A NEW SPEAKER VERIFICATION APPROACH FOR BIOMETRIC SYSTEM

J.INDRA[1] N.KASTHURI[2] M.BALASHANKAR[3] S.GEETHA MANJURI[4]

[1] *Assistant Professor (Sl.G),Dept of Electronics and Instrumentation Engineering,*
Kongu Engineering College, Tamil nadu, India

[2] *Professor, Dept of Electronics and Communication Engineering,*
Kongu Engineering College, Tamil nadu, India

[3]*EV Green tools Automation limited, Bangalore*
Karnataka,India

[4] *PG Scholar,Dept of Electrical and Electronics Engineering,*
Kongu Engineering College, Tamil nadu, India

Abstract-*In this paper a new speaker verification method is proposed. Speaker Verification for various speakers has been done and the output for feature extraction like pitch extraction, formant extraction and generation of code vectors has been presented. The speech signal is acquired by using microphone from various speakers. The pitch and formant extraction is done using two windowing techniques such as Hanning window and Hamming window. Hanning window gives better pitch extraction compared to Hamming window. This method serves as an efficient method for the speaker verification as a biometric system.*

Keywords*: Speaker verification system, formant, pitch, Hamming window, Hanning window.*

1.INTRODUCTION

Biometrics refers to the identification of humans by their characteristics or traits. Biometrics is also used in computer science as a form of identification and access control. It is also used to identify individuals in groups under surveillance. There are two distinct phases for the operation of speaker verification system: (a) an enrollment phase and (b) a verification phase. Before computing features from the speech signal, the speech signal is first digitized ,sampled and filtered. The next step in the enrollment phase is to train a speaker specific statistical model. During the verification phase an unknown utterance is authenticated against the trained speaker and background statistical model. The scores generated by both the models are normalized and integrated for the entire utterance before an acceptance or a rejection decision is made [1].

2. SPEAKER VERIFICATION SYSTEM

Speaker verification is one of the biometric systems used to identify the person by using his speech (voice) signal in a group. There are several possible applications in Security and Identification systems. Nowadays, biometrics is being used extensively in security purposes. Unlike the other biometric systems, speaker recognition does not need any direct physical contact to the system. Speaker recognition is a biometric system that provides positive verification of identity from individual's voice characteristics.

2.1 Features of speech signal

The physiological component of speaker verification is related to the physical shape of an individual's vocal track, that consists of an airway and the soft tissue cavities from vocal sounds originate. To produce speech, these components work in combination with the physical movement of the jaws, tongue and larynx and resonances in the nasal passages. The acoustic patterns of speech come from the physical characteristics of the airways. Motion of the mouth and pronunciations are the behavioral components of this system [2]. The speaker verification system analyzes the frequency content of the speech and compares characteristics such as the total energy, low-medium-high frequency energy, formant transitions, voicing detection, most prominent peak frequency, quality, duration, intensity dynamics and pitch of the signal. Here in this proposed work, the vocal track parameters such as pitch contour, formant frequencies and the prediction coefficients are concentrated.

2.1.1 Formant

Formants are defined as 'the spectral peaks of the sound spectrum of the voice'. It is the concentration of acoustic energy around a particular frequency in the speech wave [3]. There are several formants, each at a different frequency, roughly one in each 1000Hz band. In speech science and phonetics, formant is also used to mean an acoustic resonance of the human vocal tract. Formant frequencies can be extracted by using voice signal analysis. There are 7 formant frequencies available in the speech signal ranging from F1 to F7. Formant frequencies F1 to F3 contain phonetic contents and F4 to F7 contain behavioral contents. . The extraction is done for 15 speakers and the sentence used is “The Cat Sat”. The block diagram for formant extraction using LabVIEW platform is shown in the Figure 1

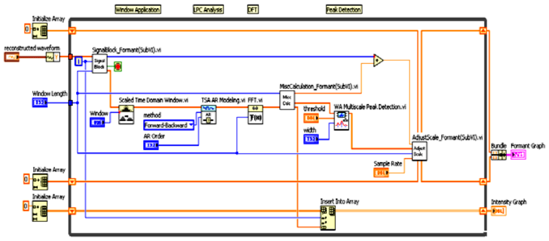


Figure-1 VI Block diagram of formant extraction

2.1.2 Pitch

Pitch, in speech, is the relative highness or lowness of a tone as perceived by the ear, that depends on the number of vibrations per second produced by the vocal cords. Pitch is the main acoustic correlate of tone and intonation [3]. The pitch contours are extracted using voice signal analysis. The extraction is done for 15 speakers and the sentence used is “The Cat Sat”. The block diagram for pitch extraction using LabVIEW platform is shown in the Figure 2.

