Voice Activity Detection in the presence of Non-stationary Background Noise

A PROJECT REPORT

submitted in partial fulfillment for the

SUMMER INTERNSHIP PROGRAM

by

Sunil Srinivasa (EE00124)
IIT Madras

under the guidance of

Dr. N. Rama Murthy
Scientist 'F'
Signal Processing Group (SPG)
Centre for Artificial Intelligence and Robotics(CAIR)

June 2003

CERTIFICATE

This is to certify that, the report titled "Voice Activity Detection in the presence of Non-stationary Background Noise" is a bona fide record of the work done by Sunil Srinivasa, EE00124, Department of Electrical Engineering, Indian Institute of Technology, Madras, under my guidance and supervision, in partial fulfillment of the requirements for the Summer Internship Program.

Dr. N. Rama Murthy
Scientist 'F' , Signal Processing Group (SPG)
Center for Artificial Intelligence And Robotics (CAIR)
Bangalore

Date: June 20, 2003
Place: Bangalore

Acknowledgments

I express my sincere gratitude to my mentor and adviser, Dr. N. Rama Murthy for his guidance, patience and encouragement throughout the development of this project. He was always available for stimulating discussions and provided the best support I could have desired. I am also indebted to him for the encouragement and the freedom I enjoyed throughout. I also acknowledge the invaluable assistance offered by Mr. Sanjeev Gupta. His cheerful and helping manner aided me in the project.

I would also like to thank all my professors in the Electrical Engineering Department at IIT Madras. I also take this opportunity to thank my parents and my brother, Sudhir. Their help never wavered.

Miscellaneous Information

This document was written in LaTeX, using MikTeX 2.2.7 (www.miktek.org) and TeXnicCenter (www.toolscenter.org). Simulation code was written in Matlab 6.1, with the DSP and Communications Toolboxes. The Matlab code in the m and html formats; and this document, in the pdf, ps and tex formats are available for download from www.geocities.com/suniliitm.

Contents

Chapter 1

Overview

Voice is the basic mode of communication that conveys the content of speech and helps to identify the production source. With the growth of technology in the field of electronics and communications, we have moved from a world of telephones to an era of mobile technology and the Internet. In the present day, use of mobile phones/cellular handsets has increased significantly. The set up and applications of these ingenious gadgets entails the thorough exploitation of the knowledge and the underlying concepts of voice communications. Hence voice communications forms an inseparable part of our daily lives.

The widespread use of cellular/wireless sets has significantly increased the use of communication systems in high-noise environments. Impairments to the generated speech could occur due to the presence of high background noise or noise picked up in the channel. This noise pick-up results in degradation of the quality or intelligibility of speech, thus deteriorating the performance of many existing signal processing algorithms such as those used in speech coding, speech recognition, speaker identification, channel coding, echo cancellation, etc. Detection of voice segments in the presence of noise becomes imperative to these systems.

Voice activity detection is the first step in all the above mentioned speech processing techniques. Detected voice segments can be further processed to offer accurate results in these systems. However, voice activity detection works well only in environments where the signal-to-noise ratio(SNR) is high. Since most vocoders and voice recognition units require high SNR, low SNR automatically degrades their performance. With the development of hands-free and voice-activated cellular phones, noise reduction techniques are necessary to improve voice quality in noisy environments. To ameliorate the quality and intelligibility of the detected speech signal, speech enhancement is carried out.

1.1 The Voice Activity Detector

Voice activity detector(VAD), as the name suggests detects voice activity. It carries out the process of identifying the boundaries(beginning and end points) of speech and silence in a conversation. In other words, the VAD finds the beginning and ending of voice spurts. The VAD works based on some set of rules incorporated in the voice activity detection algorithms.

1.2 Advantages of VAD

- Detection of voice activity helps to implement noise cancellation algorithms to specified speech segments only and thus saves the processing time in canceling noise from speech.

- Identifying and rejecting transmission of noise-only periods helps reduce Internet traffic.

- A reliable VAD detects only the voiced fragments and discards the high frequency noise samples. Hence a great deal of bandwidth reduction is achieved, while still maintaining the voice quality. Keeping bandwidth low is of prime importance in voice over packet networks.

- VAD also allows for multi-data transmission. Since voice activity is present only during a small part in conversational speech, the silence periods could be used to transmit additional data, thus improving the bit rate of the channel.

1.3 Applications of VAD

VAD is required in some communication applications such as speech recognition, speech coding, hands-free telephony, audio conferencing, echo cancellation, etc. It forms an integral part of transmission Systems today. Voice over IP systems necessarily require the incorporation of a VAD for providing toll grade voice quality. Specific applications like digital cordless telephone systems, cellular networks and the Pan-European digital cellular mobile telephone service entail the employment of a VAD.

As it is well known, a VAD achieves silence compression, which is very important in both fixed and mobile telecommunication systems. In communication systems based on variable bit-rate speech coders(eg. in multimedia or VoIP applications), it represents the most important block, reducing the average bit-rate. In a cellular radio system using the discontinuous transmission(DTX) mode(eg. GSM or UMTS systems), a VAD is able to increase the number of users as well as reduce the average power consumption in portable equipment. The recently proposed multiple access schemes, such as CDMA and enhanced TDMA for cellular and PCS systems use some form of VAD.

1.4 Objectives of this project work

- Study of several methods for VAD proposed in the literature.
- Implementation of some of the VAD algorithms using the software MATLAB.
- Performance studies of the implemented VAD algorithms using standard speech databases.
- Study of some speech enhancement techniques that employ VAD.
- Software coding of the speech enhancement procedures learnt.
- Application of these coded algorithms to some sample noisy speeches.

Chapter 2

Voice Activity Detection Algorithms

VAD algorithms take recourse to some form of speech pattern classification to differentiate between voice and silence(in general, noisy) periods. They usually extract features, specific to voice, that are highly unlikely to be found in noisy segments. They then cut off parts of the speech signal based on a calculated(or assumed) threshold quantity. The threshold value that these algorithms calculate is based on the features extracted by them. The adaptiveness of this threshold vector results in reliable and robust separation of speech(containing voice) and noisy(silent) parts from the original noisy speech samples.

VADs were first investigated for use on TASI systems. Since then, various types of VAD algorithms that trade off delay, sensitivity, accuracy and computational cost have been proposed. They are based on extracted feature vectors, such as short-time energy levels, timing, pitch, zero-crossings rate, periodicity, cepstrum and various distance measures. VAD algorithms can implemented in time-domain, frequency-domain or a combination of both.

2.1 Desirable aspects of VAD algorithms

- **A good decision rule** : A physical property of speech that can be exploited to give consistent judgment in classifying segments of the speech into silent ones or those containing voice.

- **Adaptability to changing background noise** : Adapting to non-stationary background noise improves robustness, especially in wireless telephony where the user is mobile.

- **Low computational complexity** : Internet telephony is a real-time application. Therefore the complexity of VAD algorithm must be low to suit this practical application.

- **Toll quality voice reproduction** : Voice over packet or Voice over IP (VoIP) systems have to ensure that voice quality does not significantly deteriorate due to network conditions such as packet-loss and delays. Providing toll grade voice quality through VoIP systems remains a challenge. A robust VAD must be able to reproduce toll grade voice quality.

- Saving in bandwidth is to be maximized.

2.2 VAD algorithms implemented :

Amongst several VAD algorithms proposed in literature, the following ones have been presented in this report.

VAD algorithms - Time domain

1. The Variance-based Detector

2. The Linear Energy-based Detector(LED)

3. The Adaptive Linear Energy-based Detector(ALED)

4. The Zero-Crossings Detector(ZCD)

5. The Least Squares Periodicity Estimator(LSPE)-based Detector

VAD algorithms - Frequency domain

1. The Adaptive Linear Sub-band Energy-based Detector(ALSED)
2. The Spectral Flatness Detector(SFD)
3. The Cepstral Detector

2.3 VAD implementation

For any kind of VAD, the following basic processing steps are always carried out sequentially.

1. The input speech samples are first passed through a digital band-pass filter(300-3400 Hz) to remove the out-of-band components.
2. The speech samples are partitioned into frames, of possibly 10-30 ms duration with zero padding in the last frame(s), if necessary.
3. For all the algorithms implemented in this report, the input speech is sampled at 8 kHz and partitioned into 20 ms frames, giving a total of 160 samples per frame.
4. If the VAD is designed to work in the frequency domain, each frame is windowed (possibly using a Hamming window) and 2:1(50 percent) overlap is carried out. VADs implementable in time domain do not require windowing or overlapping of frames.
5. 'Feature vector' extraction is carried out from each of the speech frames.
6. A suitable threshold vector is defined based on the calculated values of the extracted feature for the first few frames.

7. Decision as to whether the frame contains voice or noise is taken based on this threshold measure.

8. The frames rendered as 'ACTIVE' are parts of the input speech containing speech in them. Noisy frames are entitled as 'INACTIVE' frames.

Chapter 3

The Variance-based Detector

3.1 Introduction

The Variance-based Detector[2] is the simplest and one of the robust VAD found in literature. It is also implemented completely in the time domain. It is based on the simple postulate that the variance of speech is far higher compared to noise. Also, white noise has a flat variance while that of speech is non-stationary. The threshold value to be used is based on this simple variance vector.

3.2 The Variance-based algorithm

- The basic processing steps are carried out on the input speech samples.
- The feature vector used here is the variance of each frame. Hence the variance of each 20ms non-overlapping frame is calculated.
- It is observed that the variance values for noise is considerably lower compared to that of speech. Also, these values are stationary compared to that of speech. The threshold value is completely based on this measurement.
- The calculated cutoff value is the average of all the variances of the first 10 frames (0.2 s). It is assumed that no speech is present at least for this period of time.

- **Decision Rule:**

  If $\sigma(f_j) \geq k \cdot threshold$, frame is 'ACTIVE'(containing speech).

  Else, frame is 'INACTIVE'(noisy).

  where,

  $\sigma(f_j)$ is the variance calculated for the the j^{th} frame,

  k is the scaling factor for allowing a safety margin (usually taken in the range 1-2).

The experimental results for the Variance-based Detector are shown in *Figure 3.1.*

3.3 Merits of the Variance-based Detector

- It is simple to implement in real-time circuits. All that it entails is a variance calculator.

- It reliably detects low energy phonemes at considerable SNR values.

3.4 Drawbacks of the Variance-based Detector

- Low SNR causes undue clipping in the input noisy signal.

- At very low SNR values, it makes numerous incorrect decisions, since variances of speech and noise catch up with each other.

