INVESTIGATION OF THE SIGNIFICANCE OF PERIODICITY INFORMATION IN SPEAKER IDENTIFICATION

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

SEÇİL GÜRSOY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

APRIL 2008

Approval of the thesis:

USING PERIODICITY & APERIODICITY INFORMATION FOR SPEAKER VERIFICATION

submitted by **SEÇİL GÜRSOY** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. İsmet Erkmen _____
Head of Department, **Electrical and Electronics Engineering**

Assoc. Prof. Dr. Tolga Çiloğlu _____
Supervisor, **Electrical and Electronics Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Mübeccel Demirekler _____
Electrical and Electronics Engineering Dept., METU

Assoc. Prof. Dr. Tolga Çiloğlu _____
Electrical and Electronics Engineering Dept., METU

Assist. Prof. Dr. Çağatay Candan _____
Electrical and Electronics Engineering Dept., METU

Assist. Prof. Dr. Afşar Saranlı _____
Electrical and Electronics Engineering Dept., METU

Levent Alkışlar (M. Sc.) _____
ASELSAN

Date: _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Seçil Gürsoy

Signature :

ABSTRACT

INVESTIGATION OF THE SIGNIFICANCE OF PERIODICITY INFORMATION IN SPEAKER IDENTIFICATION

Gürsoy, Seçil
M.Sc., Department of Electrical and Electronics Engineering
Supervisor: Assoc. Prof. Dr. Tolga Çiloğlu

April 2008, 83 pages

In this thesis; general feature selection methods and especially the use of periodicity and aperiodicity information in speaker verification task is searched. A software system is constructed to obtain periodicity and aperiodicity information from speech. Periodicity and aperiodicity information is obtained by using a 16 channel filterbank and analyzing channel outputs frame by frame according to the pitch of that frame. Pitch value of a frame is also found by using periodicity algorithms. Parzen window (kernel density estimation) is used to represent each person's selected phoneme. Constructed method is tested for different phonemes in order to find out its usability in different phonemes. Periodicity features are also used with MFCC features to find out their contribution to speaker identification problem.

Keywords: Pitch detection, Periodicity, Aperiodicity, Speech Processing, Average Magnitude Difference Function (AMDF), Speaker Identification

ÖZ

KONUŞMACI TANIMLAMADA PERİYODİKLİK BİLGİSİNİN ÖNEMİNİN ARAŞTIRILMASI

GÜRSOY, Seçil
Yüksek Lisans Tezi, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Doç.Dr. Tolga ÇİLOĞLU

Nisan 2008, 83 sayfa

Bu tez çalışmasında konuşmacı doğrulama amacıyla kullanılan öznitelikler, özellikle de periyodiklik ve aperiyodiklik bilgisinin kullanımı araştırılmıştır. Ses sinyalinden periyodiklik ve aperiyodiklik bilgisinin elde edilmesi amacıyla bir yazılım sistemi oluşturulmuştur. Periyodiklik ve aperiyodiklik bilgisi 16 kanallı bir filtre kullanılarak, kanal çıktılarının ilgili çerçeve içerisinde o çerçeveye ait pitch bilgisine göre analiz edilmesiyle elde edilmiştir. Her bir çerçevenin pitch bilgisi yine periyodiklik algoritmaları ile hesaplanmıştır. Seçilen kişilerin analiz edilen harfleri, Parzen Window tekniği kullanılarak modellenmiştir. Oluşturulan metod, farklı sesler için kullanılabilirliğinin anlaşılması amacıyla farklı seslerde test edilmiştir. Periyodiklik bilgisi ayrıca MFCC bilgisi ile birlikte kullanılarak, konuşmacı tanıma problemine yaptığı katkı araştırılmıştır.

Anahtar kelimeler: Pitch Bulma, Periodiklik, Aperiodiklik, Ses İşleme, AMDF, Konuşmacı Tanıma

To my family, to my grandmother Mükerrem and to littlestar

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my supervisor Assoc. Prof. Dr. Tolga Çiloğlu for his guidance, advice, criticism, encouragement and insight throughout the research. During this study, I have really learnt a lot from him, both in terms of research and life.

Deepest thanks to my family for their love, trust, understanding, and every kind of support not only throughout my thesis but also throughout my life.

I would like to thank my company ASELSAN A.Ş. and my colleagues for the understanding and support on every phase of this work.

I would also like to thank Eren Akdemir, Turgay Koç and Alp Ertürk for their valuable comments, suggestions, and support.

I would like to express my gratitude to Oktay Sipahigil for his great support and sharing knowledge.

And last, I would like to thank Çağrı for his support and patience during this thesis.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

AMDF	:	Average Magnitude Difference Function
AP	:	Acoustic Parameters
CLASS	:	Classification
CM	:	Common Mode
CV	:	Coefficient of Variation
dB	:	Decibel
DCF	:	Detection Cost Function
EER	:	Equal Error Rate
GEV	:	Generalized Extreme Value
GMM	:	Gaussian Mixture Model
HOCOR	:	Haar Octave Coefficients of Residue
KL	:	Kullback-Leibler
LPCC	:	Linear Predictive Cepstral Coefficients
MFCC	:	Mel Frequency Cepstral Coefficients
ML	:	Maximum Likelihood
PDF	:	Probability Density Function
ZCR	:	Zero Crossing Rate

CHAPTER 1

INTRODUCTION

1.1 Speech Production

In humans, pushing out air from the lungs through vocal chords and mouth produces speech. Lungs act as a source of producing sound and vocal tract acts as a filter. Articulators are soft palate, tongue, lips and jaw.

From the technical point of view, the production of speech is widely described as a two-level process. In the first stage, the sound is initiated and in the second stage it is filtered. The basic assumption of the model is that the source signal is produced at the glottal level and it is linearly filtered through the vocal tract [1] (See Figure 1-1).

1.2 Speaker Recognition

Overview:

The speech signal contains many levels of information. First of all, it conveys a message via words to the listener. On the other hand, the speech conveys information about the gender, language being spoken, emotion and generally, the identity of the speaker. One branch of the speech processing is speaker recognition.

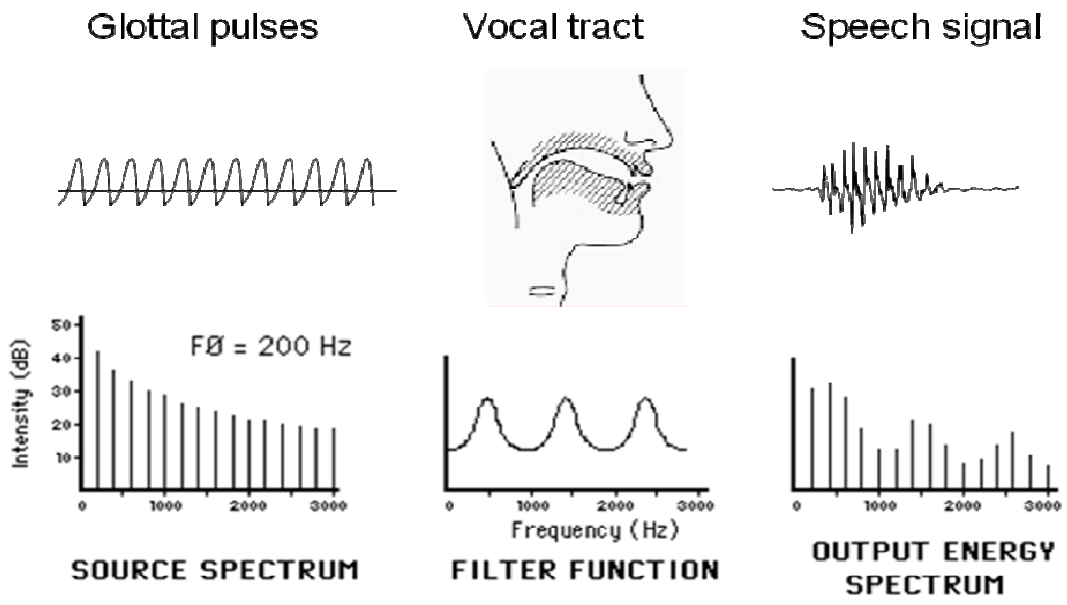

Figure 1-1: The source filter model of speech [1]

Speaker recognition encompasses two main tasks: Speaker Recognition and Speaker Verification. Speaker verification is the task of determining whether a person is who he/she claims to be. The literature abounds with different terms for speaker verification, including voice verification, speaker authentication, voice authentication, talker authentication, and talker verification. Speaker identification is the task of determining who is talking from a set of known voices or speakers. In speaker identification there is no a priori identity claim, and the system decides who the person is, what group the person is a member of, or (in the open-set case) that the person is unknown. Generally it is assumed that the unknown voice comes from a fixed set of known speakers, thus the task is often referred as closed-set identification.

A speaker known to a speaker recognition system who is correctly claiming his/her identity is labeled as a **claimant** and a speaker unknown to the system who is posing as a known speaker is labeled an **impostor**.

There are two types of errors in speaker recognition systems: false acceptances, where an imposter is accepted as a claimant, and false rejections, where claimants are rejected as impostors.

Speaker recognition is of two types:

Text-dependent: Text-dependent systems expect the speaker to say a pre-determined phrase, password, or ID. By controlling the words that are spoken, the system can look for a close match with the stored voiceprint. Text-dependent recognition is employed in applications with strong control over user input. This type of recognition has an advantage of increasing the performance of the system because of the prior knowledge of the spoken text.

Text-independent: This type of mechanism is used for recognizing any type of conversational speech or user selected phrase. Text-independent recognition system has no prior knowledge of the text spoken by the person. This is generally used in applications with less control over user input.

General overviews of speaker recognition have been given in [2], [3], [4] and [5].

Speaker recognition applications:

There are many applications to speaker recognition. In [6], these areas are grouped into three categories. These are authentication, surveillance and forensic speaker recognition.

Speaker Recognition for Authentication:

Speaker recognition for authentication allows the users to identify themselves using their voices. This can be much more convenient than carrying a key with you or remembering a PIN. There are a few distinct concepts of using the human voice for authentication, i.e. there are different kinds of speaker recognition systems for authentication purposes:

Single pass phrase system: A single pass phrase system lets the user choose a phrase that is uttered in enrollment as well as for authentication. Therefore, text dependent speaker recognition techniques can be used, which has the advantage that good recognition accuracy can be achieved with very little speech data in training as well as test.

Text prompt system: A text prompt system requires the user to utter a specific text which is generated individually for each authentication. As an example, a series of digits from "zero" to "nine" may be used. But also the generation of arbitrary phrases which are to be spoken by the person to be authenticated is conceivable. Depending on the kind of prompt, the speaker recognition technique may be text dependent as well as text independent. The disadvantage of this system is that longer speech signals have to be collected during training as well as for the authentication process, making the system convenient.

Speaker Recognition for Surveillance:

Another usage of speaker recognition is to recognize speakers by telephone or radio conversations. Security agencies use this information to recognize target speakers that are of interest for the service. The difficulty in this usage is that, there are high quantities of data and filtering mechanisms must be applied in order to find the relevant information.

Forensic Speaker Recognition:

Determining whether a given speech utterance has been produced by a particular person can help to convict a criminal or discharge an innocent in court. Semiautomatic and automatic speaker recognition technologies are present in forensic speaker recognition area and detailed information is given in [7].

Further information about speaker recognition applications can be found in [3], [6] and [7].

1.3 Overview of the Speaker Verification Process

A speaker verification system is composed of two distinct phases, a training phase and a test phase. Figure 1-2 shows a modular representation of the training phase of a speaker verification system.

The first step consists of extracting parameters from the speech signal to obtain a representation suitable for statistical modeling. This step can be named as feature extraction which maps each interval of speech to a multidimensional feature space. The second step consists of obtaining a statistical model for the speakers from the extracted parameters.

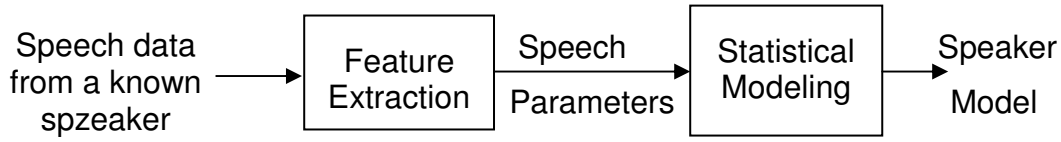

Figure 1-2: Training phase of a speaker verification system

Figure 1-3 shows a modular representation for the test phase of a speaker verification system. The entries of the system are the claimed identity and the speech samples pronounced by an unknown speaker. First, features are extracted from the speech signal using exactly the same feature extraction module as for the training phase. Then, the speaker model corresponding to the claimed identity and the training models are being compared to get a match score. The match score measures the similarity of the input speaker model to the model of the claimed speaker. Last, a decision is made to either accept or reject the claimant according to the match score.

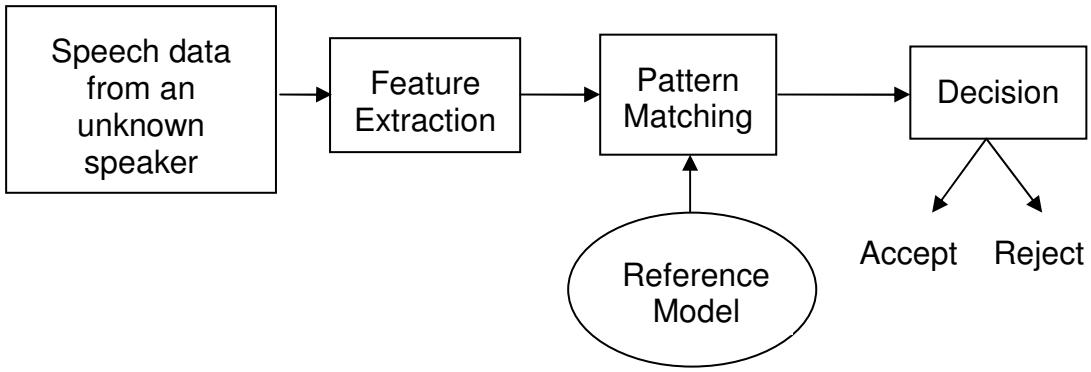

Figure 1-3: Test phase of a speaker verification system

In this thesis, we are especially interested in feature extraction step. Our aim is to study which kind of features could be used for speaker verification and to suggest new features which could be used lonely or supplementary with existing features in order to improve speaker verification task.

Feature extraction is a common step for training and testing phases of the speaker verification systems. Feature extraction converts the speech signal

into a sequence of feature vectors. The goal is to find a transformation to a relatively low-dimensional feature space that preserves useful information for speaker verification. Although it might be tempting at first to select all the extracted features to improve quality but the “curse of dimensionality” quickly becomes overwhelming. As more features are used, the feature dimensions increase, which imposes severe requirements on computation and storage in both training and testing. Also the demand for a large amount of training data to represent a speaker’s voice characteristics grows exponentially with the dimension of the feature space [2].

The features extracted for the verification process must [2], [9]

- possess high discriminative power: higher interspeaker (between-speaker) variability, low intraspeaker (within-speaker variation, due to emotion, health, state, and age) variability,
- occur frequently and naturally in speech
- be robust against noises and distortions
- be stable over time
- not be susceptible to mimicry by impostors

It is unlikely that a single feature would fulfill all the listed requirements above. Fortunately, due to the complexity of speech signals, a large number of complementary features can be extracted and combined to improve the verification accuracy. Basically we can group speech features as two types: source based features and filter (vocal tract) based features.

It is known that the speech spectrum shape encodes information about the speaker’s vocal tract shape via resonances (formants) and glottal source via pitch harmonics [3].

State-of-the-art automatic speaker recognition systems typically extract features carrying vocal tract characteristics, such as Mel Frequency Cepstral

Coefficients (MFCC) and Linear Predictive Cepstral Coefficients (LPCC). Recently, some experimental results have shown that features involving vocal cord characteristics, such as pitch, harmonics, etc., can work as supplementary features to those vocal tract ones and can improve speaker recognition performance [10], [11], [12]. Some examples of speaker recognition features and their performances are listed in Table 1-1 to give an idea. The details of the methods could be found in their references. For the listed papers in table; first column specifies the used feature(s); second column specifies the classification technique; third column tells whether text-dependent or text-independent recognition is used; fourth and fifth columns give information about the used databases and speech for training and testing respectively; and last column summarizes the performances.

Table 1-1 Examples for Speaker Recognition Algorithms

FEATURE	CLASS.	TEXT	TRAINING DATA	TEST DATA	PERFORMANCE
MFCC [15]	GMM	Ind.	• 16 Male (KING Database) • 60 sec (6000 25 dim mel cepstral vectors)	• 5 sec	Correct% = 87.3 (Model order = 16)
Cepstrum + Prosodic Variation [16]	GMM		• 250 Male, 250 Female (NIST Database) • 2 min.	• 10 sec • 5000 trials	Male: DCF[1] ($x10^3$)= 65.5 ceps DCF ($x10^3$)= 60.6 ceps +prosody Female: DCF ($x10^3$)= 62 ceps DCF ($x10^3$)= 59.6 ceps +prosody
MFCC + MPEG7 [17]	GMM	Ind.	• 150 person, NIST99		EER%: MFCC=8.58, MPEG=9.00 MFCC+SpHr=7.78 (feature

[1] Detection Cost Function (DCF): DCF = C_{fr} P(true)P(fr\true) +C_{fa}P(imposter)P(fa\imposter)
(i.e., Bayes Risk with priors P(true) = 0.01= 1-P(imposter), and false rejection and false alarm costs C_{fr} = 10; C_{fa} =1).

