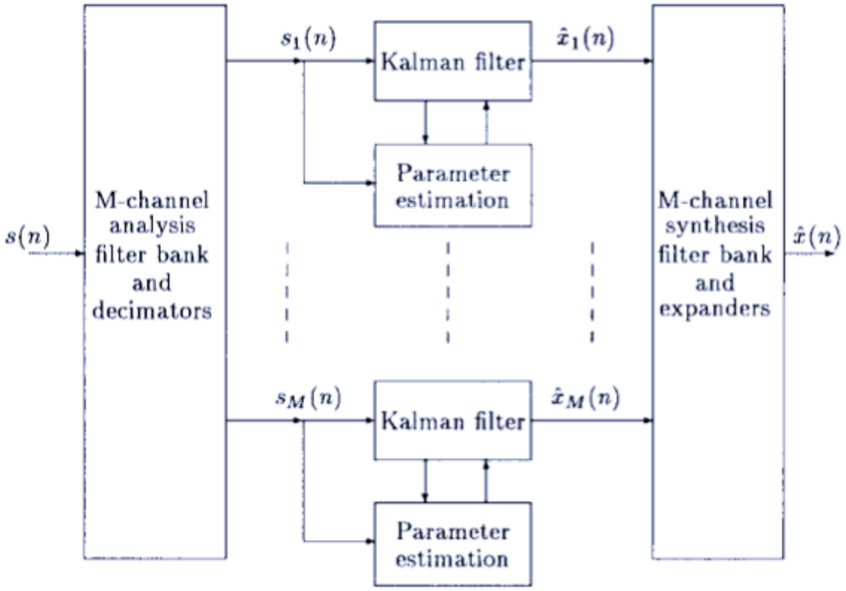


Fig4. Sub-band speech enhancement method.

Although this method is also tested out no additional gain from the speech enhancement method in Fig2 is observed. Therefore, this method is abandoned.

5.5 Evaluation Methods

The results will be evaluated by comparing the input and output SNR levels. The output SNR level is defined by $SNR = \frac{\sum_t s^2(t)}{\sum_t (s(t) - \hat{s}(t))^2}$ where $s(t)$ and $\hat{s}(t)$ are clean and enhanced speech signals respectively. Moreover, the spectrograms of the corrupted and the enhanced signals will also be observed.

6. Results

6.1 White Gaussian Noise

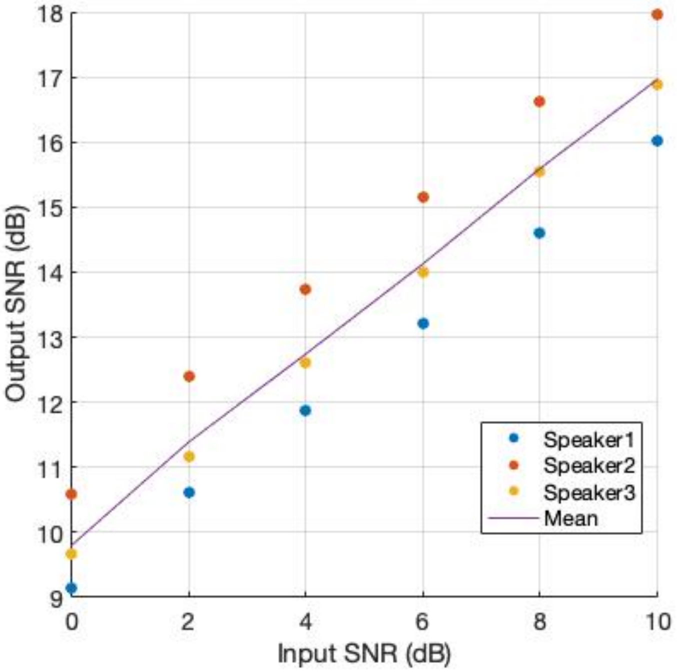


Fig5. White Noise Input-Output SNR Values After Speech Enhancement

One can see from Fig5 that our model performs very well in the case of white noise. At high SNRs it can increase the input SNR up to 8dBs and also improve the intelligibility of the speech. At low SNRs it is harder to improve the intelligibility but it can increase the SNR up to 10.5 dBs.