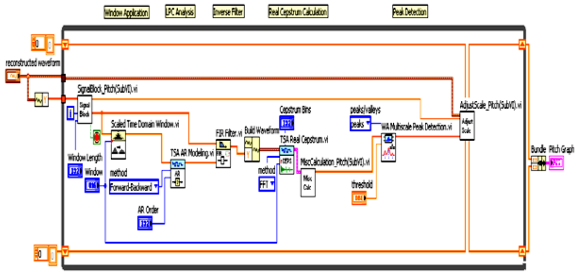


Figure-2 VI Block diagram of pitch extraction

2.2 Vector quantization

Vector Quantization (VQ) is a process of mapping vectors of a large vector space to a finite number of regions in that space. Each region is called a cluster and is represented by its centre (called a centroid). A collection of all the centroids makes up a codebook. The amount of data is significantly less, since the number of centroids is at least ten times smaller than the number of vectors in the original sample. This will reduce the amount of computations needed for comparison in later stages. Even though the codebook is smaller than the original sample, it still accurately represents a person’s voice characteristics. The only difference is that there will be some spectral distortion [2].

In an earlier feature extraction stage, the LPC cepstrum is calculated and the entire speech signal is represented as the LPC to cepstrum parameters and a large sample of these parameters are generated as the training vectors. During the training process of Vector Quantization, a codebook is obtained from these sets of training vectors. These training vectors are actually compressed to reduce the storage requirement. An element in a finite set of spectra in a codebook is called a codevector. Hence, data compression of speech is accomplished by Vector Quantization in the training phase. The speaker verification is done in testing phase by calculating the minimum Euclidian distance between the codes generated by the input speech signals and that of the stored codevectors.

3.RESULTS AND DISCUSSION

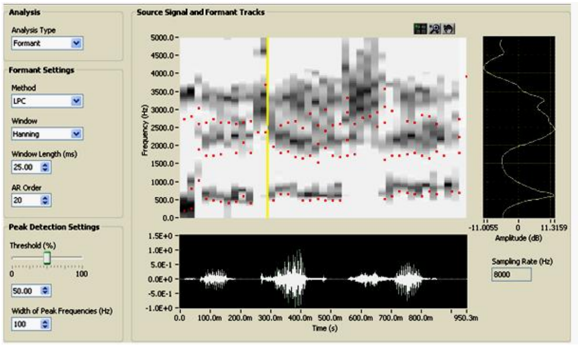
3.1 Formant extraction in speech signal

In this section the input speech signal is loaded into the voice signal analysis.vi and the formant frequencies are extracted and stored in the excel file which can be used for the future analysis. The sentence spoken by the speakers is “The cat sat” since here the phoneme “ah” is pronounced clearly by everyone. 15 speakers are trained here. The signal is allowed to pass through the sliding window of duration 25ms and the Hanning and Hamming windowing techniques are used here, since both give better results for speech signal.

3.1.1 Formant extraction using Hanning window and Hamming window

The formant frequencies are extracted by using Hanning window. Since the human speech varies from 55 Hz to

1000 Hz, the lower cutoff frequency is chosen as 1000 Hz. To get better pitch extraction, the all pole autoregressive model is having the order of 20. The window length of 20ms to 30ms is enough for text dependent speaker verification system. To get formant frequencies upto F4 the length is set as 25ms. Figures 3 and 4 shows the Formant extraction and Formant comparison using Hanning window. Figure 5 and 6 shows the Formant extraction and Formant comparison using Hamming window. The speaker's speech signal is sampled at 8000 Hz.



