Multi-microphone Speech Enhancement

Manolis Papadakis
4739418

Husain Kapadia
4707915

***Abstract*—Estimation theory is used to determine the best estimate of the clean speech signal, given noisy speech signals that are recorded using multiple microphones. Since a linear model was used to estimate the clean speech, the Best Linear Unbiased Estimator (BLUE), Least Squares (LS) Estimator and Bayesian approaches such as Linear Minimum Mean Square Error (LMMSE) Estimators were implemented. These different estimators were compared so as to determine the best estimation technique. This was achieved by computing the Empirical variances, Cramer Rao Lower Bound (CRLB) and validating the performance based on the difference between these parameters. The theory formulates that the estimate is a Minimum Variance Unbiased (MVU) Estimate and the results help understand and prove the same.**

***Keywords*—Speech Enhancement, Estimation Theory**

I. Introduction

Speech enhancement aims to improve speech quality that is corrupted in typical noisy environments. In most speech processing systems such as mobile phones, speech recognition, hearing aids noise has to be reduced first. When multiple microphones are involved in reducing the noise, this process is called multi-microphone speech enhancement. In this technique, the target signal is recorded in all microphones with different slight delays, but by compensating for these delays the replicas of the target signal can be added constructively. At the same time, signals coming from other directions (noise, interference) are combined destructively and as a result they are attenuated. In this assignment a multi-microphone speech enhancement is attempted using estimation theory to reconstruct the target signal. Then the performance of the chosen estimator is examined.

The block diagram of the multi-microphone speech enhancement system shown in figure 1 is simplified for the case of two microphones.

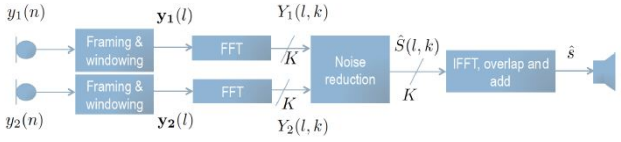

Fig. 1. Overview of a Dual-microphone speech enhancement system

Speech signals are not stationary, so they are typically framed in smaller overlapping sections and multiplied with appropriate windows of length equal to the frame size. In this case, the frame length is 20ms with 60% overlap (60% is chosen because it gives least distortion in the output) and multiplied with a Hanning window. In the frequency domain, the signal model is given by

$$Y_m(l,k) = A_m(l,k) * S(l,k) + W_m(l,k) \tag{1}$$

where Y denotes the noisy speech recorded by the microphone, S represents the clean speech that has to be estimated and W is the noise (In this case, assumed to be White Gaussian Noise) all expressed in the frequency domain. Also, A is the Acoustic Transfer function, l denotes the frequency bins within a frame, k stands for the number of time frames and m represents the number of microphones.

We can simplify our model by assuming that the Acoustic Transfer Function (ATF) satisfies:

$$A_m(l,k) = 1, \forall m, l, k \tag{2}$$

Stacking the DFT coefficients in a vector we obtain

$$\boldsymbol{Y_m}(l,k) = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} S(l,k) + \boldsymbol{W_m}(l,k) \tag{3}$$

Using this signal model we want to estimate the target speech signal. For our purposes, we assume that the noise in the frequency domain is zero mean Gaussian distributed, that is $W_m(l,k) \sim \mathcal{N}(\mu, \sigma^2)$ and stationary across time.

The first $100,000$ samples of the noisy speech signal recorded over 16 microphones is shown in figure 2:

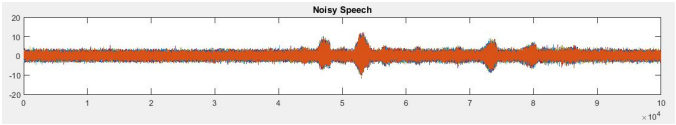

Fig. 2. Noisy Speech Signal

The first $100,000$ samples of the clean speech signal is shown in figure 3:

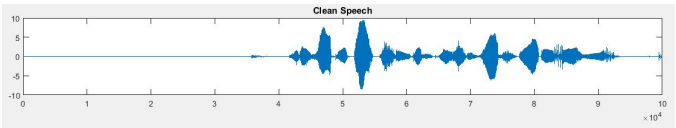

Fig. 3. Clean Speech Signal

II. Theory

This section includes the description of the various estimation techniques used and their formulations as per our model.