3.5 Experimental Results

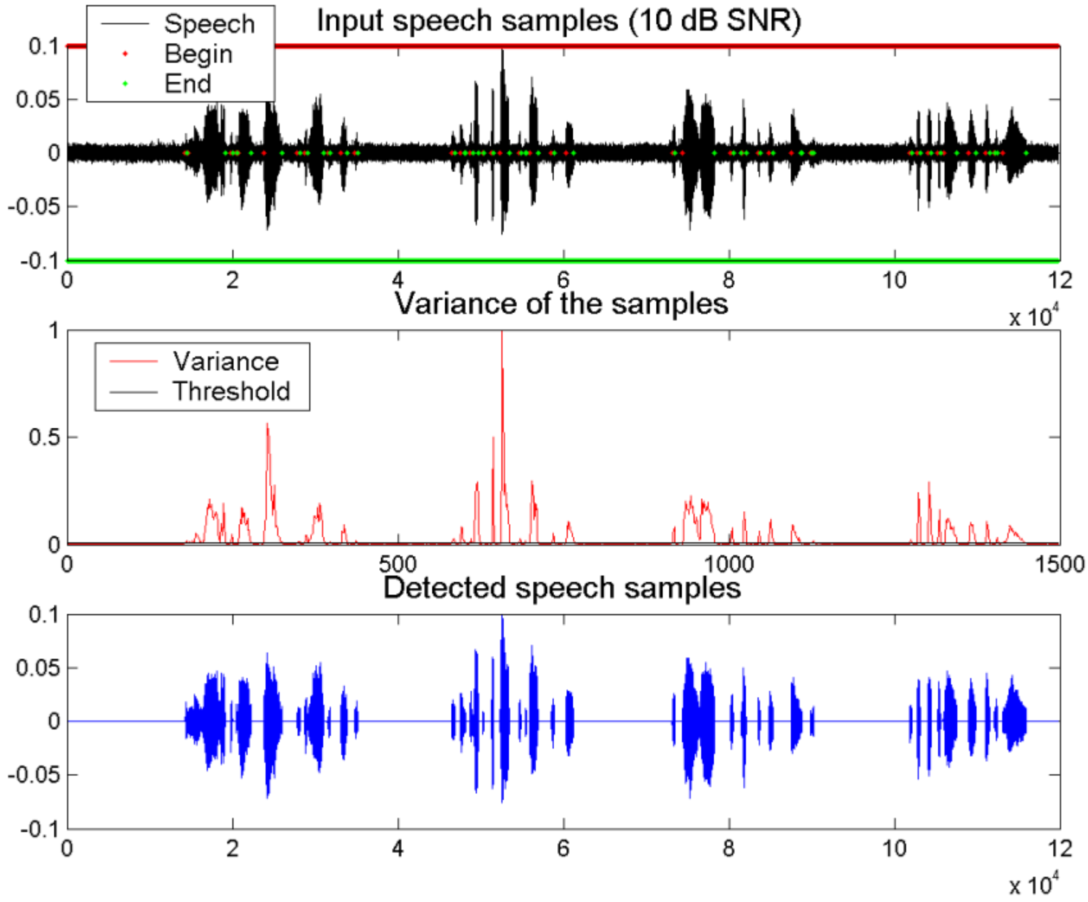

Fig.3.1: Response of the Variance-based VAD to the standard TIMIT speech.

Chapter 4

The Linear Energy-based Detector

4.1 Introduction

The LED[2] is a simple and reliable detector. It works completely in the time domain. Short-time energy is the parameter on the basis of which frames are classified as those containing speech or noisy. It is well known that energy of voice is considerably high compared to that of noise and thus suitable filtering of noise from voice can be carried out. To add robustness to this algorithm, an adaptive threshold is employed. However, adaptation is carried out only during the silent frames.

4.2 The LED algorithm

- The basic processing steps are carried out on the input speech samples.
- The feature vector used here is the energy of each frame. Hence the net energy of each 20ms non-overlapping frames is calculated.
- It is observed that the short-time energy values for noise is considerably lower compared to that of speech. Also, these values are stationary compared to that of speech. The threshold value is completely based on this measurement.
- **Achieving Adaptability:**

  To achieve robustness, the threshold value assumed must be re-computed during every inactive frame. The inactive frames are decided based on the decision taken by the variance detector. The equation which guards this rule is :

$$E_{th}(new) = E_{th}(old) \cdot (1 - p) + E_{sil} \cdot p \tag{4.1}$$

where,
$E_{th}(new)$ is the updated value of the threshold,
$E_{th}(old)$ is the previous energy threshold and
E_{sil} is the energy of the most recent silent/noisy frame.

The reference E_{th} is updated as a convex combination of the old threshold and the current noise update. p is chosen considering the impulse response of *Equation 4.1* as a first order filter($0 < p < 1$).
The Z-Transform of *Equation 4.1* is,

$$E_{th}(z) = (1 - p) \cdot z^{-1} \cdot E_{th}(z) + p \cdot E_{sil}(z) \tag{4.2}$$

The transfer function may be determined using,

$$H(z) = \frac{E_{th}(z)}{E_{sil}(z)} = \frac{p}{1 - (1 - p) \cdot z^{-1}} \tag{4.3}$$

For various values of p, the fall-time is plotted in *Figure 4.1.* It is observed that for $p = 0.2$, the fall-time (95 percent) corresponds to 15 delay units, i.e. 150ms. In effect, 15 past inactive frames influence the calculation for E_{th}.

- The initial value of is E_{th} is the average of energies of the first 10 frames (0.2 s). It is assumed that no speech is present at least for this period of time.

- **Decision Rule:**
  If $E(f_j) \geq k \cdot E_{th}$, frame is 'ACTIVE'(containing speech).
  Else, frame is 'INACTIVE'(noisy).

where,

$E(f_j)$ is the energy estimate for the the j^{th} frame,

k is the scaling factor for allowing a safety margin (usually taken in the range 1-2).

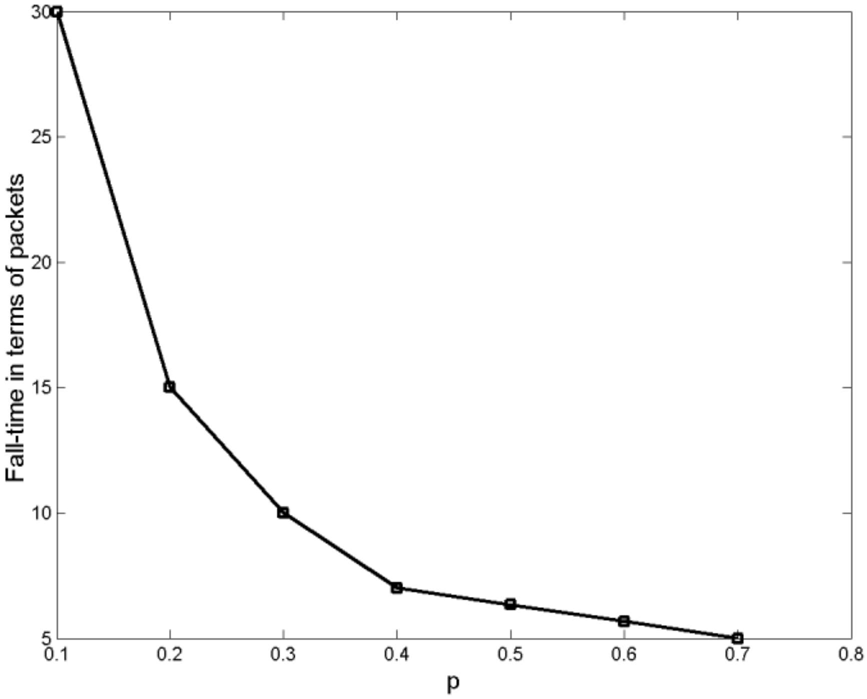

Fig.4.1: Fall-time for different values of p.

The experimental results for the LED are shown in *Figure 4.2*.

4.3 Merits of the LED

- Implementation in real-time is straightforward.
- It gives an acceptable quality of speech after compression.

4.4 Drawbacks of the LED

- This algorithm cannot give a good speech quality under varying background noise conditions. This is because the threshold is unable to keep pace with the rapidly varying background noise. Undue clippings, hence result at the beginning and ending of speech bursts.
- Non-plosive phonemes(utterances characterized by the lack of vibration of the vocal cords) like 'fish' and 'catch' are clipped completely. This is because the algorithm is exclusively based on energy calculations.
- Low SNR deteriorates the performance drastically.

4.5 Experimental Results

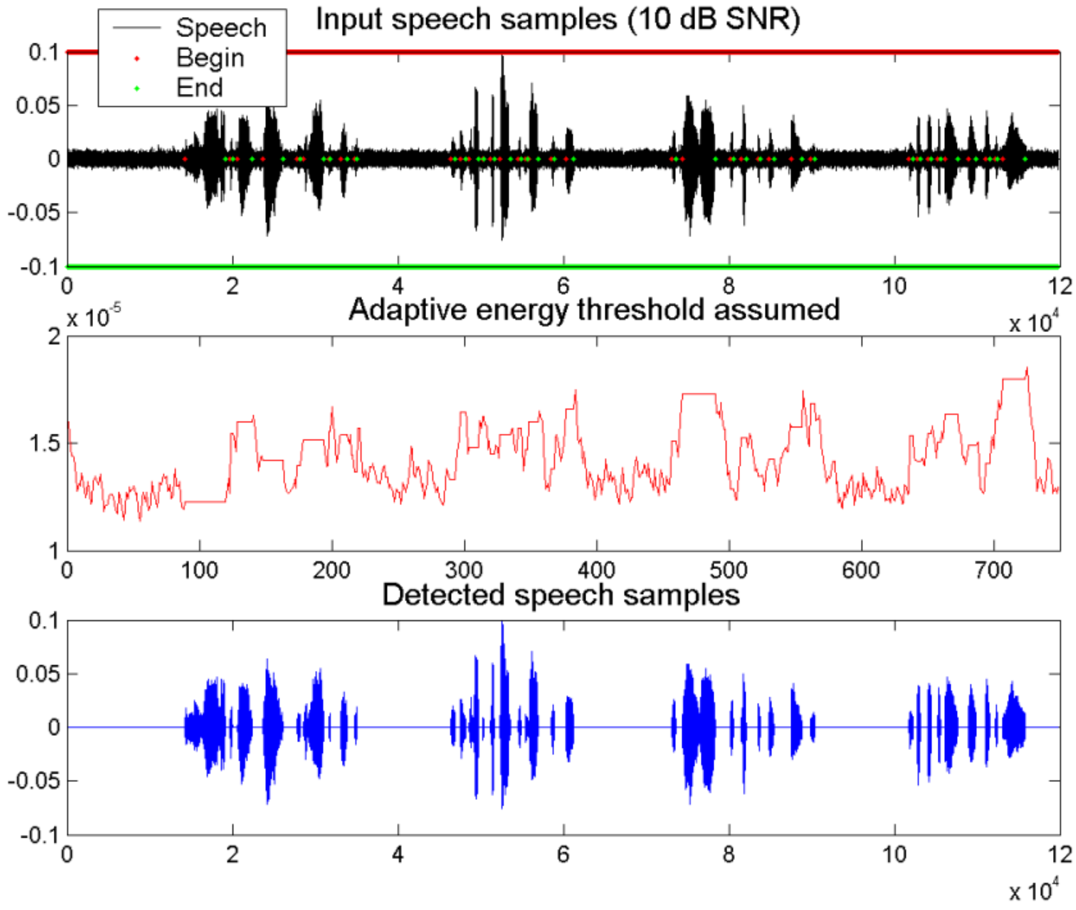

Fig.4.2: Response of the LED to the standard TIMIT speech.