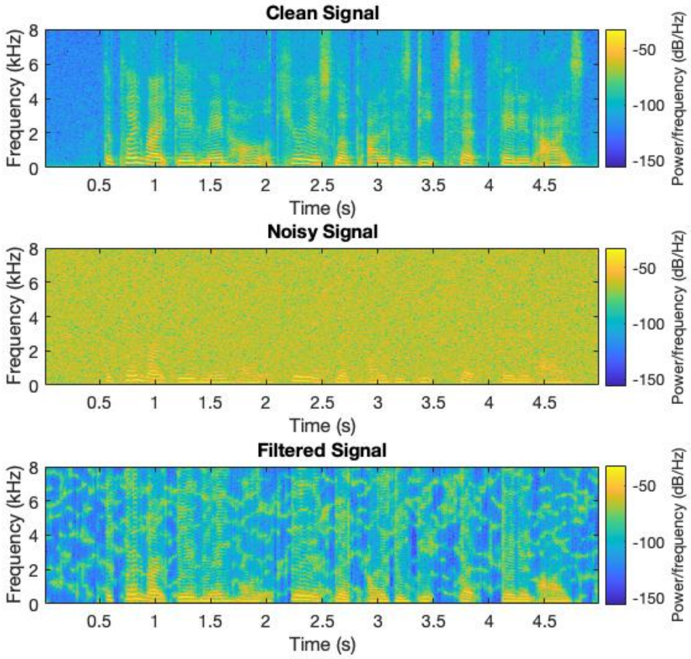


Fig6. Spectrograms at 0dB input SNR

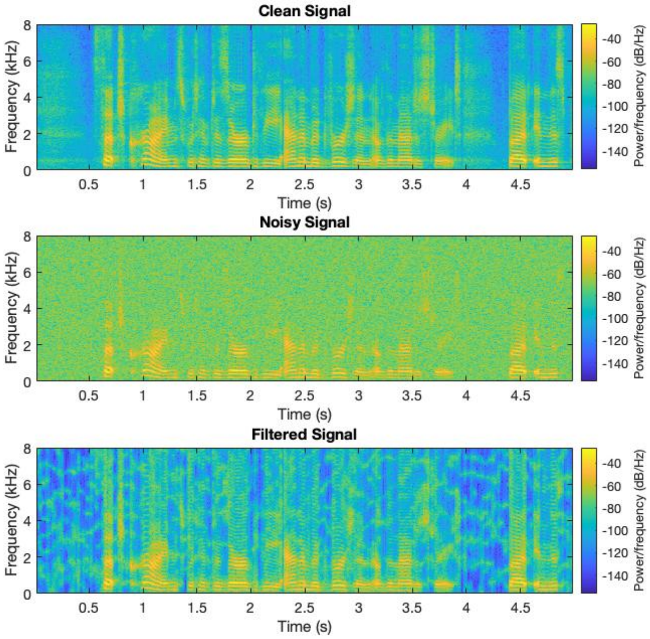


Fig7. Spectrograms at 10dB input SNR

One can see from Fig6 that although the SNR is very low our speech enhancement model manages to capture some of the speech details, such as the voiced parts of the signal at the first second. Moreover, one can see from Fig7 that the weak harmonics near the 4.5th second is retained in our filtered signal. Overall spectrograms show that our proposed method performs good at speech enhancement for white gaussian noise.

6.2 Colored Noise

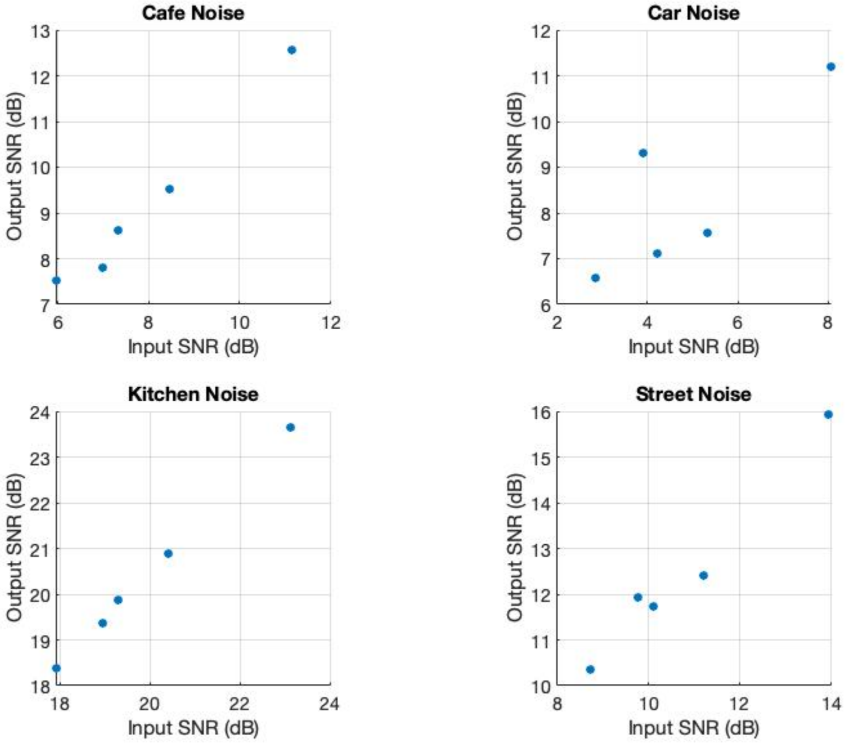


Fig8. Input-Output SNR levels for colored noise

One can see from Fig8 that although the performance of our model at colored noise is worse than at white gaussian noise the SNR values and the intelligibility of the speech signals improve.