A. Classical Approach

In this approach, we assume that the speech signal is deterministic. The goal is to find a minimum variance unbiased estimate of the clean speech. In general it is difficult to obtain a minimum variance unbiased estimator since the method depends on the Cramer Rao Lower Bound or the concept of sufficient statistics. Hence, we need to constrain our problem to be linear which may lead to sub-optimal solutions. However, in this problem for amplitude estimation of speech signal, we have a linear model as given in equation 1. Moreover, as per our assumption of the noise being White Gaussian in nature, the Best Linear Unbiased Estimator (BLUE) is the Minimum Variance Unbiased (MVU) estimator. There is no sub-optimal estimator as the MVU estimator in our case is efficient. The derivation of this is explained in the appendix.

Hence, the BLUE estimator for the model in equation 1 is given by:

$$\hat{\boldsymbol{S}}(l,k) = ({\boldsymbol{A_m}}^T {\boldsymbol{C_W}}^{-1} \boldsymbol{A_m})^{-1} {\boldsymbol{A_m}}^T {\boldsymbol{C_W}}^{-1} \boldsymbol{Y_m}(l,k) \quad (4)$$

where, $\hat{\boldsymbol{S}}(l,k)$ is the estimated speech signal in the frequency domain. This is a vector of length L $\boldsymbol{A_m}$ is the Acoustic transfer function which is a vector of all ones and has a length equal to the number of microphones m $\boldsymbol{C_W}$ is the noise co-variance matrix which has a dimension of $m \times m$ $\boldsymbol{Y_m}(l,k)$ is the noisy speech signal and has a dimension of $m \times L$

Along with the BLUE estimator, we also implemented the Least Squares (LS) estimator which is given by

$$\hat{\boldsymbol{S}}(l,k) = ({\boldsymbol{A_m}}^T \boldsymbol{A_m})^{-1} {\boldsymbol{A_m}}^T \boldsymbol{Y_m}(l,k) \quad (5)$$

The expression for Weighted Least Squares (WLS) and Maximum Likelihood (ML) estimators is same as BLUE. This means all these different estimation techniques are MVUs.

B. Bayesian Approach

Assuming that the speech signal $\boldsymbol{S}(l,k)$ is not deterministic but a random process, so we have to use a Bayesian Estimator. We have been provided with the Clean Speech sample and hence we can compute some prior information about the probability density function (pdf) of $\boldsymbol{S}(l,k)$, $p(\boldsymbol{S}(l,k))$ and we want to combine it with the measurements to make a better estimation. We first compute the mean $\boldsymbol{\mu_S(l)}$ and covariance matrix $\boldsymbol{C_S(l)}$ for each frame k, equal to the mean and covariance of the DFT of the clean signal for the corresponding frame (further assuming that random variables are uncorrelated). Then assuming Gaussian distribution for $S(l,k)$ we obtain the Bayesian linear model as shown in equation 6 in which $\boldsymbol{S}(l,k) \sim \mathcal{N}(\mu, \sigma^2 I)$ and $\boldsymbol{W}(l,k) \sim \mathcal{N}(0, C_W)$. In such a case, the MMSE estimator is equal to the LMMSE estimator and is given by

$$\hat{\boldsymbol{S}}(l,k) = \boldsymbol{\mu_S}(k) + \boldsymbol{C_S}(k) {\boldsymbol{A_m}}^T (\boldsymbol{A_m} \boldsymbol{C_S}(k) {\boldsymbol{A_m}}^T + \boldsymbol{C_W})^{-1} (\boldsymbol{Y}(l,k) - \boldsymbol{A_m} \boldsymbol{\mu_S}(k)) \quad (6)$$

where, $\boldsymbol{\mu_S}(k)$ is a vector of length L and consists of the mean value of the corresponding time frame k $\boldsymbol{C_S}(k)$ is value of the co-variance of the clean speech signal for the corresponding time frame k

III. Experiment

This section describes the flow of the process to estimate the target speech signal and the parameters that we can tweak to achieve good results.