Table 1-1 Examples for Speaker Recognition Algorithms

FEATURE	CLASS.	TEXT	TRAINING DATA	TEST DATA	PERFORMANCE
					comb) MFCC+Hr=7.78 (feature comb) MFCC+ MPEG+SpHr=6.99 MFCC+MPEG=6.99 MFCC+SpHr=7.19
MFCC + Pitch [12]	LVQ-SLP, GMM	Ind.	• 18 Female (SPIDRE) • 3 conversation	1 conversation (from different handset)	Identification rate increase: Voiced (6%), with pitch (14%)
Acoustic Parameters (AP) [18]	GMM	Ind.	• 50 to 250 all male or all female (NIST 1998) • 1 min. after silence removal 30-40 sec	30 sec. after silence removal 10-20 sec	Identification Error Rate % for population size 100, avarage: 29.44 for 8 APs, 31.11 for 26 MFCCs, 30.75 for 39 MFCCs
LPCC HOCOR	GMM		• Male subset of YOHO database	2.5 sec	EER: LPCC(24 dim.)=1.04,

Table 1-1 Examples for Speaker Recognition Algorithms

FEATURE	CLASS.	TEXT	TRAINING DATA	TEST DATA	PERFORMANCE
LPCC+HOCOR [10]					HOCOR(24 dim.)=8.74 LPCC+HOCOR(48 dim.)=0.99 (future comb.) LPCC+HOCOR(48 dim.)=0.89 (score comb.)

1.4 Voiced - Unvoiced Speech

Depending on the type of excitation, two types of sounds are produced: voiced and unvoiced sounds. **Voiced sounds** are produced by forcing air through the glottis or an opening between the vocal folds. The vocal folds vibrate in this case. Examples of voiced sound are the vowel “e” in “ses”, or “a” in ‘”can”’ (see Figure 1-4). **Unvoiced sounds** are generated by forming a constriction at some point along the vocal tract and forcing air through the constriction to produce turbulence. Vocal folds do not vibrate in this case. An example of an unvoiced sound is “s” as in “ses” (see Figure 1-5). A sound can also be simultaneously voiced and unvoiced (mixed). An example of a mixed sound is “z” in “zil”.

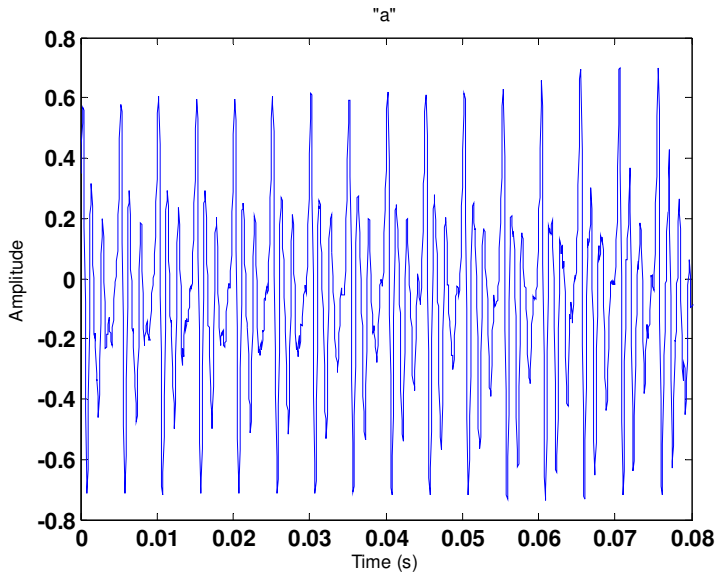

Figure 1-4: Voiced Speech

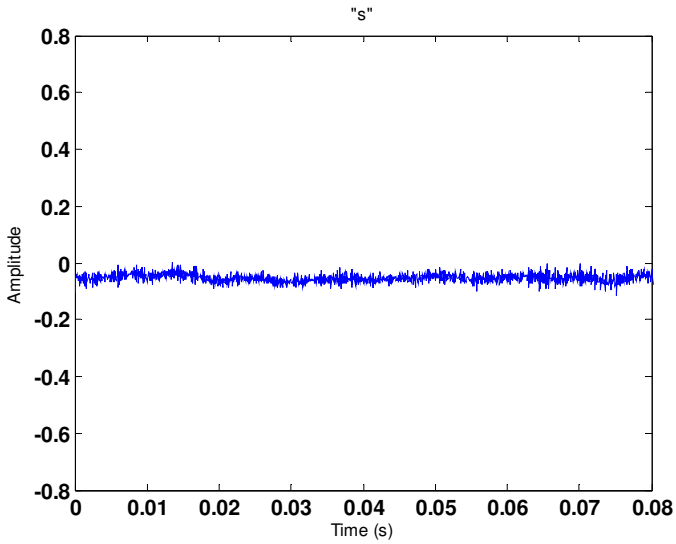

Figure 1-5: Unvoiced Speech

When producing voiced sounds, vocal folds open and close in a periodic pattern. For that reason, voiced sounds are quasi-periodic and the frequency at which vocal folds open and close is called **the fundamental frequency** or **pitch**. Table 1-2 gives the pitch characteristics of men women and children.

Table 1-2 General pitch characteristics

Pitch (Hz)	**average**	**Min.**	**Max.**
Men	125	80	200
Women	225	150	350
Children	300	200	500

1.5 The Aim of the Thesis

In this thesis, use of the periodicity and aperiodicity information of speech in speaker verification task is investigated.

In the production of speech, there are a number of sources that generate acoustic energy in the vocal tract. Periodic sounds are produced by quasi-periodic lateral movements of the vocal folds which are creating periodic energy at the glottis. Aperiodic sounds are mainly produced by creating turbulence in the flow of air through the vocal tract. Aperiodic sources include aspiration, generated at the glottis; frication, generated further forward in the vocal tract; and transient bursts produced by the rapid release of complete constrictions. All these sources are filtered by the vocal tract to generate an output signal, which will also be periodic or aperiodic depending on the source(s) [13].

We could not find any implementation that uses periodicity and aperiodicity directly as a feature for speaker verification.

Periodicity and aperiodicity information is obtained from 16 different frequency bands of speech and the obtained information is tested on selected phonemes for different persons. We compared the results for different phonemes in order to find out which phonemes are being more separated with periodicity and aperiodicity information. This is because to see if periodicity and aperiodicity information could be used lonely or supplementary with other features and which phonemes are appropriate to use these features in order to perform the speaker verification task. The constructed method is explained in chapter 2.

CHAPTER 2

THE METHOD

The constructed system to find periodicity and aperiodicity information starts with silence detection and followed by some signal processing algorithms, as detailed in Figure 2-1. The system also gives an estimate of the pitch period of the periodic component. The details of the methods used in the system are explained in the following sections.

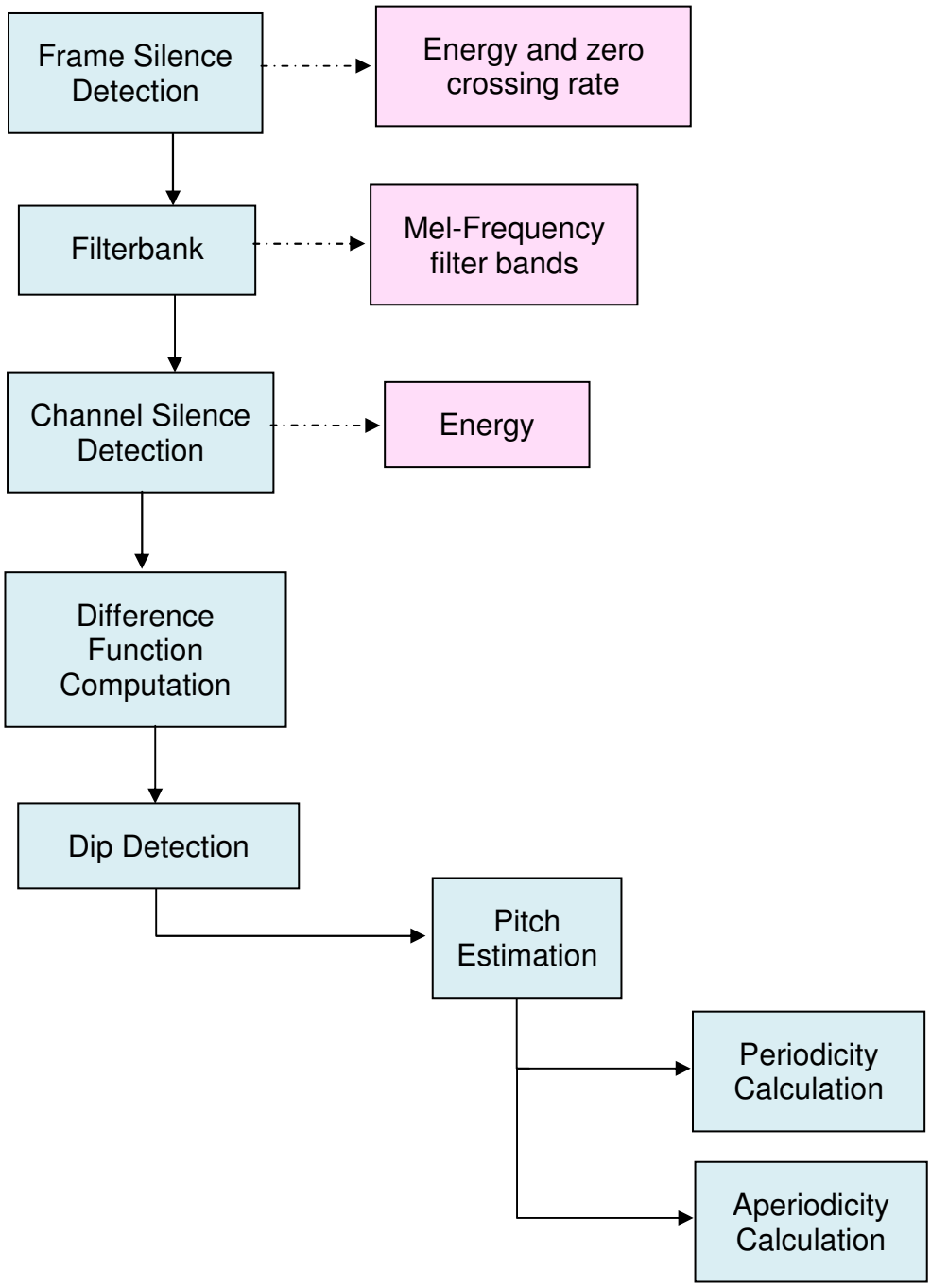

Figure 2-1: Block diagram of our system

2.1 Frame Silence Detection

The analysis begins with segmenting the speech into frames. We extracted frames of length 20 ms which overlap by 15 ms as shown in Figure 2-2.

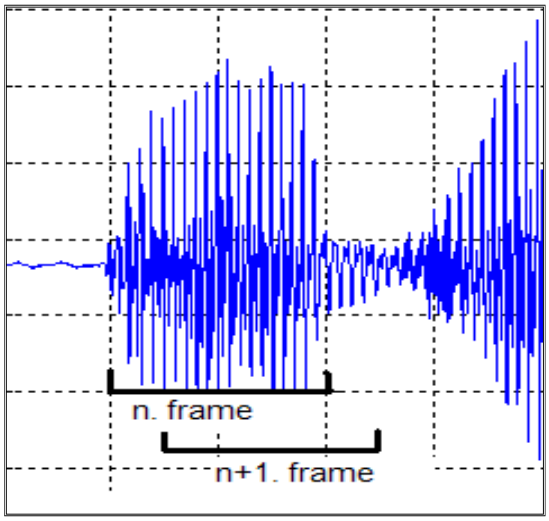

Figure 2-2 Frame blocks

After obtaining frames, silence detection begins with comparing frame energies for voiced - unvoiced decision. Energy of the n-th frame of the speech signal is calculated by the following equation:

$$E = 20log\left(\sum_{j=0}^{N-1} {x_n}^2\,(j)\right) \qquad (2.1)$$

where $x_n(j)$ is the j-th speech sample in the n-th frame and N is equal to 320 because of 20 ms frame length with 16kHz sampling rate. Usually the energy of a voiced speech frame is larger than that of an unvoiced speech frame.

Next step of the frame silence detection is to examine zero crossing rates. The zero-crossing rate is obtained by counting the sign changes (either from positive to negative or from negative to positive) in successive speech samples. The ZCR of the voiced sound is lower than the ZCR of the unvoiced sound.

A frame is judged to be nonsilent if its total energy is no more than 35 dB (threshold) below the maximum total energy computed across all of the frames in the utterance or if ZCR of that frame is smaller than 50 samples.

That is, a frame is judged to be silent if:

$$E_f < E_{max} - 35\,dB \quad or \quad ZCR_f > 50$$

E_f : energy of the frame

E_{max} : maximum energy across all frames

ZCR_f : zero crossing rate of the frame

The energy threshold (35 dB) is referenced from [13] and its suitability for our study is proven empirically. If a frame is classified as silent, then no further processing is done. The ZCR threshold is determined empirically also to improve our pitch estimation algorithm which will be discussed in section 2.4.

2.2 Filterbank

A 16 channel filter bank is applied to the speech signal. In order to provide an accurate weighting of the frequency components, channel's start and end frequencies are determined by a 16 channel Mel-scaled filter bank.

The mel scale is a perceptual scale of pitches judged by listeners to be equal in distance from one another. Human ear tends to perceive the frequencies below 1000 Hz in a linear way and frequencies above 1000 Hz in a non-linear manner. The reference point between this scale and normal frequency measurement is defined by equating a 1000 Hz tone, 40 dB above the listener's threshold, with a pitch of 1000 mels. Above about 500 Hz, larger and larger intervals are judged by listeners to produce equal pitch increments. The name mel comes from the word melody to indicate that the scale is based on pitch comparisons [14]. To convert *f* Hertz into *m* Mel use:

$$m = 1127.01048\ log_e\left(1 + \frac{f}{700}\right) \tag{2.2}$$

Or equivalently:

$$m = 2595\ log\left(1 + \frac{f}{700}\right) \tag{2.3}$$

Mel frequency filter bank is modeled by constructing the required number of triangular band-pass filters with 50% overlap. (see Figure 2-3)

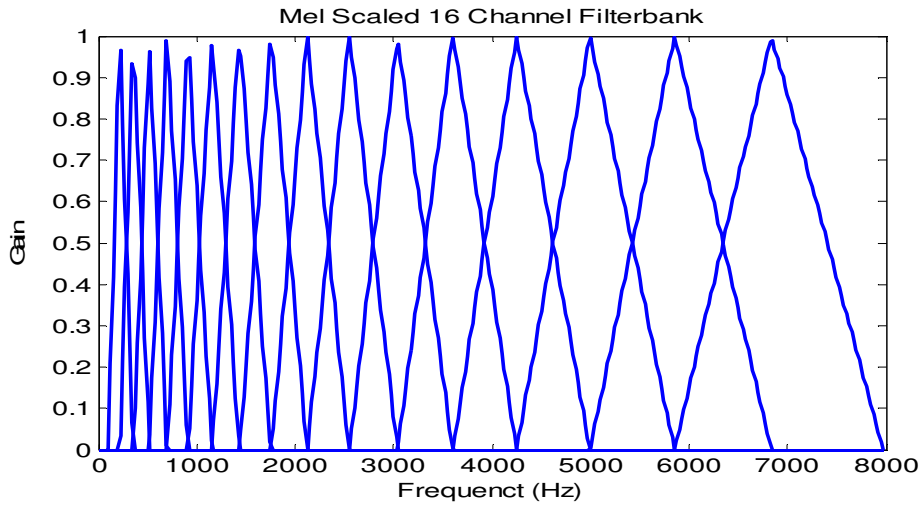

Figure 2-3: Mel scaled filterbank

In our filterbank, we used Mel scaled filterbank's start and end frequencies which are listed in Table 2-1. But instead of triangular filters, we used rectangular FIR filters with different orders for different channels. Channel orders are determined by inspection of the filter characteristics. Channel orders are also listed in Table 2-1.

Table 2-1 16 channels filter specifications

Channel.	F Start (Hz)	F End (Hz)	Filter Order
1.	100	358.92	350
2.	220.4	518.29	350
3.	358.92	701.64	300
4.	518.29	912.58	300
5.	701.64	1155.3	300
6.	912.58	1434.5	300
7.	1155.3	1755.7	300

Table 2-1 (continued)

8.	1434.4	2125.3	300
9.	1755.7	2550.5	250
10.	2125.3	3039.7	250
11.	2550.5	3602.5	250
12.	3039.7	4250.1	200
13.	3602.5	4995.1	200
14.	4250.1	5852.2	150
15.	4995.1	6838.3	150
16.	5852.2	7972.8	100

First, fifth and sixteenth channel responses are given below as examples.