Chapter 5

The Adaptive Linear Energy-based Detector

5.1 Introduction

The ALED[2] is another energy based detector. It also works completely in the time domain. Short-time energy is the parameter on the basis of which frames are classified as those containing speech or noisy. It works very similar to LED but for some enhanced adaptiveness. The sluggishness of LED is a consequence of p being insensitive to the noise statistics. Adaptability of p to the varying background noise makes the algorithm all the more hardy.

5.2 The ALED algorithm

- The basic processing steps are carried out on the input speech samples.
- The feature vector used here is the energy of each frame. Hence the net energy of each 20ms non-overlapping frames was calculated.
- It is observed that the short-time energy values for noise is considerably lower compared to that of speech. Also, these values are stationary compared to that of speech. The threshold value is completely based on this measurement.

- **Achieving Adaptability:**

  To achieve robustness, the threshold value assumed must be re-computed during every inactive frame. The inactive frames are decided based on the decision taken by the variance detector. The equation which guards this rule is :

  $$E_{th}(new) = E_{th}(old) \cdot (1 - p) + E_{sil} \cdot p \tag{5.1}$$

  where,
  $E_{th}(new)$ is the updated value of the threshold,
  $E_{th}(old)$ is the previous energy threshold and
  E_{sil} is the energy of the most recent silent/noisy frame.

  The reference E_{th} is updated as a convex combination of the old threshold and the current noise update. The value of P is varied as shown shortly.

- The initial value of E_{th} is the average of energies of the first 10 frames (0.2 s). It is assumed that no speech is present at least for this period of time.

- **Varying p :**

  A change in background noise is reckoned by comparing the variance of the present noisy frame with that of the previous one. The speech buffer contains the variance of the previous silent(noisy) frame denoted by σ_{old}. Let the variance of the latest noisy frame be σ_{new}. A profound adaptation is achieved by varying p as shown in *Table 5.1.*

The coefficients of the convex combination *Equation 5.1* now depend on the variance of the energies of inactive frames present in the buffer. We are able to make the otherwise languid E_{th} respond faster to sudden changes in the background noise. However, detection of active frames still remains energy-based.

Calculated $\frac{\sigma_{new}}{\sigma_{old}}$	p
$\frac{\sigma_{new}}{\sigma_{old}} \geq 1.25$	0.25
$1.25 \geq \frac{\sigma_{new}}{\sigma_{old}} \geq 1.10$	0.20
$1.10 \geq \frac{\sigma_{new}}{\sigma_{old}} \geq 1.00$	0.15
$1.00 \leq \frac{\sigma_{new}}{\sigma_{old}}$	0.10

Table 5.1: Value of p dependent on $\frac{\sigma_{new}}{\sigma_{old}}$

- **Decision Rule:**

  If $E(f_j) \geq k \cdot E_{th}$, frame is 'ACTIVE'(containing speech).

  Else, frame is 'INACTIVE'(noisy).

  where,

  $E(f_j)$ is the energy estimate for the the j^{th} frame,

  k is the scaling factor for allowing a safety margin (usually taken in the range 1-2).

The experimental results for the ALED are shown in *Figure 5.1.*

5.3 Merits of the ALED

- Implementation in real-time is straightforward.
- It gives an acceptable quality of speech after compression.
- This algorithm can give a good speech quality even under varying background noise conditions. This is because the threshold is able to keep pace with the rapidly varying background noise.

5.4 Drawbacks of the ALED

- Non-plosive phonemes like 'thief' and 'flower' are still found to be completely clipped as in LED.
- Low SNR deteriorates the performance drastically as in LED.

5.5 Experimental Results

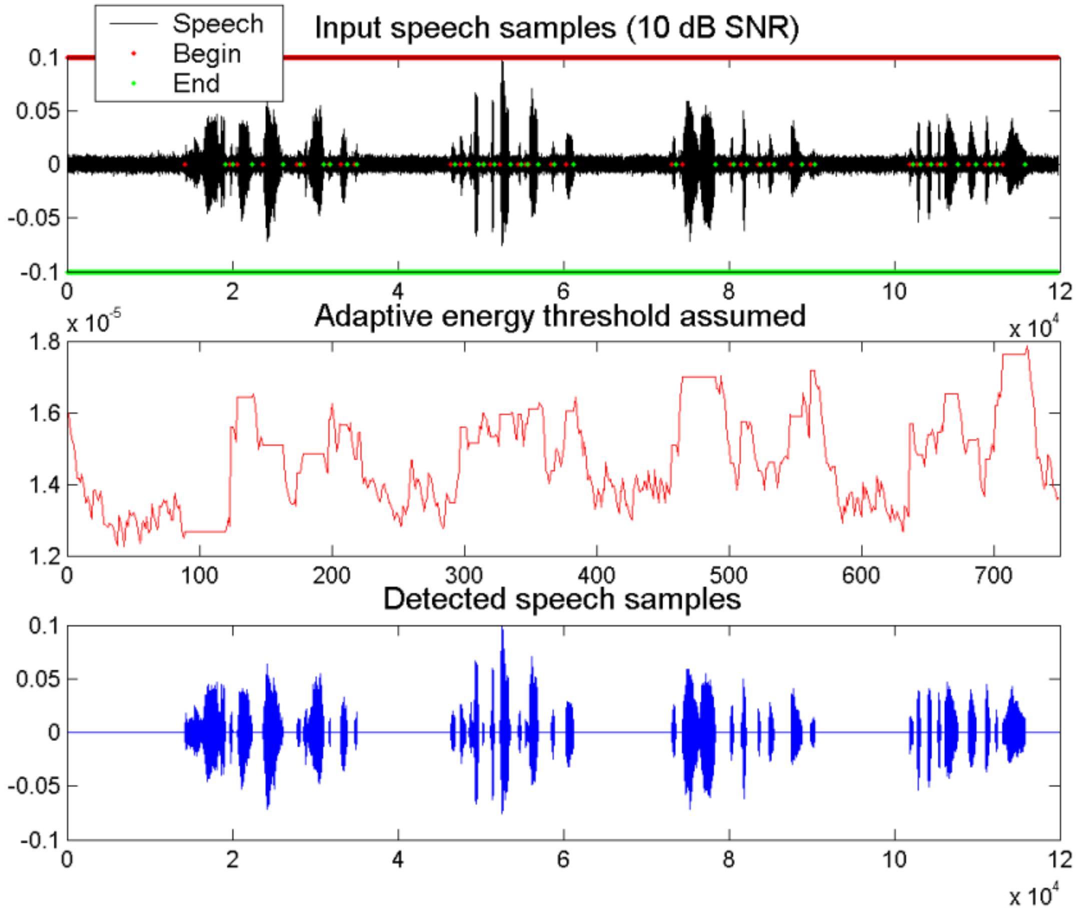

Fig.5.1: Response of the ALED to the standard TIMIT speech.

Chapter 6

The Zero Crossing Detector

6.1 Introduction

The ZCD[2] is also known as the Weak Fricatives Detector. It is implemented completely in the time domain. LED and ALED are completely energy based. Low energy phonemes are silenced completely. It is observed that high energy voiced segments are always detected by the VAD algorithms even under severe noisy conditions, while low energy frames are completely silenced. This detector overcomes this shortcoming of the energy based algorithms.

Zero-crossings is the measure of no. of times in a frame, the speech passes through zero amplitude. This algorithm is based on the fact that the zero-crossing rate of noise is considerably high compared to that of speech. Also, the no. of zero crossings for speech lies within a fixed range, whereas that for noise is random and unpredictable. Hence the threshold assumed is based on the no. of zero crossings.

6.2 The ZCD algorithm

- The basic processing steps are carried out on the input speech samples.
- The feature vector used here is the no. of zero-crossings per frame. Hence the no.of zero-crossings for each 20ms non-overlapping frames is calculated.

- It is observed that the no. of zero crossings for noise is considerably higher compared to that of speech. The threshold value is completely based on this measurement.

- The calculated threshold value is the average of all the zero crossings of the first 10 frames (0.2 s). It is assumed that no speech is present at least for this period of time.

- **Decision Rule:**
  If $N_{zcs}(f_j) \leq$ threshold, frame is 'ACTIVE'(containing speech).
  Else, frame is 'INACTIVE'(noisy).
  where,
  $N_{zcs}(f_j)$ is the No. of zero crossings in the j^{th} frame.

The experimental results for the ZCD are shown in *Figure 6.1.*

6.3 Merits of the ZCD

- It is simple to implement in real-time circuits. All that it requires is a zero-crossing counter.

- It reliably detects even low energy phonemes that were otherwise silenced by the simple energy or variance based techniques.

6.4 Drawbacks of the ZCD

- Low SNR causes undue clipping in the input noisy signal.

- It often makes incorrect decisions as noisy frames may contain the same no. of zero crossings as the speech frames or vice versa.

6.5 Experimental Results

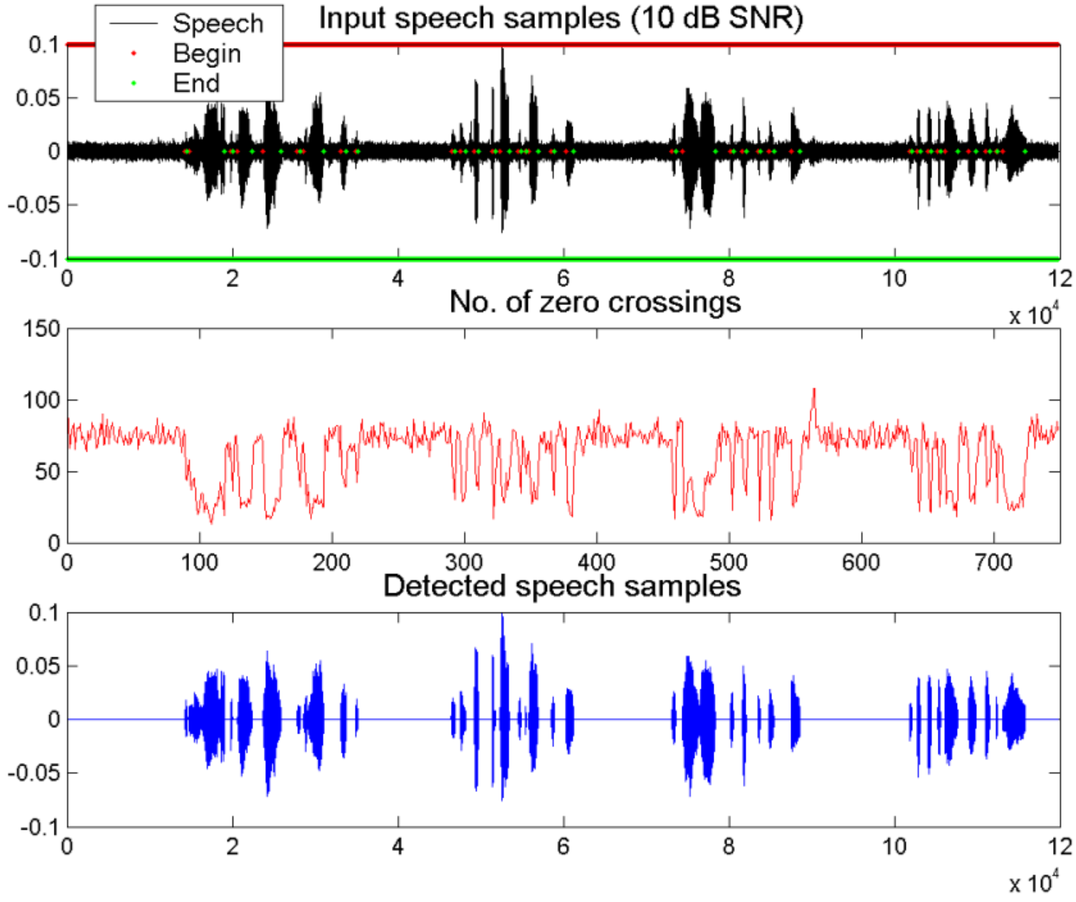

Fig.6.1: Response of the ZCD to the standard TIMIT speech.