For the experiment, we are provided with noisy speech that is sampled at 16kHz and the data is collected over $m = 16$ microphones. We consider first $100,000$ samples of the noisy speech by setting the variable N given in appendix B-A.

We compute the short time fourier transform (STFT) of the noisy signal iterated over all the microphones which returns a 3 dimensional matrix of dimensions number of Frequency bins vs. number of time frames vs. number of microphones $L \times K \times m$. We consider a time frame of length 20ms which amounts to $L = 320$ samples. We use a hanning window and 60% overlap to calculate the STFT. We use 60% overlap because it gives least distortion. This can be done by passing appropriate arguments to the function STFT as given in appendix B-B.

Now, to obtain the noise co-variance matrix, we iterate over the first $k = 200$ time frames because it consists of only noise. We generate the noise co-variance matrix of dimension $m * m$ for all k time frames and average it.

Using the noisy speech in the frequency domain and the noise co-variance matrix, we can now estimate the clean speech by using the estimate function given in appendix B-D. By selecting the type of estimate the function will output a 2 dimensional matrix of size $L \times K$ which is an estimate of the clean speech generated using either BLUE/WLS/MLE or LS or LMMSE/MAP. In case of Bayesian estimators like LMMSE/MAP the function takes the input of the mean and the variance of the clean speech calculated for the L frequency bins within every time frame. This is used as prior information.

After estimating the clean speech in the frequency domain, we need to bing it back to the time domain. This is done using the STIFT function given in appendix B-C. The function outputs a vector of length N. The STIFT is based on the overlap and add method. Thus, we obtain an estimate of the clean speech in the time domain which has reduced noise as compared to the input enhancing the quality of the speech.

To evaluate the performance of the estimator we calculate the empircial variance as a function of the number of microphones. Thus, while estimating the clean speech we use just the first microphone,then the first two and so on so forth, until we used all $M = 16$ microphones. The empirical variance is given by equation 7

$$var_{emp}(i) = \frac{1}{KL} \sum_{k=1}^{K} \sum_{l=1}^{L} |\hat{\boldsymbol{S_i}}(l,k) - \boldsymbol{S}(l,k)|^2, \forall i = 1,2,..,16 \quad (7)$$

Then, we computed the Cramer Rao Lower Bound (CRLB) per frequency band averaged across frequency. For our model given in equation 4 , the Fisher information matrix is given by the expression 8 (Derivation in the appendix)

$$CRLB(i) = \frac{1}{{\boldsymbol{A_i}}^T {\boldsymbol{C_w}}^{-1} \boldsymbol{A_i}}, \forall i = 1,2,..,16 \quad (8)$$

IV. Results

In this section, the results of the experiments will be discussed and an analysis will be conducted based on these results.

A. Classical Approach

As we said before,we first tried classical estimators to determine the clean speech signal. Because of the assumptions of the model (linear in the unknown parameter and corrupted by white Gaussian noise) Least Squares Estimator, Weighted Least Squares Estimator and Best Linear Unbiased Estimator are all given by the same expression and equal to the MVU estimator which in our case attains the CRLB, so they are all efficient estimators. To verify this, after we computed the BLUE estimator for each frame, we determined the empirical variance of the estimator.