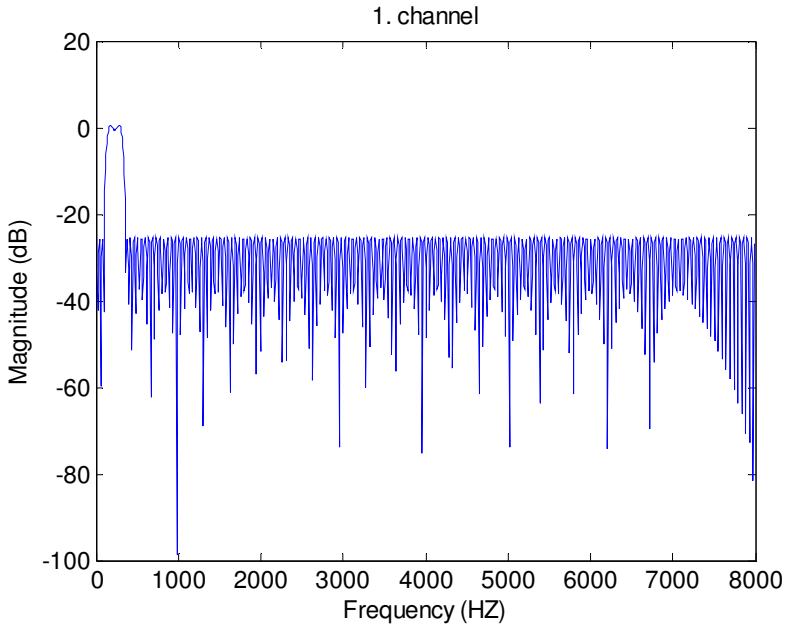

Figure 2-4: First channel

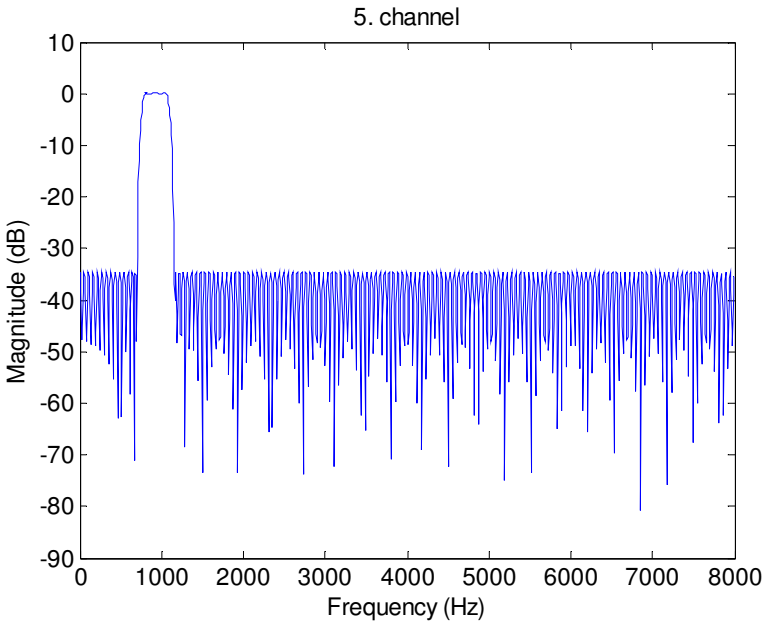

Figure 2-5: Fifth channel

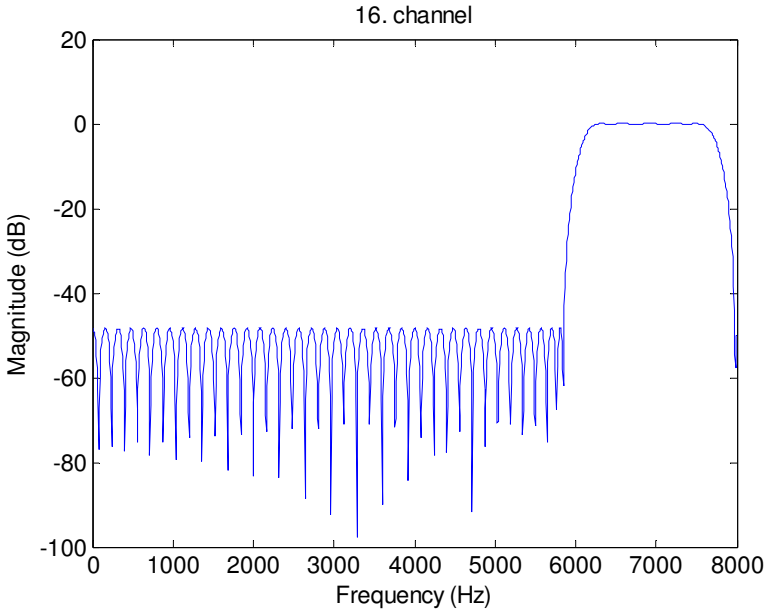

Figure 2-6: Sixteenth channel

After applying the 16 channel filter bank to our whole speech, each channel of each frame is analyzed according to its energy. For any nonsilent frame, a channel within that frame is judged to be nonsilent in case its energy is no more than 45 dB (threshold) below the maximum channel energy that has been computed up to present frame. If a channel of a frame is classified as silent, then no further processing is done with interested channel of that frame.

2.3 Difference Function Computation

Each nonsilent channel is analyzed for periodicity, aperiodicity and pitch. The raw pitch estimate of each frame is produced using Average Magnitude Difference Function (AMDF). There are several pitch finding algorithms especially autocorrelation function and difference function. The autocorrelation function consists of multiplication followed by addition which causes high computation cost. Detailed analysis of pitch estimation algorithms could be found in [19]. The average magnitude difference function is defined as:

$$AMDF(k) = \frac{1}{N-k} \sum_{n=0}^{N-1-k} |x(n) - x(n-k)|, \;\; k = 0, 1, \ldots N-1 \tag{2.4}$$

The difference function is expected to have a strong local minimum if the lag k is equal to or very close to the fundamental frequency. For each frame, the lag for which the AMDF has a global minimum is a strong candidate for the pitch period of that frame. By using only this situation we tested AMDF pitch estimation algorithm by using PRAAT software [20]. In Figure 2-7, the speech signal (sketched at the upper side), its pitch values

estimations by PRAAT (red dotted at the lower side) and AMDF function (blue dotted at the lower side) are shown.

Instead of calculating pitch estimations directly from speech signal's AMDF, we use periodicity and aperiodicity information of channels as explained in section 2.4. If a signal is periodic, then its AMDF function attains local minima at lags roughly equivalent to the pitch period and its integer multiples. Figure 2-8 shows a normalized AMDF function of a periodic channel.

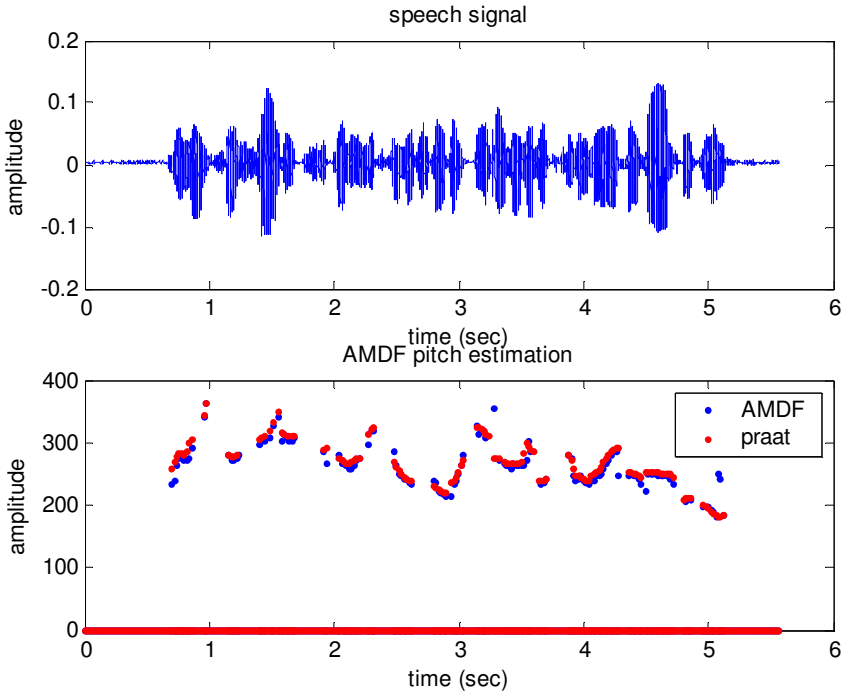

Figure 2-7: AMDF and PRAAT pitch estimations

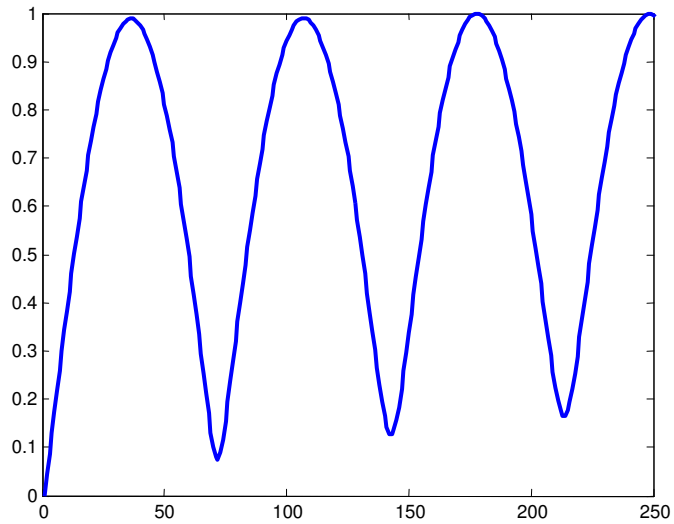

Figure 2-8: AMDF for a periodic channel

If a signal is aperiodic, then the AMDF waveform will not show such evenly spaced minimum points. Aperiodic channel's AMDF waveform has randomly distributed minimum points and its minimum points are higher than the periodic channel's minimum points. Figure 2-9 shows a normalized AMDF function of an aperiodic channel.

Hereafter the strength of the local minimum points will be referred as "dips" [13]. Dip strength values are found by subtracting the value of the minimum points from the maximum value of the AMDF. Because of the AMDF is normalized to 1, the strength of a dip can be 1 at the most. The decision regarding periodicity and aperiodicity is based on the location and strength of the dips occurring in the AMDF waveform. Dips of the channels in Figure 2-8 and Figure 2-9 are shown below respectively in Figure 2-10 and 2-11.

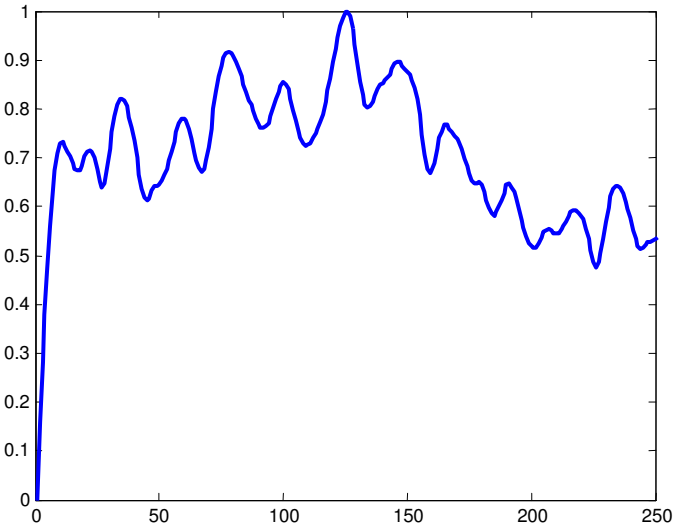

Figure 2-9: AMDF for an aperiodic channel

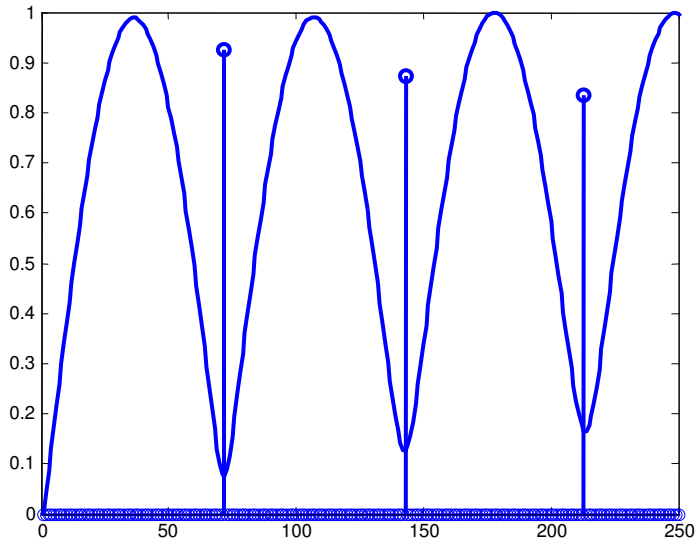

Figure 2-10: AMDF and dips for a periodic channel

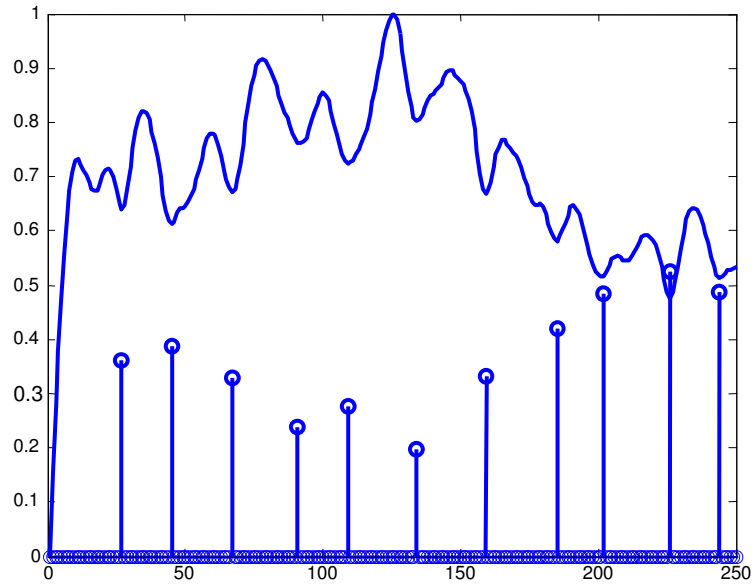

Figure 2-11: AMDF and dips for an aperiodic channel

Spacing and strength of the dips are indicators of periodicity. Note that periodic channel dips are evenly spaced and they have bigger strengths than aperiodic channel dips.

2.4 Estimation of Pitch Period

Pitch period estimation of a frame is based on the distribution of the AMDF dips across all channels within that frame. AMDF dip strengths are summed across all the channels within a frame. For a strong periodic frame, the summation of AMDF dips will constitute clusters at the pitch value and its integer multiples. For a strong aperiodic frame, the summation of AMDF dips will scatter randomly. Summations of channel dips for a periodic frame and for an aperiodic frame are shown in Figure 2-12 and 2-13 respectively.

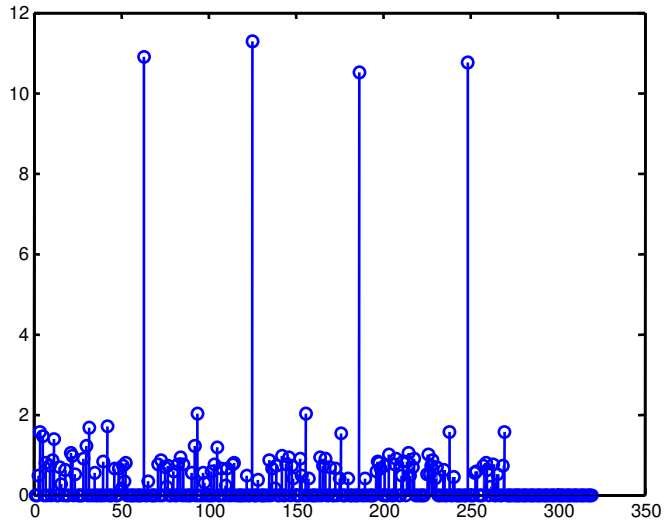

Figure 2-12: Summation of channel dips for a periodic frame

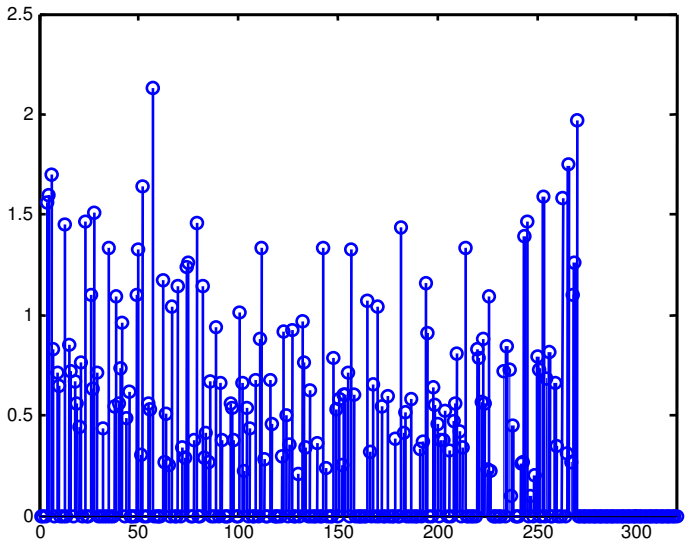

Figure 2-13: Summation of channel dips for an aperiodic frame

We estimate pitch frequency in two steps. First from the beginning of a frame, the maximum location between 45th to 93rd samples is chosen as the peak location (p) of that frame. These values correspond to 172 Hz to 356 Hz and they have been chosen according to the women voice's fundamental frequency in our database (Also see Table 1-2). With that pitch value, periodicity is calculated as explained in section 2.5. This case works for strong periodic frames. However, especially for weak periodic frames, the maximum point within the $2 \times p - 10 \; to \; 2 \times p + 10$ interval could be more determinative. As a second step we find the maximum point within $2 \times p - 10 \; to \; 2 \times p + 10$ interval, halving it and assume that the founded value is the new pitch candidate. $\left(p_{new} = \frac{\max(2 \times p - 10 : 2 \times p + 10)}{2} \right)$ We calculate periodicity value again and compare with periodicity found by p. The pitch value of the frame is chosen as the one which supports higher periodicity.

For the frame in Figure 2-12, the location corresponding to the maximum of the first cluster is 63. So for Figure 2-12, the pitch estimate is:

$$\frac{Sampling\ Frequency}{location\ of\ pitch\ estimate} = \frac{16000}{63} = 254\ Hz$$

Frames belonging to strong periodic regions have very high summation values in clusters. Frames belonging to weak periodic regions have low summation values in clusters. The reason of that is, the dips of weak periodic frames are not very strong near pitch estimates or these frames have silent channels.