Chapter 7

The Least Square Periodicity Estimator-based VAD

7.1 Introduction

This LSPE-based VAD[3] uses a periodicity measure to locate the voiced sections of speech. It is implemented completely in the time domain. This approach greatly reduces the probability of triggering on either white or impulsive noise, whether stationary or non-stationary, in very low SNR. However, the major difficulty in designing such a VAD, based on periodicity is its sensitivity to any periodic signal, which may well be interference or a background noise. Great care must be taken to stop false triggering on these signals. It is well known that the periodicity of speech is far higher than that of noise. The threshold vector is based on 'The normalized periodicity measure'.

7.2 The normalized periodicity measure

Let $s(i)$ be the input signal. $s(i)$ can be represented as:

$$s(i) = s_0(i) + n(i) \tag{7.1}$$

where,

$s_0(i)$ is the periodic component of the input signal.

$n(i)$ is the non-periodic component of the input signal.

If P_0 is assumed as its period, we have

$s_0(i) = s_0(i + kP_0)$ for any integer k.

Let $\hat{P}_0$ be our estimate for P_0.

The estimate for the periodic component of the signal $\hat{s}_0(i)$ is obtained from $\hat{P}_0$ as follows.

$$\hat{s}_0(i) = \sum_{h=0}^{K_0} \frac{s(i + h\hat{P}_0)}{K_0} \tag{7.2}$$

$$1 \leq i \leq \hat{P}_0 \ , \ P_{min} \leq \hat{P}_0 \leq P_{max}$$

where,

P_{min} and P_{max} are the minimum and maximum number of samples in a pitch period, and $K_0 = [(N - i)/\hat{P}_0] + 1$ is the number of periods of $\hat{s}_0(i)$ in the analysis frame.

The least squares method minimizes the error given by $\sum_{i=1}^{N} [s(i) - \hat{s}_0(i)]^2$ over each analysis frame. However, this method is biased towards large values of $\hat{P}_0$. To overcome this bias, we use the normalized periodicity measure or estimate. The normalized periodicity measure $R_1(\hat{P}_0)$ is defined as follows.

$$R_1(\hat{P}_0) = \frac{I_0(\hat{P}_0) - I_1(\hat{P}_0)}{\sum_{i=1}^{N} s^2(i) - I_1(\hat{P}_0)} \tag{7.3}$$

where,

$$I_1(\hat{P}_0) = \sum_{i=1}^{\hat{P}_0} \sum_{h=0}^{K_0} \frac{s(i + h\hat{P}_0)^2}{K_0} \tag{7.4}$$

and

$$I_0(\hat{P}_0) = \sum_{i=1}^{\hat{P}_0} \frac{[\sum_{h=0}^{K_0} s(i + h\hat{P}_0)]^2}{K_0} \tag{7.5}$$

7.3 LSPE processing

- The basic processing steps are carried out on the input speech samples. Also, preprocessing is carried out on the samples. A preprocessor is employed to remove any kind of interference in the signal. It usually removes the constant-frequency data and tones.

- For each 20 ms frame, $R_1(\hat{P}_0)$ is computed for the values of $\hat{P}_0$ between $P_{min} = 11$ and $P_{max} = 56$. The maximum value of $R_1(\hat{P}_0)$ obtained is the periodicity of the frame.

- Periodicity of voice is far higher compared to that of noise. The threshold value is completely based on this measurement.

- The calculated threshold value is the average of the periodicities of the first 10 frames (0.2 s). Speech is assumed to be absent during this period.

- **Decision rule:**
  If $\hat{P}(f_j) \geq$ threshold, frame is 'ACTIVE'(containing speech).
  Else, frame is 'INACTIVE'(noisy).
  where,
  $\hat{P}(f_j)$ is the periodicity estimate of the j^{th} frame.

The experimental results for the LSPE-based VAD are shown in *Figure 7.1.*

7.4 Merits of the LSPE-based VAD

- It detects reliably even at 5 dB SNR, provided that the preprocessor works robustly.

- It is very easily implementable in real-time circuits. The hardware required for this would consist of only summers and multipliers.

7.5 Drawbacks of the LSPE-based VAD

- The price paid for reliable detection is the slight loss of sensitivity in the detected samples.
- In cases where the preprocessor is unable to eliminate the interferences to the signal, numerous false triggerings take place.

7.6 Experimental Results

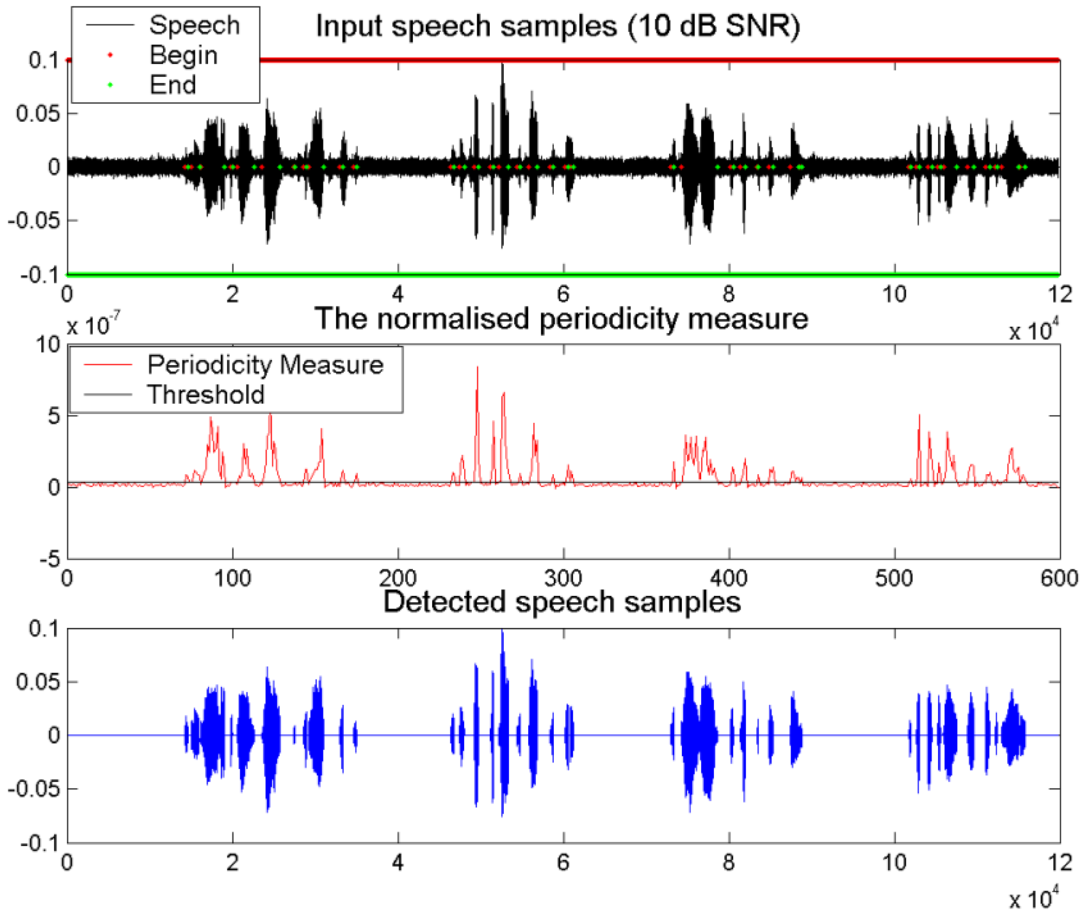

Fig.7.1: Response of the LSPE-based VAD to the standard TIMIT speech.

Chapter 8

The Adaptive Linear Sub-band Energy-based Detector

8.1 Introduction

The ALSED[2] is a frequency domain algorithm. It takes its decisions based on energy computations of the signal frame with a reference energy threshold in the frequency domain. The threshold is completely based on this energy counterpart in the frequency domain. Most of the energy in the voice signal tends to be in the lowest frequency band i.e. 0-1kHz. Selective threshold comparison in the lowest band alone gives good decisions.

8.2 The ALSED algorithm

- The basic processing steps are carried out on the input speech samples. Since it is implemented in the frequency domain, the frames are windowed with the Hamming window and overlapped in the ratio 2:1(50 percent).
- The spectrum obtained is divided into four bands of width 1kHz, i.e. the bands are 0-1kHz, 1-2kHz, 2-3kHz, 3-4kHz. The energy for each band is calculated as

$$E_n(f_j) = F_n^2(f_j) \quad for\ the\ n^{th}\ band \tag{8.1}$$

where, $F_n(f_j)$ is the discrete cosine transform(DCT) of the j^{th} frame of the n^{th} sub-band of the speech signal. DCT was used instead of DFT(discrete Fourier transform) for the following reasons.

  - It is computationally less complex as compared to DFT.
  - It is a real-valued transform.

- **Achieving Robustness:**
  To achieve robustness, the thresholds are computed recursively for each sub-band as

  $$E_{new(threshold)}(f_n) = (1-p) \cdot E_{old(threshold)}(f_n) + p \cdot E_{(new)}(f_n) \tag{8.2}$$

  $E_{(new)}(f_n)$ is the energy of the most recent silent frame, $E_{old(threshold)}(f_n)$ and $E_{new(threshold)}(f_n)$ are the previous and updated values of the threshold vector for the n^{th} sub-band.

  For achieving adaptability to the varying background noise, p is varied as shown in *Table 5.1* for each sub-band.

- The condition for presence of speech in each band is

  $$E_n(f) \geq E_{n(threshold)}(f) \tag{8.3}$$

  where,
  $E_{n(threshold)}(f)$ denotes the threshold assumed for the n^{th} sub-band. Usually, the calculated cutoff value is the average of all the spectral variances of the first 10 frames (0.2 s). It is assumed that no speech is present at least for this period of time.