From figure 4 we confirm that the BLUE estimator attains the CRLB, so it's an MVU efficient estimator. The empirical variance is slightly greater than or equal to the CRLB for every microphone. Moreover, the CRLB decreases as the number of microphones increases. This is also consistent with theory as the variance of the MVU efficient estimator is inversely proportional to the sum of inverse variances of the microphones noise as proved in the appendix A. The empirical variance also reduces as the number of microphones increases since the noise co-variance matrix includes information of different microphones which helps in better estimation. The corresponding estimation of target speech on time domain is shown in figure 5.

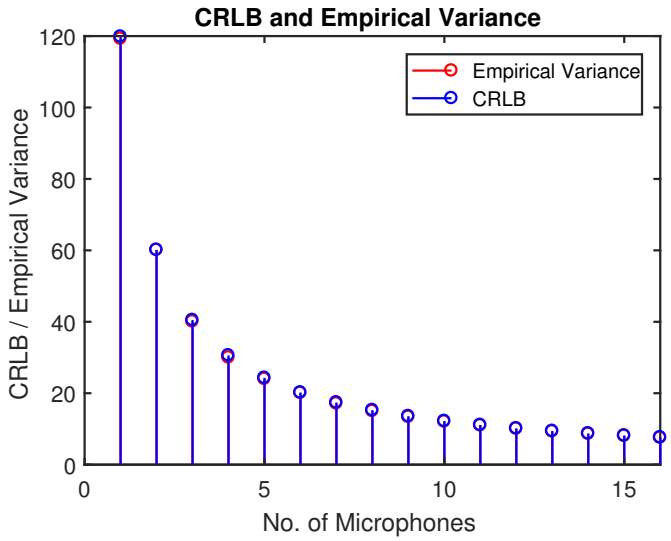

Fig. 4. Empirical Variance of BLUE and CRLB

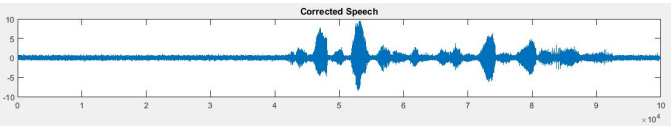

Fig. 5. Estimated target speech signal using the BLUE estimator

B. Bayesian Approach

Then, we assumed that $\hat{\boldsymbol{S}}(l,k)$ is a random process, and in particular Gaussian where its mean and co-variance were used as prior information which were obtained from the clean signal. After implementing the LMMSE estimator given by equation 6 we plotted its empirical variance versus the number of microphones involved in the estimation in figure 6.

From this it is apparent that the Empirical Variance for the Bayesian approach is less than that of the Classical Approach. This happens because we use the prior information of the clean speech signal to estimate the clean speech. Using this prior information reduces the error even further. Hence, the empirical variance is way below the CRLB and this shows that the error due to variance can further be reduced if we use prior information to estimate the clean speech. In this case, due to availability of prior information we could obtain better results. However, in reality the prior information about the clean speech signal may not be available. Thus, we need to use the classical approach to estimate these prior information and then use this estimated prior information to estimate the clean speech.

We also observe that increasing the number of microphones that are used in the estimation, the variance of the LMMSE estimator decreases as we take advantage of more information. This of course corresponds to a way cleaner signal as is shown in figure 7. Though there is still some amounts of distortion but the noise has been completely eliminated in the no speech areas due to the prior information.