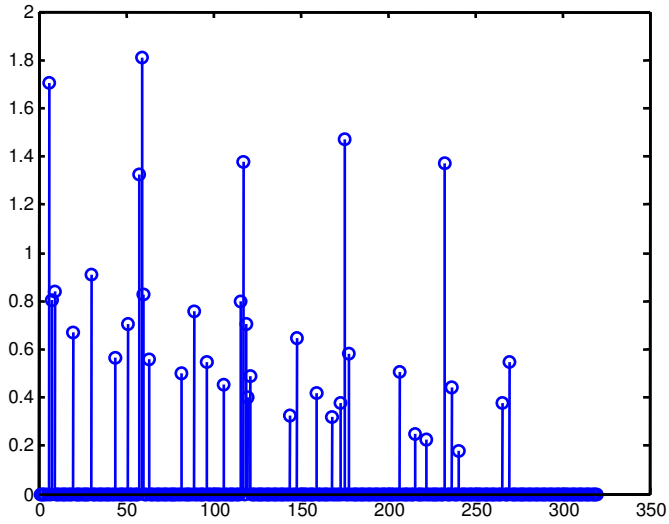

Figure 2-14: Summation of channel dips for a weak periodic frame

Strengths of aperiodic frame dips and weak periodic frame dips are comparable as seen in Figures 2-13 and 2-14. In order to distinguish an aperiodic frame from a weak periodic frame, nonzero dips are investigated. Aperiodic frames have more nonzero dips than weak periodic frames, so aperiodic frames look noisier than the weak periodic frames.

2.5 Computation of Periodicity and Aperiodicity

The distribution and strength of the AMDF dips in channels relative to the location of the pitch value and its integer multiples are used to compute the periodicity and aperiodicity values for that channel.

Periodicity measurement:

The most important indicator of the periodicity is the high strength dips which are at the pitch value and its integer multiples or very close to pitch value or its integer multiples. In order to reflect this situation on the periodicity algorithm, dips are weighted such that dips which are closer to the cluster peaks contribute more toward periodicity. This contribution decreases rapidly with the increasing distance from the cluster peaks. Consequently, dips are weighted using exponentially decaying weights according to their distance to the pitch and its integer multiples [13].

Periodic signal's AMDFs are expected to have equally spaced dips of similar strength. To take the contribution of each pitch multiple, we consider regions around each pitch multiple separately. That is, if the frame length includes C pitch multiples in its lag, then each of the regions from

$$\left[j \times p - \frac{p}{2} : j \times p + \frac{p}{2}\right] \quad (p = \text{pitch value}) \qquad \text{for} \quad j = 1,2, \ldots C$$

is analyzed separately for periodicity as shown in Figure 2-15.

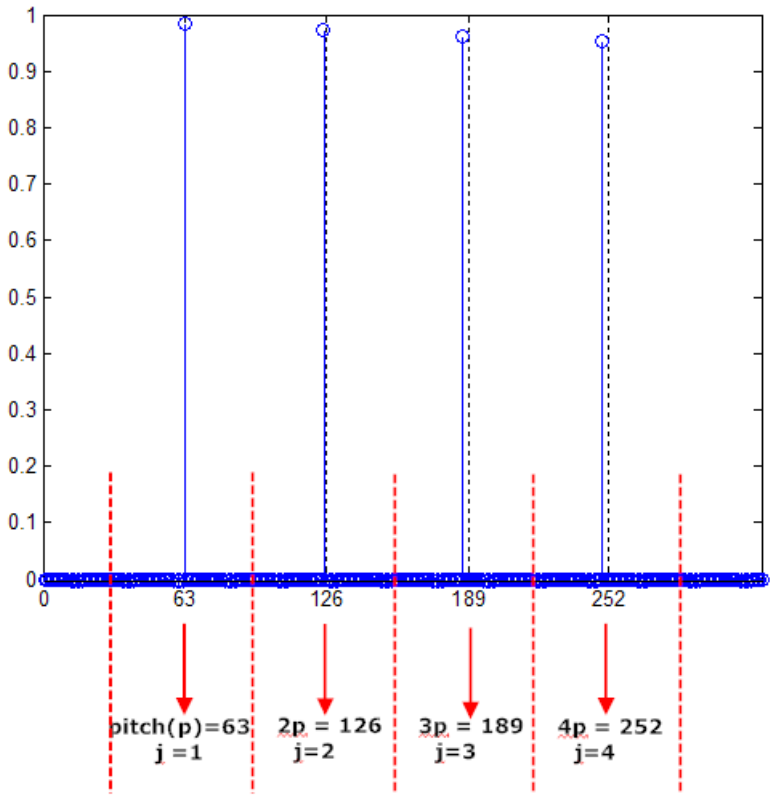

Figure 2-15: Cluster boundaries for a periodic channel to calculate periodicity

The following equation shows the calculation of the cluster periodicity for the jth cluster:

$$periodicity_j = s_j + (1 - s_j) \sum_{i=jxp-\frac{p}{2}}^{i=jxp+\frac{p}{2}} d_i \times w_i \qquad (2.5)$$

Where

p : pitch value,

s_j : strength of the dip closest to the pitch or its integer multiple in the cluster. If the cluster has a dip at the pitch value or its integer multiples,

then s_j will be equal to that dip's strength. Otherwise s_j will be equal to the strength of the dip which is closest to the pitch value or its integer multiples

d_i : strength of the dip i locations away from the peak,

w_i : value of the exponential weight function at location i. That is

$$w_i = e^{-\frac{|j \times p - i|}{7}} \tag{2.6}$$

If a dip is near the pitch value or one of its integer multiple, then it will conserve most of its value toward periodicity. When the dips get far away from the pitch value or one of its integer multiples, the contribution toward periodicity will decay exponentially. The "7" in the denominator of the formula (2.6) is found empirically. For example; Figures 2-16, 2-17 and 2-18 show three different channel AMDFs. The periodicity values found with three different weight formulas for these three channels are given in Table 2-2.

Table 2-2 Periodicity values for different weight values

Weight / Ch.	$w_i = e^{-\frac{\lvert j \times p - i \rvert}{3}}$	$w_i = e^{-\frac{\lvert j \times p - i \rvert}{7}}$	$w_i = e^{-\frac{\lvert j \times p - i \rvert}{10}}$
Ch. 1	0.86742	0.98882	0.99111
Ch. 2	0.81005	0.92248	0.9965
Ch. 3	0.87393	0.87407	0.89339

It is expected that, Channel 1 have the maximum periodicity among these three channels.

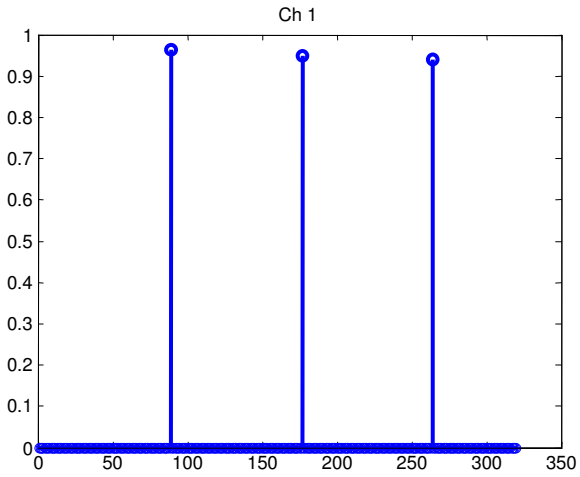

Figure 2-16: AMDF dips for Channel 1

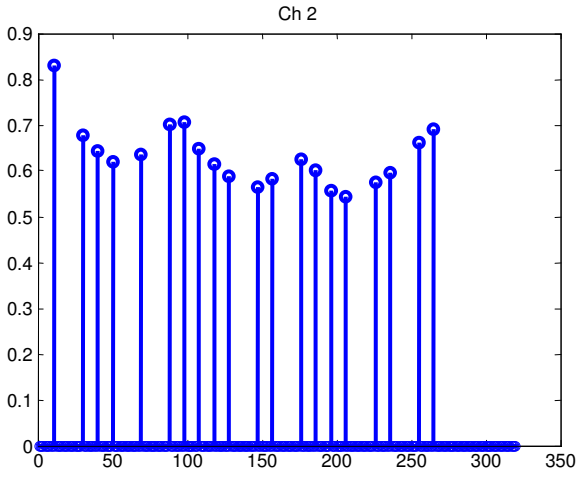

Figure 2-17: AMDF dips for Channel 2

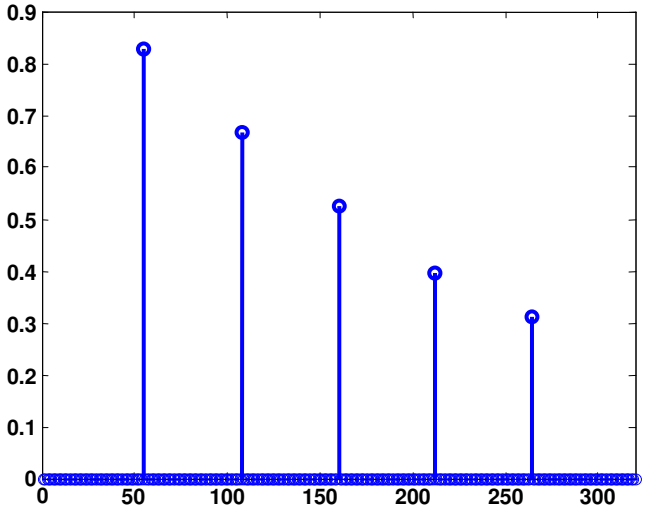

Figure 2-18: AMDF dips for Channel 3

$w_i = e^{-\frac{|j \times p - i|}{3}}$ formulation produced maximum periodicity for Ch.3 and $w_i = e^{-\frac{|j \times p - i|}{10}}$ formulation produced maximum periodicity for Ch.2. So, $w_i = e^{-\frac{|j \times p - i|}{7}}$ formulation is used to calculate periodicity.

To find the overall channel periodicity, the average periodicity across all clusters within a channel is calculated by equation (2.7)

$$periodicity_{channel} = \frac{1}{C} \sum_{j=1}^{j=C} periodicity_j \qquad (2.7)$$

Aperiodicity measurement:

Important indicators of the aperiodicity are; dips are far from the pitch period and its integer multiples; dips are scattered along the frame length and they are small in amplitude. In order to reflect this situation on aperiodicity algorithm, dips are weighted such that dips far from the cluster peaks contribute more toward aperiodicity. This contribution decreases rapidly with decreasing distance from the cluster peaks. Consequently, dips are weighted according to their location and exponentially decaying weights are used [13]. Aperiodicity measurement is defined as the sum of these weighted dips instead of the mean across the clusters. That is because aperiodicity is directly related to number of spurious dips.

Note that the dips which are minimum 10 samples far away from the pitch value or its multiple taken into account during aperiodicity measurement.

$$aperiodicity = \sum_{i=1}^{i=N} d_i \times w_i \qquad (2.8)$$

$$for\ \forall i\ that\ satisfies\ \min(|i - p \times j|) > 10\ \ for\ j = 1,2,..C$$

d_i : strength of the dip at location i,

C : total number of clusters

w_i : value of the exponential weight function at location i. That is

$$w_i = e^{-\frac{|p-distance|}{10}} \qquad (2.9)$$

$distance$: minimum of the distances of i to p and its integer multiples,so

$$distance = \min(i - p \times j) \quad for \;\; j = 1,2,..C \tag{2.10}$$

The “10” in the denominator of (2.9) is found empirically like periodicity formulation which is explained with table 2-2.

Some examples for the channel dip characteristics and their periodicity and aperiodicity values are shown in Figures 2-19 to 2-22

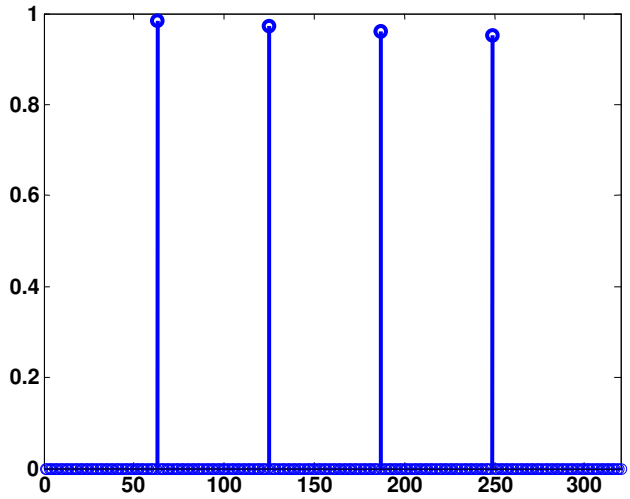

Figure 2-19 Dips for a periodic channel

Periodicity = 0.99153, Aperiodicity = 0

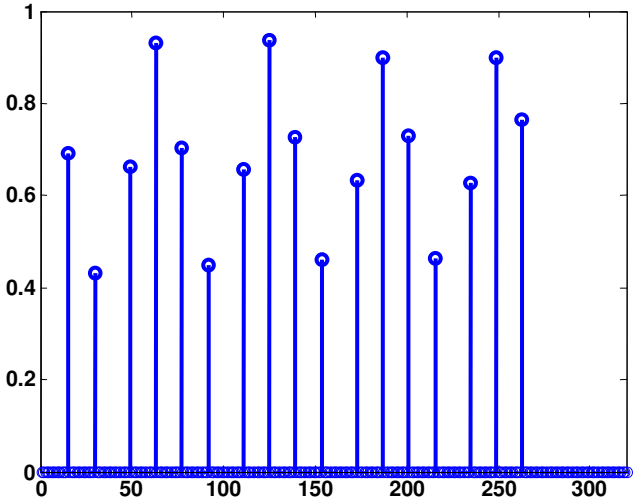

Figure 2-20 Dips for a periodic channel

Periodicity = 0.99415, Aperiodicity = 0.25824

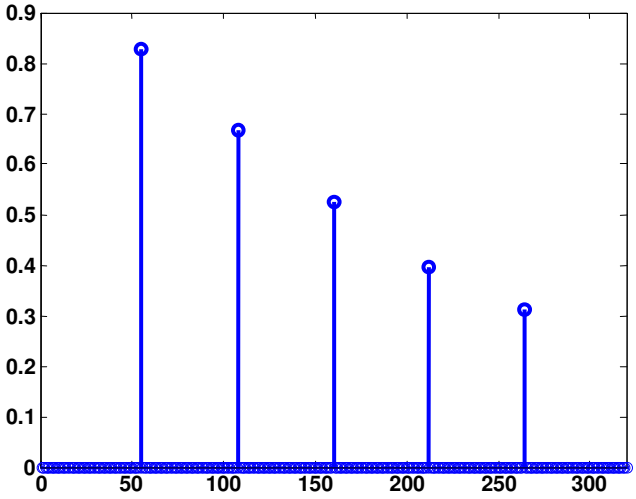

Figure 2-21 Dips for a weak periodic channel

Periodicity = 0.73139, Aperiodicity = 0

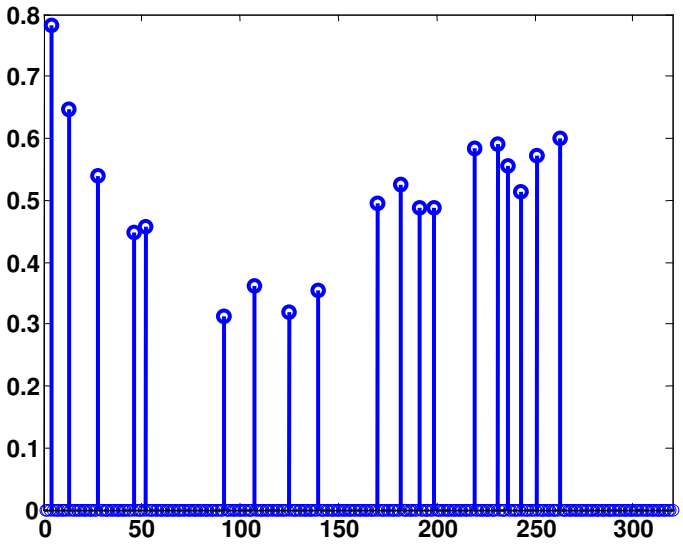

Figure 2-22 Dips for an aperiodic channel

Periodicity = 0.69559, Aperiodicity = 0.85928

If we compare the results, channel in Figure 2-19 is purely periodic, its dips are located on the pitch multiples or very close to pitch multiples. Channel in Figure 2-20 is another periodic channel. It has some aperiodicity value, because of the dips which are located far from pitch value. Figure 2-21 is a weak periodic channel. Its periodicity is lower than the previous channel's periodicity and it has no aperiodicity value as expected. The maximum aperiodicity value belongs to the aperiodic channel as expected (Figure 2-22). We see that, periodicity value of the aperodic frame is very close to the periodicity of the weak periodic frame. This may seem interesting but actually that was another expected result because the number of nonzero dips in the aperiodic frame is more than the number of nonzero dips in the weak periodic frame. On the other hand the aperiodicity value is bigger than the periodicity value in the aperiodic frame and this situation occurred only for the aperiodic frame.

2.6 Database

Sabancı University records are used for testing the constructed algorithm. This database contains total 70 speakers both male and female. Each speaker has approximately 120 sentences. Female speakers are selected and their speeches are processed in this thesis.

2.7 Classification

For selected phonemes, periodicity and aperiodicity values are investigated to decide the suitable model. For ten people, “m”, “a”, “yo” and “e” phonemes are extracted.