Analysis
Analysis Type
Formant
Formant Settings
Method
LPC
Window
Hanning
Window Length (ms)
25.00
AR Order
20
Peak Detection Settings
Threshold (%)
50.00
Width of Peak Frequencies (Hz)
100
Sampling Rate (Hz)
8000

Figure-3 Formant extraction using Hanning window

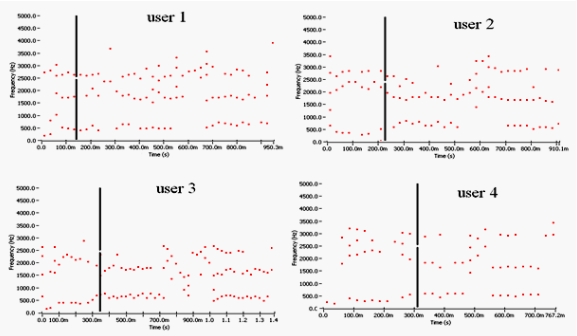
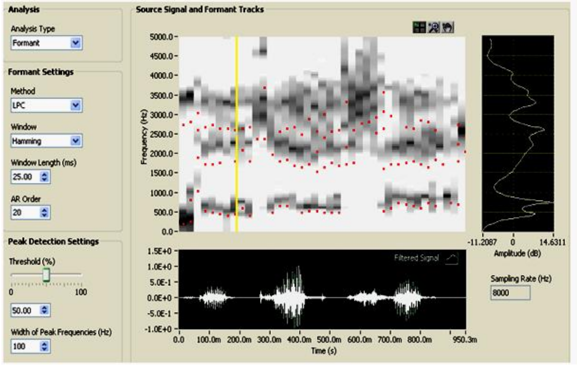


Figure-4 Formant comparison for various speakers using Hanning window



Analysis
Analysis Type
Formant
Formant Settings
Method
LPC
Window
Hamming
Window Length (ms)
25.00
AR Order
20
Peak Detection Settings
Threshold (%)
50.00
Width of Peak Frequencies (Hz)
100
Sampling Rate (Hz)
8000

Figure-5 Formant extraction using Hamming window

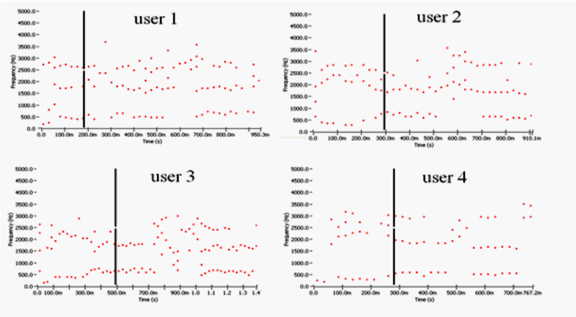


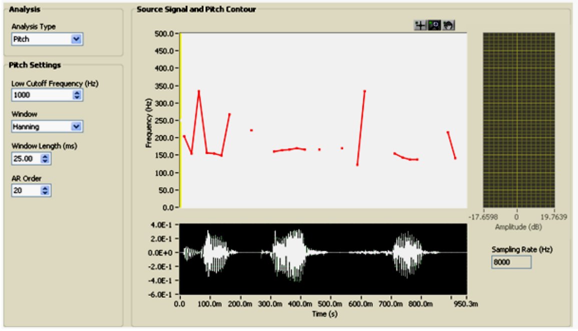
Figure-6 Formant comparison for various speakers using Hamming window

3.2 Pitch extraction in speech signal

In this section the input speech signal is loaded into the voice signal analysis.vi and the pitch is extracted and stored in the excel file which can be used for the future analysis. The sentence spoken by the speakers is "The cat sat", since here the phoneme "ah" is pronounced clearly by every one. There are 15 speakers used here. The signal is allowed to pass through the sliding window of duration 25ms and the windowing technique used here are Hanning and Hamming windows, since both give better results for speech signal.

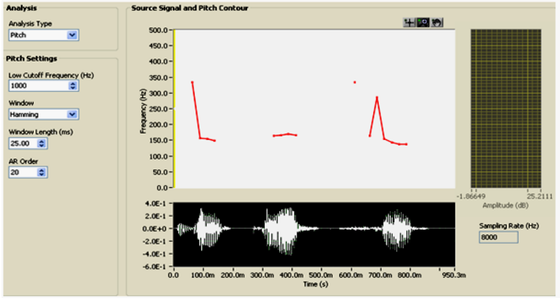
3.2.1 Pitch extraction using Hanning window and Hamming window

The formant frequencies are extracted by using Hanning window. The lower cutoff frequency is chosen as 1000 Hz. To get better pitch extraction the all pole autoregressive model is having the order of 20. The window length used here is 25ms. Figures 7 and 8 shows the extraction of pitch contour using Hanning window and Hamming window. Figures 9 and 10 shows the comparison of pitch for various speakers using Hanning window and Hamming window. The speaker's speech signal is sampled at 8000 Hz.