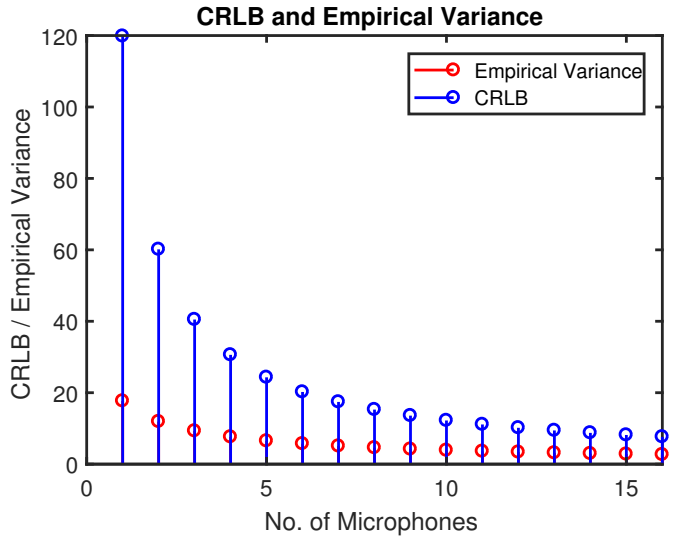

Fig. 6. Empirical Variance of LMMSE and CRLB

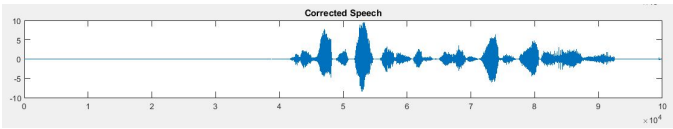

Fig. 7. Estimated target speech signal using the MMSE estimator

V. Conclusion

In a multi-microphone speech enhancement system, we attempt to combine noisy recordings from different microphones in order to get the best possible estimate of the target speech. Assuming a linear model in Gaussian Noise, in classical estimation where the unknown parameter is assumed deterministic, the MVU estimator is identical to the BLUE estimator, and the WLS estimator and furthermore to the LS estimator when additionally the noise is white. Under these assumptions, all these estimators are efficient meaning that their emprirical variances attain the CRLB.

When the unknown parameter is assumed to be random, then we should use a prior information about its probability density function. In Bayesian estimation, the estimator that gives the minimum mean square error averaged over the joint pdf $p(\boldsymbol{x}, \theta)$ is the MMSE estimator and for the linear Gaussian model is identical to the LMMSE estimator. Using a reasonable choice for the prior pdf, we can drop the variance of the estimator below the CRLB.

In both approaches we notice that the CRLB and the empirical variances reduce as we go on increasing the number of microphones. However, this too has a limit to it as we see that beyond a point the difference in the subsequent readings reduces. This shows that there will always be some amount of distortion or noise present in the estimated output.

References

[1] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. University of Rhode Island: Prentice Hall, 2004.

[2] R. Hendricks, "Et4386 estimation & detection assignment: Multi-microphone speech enhancement," a PDF file containing the assignment related to this essay. [Online]. Available: http://ens.ewi.tudelft.nl/Education/courses/et4386/Projects1/speech_enhancement/speech_enhancement.pdf

APPENDIX A
DERIVATION OF CRLB

The probability density function for the model given in equation 1 can be expressed as

$$\mathcal{P}(\boldsymbol{Y_m}(l,k);\boldsymbol{S}(l,k)) = \frac{1}{(2\pi)^{L/2}|det(\boldsymbol{C_w})|^{1/2}} e^{[-\frac{1}{2}[\boldsymbol{Y_m}(l,k)-\boldsymbol{A_m}\boldsymbol{S}(l,k)]^T \boldsymbol{C_w}^{-1}[\boldsymbol{Y_m}(l,k)-\boldsymbol{A_m}\boldsymbol{S}(l,k)]]} \tag{9}$$

Taking log on both sides we obtain the log-likelihood function

$$\log \mathcal{P}(\boldsymbol{Y_m}(l,k);\boldsymbol{S}(l,k)) = -\frac{L}{2}\log 2\pi - \frac{1}{2}\log|det(\boldsymbol{C_w})| - \frac{1}{2}[\boldsymbol{Y_m}(l,k)-\boldsymbol{A_m}\boldsymbol{S}(l,k)]^T \boldsymbol{C_w}^{-1}[\boldsymbol{Y_m}(l,k)-\boldsymbol{A_m}\boldsymbol{S}(l,k)] \tag{10}$$