Extraction of selected phonemes from sentences is done with Audacity 1.3 Beta (Unicode) program [21]. For each person and selected phoneme, we have approximately 120 frames for training. Note that, each individual sample has approximately 5 to 10 frames for a selected phoneme. Therefore, 100 frames means approximately 700 - 800 ms with 20 ms frame length and 15 ms overlap. Extraction of selected phonemes is followed by finding the periodicity and aperiodicity values for that phonemes and then gathering these values. Periodicity and aperiodicity values of frames are collected in a matrix to obtain an overall periodicity and aperiodicity database for selected phoneme of a specific person. Obtained matrix has 16 columns for 16 channels and its rows indicate the frames. Histogram of each channel’s periodicity and aperiodicity values are inspected in order to understand their distributions. Normalized periodicity histograms for three people’s “a” phoneme is given in Figure 2-23, 2-24 and 2-25 for different channels. Figure 2-23 shows three person’s 1^{st} and 4^{th} channel’s periodicity histograms, Figure 2-24 shows 7^{th} and 10^{th}, and last Figure 2-25 shows 13^{th} and 16^{th} channel’s periodicity histograms.

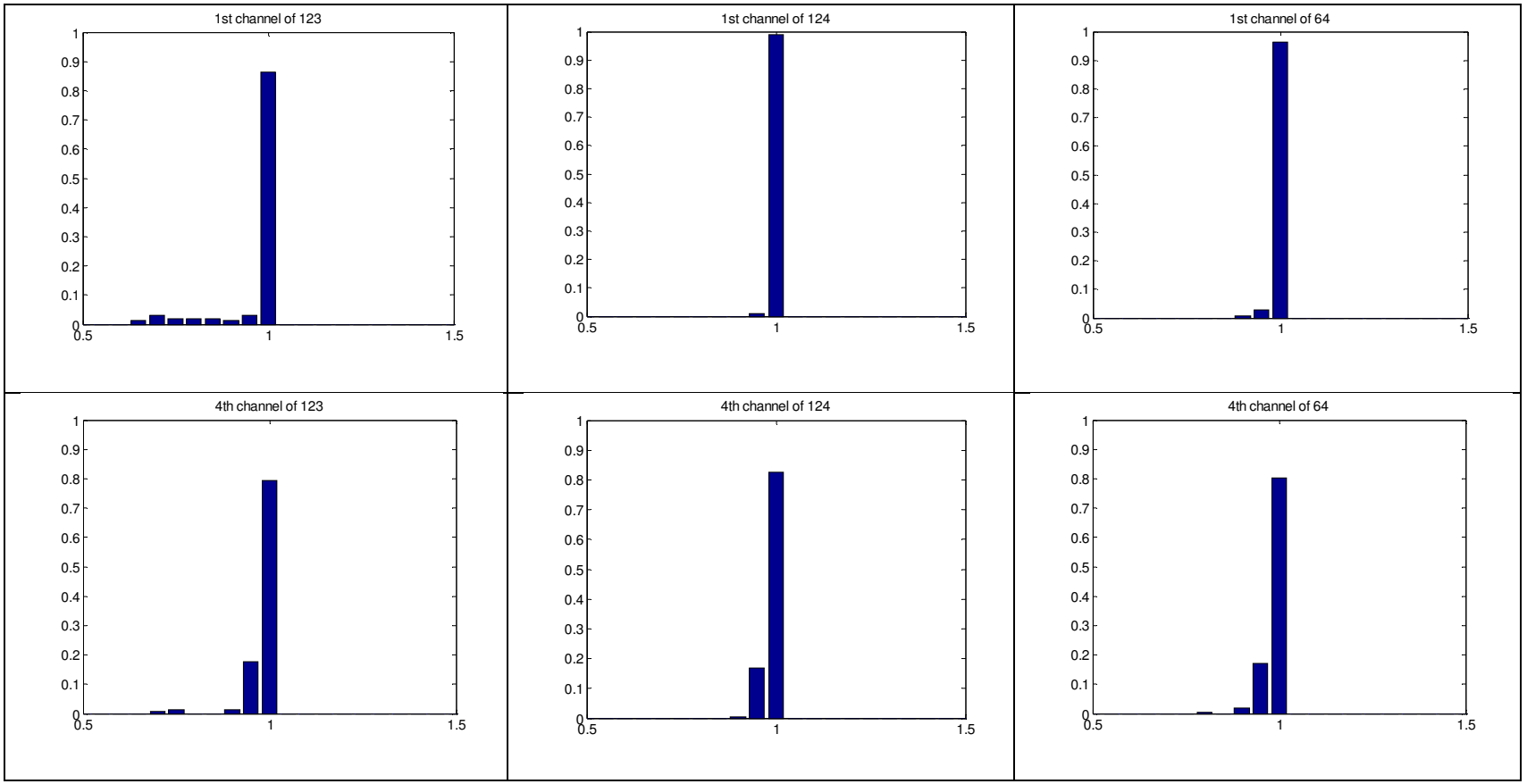

Figure 2-23 Normalized periodicity histograms of "a" for 3 different persons, channels 1 and 4

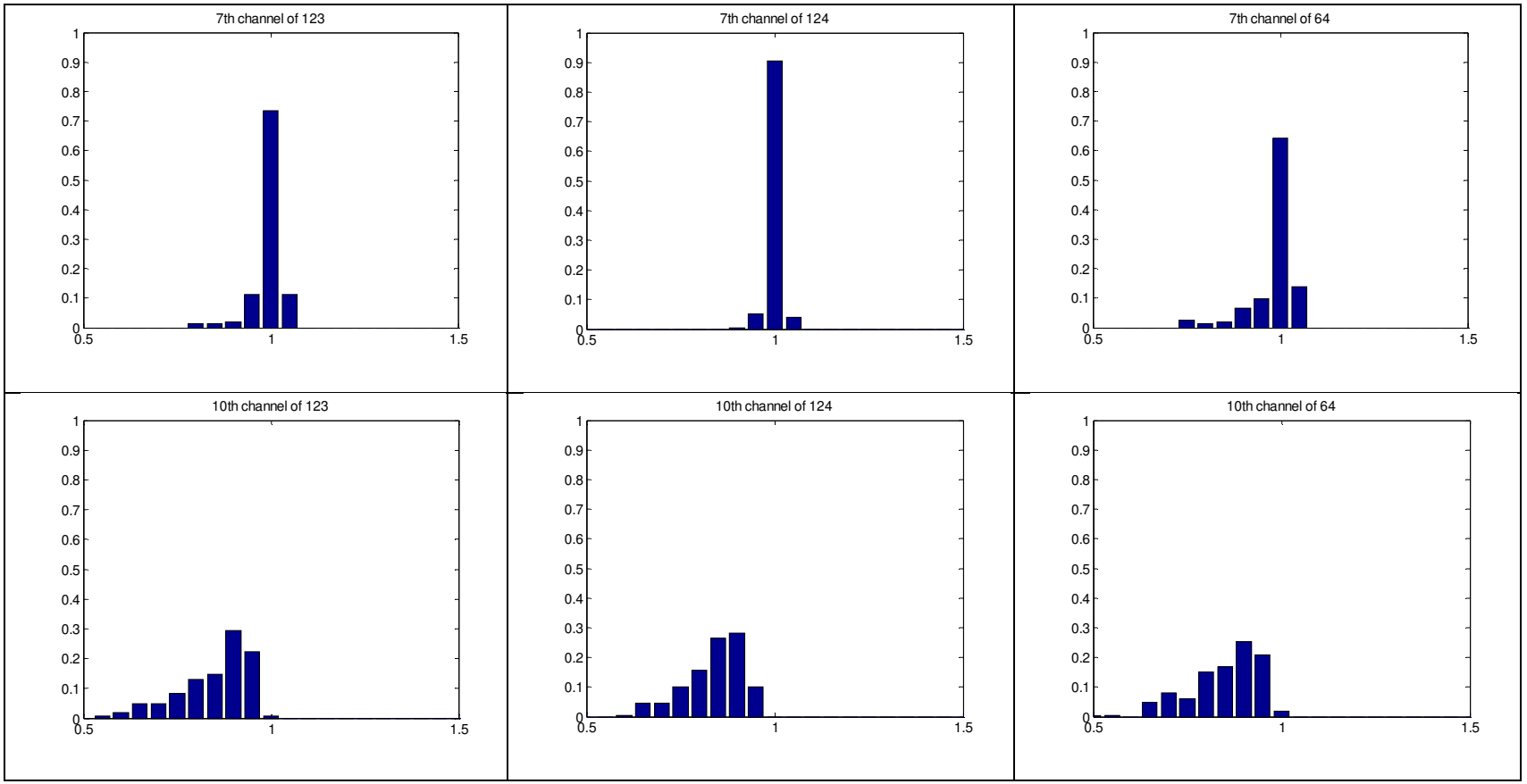

Figure 2-24 Normalized periodicity histograms of "a" for 3 different persons, channels 7 and 10

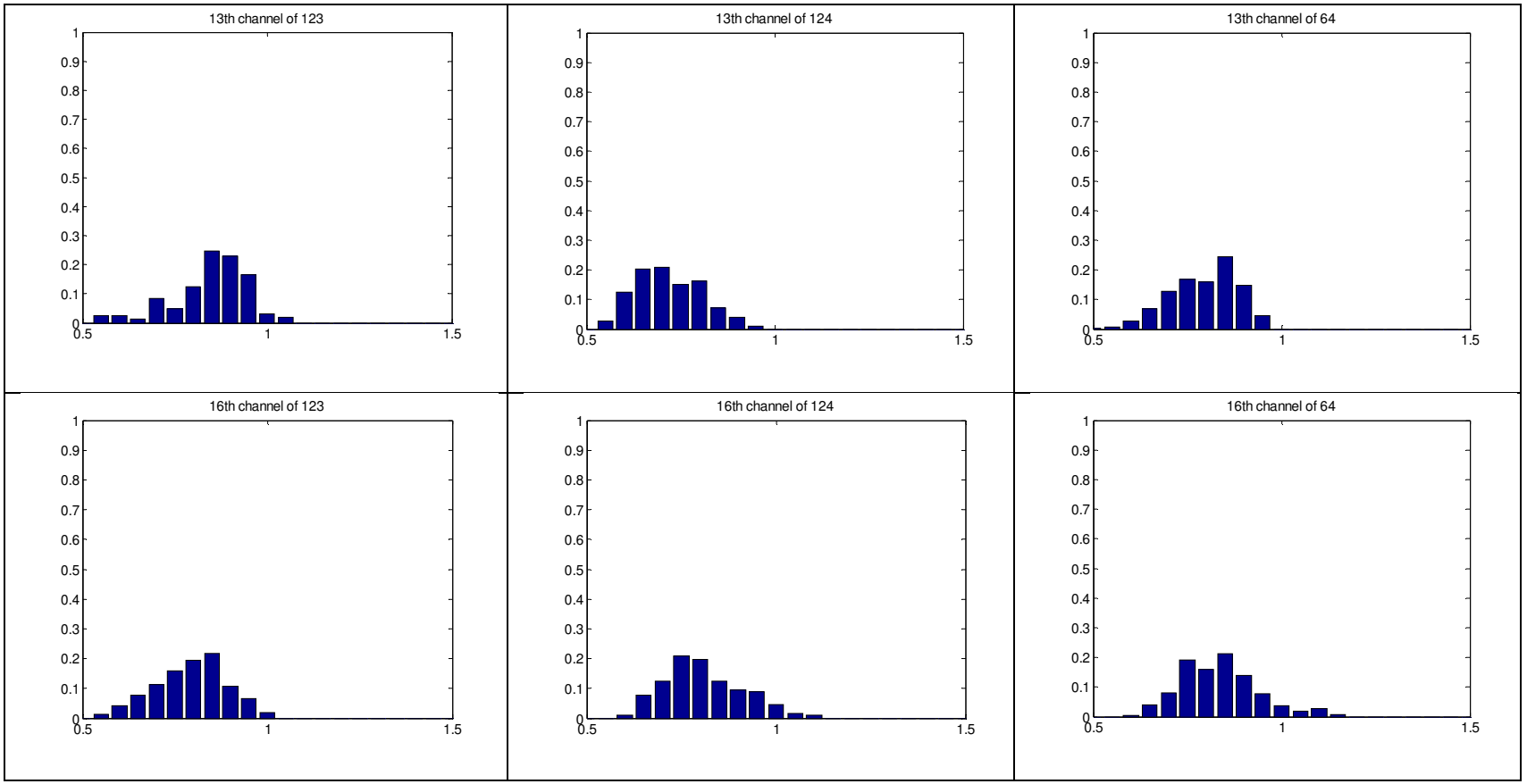

Figure 2-25 Normalized periodicity histograms of "a" for 3 different person, channels 13 and 16

In order to find the appropriate distribution function, trial version of Easy Fit (ver.4.0) [23] program is used. Easy Fit is a data analysis and simulation application allowing to fit probability distributions to sample data and select the best model. For selected phonemes, each channel is analyzed with Easy Fit and it showed that Generalized Extreme Value Distribution is appropriate for all channels.

Generalize Extreme Value (GEV) distribution formula and its characteristics are given below:

Parameters

k- continuous shape parameter

σ- continuous scale parameter (σ>0)

μ- continuous location parameter

Domain

$1+k\frac{(x-\mu)}{\sigma} > 0$ for $k \neq 0$

$-\infty < x < +\infty$ for $k = 0$

Probability Density Function

$$f(x) = \begin{cases} \frac{1}{\sigma}\exp\left(-(1+kz)^{-1/k}\right)(1+kz)^{-1-1/k} & k \neq 0 \\ \\ \frac{1}{\sigma}\exp(-z-\exp(-z)) & k = 0 \end{cases} \quad (2.11)$$

Cumulative Distribution Function

$$F(x) = \begin{cases} \exp\left(-(1+kz)^{-1/k}\right) & k \neq 0 \\ \exp(-\exp(-z)) & k = 0 \end{cases} \quad (2.12)$$

Where $z \equiv \frac{x-\mu}{\sigma}$

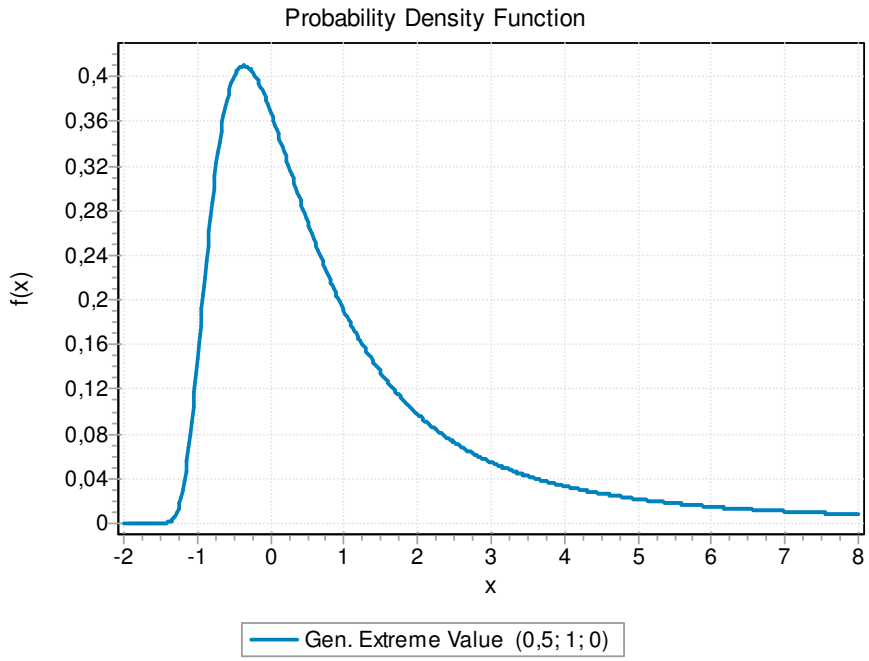

Figure 2-26 Generalized Extreme Value Distribution

Figure 2-27 shows a channel histogram of our data and the fitted GEV distribution. ($k = -1.82827$ $\sigma = 0.009688$ and $\mu = 0.99216$)

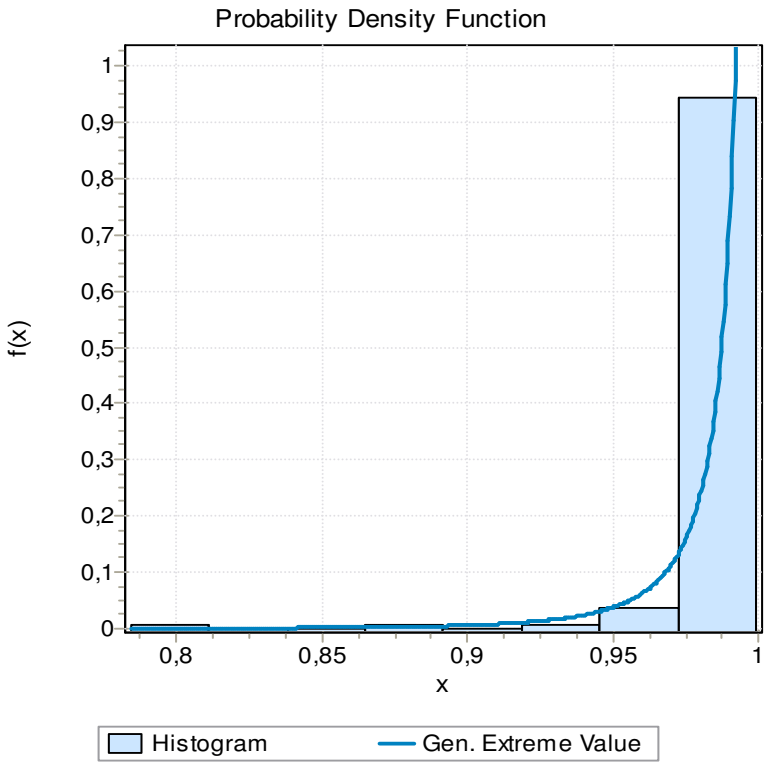

Figure 2-27 Generalized Extreme Value Function fitted a channel

$(k = -1.82827 \ \sigma = 0.009688 \ and \ \mu = 0.99216)$

The disadvantage of GEV is its values over unobserved values are not reasonable. An example of that situation is shown in the Figure 2-28. In order to get accurate distributions, we have to model each channel using its maximum point and equating the value of the unobserved data to 0. This is a very time consuming method, so we did not use it.