Analysis Type: Pitch
Low Cutoff Frequency (Hz): 1000
Window: Hanning
Window Length (ms): 25.00
AR Order: 20
Sampling Rate (Hz): 8000

Figure-7 Pitch extraction using Hanning window



Analysis Type: Pitch
Low Cutoff Frequency (Hz): 1000
Window: Hamming
Window Length (ms): 25.00
AR Order: 20
Sampling Rate (Hz): 8000

Figure-8 Pitch extraction using Hamming window

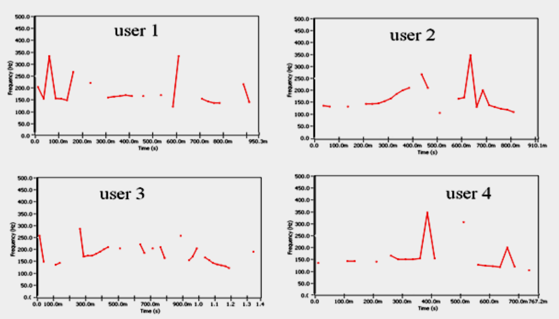


Figure-9 Pitch comparison for various speakers using Hanning window

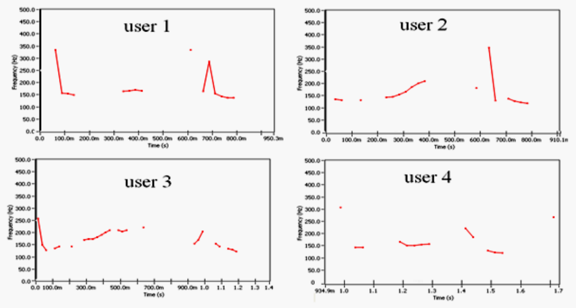


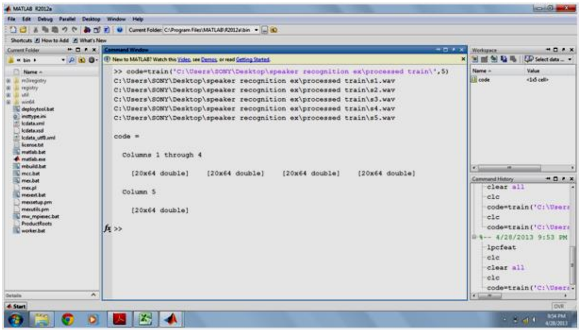
Figure-10 Pitch comparison for various speakers using Hamming window

Pitch and formants extraction using Hanning window yields better details than Hamming window. Also, the number of formant frequencies and pitch contours obtained are more in Hanning window, when compared to the Hamming window.

3.3 Generating Code vectors using Vector quantization

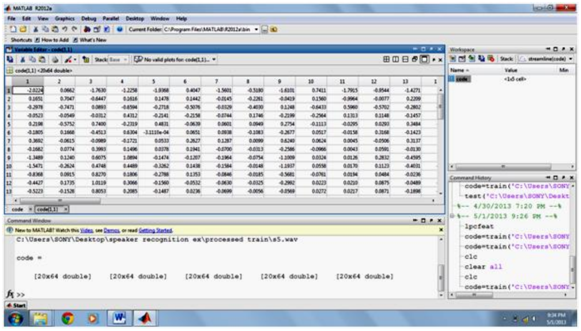
For resizing the filter coefficients, Vector Quantization is used and it generates the code vectors and stores these speaker models in the vector space based upon the centroids generated by the code vectors.

Figure 11 shows the MATLAB output window which contains the generated code vectors. Figure 12 shows the codes generated for single speaker after vector quantization.