Taking partial derivatives with respect to $\boldsymbol{S}(l,k)$ on both sides

$$\frac{\partial \log \mathcal{P}(\boldsymbol{Y_m}(l,k);\boldsymbol{S}(l,k))}{\partial \boldsymbol{S}(l,k))} = \boldsymbol{A_m}^T \boldsymbol{C_w}^{-1}[\boldsymbol{Y_m}(l,k)-\boldsymbol{A_m}\boldsymbol{S}(l,k)] \tag{11}$$

Taking partial derivatives again with respect to $\boldsymbol{S}(l.k)$ on both sides

$$\frac{\partial^2 \log \mathcal{P}(\boldsymbol{Y_m}(l,k);\boldsymbol{S}(l,k))}{\partial^2 \boldsymbol{S}(l,k))} = -\boldsymbol{A_m}^T \boldsymbol{C_w}^{-1}\boldsymbol{A_m} \tag{12}$$

Thus, the Cramer Rao Lower Bound (CRLB) is given by

$$var(\hat{\boldsymbol{S}}(l,k))) = \frac{1}{-E[\frac{\partial^2 \log \mathcal{P}(\boldsymbol{Y_m}(l,k);\boldsymbol{S}(l,k))}{\partial^2 \boldsymbol{S}(l,k))}]} = \frac{1}{\boldsymbol{A_m}^T \boldsymbol{C_w}^{-1}\boldsymbol{A_m}} \tag{13}$$

To verify if the estimator is efficient, recall equation 11 and multiply and divide by $\boldsymbol{A_m}^T \boldsymbol{C_W}^{-1}\boldsymbol{A_m}$ Thus, we obtain

$$\frac{\partial \log \mathcal{P}(\boldsymbol{Y_m}(l,k);\boldsymbol{S}(l,k))}{\partial \boldsymbol{S}(l,k))} = -(\boldsymbol{A_m}^T \boldsymbol{C_w}^{-1}\boldsymbol{A_m})[(\boldsymbol{A_m}^T \boldsymbol{C_w}^{-1}\boldsymbol{A_m})^{-1}\boldsymbol{A_m}^T \boldsymbol{C_w}^{-1}\boldsymbol{Y_m}(l,k) - \boldsymbol{S}(l,k)] \tag{14}$$

Hence, the MVU estimator is given by

$$\hat{\boldsymbol{S}}(l,k) = (\boldsymbol{A_m}^T \boldsymbol{C_W}^{-1}\boldsymbol{A_m})^{-1}\boldsymbol{A_m}^T \boldsymbol{C_W}^{-1}\boldsymbol{Y_m}(l,k) \tag{15}$$

APPENDIX B
MATLAB CODE

A. Speech Enhancement

demo.m

```
clc;
close all;

%% initializations
load('Data.mat')
fs = 16000;                     %sampling frequency
N = 100000;                     %length of speech
m = nrmics;                     %number of mics
y = Data(1:N,1:m);              %noisy speech
s = Clean(1:N);                 %clean speech
l = 20;                         %frame length in ms
o = 60;                         %percent overlap

S = stft(s, 3, l, o, 1, fs);    %Clean speech in Frequency domain
S_e = zeros(size(S));
```

```
var_emp = zeros(1, m);
mse = zeros(1, m);
CRLB = zeros(1, m);
CRLBpf = zeros(1, size(S,1));

for i = 1:m
    Cw = zeros(i);

    %% STFT with overlap
    Y = stft(y, 3, 1, o, i, fs);

    %% Noise Covariance
    P1 = permute(Y, [1 3 2]);
    for j = 1:200
        U1 = P1(:,:,j);
        Cw = (j*Cw + cov(U1))/(j+1);
    end

    Ct = var(S);
    mt = mean(S);

    %% Estimation
    %type
    %1 for BLUE / WLS / MLE
    %2 for LS
    %3 for LMMSE / MAP
    type = 3;
    S_e = estimate(Y, type, Cw, i, mt, Ct);

    %% Evaluation in frequency domain
    var_emp(i) = sum(sum(abs(S_e - S).^2))/(size(Y,1)*size(Y,2));

    a = ones(1, i);
    CRLB(i) = real(1/(a*inv(Cw)*a'));

    %% STIFT with overlap add
    s_e = stift(S_e, 3, 1, o, 1, fs);

    %% Evaluation in time domain
    mse(i) = mean((s_e - s(1:length(s_e))).^2);
end

%% Plots
figure()
subplot(3,1,1)
plot(y), title('Noisy Speech');
subplot(3,1,2)
plot(s_e), title('Corrected Speech');
subplot(3,1,3)
plot(s), title('Clean Speech');

figure()
stem(var_emp, 'r', 'DisplayName', 'Empirical Variance')
xlabel('No. of Microphones')
ylabel('CRLB / Empirical Variance')
hold on;
stem(CRLB, 'b', 'DisplayName', 'CRLB')
title('CRLB and Empirical Variance');
legend('show');

figure()
stem(mse), title('Mean Square Error');
```

```
78
79 %% Sound
80 sound(0.1*s_e, fs)
```