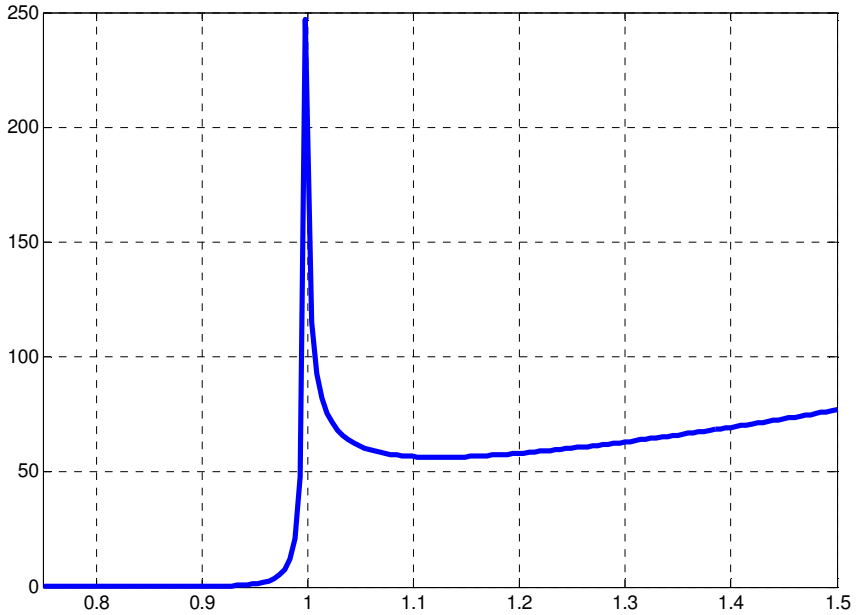

Figure 2-28 Generalized Extreme Value Function fitted a channel (unobserved values)

$$(k = -1.82827 \ \sigma = 0.009688 \, and \, \mu = 0.99216)$$

Another way of estimating the probability density function of a random variable is Kernel Density Estimation (Parzen Window). As an illustration, given some data about a sample of a population, kernel density estimation makes it possible to extrapolate the data to the entire population. Given an instance of the random sample x, Parzen-windowing estimates the PDF $\mathrm{p}(\mathrm{x})$ from which the sample was derived. Suppose that we want to estimate the value of the PDF at point x. Then, we can place a window function at x and determine how many observations $\mathrm{x_i}$ fall within our window or, rather, what is the contribution of each observation to this window. The PDF value is then the sum total of the contributions from the observations to this window. The Parzen-window estimate is defined as:

$$p_n(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h_n}\varphi\left(\frac{x - x_i}{h_n}\right) \tag{2.13}$$

Where $\varphi(u)$ is the window function. Gaussian window function is used, that is:

$$\varphi(u) = \frac{1}{\sqrt{2\pi}}e^{-u^2/2} \tag{2.14}$$

And

h_n: window width

The multivariate kernel density estimator in the d-dimensional case is defined as [24]

$$p_n(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h_1 \dots h_d}\varphi\left(\frac{x_i - x_{i_1}}{h_1}, \dots, \frac{x_i - x_{i_d}}{h_d}\right) \tag{2.15}$$

A common approach to build multidimensional kernel functions is to use a product kernel [24]

$$\varphi(u_1, \dots, u_d) = \prod_{i=1}^{d}\varphi(u_i) \tag{2.16}$$

where φ is a one-dimensional kernel function.

Figures 2-29, 2-30 and 2-31 show our typical histogram and its kernel density estimate:

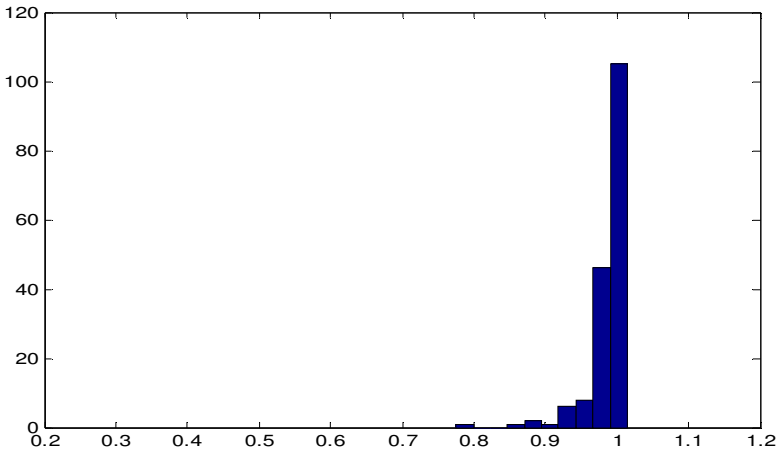

Figure 2-29 Histogram of a channel

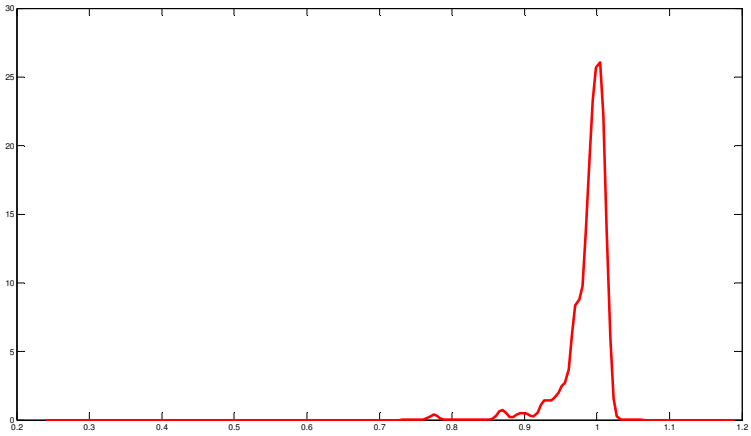

Figure 2-30 Parzen density estimate of the channel in Figure 2-29

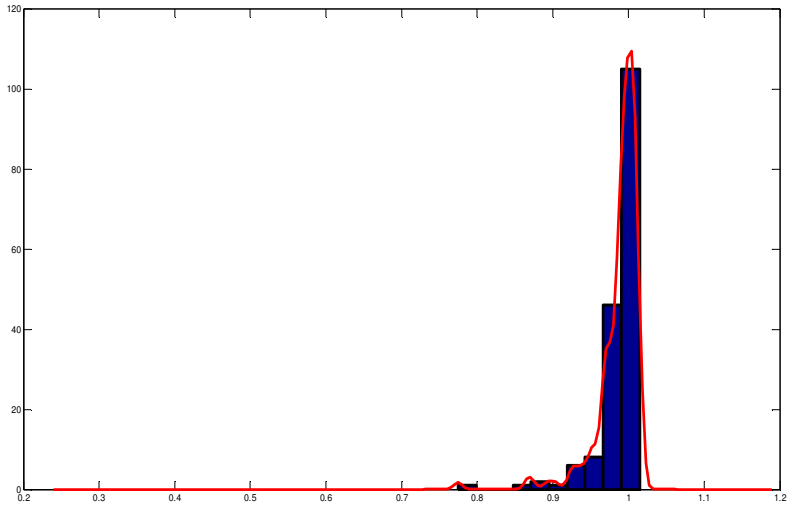

Figure 2-31 Channel Histogram and its parzen density estimate (Fig. 2-29 and Fig. 2-30 together)

2.8 Testing

For each person and for each phoneme, approximately 25 frames are extracted for testing. Words selected for testing are different from the words used for training.

After obtaining periodicity and aperiodicity values for the test frames, 2 different algorithms are used for comparing test model with training models. They are Kullback-Leibler Distance and Maximum Likelihood methods.

2.8.1 Kullback-Leibler Distance

The Kullback-Leibler distance (KL-distance) is a measure of the difference between two probability distributions. It is defined as

$$KL\,(p,q) = \sum_{i} p(i) ln \frac{p(i)}{q(i)} \tag{2.17}$$

Where "p" represents data, observations, or a precise calculated probability distribution, and "q" represents a theory, a model, a description or an approximation of "p" [22].

It is clear that when $\mathrm{p}(x) = \mathrm{q}(x);\ \mathrm{KL}\,(\mathrm{p},\mathrm{q}) = 0$. So the KL distance between the same person's test and training distributions is expected to be smaller than the KL distance between the different person's test and training distributions.

2.8.2 Maximum Likelihood

The likelihood of a sample point given a model is the value of the probability density function for that point.

$$p(X) = \prod_{i=1}^{N} p(x_i) \tag{2.18}$$

Computing the log-likelihood turns the product into a sum:

$$\log\ p(X) = \sum_{i=1}^{N} \log p(x_i) \tag{2.19}$$

For the multidimensional kernel density case, insert equation (2.15) and (2.16) in equation (2.19):

$$p_n(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h_1 \dots h_d}\prod_{j=1}^{d}\frac{1}{\sqrt{2\pi}}exp\left(-\frac{\left(\frac{x_i - x_{i_j}}{h_j}\right)^2}{2}\right)$$

(2.20)

The training model, that produces the highest likelihood value for a test distribution is assumed to be the best fitting model for that test model. This means that a person's test distribution is expected to have the maximum likelihood with the same person's training distribution.

2.9 Mel Frequency Cepstrum Coefficients (MFCC)

Periodicity features are used with MFCCs as supplementary features.
Section 3.2 and 3.3 give the test results for contribution of the periodicity features to MFCCs. The test has been carried out with both selected phonemes and sentences.

Block diagram of the MFCC processor is given in Figure 2-32. Extracting the MFFC features starts with segmenting the speech into frames. (See 2.1) The next step in the processing is to window each individual frame to minimize the signal discontinuities at the beginning and at the end of each frame. Typically the Hamming window is used, which has the form:

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right), \qquad 0 \le n \le N-1 \qquad (2.21)$$

The next processing step is the Fast Fourier Transform, which converts each frame of N samples from the time domain into the frequency domain. After obtaining the spectrum, signal is filtered using Mel-spaced filterbank. (See 2.2) In the final step, mel spectrum is converted back to time using Discrete

Cosine Transform. The result is called the mel frequency cepstrum coefficients (MFCC).

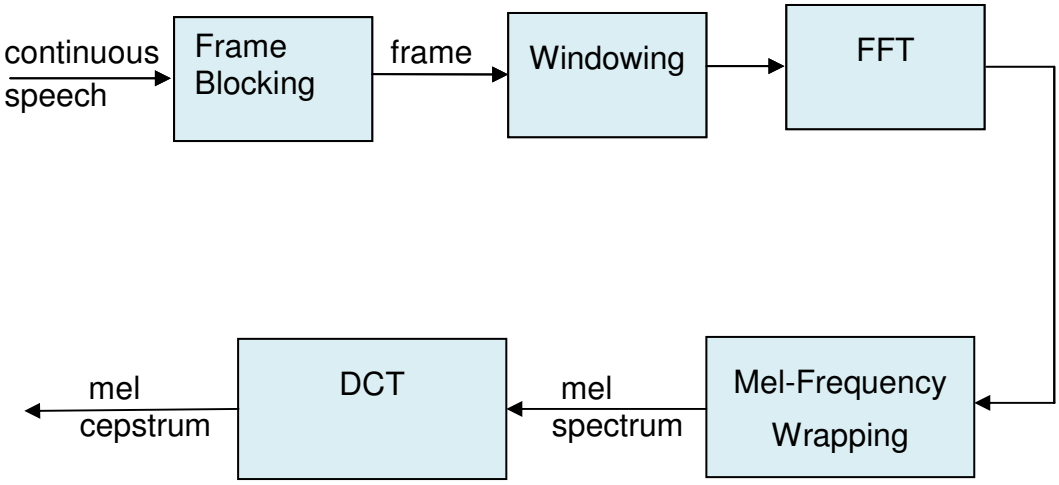

Figure 2-32 Block Diagram of the MFCC Processor

MFCC features are obtained directly using the HTK Tool. HTK is developed by Cambridge University and it is a toolkit for building Hidden Markov Models (HMMs). HMMs can be used to model any time series and the core of the HTK is similarly general-purpose. However, HTK is primarily designed for building HMM-based speech processing tools, in particular recognisers. Much of the functionality of HTK is built into the library modules. It has 4 main phases: data preparation, training, testing and analysis. [25]

Some important properties of HTK are listed below:

- HTK functions are accessed from the command-line
- Tools provided for estimating HMM parameters

- Testing can be done with Viterbi decoder
- Requires transcriptions and audio files.
- Performs Baum-Welch estimation of HMM parameters
- Allows for various parameter tying schemes and mixture incrementing
- Performs Viterbi-based recognition using the Token-passing algorithm

The details of the HTK Tool could be found in [25].

CHAPTER 3

EXPERIMENTS

3.1 Results for Periodicity & Aperiodicity of Phonemes

Test results for 10 persons' "m", "a", "yo" and "e" phonemes are given in the confusion matrices below. 10 persons are coded as: 07, 08, 10, 63, 64, 66, 123, 124, 126 and 181 respectively.

In order to get more test results, cross validation method is used. That is, test data has been shifted with small increments on the overall data and training has been repeated for each test. This method helped us to obtain confusion matrices. Values in the confusion matrices show the percentage of the results. It is expected that the diagonal elements have the largest values. "**Bold**" values are the largest values in the tables below and indicate the decided person (column) for the test data (row).

Table 3-1 to 3-8 are periodicity feature results for "m", "a", "yo" and "e" phonemes respectively. Tables 3-9 to 3-16 give the aperiodicity feature results. For each phoneme, test results found by using KL distance and Maximum Likelihood methods are also given in the tables. Tables 3-17 to 3-20 give the total test results for periodicity and aperiodicity features for all phonemes using KL distance and ML testing methods. "Red" values indicate the unexpected results, errors (see 2.8)

Table 3-1 Confusion matrix for periodicity of "m" using KL distance

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**72.73**	0	0	0	0	4.55	0	13.64	9.09	0
08	0	**92**	0	0	0	4	0	4	0	0
10	16	0	**44**	0	0	0	36	4	0	0
63	0	10	0	**50**	0	5	0	0	35	0
64	0	0	8.33	4.17	**79.17**	0	4.17	4.17	0	0
66	19.23	0	0	0	0	**69.23**	0	3.85	7.69	0
123	0	0	0	0	0	0	**95.65**	4.35	0	0
124	0	0	3.45	0	0	0	3.45	**82.76**	10.35	0
126	9.09	0	0	4.55	4.55	0	0	0	**81.82**	0
181	0	0	0	0	0	0	0	0	0	**100**

Table 3-2 Confusion matrix for periodicity of "m" using ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	22.73	0	13.6	**54.55**	0	0	0	0	9.09	0
08	0	0	24	**72**	0	0	0	0	4	0
10	0	0	**36**	20	12	4	4	0	0	24
63	5	0	0	25	5	0	5	5	15	**40**
64	4.17	0	0	8.33	**62.5**	0	0	0	0	25
66	0	0	3.85	**84.62**	0	3.85	0	0	7.69	0
123	0	0	0	21.74	**39.13**	21.74	13.04	4.35	0	0
124	0	0	**41.38**	27.59	0	0	0	31.03	0	0
126	0	9.09	0	**50**	0	4.55	0	0	27.27	9.09
181	22.73	0	13.64	**54.55**	0	0	0	0	9.09	0

Table 3-1 shows that all of the 10 persons are recognized using “m” phoneme with KL distance method. Minimum performance is 44% and that is the result for person “10”. This means that for 44 of the total 100 tests, test phonemes for “10” produced minimum KL distance with “10”’s training phonemes. Test phonemes for person “181” produced minimum KL distance with her own training phonemes for all tests, so the result for “181” is 100% as shown in the Table 3-1.

Recognition performance decreased with ML method. Most of the test phonemes couldn’t be recognized with this method.

Table 3-3 Confusion matrix for periodicity of “a” using KL distance

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**50**	0	4.17	0	0	20.83	0	25	0	0
08	0	**63.33**	0	0	0	10	20	3.33	3.33	0
10	0	0	**79.3**	0	0	20.69	0	0	0	0
63	0	3.33	0	**53.33**	0	0	0	0	13.33	0
64	0	9.76	0	0	**41.46**	7.32	21.95	0	19.52	0
66	0	3.33	0	0	0	**93.33**	3.33	0	0	0
123	3.45	0	0	0	10.35	**44.83**	10.35	0	31.03	0
124	23.33	0	0	0	0	0	0	**76.67**	0	0
126	0	0	0	0	0	0	4.17	25	**70.83**	0
181	14.29	28.57	3.57	0	0	**35.71**	0	14.29	0	3.57

Table 3-4 Confusion matrix for periodicity of "a" using ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	41.67	0	0	**58.33**	0	0	0	0	0	0
08	0	3.33	0	**96.67**	0	0	0	0	0	0
10	0	0	37.93	**62.07**	0	0	0	0	0	0
63	0	0	0	**100**	0	0	0	0	0	0
64	0	0	0	**80.49**	19.51	0	0	0	0	0
66	0	0	0	**90**	0	0	0	0	0	10
123	0	20.69	0	**68.97**	0	0	10.35	0	0	0
124	0	0	0	**66.67**	0	0	0	33.33	0	0
126	0	0	0	**100**	0	0	0	0	0	0
181	7.14	3.57	0	25	7.14	0	0	3.57	0	**53.57**

Table 3-3 shows that, "a" phoneme could not be recognized only for "123" and "181" using KL distance. Test phonemes for both of them produced minimum KL distances with person "66". Recognition performance decreased using ML method. Table 3-4 shows that most of the test phonemes are recognized as person "63" and this is because of the wide distribution of the "63"'s "a" phonemes. In ML method, if the test data has small variance and its mean value is slightly different from the same person's training data, another training data with larger variance may produce higher likelihood values.

