ADDIS A BABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

FACULTY OF INFORMATICS

DEPARTMENT OF INFORMATION SCIENCE

CONCATENATIVE TEXT-TO-SPEECH (TTS) SYNTHESIS FOR THE AMHARIC LANGUAGE

A Thesis Submitted to the School of Graduate Studies of Addis Ababa University in Partial Fulfillment of the Requirements for The Degree of Masters of Science in Information Science

BY

HENOCK LULSEGED

JUNE 2003

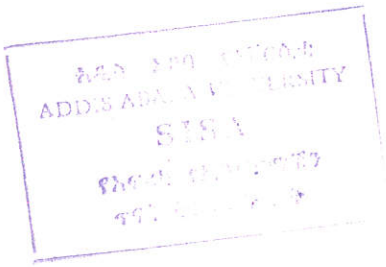
ADDIS ABABA UNIVERSITY

SCHOOL OF GRADUATE STUDIES

FACULTY OF INFORMATICS

DEPARTMENT OF INFORMATION SCIENCE

CONCATENATIVE TEXT-TO-SPEECH (TTS) SYNTHESIS FOR THE AMHARIC LANGUAGE



BY

HENOCK LULSEGED

Signature of the Board of Examiners for Approval

_____	_____
_____	_____
_____	_____
_____	_____

DEDICATION

To my mom, who has always been there for me

ACKNOWLEDGEMENT

First and most of all, I am very much indebted to my advisors, Ato Solomon Berhanu, Ato Ermias Abebe, and Ato Kinfe Taddese, who have been giving me their constructive suggestions and helping me in reviewing this text.

My special gratitude goes to my best friends Fetahi Zebenigus and Eskindir Ayallew who had been helping me out with printing facilities. I would also like to thank all my friends at the faculty of technology for their support and encouragement

Last but not least, I would like to thank my family members and my friends at SISA for their support and cooperation in every possible way.

TABLE OF CONTENTS

LIST OF FIGURES, TABLES & APPENDICES

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

ABSTRACT

In this study, the potential of developing an Amharic TTS system using the TD-PSOLA algorithm has been investigated. In doing this thesis work, the Delphi programming language and the MATLAB software have been used. Additionally, a spectrographic analysis tool called Praat had been used for the purpose of data preparation.

All the acoustic speech units have been extracted from a corpus recorded at a sampling rate of 11,025, and the whole of the corpus had been recorded at one time. Two acoustic unit types have been extracted from the corpus data: diphones and CV-Syllables. CV-Syllables are suitable for the Amharic language because most of the symbols in the Amharic writing system represent a CV-Syllable, and this makes tasks like grapheme-to-phoneme transcription easy. Due to time constraints only a limited number of CV-Syllables and diphones have been extracted from the corpus.

Testing performance of TTS systems is one of the difficult tasks because there is no single measure to pinpoint the quality of the system. Although no standard test is available, a number of testing methods have been developed. The Open Rhyme Test (ORT) and Mean Opinion Score (MOS) test have been used in this work to test performance.

The results obtained from the experiment are promising and indicative of the possibility of producing high quality TTS system for Amharic using other advanced algorithms than the one used in this work.

CHAPTER ONE

INTRODUCTION

1.1. BACKGROUND

The advent of microcomputers with fast processing speeds, succeeded by the present information and communication technology has declared the start of the information age and resulted in major advancements in fulfilling the ever unsatisfied information need of mankind. Using the currently available technology, mankind can reach to a pile of useful information anywhere in the world as if it is at the fingertips. The influence of geographical distance on information sharing has become nearly history because anyone sitting at a terminal can reach to any information source or any other person with a few clicks of mouse buttons.

Although the advancements have shown significant results, the increasing demand of information is adding the burden on currently available technologies to provide universal access. Users do not know (or do not need to know) what goes on behind the curtain when they access information. They only interact with the front-end interface.

Front-end interfaces abstract processes and unnecessary details from the user, thereby making the task of users fairly simple. The development of user friendly interfaces has been and still is one of the many problem areas of research in the world. One approach to address this problem is to impart human like capabilities onto machines, so that they can speak and hear, just like the users with whom they interact (Lemmetty, 1999).

The question of having human like capabilities imparted on machines is not only limited to addressing the user friendliness of interfaces, but also takes into consideration the information need of handicapped people. One of such human like capabilities is the ability to speak (Ibid).