- **Decision Rule:** A frame is declared to be ACTIVE(containing speech) if the lowest band(0-1kHz) is ACTIVE and any two of the remaining three bands are ACTIVE. Otherwise, it is termed as INACTIVE(noisy).

  The experimental results for the ALSED are shown in *Figure 8.1*.

8.3 Merits of the ALSED

- It gives an acceptable quality of speech after compression.
- Selective threshold comparison in the lowest sub-band alone provides good decisions.

8.4 Drawbacks of the ALSED

- Low SNR causes undue clipping in the input noisy signal.
- Low energy phonemes can't be detected even at considerable SNR values.

8.5 Experimental Results

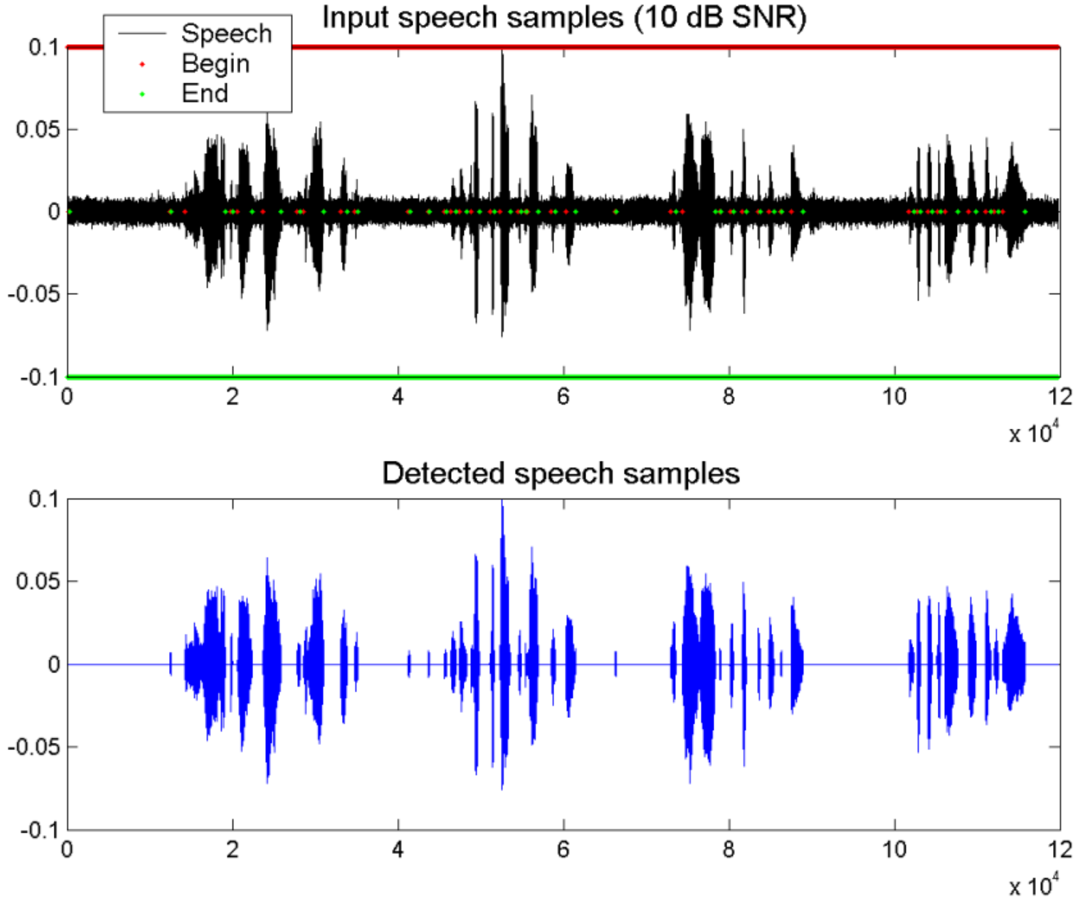

Fig.8.1: Response of the ALSED to the standard TIMIT speech.

Chapter 9

The Spectral Flatness Detector

9.1 Introduction

The SFD[2] is a reliable detector implemented completely in the frequency domain. It works robustly even at low SNR values. White noise has a flat spectrum while voiced signals have non-stationary spectrum with more spectral content in the lower frequencies. Thus high variance implies speech content while low variance implies noise alone. The threshold assumed is based on the spectral variance of the signal samples.

9.2 The SFD algorithm

- The basic processing steps are carried out on the input speech samples. Since it is implemented in the frequency domain, the frames are windowed with the Hamming window and overlapped in the ratio 2:1(50 percent).
- The feature vector used here is the spectral variance of each frame. Hence the spectral variance of each 20ms non-overlapping frames is calculated. The spectral variance of the j_{th} frame is given by :

  $$\sigma(f_j) = VAR[X(f)]; \tag{9.1}$$

  where,
  $X(f) = DCT(x(n))$; DCT stands for discrete cosine transform.

- It is observed that the spectral variance values for noise is considerably lower compared to that of speech. Also, these values are flatter compared to that of speech. The threshold value is completely based on this measurement.

- The calculated cutoff value is the average of all the spectral variances of the first 10 frames (0.2 s). It is assumed that no speech is present at least for this period of time.

- **Decision Rule:**
  If $\sigma(f_j) \geq k \cdot threshold$, frame is 'ACTIVE'(containing speech).
  Else, frame is 'INACTIVE'(noisy).
  where,
  $\sigma(f_j)$ is the spectral variance calculated for the the j^{th} frame,
  k is the scaling factor for allowing a safety margin (usually taken in the range 1-2).

The experimental results for the SFD are shown in *Figure 9.1*.

9.3 Merits of the SFD

- It is simple to implement in real-time. All that it entails is a variance calculator.

- It reliably detects low energy phonemes at considerable SNR values, as it makes use of a statistical approach to the energy distribution in the spectra.

9.4 Drawbacks of the SFD

- Low SNR causes undue clipping in the input noisy signal.

- At very low SNR values, it makes numerous incorrect decisions, since variances of speech and noise catch up with each other.

9.5 Experimental Results

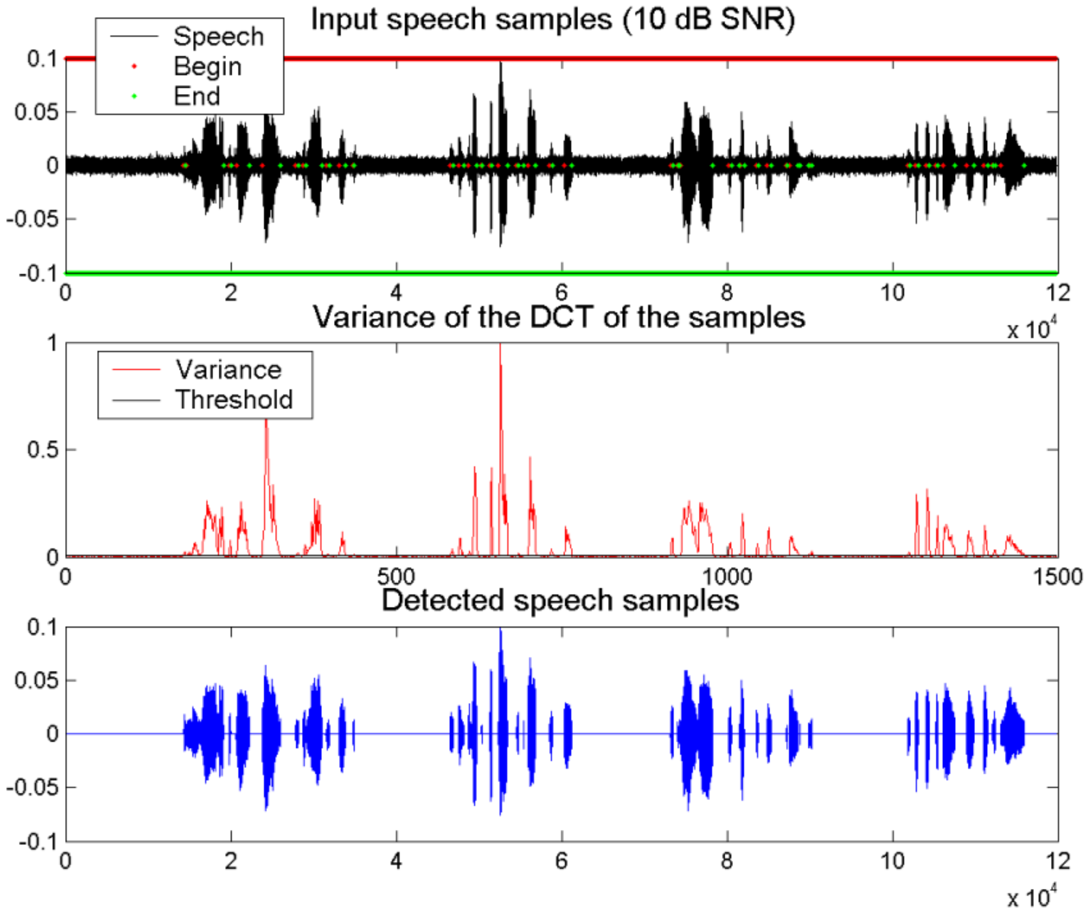

Fig.9.1: Response of the SFD to the standard TIMIT speech.

Chapter 10

The Cepstral Detector

10.1 Introduction

The Cepstral Detector[4] is based on the cepstral features of speech signals. It is a frequency domain technique. It is seen that the algorithms based on simple feature vectors like energy, variance and zero-crossing fail to produce effective results. The cepstral algorithm overcomes this shortcoming of other algorithms. It is suitable for VAD applications because of the fact that variance of cepstrum for voice is far greater than voice. In addition, unlike other techniques, cepstral coefficient values don't depend on absolute amplitude levels of a signal. Also, cepstral coefficients represent the frequencies present in speech clearly and effectively. Therefore, ambiguities and errors in feature vector calculations are minimized. Hence, this is a widely used technique.

10.2 Cepstrum:

For a real signal $X(t)$, complex cepstrum $X_c(n)$ is given as follows,

$$X_c(n) = \frac{1}{2\pi} \int\limits_{-\pi}^{\pi} \log |X(\omega)| e^{jn\omega} d\omega \qquad (10.1)$$

where,

$X(\omega)$ is the Fourier transform of the speech signal.

i.e. complex cepstrum $= IFT(\log|FT(X)|)$ where,
X is the speech signal,
FT and IFT denote the Fourier and inverse Fourier transforms.

For a sequence $x(n)$, discrete cepstrum $X_N(n)$ is defined as :

$$X_N(n) = \frac{1}{N}\sum_n X_l(k)e^{\frac{j2\pi n}{N}} \tag{10.2}$$

where,
$X_l(k)$ is the logarithm of the modulus of the Fourier transform $X(k)$ of the speech signal
i.e. $X_l(k) = \log|X(k)|$.
Time index n in cepstral domain is called 'quefrency'.

10.3 Properties of cepstrum:

- For a sequence $X(n)$ with rational Z-transform, complex cepstrum decays as fast as $\frac{1}{mod(n)}$, where n is the quefrency value. This shows that cepstrum values concentrate at $n = 0$.
- For a sequence consisting of train of impulses of constant period, complex cepstrum consists of an impulse train of a constant period equal to the pitch period.
- For minimum phase sequences obtained by linear prediction, a real cepstrum can be sufficient for analysis purpose.