MATLAB R2012a
File Edit Debug Parallel Desktop Window Help
Current Folder: C:\Program Files\MATLAB\R2012a\bin
Shortcuts How to Add What's New
Command Window
Workspace
Command History
Start

```
>> code=train('C:\Users\SONY\Desktop\speaker recognition ex\processed train\',5)
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s1.wav
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s2.wav
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s3.wav
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s4.wav
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s5.wav

code =

  Columns 1 through 4

    [20x64 double]    [20x64 double]    [20x64 double]    [20x64 double]

  Column 5

    [20x64 double]

fx >>
```

Figure-11 Output window containing generated code vectors


MATLAB R2012a
File Edit View Graphics Debug Parallel Desktop Window Help
Current Folder: C:\Program Files\MATLAB\R2012a\bin
Shortcuts How to Add What's New
Variable Editor - code{1,1}
Stack: Base
No valid plots for: code{1,1}...
Workspace
Command History
Command Window
code
code{1,1}
Start

code{1,1} <20x64 double>

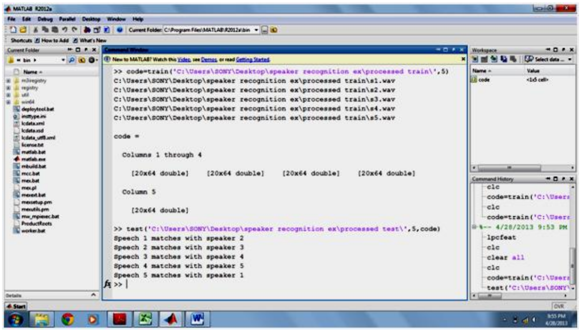
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	-2.0224	0.0662	-1.7630	-1.2258	-1.9368	0.4047	-1.5601	-0.5180	-1.6101	0.7411	-1.7915	-0.8544	-1.4271
2	0.1651	0.7047	-0.6447	0.1616	0.1478	0.1442	-0.0145	-0.2261	-0.0419	0.1560	-0.9964	-0.0077	0.2209
3	-0.2978	-0.7471	0.0893	-0.8594	-0.2718	-0.5076	-0.0329	-0.4030	0.1248	-0.6433	0.5960	-0.5702	-0.2802
4	-0.0523	-0.0549	-0.0312	0.4312	-0.2141	-0.2158	-0.0744	0.1746	-0.2199	-0.2564	0.1313	0.1148	-0.1457
5	0.2198	-0.5752	0.7400	-0.2319	0.4831	-0.0639	0.0601	0.0949	0.2754	-0.1113	-0.0295	0.0293	0.3484
6	-0.1805	0.1668	-0.4513	0.6304	-3.1110e-04	0.0651	0.0938	-0.1083	-0.2677	0.0517	-0.0158	0.3168	-0.1423
7	0.3692	-0.0615	-0.0989	-0.1721	0.0533	0.2627	0.1287	0.0099	0.6240	0.0624	0.0045	-0.0506	0.3137
8	-0.1682	0.0774	0.3993	0.1496	0.0378	0.1941	-0.0700	-0.0313	-0.2586	-0.0966	0.0043	0.0591	-0.0130
9	-1.3489	0.1240	0.6075	1.0894	-0.1474	-0.1207	-0.1964	-0.0754	-1.1009	0.0324	0.0126	0.2832	-0.4595
10	-1.5471	-0.2624	0.4748	0.4489	-0.3262	0.1438	-0.1584	-0.0148	-1.1937	0.0558	0.0170	0.1123	-0.4031
11	-0.8368	0.0915	0.8270	0.1806	-0.2788	0.1353	-0.0846	-0.0185	-0.5681	-0.0761	0.0194	0.0484	-0.0236
12	-0.4427	0.1735	1.0119	0.3066	-0.1560	-0.0532	-0.0630	-0.0325	-0.2992	0.0223	0.0210	0.0875	-0.0489
13	-0.5223	-0.1528	0.8053	0.2085	-0.1487	0.0236	-0.0699	-0.0056	-0.0569	0.0272	0.0217	0.0871	-0.1898

```
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s5.wav

code =

    [20x64 double]    [20x64 double]    [20x64 double]    [20x64 double]    [20x64 double]

fx >>
```