B. Short Time Fourier Transform

stft.m

```
1  function Y = stft(x, win, frame_len, overlap, mic, fs)
2      L = frame_len*fs/1000;                  %frame length                    %
           percent overlap
3      D = (1 - 0.01*overlap)*L;               %start index for overlap
4      K = round(1 + floor((length(x)-L)/D));%number of sections
5      Y = zeros(L, K, mic);
6
7      switch win
8          case 1
9              w = ones(L,1);
10         case 2
11             w = hamming(L);
12         case 3
13             w = hanning(L);
14         case 4
15             w = bartlett(L);
16         case 5
17             w = blackman(L);
18     end
19
20     for j = 1:mic
21         n1 = 1;                             %start index
22         for i=1:K
23             xw(:, j) = x(n1:n1+L-1, j).*w;
24             Y(:, i, j) = fft(xw(:, j), L);
25             n1 = n1 + D;
26         end
27     end
28
29 end
```

C. Short Time Inverse Fourier Transform

stift.m

```
1  function y_n = stift(X, win, frame_len, overlap, mic, fs)
2      L = frame_len*fs/1000;          %frame length                    %percent
           overlap
3      D = (1 - 0.01*overlap)*L;       %start index for overlap
4      K = size(X, 2);                 %number of sections
5      N = round(L + (K-1)*D);
6      y_n = zeros(N, mic);
7
8      switch win
9          case 1
10             w = ones(L,1);
11         case 2
12             w = hamming(L);
13         case 3
14             w = hanning(L);
15         case 4
```

```
            w = bartlett(L);
        case 5
            w = blackman(L);
    end

    for j = 1:mic
        n1 = 1;                         %start index
        for i=1:K
            Xw = X(:, i, j);
            Xw = [Xw; conj(Xw(end:-1:2))];
            y_n(n1:n1+L-1, j) = y_n(n1:n1+L-1, j) + real(ifft(Xw, L)).*w;
            n1 = n1 + D;
        end
    end

end
```

D. Estimators

estimate.m

```
function S = estimate(Y, type, Cw, mic, mt, Ct)
    a = ones(1,mic);
    P = permute(Y, [3 1 2]);

    for j = 1:size(Y,2)
        Z = P(:,:,j);
        switch(type)
            case 1
                % BLUE / WLS /MLE
                B = (a*inv(Cw)*Z)/(a*inv(Cw)*a');
            case 2
                % LS
                B = pinv(a)'*Z;
            case 3
                % LMMSE / MAP
                mu = mt(j)*ones(1,size(Y,1));
                C = a'*Ct(j)*a + Cw;
                B = mu + Ct(j)*a*inv(C)*(Z - a'*mu);
        end
        S(:, j) = B.';
    end

end
```