10.4 Linear prediction analysis:

The basic idea behind linear prediction analysis is that each speech sample can be represented as a linear combination of the past speech sample values. Such representation leads to all-pole model of short-time speech spectrum. Linear prediction process is given as follows.

$$x(n) = a_1x(n-1) + a_2x(n-2) + a_3x(n-3) + ... + a_mx(n-m) \tag{10.3}$$

Equivalently,

$$x(n) = \sum_{k=1}^{m} a_kx(n-k) \tag{10.4}$$

where,

$a_1, a_2, ...a_m$ are defined as the m linear predictor coefficients(LPC) of the speech signal.

Linear prediction analysis is a modeling technique that can be effective for speech signals. It gives good computational speed and accuracy. The limitation of this technique is that it does not model zeros in a system, which are actually present in vocal tract transfer function. This leads to approximations in the use of this technique. Conversion of linear prediction coefficients to cepstral coefficients, however, ameliorates the approximations assumed.

10.5 Conversion of linear prediction to cepstral coefficients

LPC are converted to cepstral coefficients by the following equation.

$$c_n = l_n + \sum_{k=1}^{m}(1 - \frac{k}{n}) \cdot l_k \cdot c_{n-k} \tag{10.5}$$

where,

l_n represents the n^{th} LPC and

c_n is the n^{th} cepstral coefficient.

Major advantages of this method are:

- All-pole estimation is less affected by the effect of high pitch aliasing, due to reduction in waveform periodicity.
- Zero estimation can be performed, which is missed in all-pole modeling of speech. This reduces approximations to vocal tract response in a speech production system.

10.6 The Cepstral algorithm

- The basic processing steps are carried out on the input speech samples. Since this is a frequency domain algorithm, each frame is windowed with a Hamming window and 2:1(50 percent) overlap is carried out.
- The feature vector used here is the variance of the cepstral coefficients of each frame. LPC are first calculated and converted to cepstral coefficients. About twelve coefficients are sufficient as they can represent the speech effectively. Cepstral variance, which is the variance of the cepstral coefficients is calculated for each frame.
- It is observed that the cepstral variance for noise is considerably higher compared to that of speech. The threshold value is completely based on this measurement.
- The calculated threshold value is the average of the cepstral variances of the first 10 frames(0.2 s). It is assumed that no speech is present at least for this period of time.
- **Decision Rule:**

  If $Cepvar(f_j) \geq$ threshold, frame is 'ACTIVE'(containing speech).

Else, frame is 'INACTIVE'(noisy).

where,

$Cepvar(f_j)$ is the cepstral variance for the j^{th} frame.

The experimental results for the Cepstral Detector are shown in *Figure 10.1*.

10.7 Merits of the Cepstral Detector

- This method is advantageous as cepstra can effectively model the vocal tracts. Start of voice patches is accurately detected in most cases.
- It can detect accurately even in the presence of non-stationary noise.

10.8 Drawbacks of the Cepstral Detector

- Low SNR leads to numerous false detections.
- Also, excessive fricatives in a sentence affect VAD results.

10.9 Experimental Results

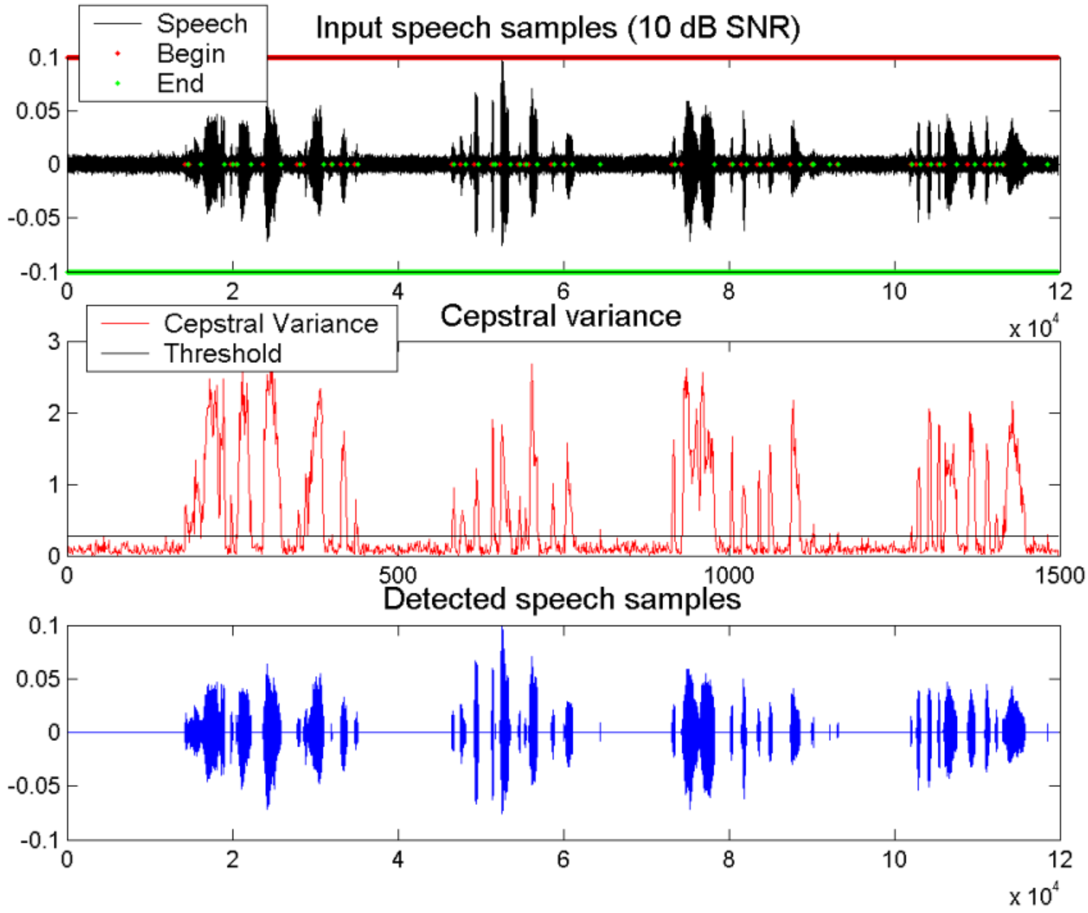

Fig.10.1: Response of the Cepstral VAD to the standard TIMIT speech.

Chapter 11

Comparison of the VAD Algorithms

The VAD algorithms discussed were put into testing using two standard databases:

1. The TIMIT speech database[9]
2. The CALLBASE digit database[10]

The test templates varied in loudness, speech continuity, content, accent and speed. Both male and female recordings were used for testing. Testing was carried out under SNR conditions varying from 10 dB all the way down to -5 dB. Noise added was Additive White Gaussian(AWGN) in nature. All the testing procedures were carried out in MATLAB environment.

Figure 11.1 and *Figure 11.2* represent the performance of the VADs under different SNR conditions for the two databases.

The performance of the algorithms was studied on the basis of the percentage of correctly detected samples.

11.1 Results of the project work on VAD

- Almost all the VADs worked well even at 10 dB SNR. Below this level, their performance deteriorated.
- Faithful detections were observed when the length of the words were short. As the sentence length increased, the performance deteriorated.

- As the SNR decreased, performance went worse. Errors such as missing of voiced parts and late detection of next voiced patches were seen.
- It was observed that the value of threshold assumed was extremely critical for the performance of the VAD. Adaptation of threshold gave more faithful results.
- As seen from the charts, the Variance-based Detector and the SFD performed outstandingly. Ironically, the simplest algorithm based on variance performed the best.

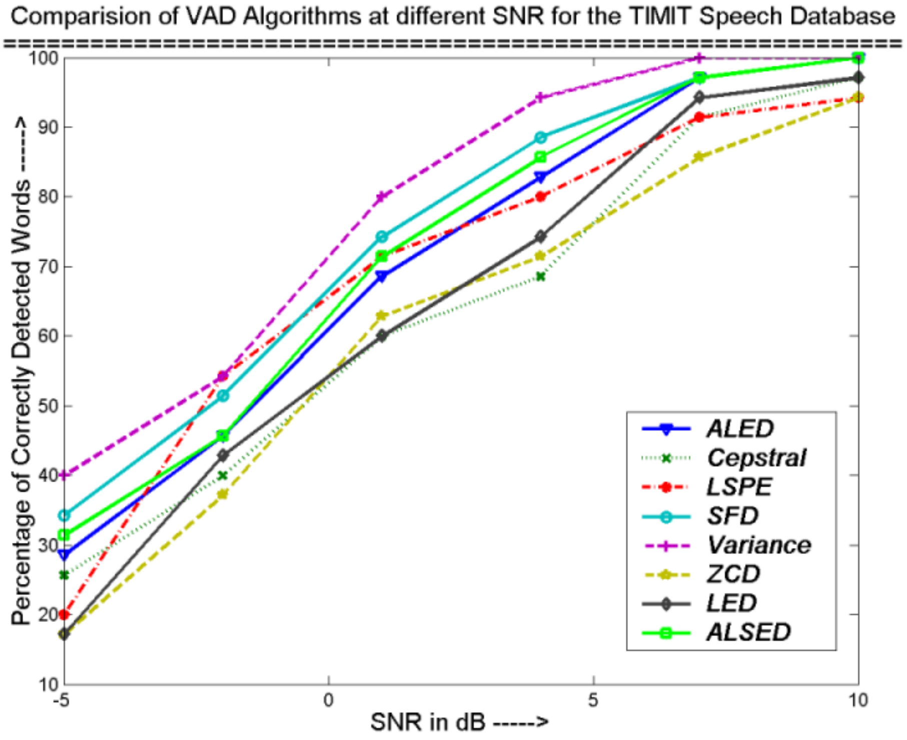

Fig.11.1: Comparison of the VADs for the TIMIT speech database.