Figure-12 Generated codes for single user using vector quantization

3.4 Speaker Verification during testing phase

During training phase the code vectors are stored in the vector space. The codes generated during the testing phase are also located in the same vector space based on the centroid values generated. Now speaker verification is done by comparing the minimum Euclidean distance between the two code vectors. Here the analysis is shown for five speakers. Figure 13 shows the output of the speaker verification system.



```
>> code=train('C:\Users\SONY\Desktop\speaker recognition ex\processed train\',5)
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s1.wav
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s2.wav
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s3.wav
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s4.wav
C:\Users\SONY\Desktop\speaker recognition ex\processed train\s5.wav

code =

  Columns 1 through 4

    [20x64 double]    [20x64 double]    [20x64 double]    [20x64 double]

  Column 5

    [20x64 double]

>> test('C:\Users\SONY\Desktop\speaker recognition ex\processed test\',5,code)
Speech 1 matches with speaker 2
Speech 2 matches with speaker 3
Speech 3 matches with speaker 4
Speech 4 matches with speaker 5
Speech 5 matches with speaker 1
>> 
```

```
clc
code=train('C:\Users
clc
code=train('C:\Users
%-- 4/28/2013 9:53 PM
lpcfeat
clc
clear all
clc
code=train('C:\Users
test('C:\Users\SONY\
```

Figure-13 Output of the Speaker Verification system

4. CONCLUSION

Speaker Verification for various speakers has been done and the outputs for feature extraction like pitch extraction, formant extraction and generation of code vectors are obtained. The speech signal is acquired by using microphone from various speakers. The pitch and formant extraction is done using two windowing techniques and the Hanning window gives better pitch extraction. The Speaker Verification done by Vector Quantization performs better. The future work includes: increasing the robustness of the system; recognizing the speaker by identifying the unauthorized speakers. This speaker verification approach is efficient for biometric systems.

REFERENCES

[1] Amin Fazel, Shantanu Chakrabartty. (2011), 'An Overview of Statistical Pattern Recognition Techniques for Speaker Verification', IEEE circuits and systems magazine, second quarter 2011, pp.62-81.

[2] Lawrence Rabiner, Biing-Hwang Juang, 'Fundamentals of speech recognition', Prentice-Hall international, Inc, 2nd edition 2001.

[6] Vanishree Gopalakrishna, Nasser Kehtarnavaz (2012), 'Real-Time Automatic Tuning of Noise Cancelling System', 12th Int. Conference on Digital Audio Effects(DAFx-09), Como, Italy.

[3] Adnene Cherif, Lamia Bouafif, Turkia Dabbabi. (2001), 'Pitch detection and formant analysis of Arabic speech processing', Elsevier, Applied Acoustics 62, pp.1129–1140.

[4] David Pearce, Hans-Günter Hirsch. (2000), 'The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions' ICSLP 2000, Beijing, China.

[5] Huang, Xuedong, Alex Acero, Hsiao-Wuen Hon. (2001). "Spoken Language Processing", Prentice Hall PTR. ISBN 0-13-022616-5 pp. 325.

[7] Jong-Hwan Lee, Ho-Young Jung, Te-Won , Soo-Young Lee. (2000), 'Speech Feature Extraction Using Independent Component Analysis', IEEE international conference on Acoustics, Speech and signal processing, ICASSP'00, Vol 3, pp. 1631 - 1634.

[8] Kumar Rakesh, Subhangi Dutta and Kumara Shama. (2011), 'Gender recognition using speech processing techniques in LabVIEW', IJAET Vol.1, Issue 2, pp.51-63.

[9] Man-Wai Mak, Sun-Yuan Kung, (2000), 'Estimation of Elliptical Basis Function Parameters by the EM Algorithm with Application to Speaker Verification', IEEE Transactions on Neural Networks, Vol.11, No.4, pp. 961-969.

[10]Matthias Wolfel. (2009), 'Enhanced Speech Features by Single-Channel Joint Compensation of Noise and Reverberation' IEEE transactions on audio, speech, and language processing, Vol. 17, No. 2, pp.312-323.

[11]Ozlem Kalinli, Michael L. Seltzer, Jasha Droppo, Alex Acero. (2010), 'Noise Adaptive Training for Robust Automatic Speech Recognition' IEEE Transactions on audio, speech, and language processing, Vol. 18, No.8, pp.1889-1901.

[12]Ronan Flynn, Edward Jones. (2010), 'Robust distributed speech recognition in noise and packet loss conditions' Elsevier Digital Signal Processing 20(2010), pp.1559–1571.