Table 3-5 Confusion matrix for periodicity of "yo" using KL Distance

Train / Test	07	08	10	63	64	66	123	124	126	181
07	27.78	0	0	0	0	11.11	0	**38.89**	22.22	0
08	0	**40**	0	0	0	0	15	**40**	5	0
10	0	0	**40.91**	0	0	4.55	4.55	27.27	22.73	0
63	0	0	0	8.70	4.35	4.35	**34.78**	13.04	**34.78**	0
64	0	0	0	0	**40.63**	0	12.5	15.63	31.25	0
66	0	0	0	0	0	**65.22**	17.39	0	17.39	0
123	0	0	0	0	0	0	**91.30**	0	8.70	0
124	0	0	2.94	0	0	2.94	11.77	**67.65**	14.71	0
126	0	0	0	0	0	0	8.33	0	**91.67**	0
181	0	0	0	8.57	0	17.14	0	**60**	2.86	11.43

Table 3-6 Confusion matrix for periodicity of "yo" using ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**38.89**	11.11	0	0	0	16.67	5.56	16.67	0	11.11
08	0	**90**	0	0	0	0	0	0	0	10
10	4.55	9.09	22.73	0	22.73	9.09	0	**27.27**	4.55	0
63	**26.09**	13.04	0	21.74	4.35	0	8.70	0	0	**26.09**
64	0	43.75	0	0	**50**	0	0	0	6.25	0
66	13.04	0	**43.48**	0	0	13.04	0	4.35	0	26.09
123	26.09	0	8.70	26.09	**30.44**	0	0	0	0	8.70
124	20.59	23.53	0	0	17.65	0	0	0	0	**38.24**
126	37.5	0	0	0	20.83	0	0	0	0	**41.67**
181	5.71	**48.57**	0	0	0	11.43	0	0	0	34.29

Table 3-5 shows that, “yo” phonemes of person “07”, “63” and “181” could not be recognized using KL distance. Also KL distances for “08” test phonemes are equal to training phonemes of “08” and “124”.

Table 3-7 Confusion matrix for periodicity of “e” using KL Distance

Train / Test	07	08	10	63	64	66	123	124	126	181
07	12.90	0	25.81	0	0	0	0	**61.29**	0	0
08	0	**60.71**	25	3.57	0	10.71	0	0	0	0
10	0	0	**82.93**	0	0	0	17.07	0	0	0
63	0	25.81	16.13	**51.61**	0	0	0	0	6.45	0
64	0	2.5	2.5	**25**	0	20	**25**	15	0	10
66	0	0	0	0	0	**97.22**	2.78	0	0	0
123	0	0	**79.31**	0	0	0	13.79	6.90	0	0
124	0	0	43.59	0	0	0	0	0	**56.41**	0
126	0	3.23	9.68	**32.26**	0	0	19.36	3.23	**32.26**	0
181	0	3.03	0	33.33	3.03	0	0	**57.58**	0	3.03

Table 3-8 Confusion matrix for periodicity of "e" using ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**35.48**	0	12.90	16.13	12.90	6.45	6.46	9.68	0	0
08	14.29	14.29	0	3.57	**42.86**	0	0	17.86	3.57	3.57
10	26.83	0	0	12.20	**60.98**	0	0	0	0	0
63	0	9.68	0	12.90	**58.07**	0	0	0	19.36	0
64	0	0	0	0	**87.5**	12.5	0	0	0	0
66	0	0	0	**63.89**	33.33	0	0	0	2.78	0
123	20.69	0	0	6.90	**41.38**	3.45	3.45	6.90	17.24	0
124	**58.97**	0	0	0	23.08	0	0	5.13	12.82	0
126	0	22.58	0	**29.03**	**29.03**	3.23	0	16.13	0	0
181	0	9.09	0	0	0	18.18	9.09	0	9.09	**54.55**

Table 3-7 shows that, half of the 10 persons are recognized with "e" phonemes' periodicity features using KL distance method. Recognition performance decreased with ML method again. Table 3-8 shows that most of the test phonemes are recognized as person "64" and this is because the wide distribution of the "e" phonemes of the "64".

Recognition performance for the "m" phoneme's periodicity features is better than other phonemes as shown in the tables above. "m" is a nasal phoneme which is formed by blocking the oral passage and allowing the air to escape through the nose. Effect of nose on producing the "m" phoneme may increase the recognition performance.

Recognition performances for "m", "a", "yo" and "e" phonemes's aperiodicity features are shown in the Tables 3-9 to 3-18.

Table 3-9 Confusion matrix for aperiodicity of "m" using KL Distance

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**59.09**	0	0	0	4.55	0	31.82	4.55	0	0
08	20	**72**	0	0	4	4	0	0	0	0
10	**60**	0	32	0	4	0	4	0	0	0
63	**60**	5	0	0	20	5	0	10	0	0
64	**62.5**	0	0	0	37.5	0	0	0	0	0
66	**69.23**	0	0	0	0	26.92	0	3.85	0	0
123	**52.17**	0	0	0	4.35	13.04	30.44	0	0	0
124	**47.62**	0	4.76	0	19.05	0	0	28.57	0	0
126	**50**	4.55	0	0	9.09	0	0	4.55	31.82	0
181	21.74	4.35	0	0	4.35	4.35	0	4.35	**30.44**	**30.44**

Table 3-10 Confusion matrix for aperiodicity of "m" using ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	13.64	45.46	0	0	4.55	0	0	31.82	0	4.55
08	0	**68**	0	32	0	0	0	0	0	0
10	0	0	20	**80**	0	0	0	0	0	0
63	0	15	0	**85**	0	0	0	0	0	0
64	0	0	0	29.17	**70.83**	0	0	0	0	0
66	0	15.39	0	**42.31**	0	23.08	0	11.54	0	7.69
123	0	21.74	4.35	**39.13**	4.35	0	30.44	0	0	0
124	9.52	0	0	**47.62**	9.52	0	0	33.33	0	0
126	0	**54.55**	0	13.64	0	0	0	4.55	27.27	0
181	0	17.39	0	21.74	21.74	0	0	0	0	**39.13**

Table 3-11 Confusion matrix for aperiodicity of "a" using KL Distance

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**91.67**	0	8.33	0	0	0	0	0	0	0
08	6.67	16.67	**53.33**	0	13.33	0	0	6.67	3.33	0
10	0	0	**100**	0	0	0	0	0	0	0
63	0	3.33	**76.67**	3.33	6.67	0	0	0	10	0
64	0	0	**63.41**	0	36.59	0	0	0	0	0
66	0	0	**36.67**	0	0	**50**	0	0	13.33	0
123	6.90	3.45	**86.21**	0	0	0	3.45	0	0	0
124	20	0	**70**	0	0	0	0	10	0	0
126	25	0	20.83	0	4.17	0	0	0	**50**	0
181	0	0	**71.43**	0	3.57	3.57	0	0	17.86	3.57

Table 3-12 Confusion matrix for aperiodicity of "a" using ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	25	45.83	0	16.67	8.33	0	4.17	0	0	0
08	0	46.67	0	**53.33**	0	0	0	0	0	0
10	0	0	37.93	**58.62**	0	0	3.45	0	0	0
63	3.33	0	0	**86.67**	0	3.33	0	0	6.67	0
64	0	3.57	0	**60.71**	35.71	0	0	0	0	0
66	0	0	0	**50**	0	**50**	0	0	0	0
123	0	0	0	**75**	0	0	25	0	0	0
124	0	0	0	0	0	0	0	**100**	0	0
126	0	33.33	0	20.83	0	0	0	0	**41.67**	4.17
181	5.26	5.26	5.26	5.26	0	0	0	15.79	0	**63.16**

Table 3-13 Confusion matrix for aperiodicity of "yo" using KL Dist.

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**72.22**	0	0	11.11	0	5.56	5.56	5.56	0	0
08	0	**50**	5	10	5	0	10	15	5	0
10	4.55	0	**54.55**	0	9.09	0	13.64	0	18.18	0
63	4.35	0	0	0	8.70	17.39	0	4.35	**65.22**	0
64	15.63	0	9.38	0	0	**31.25**	0	12.5	25	6.25
66	0	0	0	0	0	**100**	0	0	0	0
123	8.70	0	0	4.35	0	13.04	**56.52**	0	17.39	0
124	11.77	0	2.94	8.82	0	0	11.77	**44.12**	20.59	0
126	4.17	0	0	4.17	0	4.17	0	0	**87.5**	0
181	5.71	0	0	0	0	0	2.86	0	25.71	**65.71**

Table 3-14 Confusion matrix for aperiodicity of "yo" using ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	0	50	0	44.44	0	0	0	5.56	0	0
08	0	**80**	0	5	0	0	0	0	0	15
10	0	0	**59.09**	22.73	4.55	0	0	13.64	0	0
63	0	**52.17**	0	26.09	0	0	0	0	0	21.74
64	0	**53.13**	0	46.88	0	0	0	0	0	0
66	0	0	0	**73.91**	0	26.09	0	0	0	0
123	0	**56.52**	0	4.35	4.35	0	34.78	0	0	0
124	0	**52.94**	0	35.29	0	0	0	11.77	0	0
126	0	**50**	0	16.67	0	0	12.5	12.5	8.33	0
181	0	**42.86**	0	**42.86**	0	0	0	2.86	0	11.43

Table 3-15 Confusion matrix for aperiodicity of "e" using KL Distance

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**83.87**	0	9.68	0	0	6.45	0	0	0	0
08	3.57	0	0	21.43	14.29	**35.71**	0	14.29	0	10.71
10	29.27	0	**39.02**	0	4.88	26.83	0	0	0	0
63	0	0	6.45	**77.42**	12.90	0	3.23	0	0	0
64	5	0	**45**	15	2.5	10	12.5	0	0	10
66	11.11	13.89	5.56	2.78	2.78	**61.11**	0	2.78	0	0
123	10.35	0	13.79	6.90	0	3.45	**62.07**	3.45	0	0
124	43.59	0	5.13	0	0	0	0	**51.28**	0	0
126	0	3.23	6.45	**41.94**	19.36	29.03	0	0	0	0
181	0	3.03	0	9.09	0	3.03	0	0	0	**84.85**

Table 3-16 Confusion matrix for aperiodicity of "e" using ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	19.36	41.94	0	32.26	0	0	3.23	0	3.23	0
08	0	**46.43**	0	32.14	3.57	0	0	7.14	0	10.71
10	0	0	36.59	17.07	0	0	**46.34**	0	0	0
63	0	0	0	0	**100**	0	0	0	0	0
64	0	0	0	25	0	**50**	2.5	10	12.5	0
66	0	0	0	**100**	0	0	0	0	0	0
123	0	17.24	0	**41.38**	0	0	**41.38**	0	0	0
124	0	0	0	17.95	0	12.82	0	**69.23**	0	0
126	0	22.58	0	6.45	0	0	0	0	**71**	0
181	0	18.18	27.27	0	6.06	0	0	0	0	**48.48**

Recognition performance for the “m” phoneme is decreased significantly by using aperiodicity features (see Table 3-1 and 3-9). On the other hand, recognition performance of the aperiodicity features for the “yo” phoneme is not worse than the periodicity features (see Table 3-5 and Table 3-13). But generally, Tables 3-9 to 3-18 show that, periodicity features provide better performance than aperiodicity features.

Test results obtained for different phonemes are combined to have a general idea for the performance of periodicity and aperiodicity features. Overall results for “m”, “a”, “yo” and “e” phonemes are given in the below confusion matrices:

Table 3-17 All phonemes periodicity with KL

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**40.85**	0	7.49	0	0	9.12	0	34.70	7.83	0
08	0	**64.01**	6.25	0.89	0	6.18	8.75	11.83	2.08	0
10	4	0	**61.78**	0	0	6.31	14.41	7.82	5.68	0
63	0	9.78	4.03	**40.91**	1.09	2.34	8.70	3.26	22.39	0
64	0	3.06	2.71	7.29	**40.31**	6.83	15.90	8.70	12.69	2.50
66	4.81	0.83	0	0	0	**81.25**	5.87	0.96	6.27	0
123	0.86	0	19.83	0	2.59	**35.12**	28.86	2.81	9.93	0
124	5.83	0	12.50	0	0	0.74	23.63	**36.94**	20.37	0
126	2.27	0.81	2.42	9.20	1.14	0	7.96	27.51	**48.69**	0
181	3.57	7.90	0.89	10.48	0.76	13.21	0	**32.97**	0.71	29.51

Table 3-18 All phonemes periodicity with ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**34.69**	2.78	6.63	32.25	3.23	5.78	3.00	6.59	2.27	2.78
08	3.57	26.91	6	**43.06**	10.71	0	0	4.46	1.89	3.39
10	7.84	2.27	**24.17**	23.57	23.93	3.27	1	6.82	1.14	6
63	7.77	5.68	0	**39.91**	16.85	0	3.42	1.25	8.59	16.52
64	1.04	10.94	0	22.21	**54.88**	3.13	0	0	1.56	6.25
66	3.26	0	11.83	**59.63**	8.33	4.22	0	1.09	2.62	9.02
123	11.69	5.17	2.17	**30.92**	27.74	6.30	6.71	2.81	4.31	2.17
124	19.89	5.88	10.35	**23.56**	10.18	0	0	17.37	3.21	9.56
126	9.38	7.92	0	**44.76**	12.47	1.94	0	4.03	6.82	12.69
181	15.17	15.31	0	6.25	1.79	7.40	2.27	0.89	2.27	**48.64**

Table 3-17 and 3-18 show that, periodicity features recognized 8 persons for “m”, “a”, “yo” and “e” phonemes with KL Distance method and 5 persons with ML method. For ML method, unrecognized 5 persons (“08”, “66”, “123”, “124” and “181“) are decided as person “63”. This is because of the wide distribution of the person “63”’s phonemes as mentioned before.

Table 3-19 All phonemes aperiodicity with KL

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**76.71**	0	4.50	2.78	1.14	3.00	9.34	2.53	0	0
08	7.56	**34.67**	14.58	7.86	9.15	9.93	2.50	8.99	2.08	2.68
10	23.45	0	**56.39**	0	4.49	6.71	4.41	0	4.55	0
63	16.09	2.08	**20.78**	20.19	12.07	5.60	0.81	3.59	18.80	0
64	20.78	0	**29.45**	3.75	19.15	10.31	3.13	3.13	6.25	4.06
66	20.09	3.47	10.56	0.69	0.69	**59.51**	0	1.66	3.33	0
123	19.53	0.86	25	2.81	1.09	7.38	**38.12**	0.86	4.35	0
124	30.74	0	20.71	2.21	4.76	0	2.94	**33.49**	5.15	0
126	19.79	1.94	6.82	11.53	8.15	8.30	0	1.14	**42.33**	0
181	6.86	1.84	17.86	2.27	1.98	2.74	0.71	1.09	18.50	**46.14**

Table 3-20 All phonemes aperiodicity with ML

Train / Test	07	08	10	63	64	66	123	124	126	181
07	14.50	**45.81**	0	23.34	3.22	0	1.85	9.34	0.81	1.14
08	0	**60.28**	0	30.62	0.89	0	0	1.79	0	6.43
10	0	0	38.40	**44.61**	1.14	0	12.45	3.41	0	0
63	0.83	16.79	0	**49.44**	25	0.83	0	0	1.67	5.43
64	0	14.17	0	**40.44**	26.64	12.50	0.63	2.50	3.13	0
66	0	3.85	0	**66.56**	0	24.79	0	2.88	0	1.92
123	0	23.88	1.09	**39.96**	2.17	0	32.90	0	0	0
124	2.38	13.24	0	25.22	2.38	3.21	0	**53.58**	0	0
126	0	**40.12**	0	14.40	0	0	3.13	4.26	37.06	1.04
181	1.32	20.92	8.13	17.47	6.95	0	0	4.66	0	**40.55**

Results for the overall phonemes' aperiodicity features are similar to periodicity features with KL distance method. For ML method, 6 persons could not be recognized using aperiodicity features and most of them ("07", "10", "64", "66" and "123") are decided as person "63" like the periodicity results in Table 3-18.

3.2 Combination of Periodicity & MFCC for Specific Phonemes

For selected phonemes, MFCC and periodicity features are combined and their performance is analyzed using the likelihood method. Combined features have 28 dimensions, 16 for periodicity features and 12 for MFCCs. Likelihood method is selected because it provides to examine joint characteristics of the combined features. Only "a" and "yo" phonemes' results are shown below, because MFCC performances for "e" and "m" phonemes are 100%. Table 3-21 and Table 3-23 show the performance of MFCC features for "a" and "yo" phonemes respectively and Table 3-22 and Table 3-24 show the contribution of the periodicity features to MFCC for "a" and "yo" phonemes.