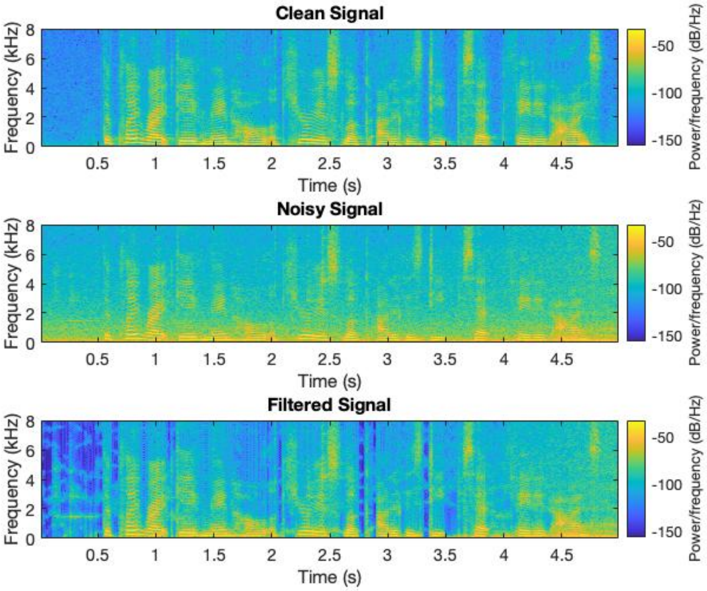


Fig9. Spectrograms Using Street Noise

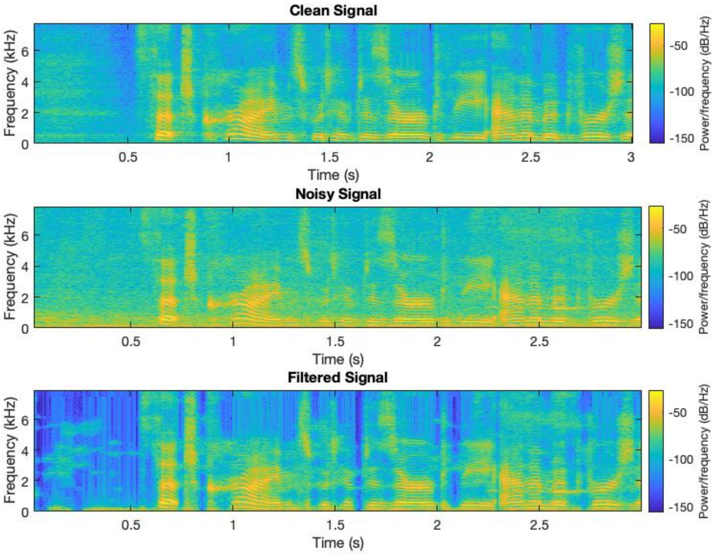


Fig10. Spectrograms Using Cafe Noise

Again in Fig9 and Fig10 one can clearly see the noise reduction for colored noise examples. In Fig10 there is a high frequency component at 2.5th second which is not removed after filtering. This is due to the fact that this noise component is abrupt and therefore its parameters are not estimated in as noise-only for this noise doesn't exist. Therefore filtering couldn't be done for this noise and it still remains in the filtered signal.

References

[1] Paliwal, K., & Basu, A. (1987, April). A speech enhancement method based on Kalman filtering. In *ICASSP'87. IEEE International Conference on Acoustics, Speech, and Signal Processing* (Vol. 12, pp. 177-180). IEEE.

[2] Gannot, S., Burshtein, D., & Weinstein, E. (1998). Iterative and sequential Kalman filter-based speech enhancement algorithms. *IEEE Transactions on speech and audio processing*, *6*(4), 373-385.

[3] Goh, Z., Tan, K. C., & Tan, B. T. G. (1999). Kalman-filtering speech enhancement method based on a voiced-unvoiced speech model. *IEEE Transactions on speech and audio processing*, *7*(5), 510-524.

[4] Ishaq, R., Zapirain, B. G., Shahid, M., & Lövström, B. (2013, May). Subband modulator Kalman filtering for single channel speech enhancement. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 7442-7446). IEEE.

[5] Panayotov, V., Chen, G., Povey, D., & Khudanpur, S. (2015, April). Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)* (pp. 5206-5210). IEEE.

[6] Hu, G., & Wang, D. (2010). A tandem algorithm for pitch estimation and voiced speech segregation. *IEEE Transactions on Audio, Speech, and Language Processing*, *18*(8), 2067-2079.

[7] Dean, D., Sridharan, S., Vogt, R., & Mason, M. (2010). The QUT-NOISE-TIMIT corpus for evaluation of voice activity detection algorithms. In *Proceedings of the 11th Annual Conference of the International Speech Communication Association* (pp. 3110-3113). International Speech Communication Association.

[8] Koo, B., Gibson, J. D., & Gray, S. D. (1989, May). Filtering of colored noise for speech enhancement and coding. In *International Conference on Acoustics, Speech, and Signal Processing,* (pp. 349-352). IEEE.

[9] Wu, W. R., & Chen, P. C. (1998). Subband Kalman filtering for speech enhancement. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, *45*(8), 1072-1083.

[10] Yu, H., Zhu, W. P., & Champagne, B. (2020). Subband Kalman Filtering with DNN Estimated Parameters for Speech Enhancement. In *INTERSPEECH* (pp. 2697-2701).