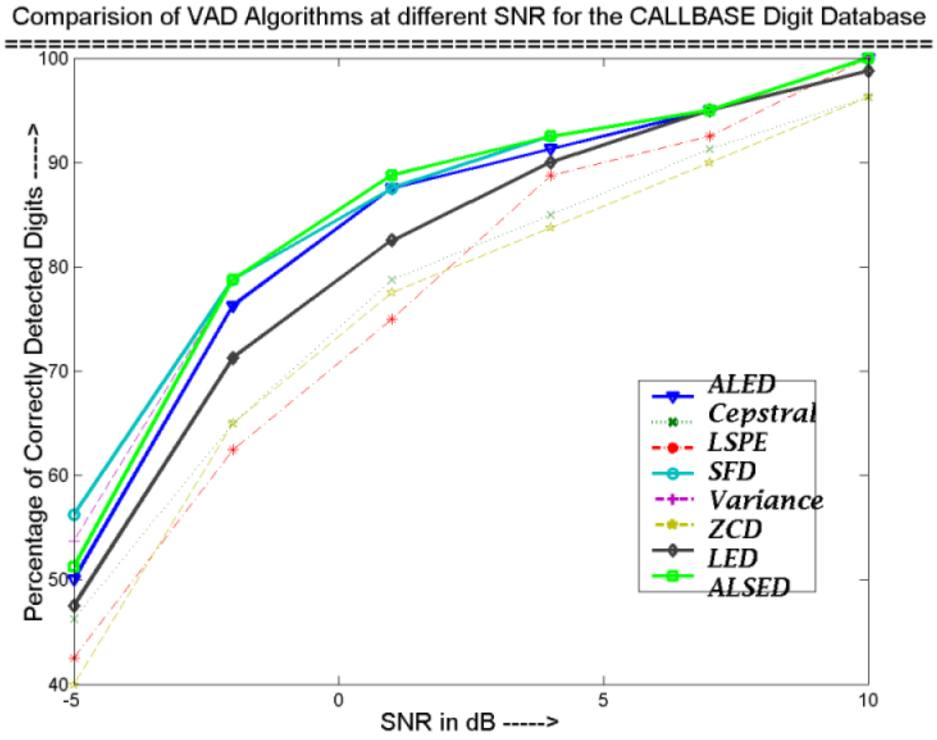

Fig.11.2: Comparison of the VADs for the CALLBASE digit database.

Chapter 12

Other VAD Algorithms

Several other VAD algorithms have been proposed in literature. Some of them are:

- VAD working on the fusion of two or more basic VADs.
- VAD based on the Viterbi algorithm.
- VAD using Bayesian adaptation with conjugate normal distributions.
- VAD based on a certain statistical model.
- VAD working on the principle of a hidden Markov model.
- VAD based on the higher-order statistics(HOS) of speech.
- VAD by tracking power envelope dynamics.
- VAD based on the perpetual wavelet packet transform.
- VAD working on the principle of CAPDM architecture.
- VAD based on time delay estimation and fuzzy activity classification.

Some well-known VADs used in practice these days are:
The GSM VAD, The TOS VAD, The IGSM VAD, The EVRC VAD, The ITU-T VAD, The ETSI VAD, G.729 and The AMR VAD.

Chapter 13

Noise Reduction Techniques

In many speech processing techniques, speech has to be processed in the presence of undesirable background noise. Noise reduction becomes imperative to improve noise quality and intelligibility. The purpose of many speech enhancement algorithms is to reduce background noise, improve speech quality, or suppress undesired interference. Several speech enhancement algorithms depend on a VAD to facilitate the estimation of the characteristics of noise.

There are three general classes of speech enhancement techniques[8]: subtraction of interference, suppression of harmonic frequencies and re-synthesis using vocoders. The first class of techniques suppresses noise by subtracting a noise spectrum. The second type of speech enhancement is based on the periodicity if noise. These methods employ fundamental frequency tracking using adaptive comb filtering of the harmonic noise. The third class of techniques is based on speech modeling using iterative methods. These systems focus on estimating model parameters that characterize the speech signal, followed by re-synthesis of the noise-free signal. They require a prior knowledge of noise and speech statistics and generally results in iterative enhancement schemes.

Noise subtraction algorithms can also be partitioned depending on whether a single-channel or dual-channel(or multiple-channel) approach is used.

Figure 13.1 shows the block diagram of a general speech enhancement algorithm employing a VAD.

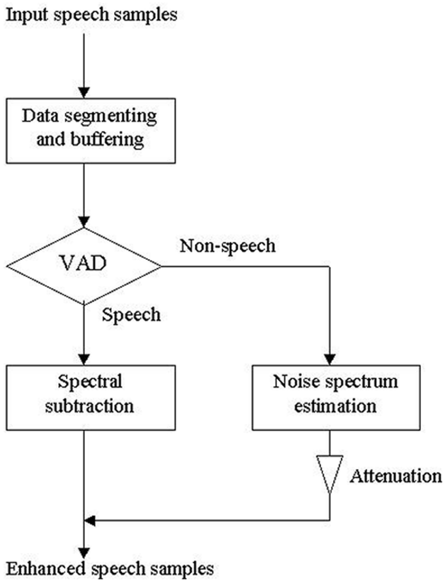

Fig.13.1: Block diagram of a general speech enhancement algorithm.

In the following chapters, two speech enhancement algorithms are presented. They are:

1. Speech Enhancement using a Minimum Mean Square Error Log-Spectral Amplitude Estimator(MMSE-LSAE)

2. Speech Enhancement by Adaptive Wavelet Packet

Chapter 14

Speech Enhancement using a MMSE-LSAE

14.1 Introduction

This algorithm[5] capitalizes on the major importance of the short-time spectral amplitude(STSA) of the speech signal in its perception, and utilizes a minimum mean-square error(MMSE) STSA estimator for enhancing the noisy speech. It is well known that a distortion measure, which is based on the mean-square error of the log-spectra is more suitable for speech processing. Such a distortion measure is therefore extensively used for speech analysis and recognition. For this reason, it is of great interest to examine the STSA estimator, which minimizes the mean square error of the log spectra in enhancing speech.

14.2 The MMSE-LSAE

Let $x(t)$ denote the speech signal and $n(t)$, the noise process. Let the analysis frame be from $0 \leq t \leq T$.
The noisy speech $y(t) = x(t) + n(t)$, $0 \leq t \leq T$.

The Fourier expansion coefficients of the speech process, as well as of the noise process, are modeled as statistically independent Gaussian random variables. This model utilizes asymptotic statistical properties(as $T \rightarrow \infty$) of spectral components. In particular, the Gaussian model is motivated by the central limit theorem and the statistical independence assumption is motivated by the fact that the correlation between the spectral components reduce as the analysis interval length increases.

Let $X_k = A_k e^{j\alpha_k}$, N_k and $Y_k = R_k e^{j\vartheta_k}$ denote the k^{th} Fourier expansion coefficient of the speech signal, the noise process and the noisy observations, respectively in the analysis interval[0,T]. According to the formulation of the estimation problem, we are looking for the estimator $\hat{A}_k$, which minimizes :

$$E[(\log A_k - \log \hat{A}_k)^2] \tag{14.1}$$

given the noisy observations[y(t),0 $\leq t \leq T$]. This estimator is easily shown to be

$$\hat{A}_k = exp[E(\ln A_k | y(t))], 0 \leq t \leq T \tag{14.2}$$

Note that the estimator*(Equation 14.2)* results also if we choose to minimize the mean-square of the log power spectra given by

$$E[(\log A_k^2 - \log \tilde{A}_k^2)^2] \tag{14.3}$$

where $\tilde{A}_k^2$ denotes the estimator of A_k^2, and use

$$\hat{A}_k = \sqrt{\tilde{A}_k^2} \tag{14.4}$$

After rigorous mathematical calculations, it can be shown that

$$\hat{A}_k = \frac{\xi_k}{1+\xi_k} exp[\frac{1}{2}\int\limits_{v_k}^{\infty} \frac{e^{-t}}{t}] \cdot R_k \tag{14.5}$$

which is the MMSE LOG-STSA estimate for A_k.
where,

R_k denotes the Fourier transform of the noisy signal,

ξ_k is the priori SNR,

v_k is defined as

$$v_k = \frac{\xi_k}{1+\xi_k}\lambda_k \tag{14.6}$$

where,

$$\frac{1}{\lambda_k} = \frac{1}{\lambda_x(k)} + \frac{1}{\lambda_n(k)} \tag{14.7}$$

$\lambda_x(k)$ and $\lambda_n(k)$, denoting the variances of the k^{th} spectral component of the speech signal and noise.

The estimate values of $\lambda_x(k)$ and $\lambda_n(k)$ are found with the help of a reliable VAD. Since the VAD detects the voice-only segments, the rest denotes noise. The noise-only component of speech is got by subtracting the detected speech from the original speech. Hence the variance of noise could be found out. By using the variance of noise along with the correlated value between noise and speech, the variance of speech is estimated.

14.3 The denoising process

- The input speech samples are partitioned into 20ms frames.
- Each frame is windowed(Hamming) and 50 percent overlapping is carried out. It is then transformed into the spectral domain.
- The value of v_k is estimated using the MMSE-LOG estimator as described in the previous section for each frame.
- The estimate $\hat{A}_k$ is calculated using the *Equation 14.5.*
- The enhanced speech is constructed by taking the inverse transform of the estimated coefficients and using the principle of overlap-add on each window.

14.4 Performance Evaluation

The algorithm incorporating the method of enhancement using MMSE-LSAE was tested using the TIMIT speech file. The results are shown in *Figure 14.1.* The clean speech(in blue) refers to the TIMIT speech samples. The noisy samples(in red) are obtained when noise(AWGN) is added to the clean samples. The enhanced samples(in black) are shown at the bottom.

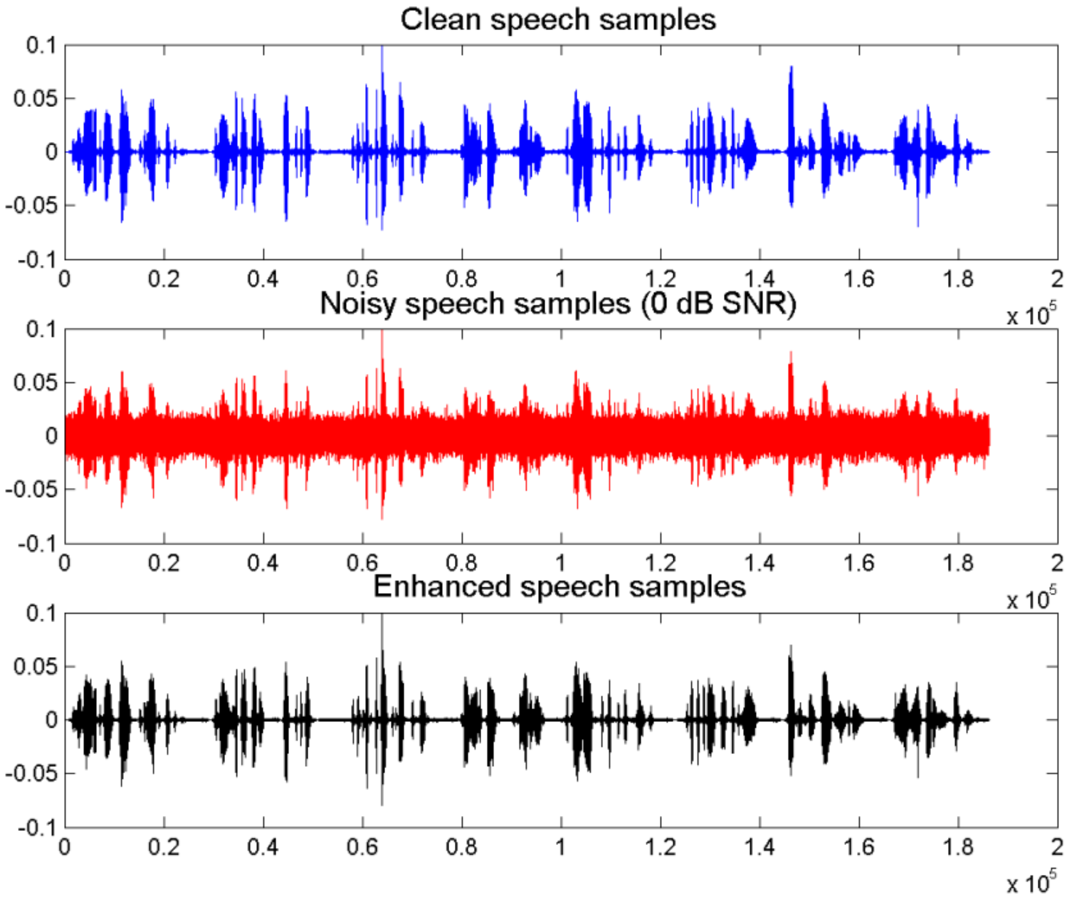

Fig.14.1: Experimental Results for the TIMIT speech.