The idea of synthesizing speech artificially in machines has been with human beings for quite some time. But the realization of such machines has been practical within the last 50 years. Even more recently, it is in the last 20 years or so that practical examples had been observed (Black & Lenzo, 2003).

Nevertheless, the task of making an application speak or understand its users' commands – which has been regarded as science fiction once upon a time – could be accomplished with relative ease and good performance with today's computers and speech technology (Long, 2001).

1.1.1. OVERVIEW OF SPEECH SYNTHESIS

Speech is the usual and common mode of communication between human beings. When a person hears speech in his/her own language, the person hears the individual words and sounds. This can be easily proven in that speech can be written using discrete letters and spaces between words. But this is not true if the person hearing the speech is not familiar with the language. In this case all the words and sounds seem to run together in a continuous stream (Rodman, 1999).

This creates a debate on whether speech is discrete or continuous. According to Rodman (1999), speech is both discrete and continuous. This is from two different points of view. On a physical level, speech is continuous (except where the speaker pauses to take a breath). But from psychological level, speech can be considered as discrete because it is perceived as composed of discrete sounds (Ibid).

1.1.2. OVERVIEW OF TEXT-TO-SPEECH (TTS) SYSTEMS

Text-To-Speech (TTS) systems are well known for contributing much in the man-machine communication environment. A TTS system takes a human readable text as an input and correspondingly delivers speech utterances as an output. A TTS system is not merely expected to play back prerecorded speech sounds, as in the case of voice response systems (Dutoit, 1997).

Voice response systems are systems that produce artificial speech by simply concatenating isolated words or other bigger units like phrases. Such systems are often recommended when a limited vocabulary is required and the sentences to be pronounced have a common restricted structure, as in the case of some announcement systems in train stations (Ibid).

But this is not the case with TTS systems. In a sense, TTS systems are able to produce new sentences; not only prerecorded words and phrases. Besides, there are a number of tasks (such as Natural language processing and signal processing) involved in TTS systems than mere concatenation of words and phrases (Dutoit, 1997).

The speech uttered by a TTS system is expected to be intelligible and natural (Minghui, 2000). Both intelligibility and naturalness depend on two factors: *segmental quality* and *suprasegmental quality*. Segmental quality refers to the ability of the machine to produce natural sounding speech utterances given the required high-quality information. On the other hand, suprasegmental quality refers to the richness of prosodic features that the machine is capable of exploiting (Dutoit, 1997).

Although the quality of the speech synthesized by the currently available TTS systems is adequate enough to apply it in applications such as multimedia and telecommunications, it is not

1.2. STATEMENT OF THE PROBLEM AND JUSTIFICATION OF THE STUDY

TTS systems are needed rather for universal set of problems. Some of these problems are mentioned below.

- ***Aid to people with disabilities***: -Probably the most important field of application of a TTS system is its aid in the communication of visually disabled people. TTS systems can be of invaluable support with this respect in that it somehow addresses the visually impaired people's need to get as much information as there exists in written texts (Dutoit, 1997).

  People who have hearing difficulties often have a difficulty to speak too. Therefore, synthesized speech gives such people the ability to communicate with people who do not understand the sign language (Lemmetty, 1999).

- ***Educational purposes***: -The application of TTS systems may further go up to educational level. Computers with speech synthesizer can teach 24 hours a day and 365 days a year. Especially those interactive educational applications need a TTS system. A high quality TTS system can be programmed for special tasks like spelling and pronunciation teaching of various languages (Dutoit, 1997).

- ***Telecommunication services***: - TTS systems can also be used in communications and multimedia areas. With the present day's quality of synthesized speech, it is possible to use TTS systems in telephone inquiry systems. Obviously electronic mail (e-mail) has become very popular over the past few years. Sometimes, however, it would be impossible to read e-mail messages if there is no proper computer available. In such cases, TTS systems can help to read e-mail messages via plain telephone lines (Lemmetty, 1999).

- ***Man-machine communication***: - Generally, TTS systems can be used in any kind of human-machine interactions. Warning alarm systems, for instance, can be used to give more accurate information of the situation rather than the warning lights and buzzers which give plain alert signals (Lemmetty, 1999).

- ***Fundamental and applied research***: - TTS synthesizer can serve as laboratory tools for linguists. Since they are completely under control, repeated experiments would definitely provide identical results. This makes TTS systems a good candidate to investigate issues like the efficiency of prosodic models. But this is not the case with human beings (Dutoit, 1997).