Table 3-21 Performance of the MFCC Features for "a" phoneme

Train / Test	07	08	10	63	64	66	123	124	126	181
07	87.5	8.33	0	0	0	0	4.17	0	0	0
08	0	90	0	0	0	0	3.33	0	3.33	3.33
10	0	3.45	96.55	0	0	0	0	0	0	0
63	0	3.33	0	86.67	3.33	0	0	0	0	6.67
64	0	0	0	0	**100**	0	0	0	0	0
66	0	0	0	0	0	**100**	0	0	0	0
123	0	0	0	0	0	0	**100**	0	0	0
124	0	0	0	0	0	0	0	**100**	0	0
126	0	4.17	0	0	0	0	0	0	95.83	0
181	0	0	0	0	0	0	0	0	0	**100**

Table 3-22 Performance of the Periodicity & MFCC Features for "a" phoneme

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**100**	0	0	0	0	0	0	0	0	0
08	0	**100**	0	0	0	0	0	0	0	0
10	0	0	**100**	0	0	0	0	0	0	0
63	0	0	0	**100**	0	0	0	0	0	0
64	0	0	0	0	**100**	0	0	0	0	0
66	0	0	0	0	0	**100**	0	0	0	0
123	0	0	0	0	0	0	**100**	0	0	0
124	0	0	0	0	0	0	0	**100**	0	0
126	0	0	0	0	0	0	0	0	**100**	0
181	0	0	0	0	0	0	0	0	0	**100**

Table 3-21 shows that MFCC features recognized all of the 10 persons. "**blue**" values show the performance of the MFCC features that are smaller than 100%.

Table 3-22 shows that the combination of periodicity and MFCC features increased the performance of the MFCC features alone. In this case all of the 10 persons are recognized with 100%

Table 3-23 Performance of the MFCC Features for "yo" phoneme

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**100**	0	0	0	0	0	0	0	0	0
08	0	**100**	0	0	0	0	0	0	0	0
10	0	0	**100**	0	0	0	0	0	0	0
63	0	0	0	**100**	0	0	0	0	0	0
64	0	0	0	0	**100**	0	0	0	0	0
66	0	0	0	0	0	**100**	0	0	0	0
123	0	0	0	0	0	0	**100**	0	0	0
124	0	0	0	0	0	0	0	**100**	0	0
126	0	0	4.17	0	0	0	0	0	**95.83**	0
181	0	0	0	0	0	0	0	0	0	**100**

Table 3-24 Performance of the Periodicity & MFCC Features for "yo" phoneme

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**100**	0	0	0	0	0	0	0	0	0
08	0	**100**	0	0	0	0	0	0	0	0
10	0	0	**100**	0	0	0	0	0	0	0
63	0	0	0	**100**	0	0	0	0	0	0
64	0	0	0	0	**100**	0	0	0	0	0
66	0	0	0	0	0	**100**	0	0	0	0
123	0	0	0	0	0	0	**100**	0	0	0
124	0	0	0	0	0	0	0	**100**	0	0
126	0	0	0	0	0	0	0	0	**100**	0
181	0	0	0	0	0	0	0	0	0	**100**

Table 3-23 and Table 3-24 show the improvement of the combination of periodicity and MFCC features. Recognition performance for person "126" is 95.83% with MFCC features alone and this value increased to 100% by combining the periodicity features and MFCCs.

3.3 Combination of Periodicity & MFCC for Text-Independent Case

Results in section 3.1 and 3.2 show the periodicity features' speaker identification performance and their contribution to MFFCs for selected phonemes. These results encouraged us to investigate the effect of periodicity features on MFFCs in whole sentences. This situation is closer to the real life situations. Steps of this test are explained below.

3.3.1 Scenario

As a first step, all periodicity and MFCC features are extracted and combined to get 28 dimensional feature vectors. MFCC features are extracted by HTK as explained in section 2.9.

The database has 120 sentences for each speaker. 100 sentences of this database are used for training and 20 sentences are used for testing for each person (one sentence is approximately 2.5 seconds). 20 sentences for test are shifted 4 times and training has repeated 4 times for each person. 4 testing group, each has 20 sentences, provided to take 80 tests for each person. In the training phase HTK is used. HTK uses Gaussian Mixture Models during training.

3.3.2 Gaussian Mixture Models [15]

Gaussian mixture model (GMM) is a sophisticated statistical model, which can be viewed as a universal estimator. GMM has been applied to speaker recognition to model speaker's characteristics.

Using a GMM model, for speaker identification, a group of S : $1,2,\ldots..,S$ speakers can be represented by their unique model parameters λ_1, λ_2 , ... λ_s (see Figure 3-1). Identifier of each speaker's λ can be represented as a combination of three parameters:

p_i : mixture weights for $i = 1,..M$ where M is the number of component densities,

$\overline{\mu_i}$: mean vector with D - dimensional normal distribution

Σ_i : covariance matrix.

Collectively λ is represented as $\lambda = \{p_i, \bar{\mu}_i, \Sigma_i\}$ for $i = 1, .. M$.

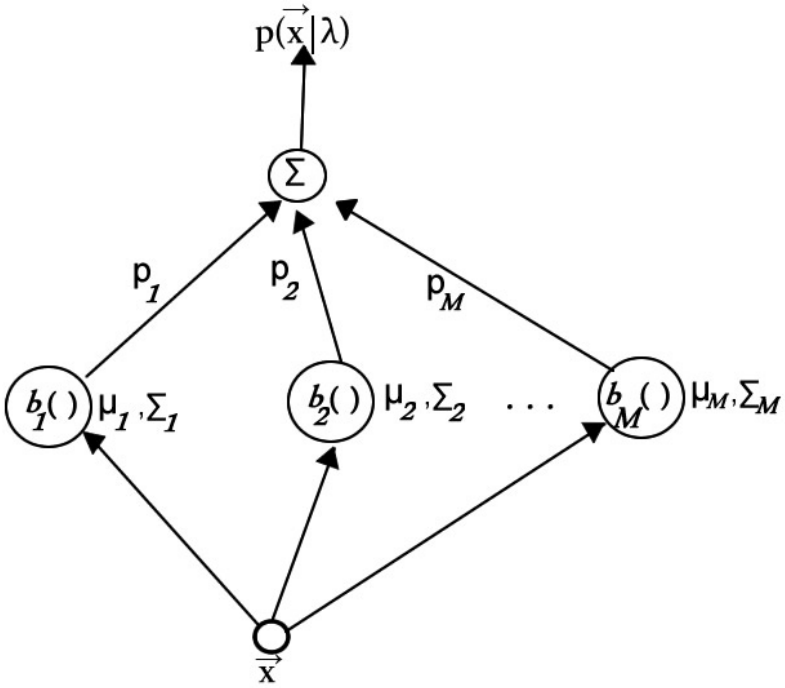

Figure 3-1 M Component Gaussian Mixture Density [15]

GMM training is done by expectation-maximization (EM) algorithms. There are two critical points during training.

First one is the *Variance Limiting:* when training a nodal variance GMM (one covariance matrix per Gaussian component), it has been observed that variance elements can become quite small in magnitude. These small variances produce a singularity in the model's likelihood function and can degrade identification performance.

Second critical point is the *Model Order:* determining the number of components "M" in a mixture to model a speaker is an important and difficult problem. There is no theoretical way to estimate the number of mixture components a *priori*. Choosing few mixture components can produce a speaker model which does not accurately model the distinguishing characteristics of a speaker's distribution. Choosing many components can reduce the performance when there are a large number of model parameters relative to the available training data and too many parameters can also result excessive computational complexity both in training and classification. [15]

Training GMMs has been done using HTK. Several options have been tried for different variables to get trustworthy results. Unvoiced parts of the speech signal are dismissed so training and testing is done only with voiced signal. Different mixture numbers are tried because of the reasons explained above. The best results obtained with 40 mixtures. Test results using only MFCC features is given in Table 3-25 and the results for periodicity and MFCC combined features are shown in Table 3-26.

Table 3-25 Performance of the MFCC Features

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**53.75**	1.25	0	1.25	6.25	0	11.25	26.25	0	0
08	2.5	**61.25**	0	0	0	2.5	0	32.5	0	1.25
10	5	0	27.5	0	16.25	2.5	3.75	**45**	0	0
63	0	0	0	**51.25**	8.75	2.5	3.75	33.75	0	0
64	0	0	0	0	**95**	0	1.25	3.75	0	0
66	0	0	0	1.25	1.25	**76.25**	7.5	12.5	0	1.25
123	0	0	0	0	17.5	0	**56.25**	26.25	0	0
124	0	0	0	0	2.5	2.5	11.25	**83.75**	0	0
126	1.25	20	0	2.5	12.5	0	16.25	**26.25**	20	1.25
181	2.5	2.5	0	3.75	0	0	2.5	33.75	1.25	**53.75**

Table 3-26 Performance of the Periodicity & MFCC Features

Train / Test	07	08	10	63	64	66	123	124	126	181
07	**66.25**	0	0	0	2.5	18.75	2.5	8.75	0	1.25
08	0	**78.75**	0	0	0	5	1.25	15	0	0
10	5	0	**41.25**	0	30	12.5	0	11.25	0	0
63	0	0	0	**55**	8.75	18.75	2.5	15	0	0
64	0	0	0	0	78.75	12.5	6.25	2.5	0	0
66	0	0	0	0	1.25	**90**	2.5	3.75	0	2.5
123	0	0	0	0	3.75	8.75	**77.5**	10	0	0
124	0	0	0	0	0	10	3.75	**86.25**	0	0
126	0	11.25	0	1.25	10	8.75	10	8.75	**50**	0
181	0	0	0	1.25	0	8.75	6.25	6.25	0	**77.5**

Table 3-25 shows the performance of the MFCC features alone and it is obvious that person “10” and “126” could not be recognized using MFCC features with our test and training data.

Combining periodicity and MFCC features provided recognition for each person and recognition performance increased except for “64” as shown in Table 3-26. Periodicity provided average of 12.25% improvement for 40 Gaussian mixtures.

Lower model orders decreased the performance of the system because of the inadequate modeling. Higher model orders decreased the contribution of periodicity features because of the insufficient training data. Average improvement of periodicity features with 50 Gaussian mixtures is 5.59% which is lower than the 40 mixtures case.

CHAPTER 4

CONCLUSIONS

In this thesis, significance of the periodicity and aperiodicity information of speech signal is analyzed in speaker identification problems. In Chapter 1, a brief introduction is given to the characteristics of speech and the steps of the speaker recognition algorithm are explained. We also gave some examples of the features used in the speaker recognition applications. In Chapter 2, the constructed system to obtain the periodicity and aperiodicity information from the speech signal is explained. Obtained periodicity and aperiodicity information is tested on the specific phonemes for 10 females. Parzen window density estimation method is used for modeling the test frames and the training frames. The shortcoming of the Parzen window density estimation is the computational difficulty of the estimation in higher dimensions. For the 16-dimensional periodicity and aperiodicity features, calculation of the joint density estimation could not be possible with parzen window estimate, so each channel's periodicity and aperiodicity features are modeled independently. Finally, testing methods, Kullback-Leibler Distance method and Maximum Likelihood method are given in Chapter 2. In Chapter 3, firstly the results of the experiments for the periodicity and aperiodicity features' performance for 10 females on the specific phonemes are given. Results show that periodicity and aperiodicity information make contributions to the speaker identification problem. For the specific phonemes, periodicity features are also combined with the 12-dimensional MFCC features and the contribution of the periodicity features to the MFFC features are tested. A

significant improvement is observed by using periodicity features as supplementary features to MFCCs. Chapter 3 also explains the experiments for the text independent case.

Training for the text independent case is performed by HTK Tool and HTK represents distributions by Gaussian Mixture Densities. As mentioned before, model order is a very critical variable during GMM training and it depends on the available training data set. One of the shortcomings of our experiments for text independent case is that the insufficiency of the training data set. Also we had to reduce the data set by excluding the silent frames because of the variance limiting problem. A larger training set supplies the increment of Gaussian mixture number and the models could represent the speakers better. More accurate test results for the contribution of the periodicity features to MFCCs could be taken with a larger training data set.

Another shortcoming of our experiments for text independent case is that the variance limiting problem. Periodicity features have very small variances up to 10^{-6} which may produce a singularity in the model's likelihood function and degrade the identification performance. To overcome this shortcoming, a floor variance can be assigned for the HTK algorithm. However, this situation may affect the structure of the periodicity features and may cause degradation on the identification performance.

As mentioned in the thesis, a lower feature dimension is desirable in speaker recognition. The demand for a large amount of training data to represent a speaker's voice characteristics grows exponentially with the dimension of the feature space. We had 16 dimensional feature vectors for periodicity and aperiodicity information separately. As a future work, feature dimensions would be reduced by eliminating the features which are making less contribution to speaker identification.

As another future work, aperiodicity features can be used supplementary with other features. Performance of the combination of the periodicity and aperiodicity features, with a lower feature dimension, can also be investigated independently and this information could be used with different features.

REFERENCES

[1] Douglas A. Reynolds, "Automatic Speaker Recognition, Acoustics and Beyond" *presentation, CLSP Workshop 2002, July 2002*,from http://www.clsp.jhu.edu/ws2002/preworkshop/reynolds.pdf

[2] Joseph P. Campbell, "Speaker Recognition : A Tutorial", *Proceedings of the IEEE*, vol. 85, No. 9, pp. 1437-1462 September 1997

[3] Douglas A. Reynolds, "*An Overview of Automatic Speaker Recognition Technology*" In ICASSP 2002, Orlando, USA.

[4] Judith A. Markowitz, J. Markowitz Consultants, "Speaker Recognition", *Information Security Technology Report*, Vol. 3, No.1 pp. 14-20, 1998

[5] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall Signal Processing Series. Prentice Hall, Englewood Cliffs, NJ, 1993.

[6] http://www.speaker-recognition.org January 1, 2008

[7] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S.Meignier, T. Merlin, J. Ortega-Garcia, D. Petrovska-Delacretaz, and D. A. Reynolds, "A Tutorial on Text-Independent Speaker Verification", *EURASIP J. on Applied Signal Processing* 2004:4, 430–451.

[8] Pravinkumar Premakanthan and Wasfy. B. Mikhael, “Speaker Verification/Recognition and the Importance of Effective Feature Extraction: A Review”, *IEEE Midwest Symposium on Circuits and Systems*, 2001,Ohio. Page(s): 57 - 61 vol.1

[9] Jayant M. Naik, “Speaker Verification: A Tutorial”, *IEEE Communications Magazine*, January 1990, pp 42-48

[10] Nengheng Zheng and P.C. Ching, Using Haar Transformed Vocal Source Information For Automatic Speaker Recognition, *ICASSP 2004 IEEE*, pp77-80

[11] Andre G. Adami, Radu Mihaescu, Douglas A. Reynolds, John J. Godfrey, Modeling Prosodic Dynamics For Speaker Recognition, *ICSLP 1998*, Sydney, pp.3189-3192

[12] Hassan Ezzaidi, Jean Rouat and Douglas O’Shaughnessy, Combining Pitch and MFCC for speaker recognition systems*, In The Speaker Recognition Workshop*, pp. 207-212, June 2001, Crete, Greece.

[13] Om Deshmukh, Carol Y. Espy-Wilson, Ariel Salomon, and Jawahar Singh, Use of Temporal Information: Detection of Periodicity, Aperiodicity, and Pitch in Speech, *IEEE Transactions On Speech And Audio Processing*, 2005

[14] Wikipedia-The Free Encyclopedia, Mel Scale http://en.wikipedia.org/wiki/Mel_scale November 17 2007

[15] D. A. Reynolds and R. C. Rose, “Robust Text - Independent Speaker Identification Using Gaussian Mixture Speaker Models”, *IEEE Trans. on Speech and Audio Processing*, vol.3, No.1, pp.72-83,January 1995

[16] K. Sonmez, E. Shriberg, L. Heck, and M. Weintraub, “Modeling Dynamic Prosodic Variation for Speaker Verification,” *In Proc. of ICSLP*, Vol. 7, pp. 3189-3192, 1998.

[17] Hakan Altınçay, Cem Ergün, Tolga Çiloğlu "Fusion of MFCC and MPEG-7 Attributes for Speaker Verification", *In Proceedings of SIU'2006 National Conference on Signal Processing and its Applications*, 17-19 April 2006, Antalya, Turkey

[18] Carol Y. Espy-Wilson, Sandeep Manocha and Srikanth Vishnubhotla, "A New Set of Features for Text-Independent Speaker Identification" *Interspeech 2006*, 17-21 September.

[19] Alain de Chevengne and Hideki Kawahara, "YIN, a fundamental frequency estimator for speech and music", *The Journal of the Acoustical Society of America 111*, pp 1917-1930, 2002

[20] Praat: doing phonetics by computer http://www.fon.hum.uva.nl/praat/ , December 15, 2007

[21] Audacity The Free, Cross-Platform Sound Editor http://audacity.sourceforge.net/ , December 17, 2007

[22] Melih Günay, Representation Of Covariance Matrices In Track Fusion Problems, A Thesis Submitted To The Graduate School Of Natural And Applied Sciences Of Middle East Technical University, November 2007

[23] EasyFit :: Distribution Fitting Made Easy http://www.mathwave.com/easyfit-distribution-fitting.html, April 02, 2008

[24] B. Smolka, K.N. Plataniotis, R. Lukac, A.N. Venetsanopoulos, "New Class of Impulsive Noise Reduction Filters Based on Kernel Density", *Proceedings of the 28th IEEE International Conference on Acoustic, Speech & Signal Processing ICASSP 2003 in Hong-Kong, Hong-Kong, April 6-10, 2003*

[25] HTK Speech Recognition Toolkit, http://htk.eng.cam.ac.uk, April 10, 2008