Chapter 15

Speech Enhancement by Adaptive Wavelet Packet

15.1 Introduction

Wavelet transform has been used intensively as a powerful tool for noise reduction in several speech algorithms[7]. But each algorithm incorporates the method of noise estimation in a different way. This algorithm estimates noise based on *spectral entropy* using *histogram of intensity*. Also, to alleviate time-frequency discontinuities, it employs a modified hard-thresholding method, based on the μ-law logarithm for enhancing speech. This algorithm is implementable, not only in the presence of non-stationary noise, but also in situations involving colored noise.

15.2 Denoising by wavelet thresholding

The conventional wavelet based denoising algorithm can be summarized as follows. Let s denote clean speech with the finite length N and x the corrupted speech by white Gaussian noise n with variance σ^2. We have,

$$x = s + n \tag{15.1}$$

If W denotes a wavelet transform matrix, *Equation 15.1* in the wavelet domain can be given by,

$$X = S + N \tag{15.2}$$

where,
$X = Wx$, $S = Ws$ and $N = Wn$.
Let $\hat{S}$ be an estimated speech of S, based on the noisy observation X in the wavelet domain. The speech $\hat{S}$ can be estimated by

$$\hat{S} = THR(X, T) \tag{15.3}$$

where, $THR(;)$ denotes a thresholding function and T a threshold value.

Thresholding can be performed as a hard one or as a soft one, defined as follows:

$$\begin{aligned} THR_h(X,T) &= X, \ |X| > T, \\ & \quad 0, \ |X| \leq T. \end{aligned} \tag{15.4}$$

$$\begin{aligned} THR_s(X,T) &= sgn(X)(|X| - T), \ |X| > T, \\ & \quad 0, \ |X| \leq T. \end{aligned} \tag{15.5}$$

A universal threshold T for the discrete wavelet packet transform case is

$$T = \hat{\sigma}_k \sqrt{2 \log(N \log_2(N)} \tag{15.6}$$

where,
$\hat{\sigma}_k$ is the noise level for the k^{th} node and N, the sample size(Here, node refers to each sub-band, when the wavelet-tree decomposition of the noisy signal is carried out).

15.3 Noise estimation based on spectral entropy using histogram of intensity

Generally, median-absolute-deviation(MAD) based noise level estimation is adopted for wavelet thresholding. This method has an assumption that the noise has Gaussian distribution. However, this is not always valid in practice. So an estimation based on spectral entropy is employed, which is as follows[6].

1. Estimate spectral pdf through histogram of wavelet packet coefficients for each node. Histogram is composed of B bins.

2. Calculate the normalized spectral entropy.

$$Entropy(n) = -\sum_{b=1}^{B} P \cdot log_B(P) \tag{15.7}$$

   with,

   n = 1,2,...No. of best nodes

   $P = \frac{No.\ of\ Wavelet\ Packet\ Coefficients\ c_k\ in\ bin\ b\ and\ node\ k}{Node\ size\ in\ adapted\ wavelet\ packet\ tree}$

3. Estimate *spectral magnitude intensity by histogram* and standard deviation of noise for node dependent wavelet thresholding.

4. Define an auxiliary threshold α.

$$\alpha(n) \ = \ Entropy(n) \cdot (node\ size) \ \cdot \beta \tag{15.8}$$

   where the range of β is from 0.7 to 0.9. It is usually taken as 0.8.

5. Finally, estimate the standard deviation of noise for node-dependent wavelet thresholding.

$$\hat{\sigma}_k \ = \ [No.\ of\ bins\ in\ node\ k\ bigger\ than\ \alpha(n)] \cdot bin\ width \tag{15.9}$$

6. One of the major problems of hard/soft thresholding is time-frequency discontinuities. This leads to annoying artifacts and further degradation of output speech. To resolve this problem, μ-law algorithm is adopted as a non-linear

function.

$$\begin{aligned} THR_{\mu-law}(X,T) &= X,\ |X| > T, \\ & \quad T \cdot (\frac{1}{\mu}[(1+\mu^{|\frac{X}{T}|}) - 1] \cdot sqn(x)),\ |X| \leq T. \end{aligned} \quad (15.10)$$

μ is usually taken to be 256. T is as defined by *Equation 15.6.*

15.4 The denoising process

- The input speech samples are partitioned into 20ms frames.
- Each frame is decomposed into a tree structure(with 3 nodes) and the wavelet packet coefficients for each node of the tree are calculated.
- Noise estimation is carried out as discussed in *Section 15.3.* The threshold is calculated for each node by using *Equation 15.6.*
- The enhanced speech samples are obtained by using the μ-law thresholding as given by *Equation 15.10.*

15.5 Performance Evaluation

The algorithm incorporating the method of enhancement using adaptive wavelet packet was tested using the TIMIT speech file. The results are shown in *Figure 15.1*. The clean speech(in blue) refers to the TIMIT speech samples. The noisy samples(in red) are obtained when noise(AWGN) is added to the clean samples. The enhanced samples(in black) are shown at the bottom.

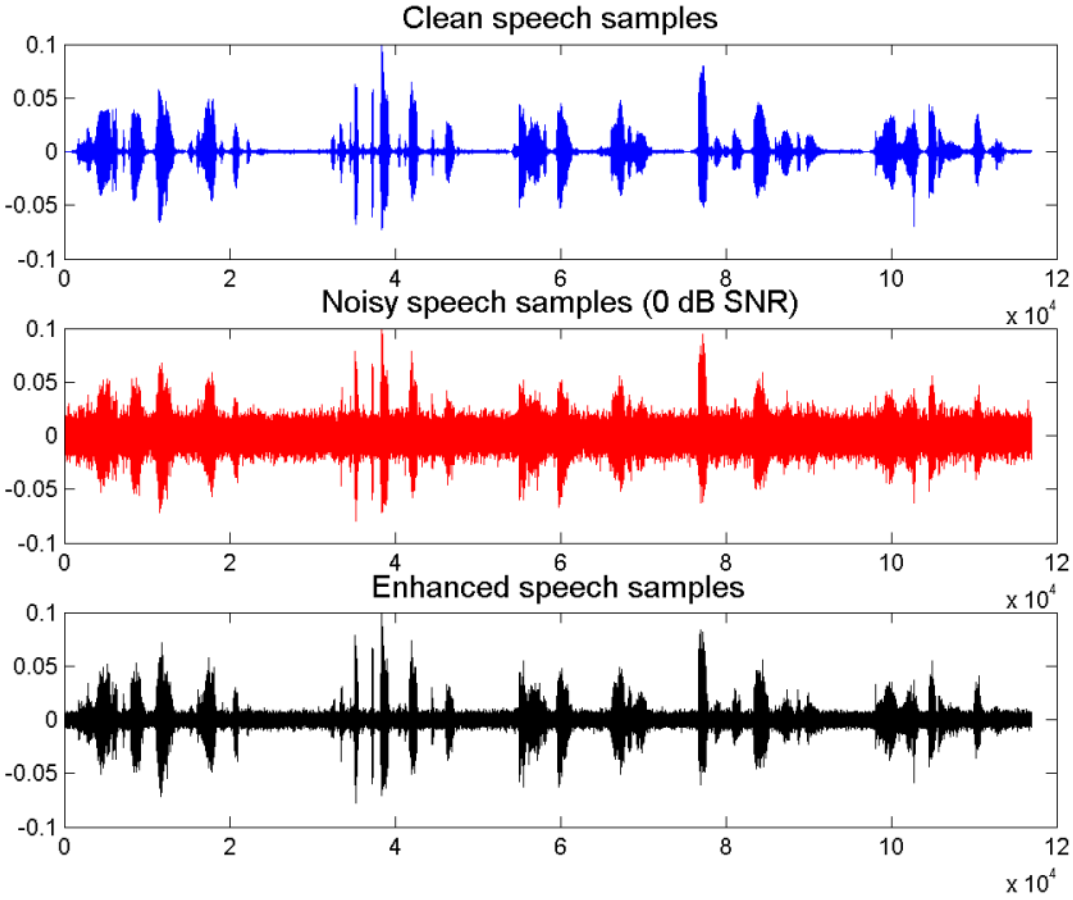

Fig.15.1: Experimental Results for the TIMIT speech.

References and Bibliography.

1. S. Gokhun Tanyer and Hamza Ozer, *"VAD in Non-stationary Noise."*, IEEE Transactions on Speech and Audio Processing, July 2002, Vol.8, No. 4, pp. 478-482.

2. A. Sangwan, Chiranth M. C, R. Shah, V. Gaurav, R. V. Prasad and H. S. Jamadagni, *"Comparison of VAD Algorithms for VoIP."*, Proceedings of the Seventh International Symposium on Communications and Computers, ISCC-2002, pp. 530-535.

3. R. Tucker, *"VAD using a Periodicity Measure."*, IEE Proceedings-I,Vol. 139, No. 4, Aug. 1992, pp. 377-380.

4. J. A. Haigh and J. S. Mason, *"Robust VAD using Cepstral Features."*, IEEE TEN-CON, 1993, pp. 321-324.

5. Y. Ephraim and D. Malah, *"Speech Enhancement Using a Minimum Mean-Square Error Log-Spectral Amplitude Estimator."*, IEEE Trans. on Acoustics, Speech and Signal Processing(ASSP), Vol. 33, No. 2, Apr. 1985, pp. 443-445.

6. S. Chang, Y. Kwon, S. Yang and I. Kim, *"Speech Enhancement for Non-Stationary Environment By Adaptive Wavelet Packet."*, Proceedings of the IEEE Conference on ASSP(ICASSP), 2002, Vol. 1, pp. 561-564.

7. I. M. Johnstone and B. W. Silverman, *"Wavelet Threshold Estimators for Data with Correlated Noise."*, J.Roy. Statistics, Soc. B, 1997, Vol. 59, pp. 319-351.

8. A. M. Kondoz, *"Digital Speech."*, WILEY Publications, Ch. 10, pp. 337-341.

9. TIMIT Acoustic Phonetic Continuous Speech Corpus, http://www.timit.com.

10. Australian English Telephone Digit Database, http://www.callbase.com.