The area of TTS synthesis is very wide and has got many applications. With the aid of TTS systems a number of drastic changes can be attained in the way we live. Keeping the invaluable support that TTS systems give to disabled people on one side, creating an easy environment to interact with machines allows the smooth running of life. Using the coupled effort of TTS synthesis and speech recognition, an easy, two-way kind of communication between man and machines could be established.

A number of TTS systems have been developed so far. Some of them are flexible in that they support more than one language (multilingual TTS systems). Others support only one specific language. Systems like *MBrola, Klatt, Festival, HTK, IPOX, rsynth* and *PlainTalk* are some of the famous speech synthesizers and TTS systems that could be cited as an example.

Amharic is one of the widely spoken languages in Ethiopia. The amount of digital data in the Amharic language is growing everyday. There are a lot of digital (electronic) documents in the Amharic language that deliver a great deal of information to users. News items are prepared in Amharic and posted on the web for users. Even writing an e-mail message using Amharic

symbols has been possible these days. Along with such advancements, the possibility of incorporating accessories like Amharic text or e-mail readers should be given attention.

But there had not been many researches done on the development of a TTS system for the Amharic language. Laine (1998) had attempted to develop a TTS system for the Amharic language using the linear predictive coding (LPC) method. In his work, Laine used diphones as speech units, and his work was up to sentence level.

However, Laine did not take prosodic effects into account. Besides he failed to consider other techniques like the Pitch Synchronous Overlap ADD (PSOLA), which give much better quality than the LPC technique. Nevertheless, he has recommended the possibility of considering prosodic effects using other methods than LPC

In the light of these facts, the thesis work is a mere attempt to discover the potential of developing concatenative type of TTS synthesis for the Amharic language, by taking prosodic effects into consideration. Two acoustic units are to be considered in this work. These are diphones and syllables. The performance of these two will be compared and contrasted based on test results.

1.3. OBJECTIVES OF THE STUDY

1.3.1. GENERAL OBJECTIVES

The general objective of this study is to investigate the possibility of developing a prototype of concatenative TTS system for the Amharic language, taking prosodic effects into consideration; and then test the result on two speech unit types (diphones and syllables).

1.3.2. SPECIFIC OBJECTIVES

The specific objectives of this research are:

- To review related literature on the concept of TTS systems.
- To review related literature on different Natural Language Processing (NLP) and Digital Signal Processing (DSP) techniques used in TTS systems.
- To review literature on the phonology of the Amharic language and compile a list of diphones and syllables.
- To design and build a prototype and test it.
- To report the results obtained and forward conclusions and recommendations.

1.4. METHODS

1.4.1. LITERATURE REVIEW

Various related literature in the area of TTS systems had been reviewed. Especially, literature in areas of *Natural Language Processing (NLS)* and *Digital Signal Processing (DSP),* as well as different speech synthesis techniques have been reviewed.

1.4.2. DATA COLLECTION AND PREPARATION

Data for concatenative speech synthesis is a set of speech segments. These speech segments are obtained from prerecorded data corpus. Therefore the data collection procedure consists of two major tasks. One is the recording of the speech corpus. The other is the extraction of the desired speech segments (diphones and CV-syllables in this case) from the corpus.

Before extracting the acoustic units, though, the amplitude (energy) of the recorded corpus should be normalized using equalization algorithm. Such variations come from inconsistence in speaker's voice. If left without any correction, the variation in energy between the corpus members may result in quality degradation in the final output.

A speech analysis tool called Praat has been used to record the corpus, normalize the corpus, and extract the acoustic units. This software is developed by Paul Boersma and David Weenink for the purpose of doing phonetics by computer, and is freely distributed. It incorporates a number of functions, such as spectral analysis, sound recording, sound segmentation, amplitude equalization, segment concatenation, and many more.

1.4.3. TECHNIQUES AND PROTOTYPE DEVELOPMENT TOOLS

The TTS system is developed using concatenative synthesis method. PSOLA (Pitch Synchronous Overlap Add) method is used in the synthesis part. This method is preferred because it gives a good quality of synthesized speech. The PSOLA method is also good at handling intonation and duration changes because it enables the direct modification of pitch and duration of the synthesized speech (Dutoit, 1997).

The Borland Delphi programming language has been used to develop the prototype TTS system. This development environment is chosen because of its familiarity to the conductor of the research and because of its simplicity in developing user interfaces. Apart from Delphi, the MATLAB (Matrix Laboratory) software is used, especially to handle the signal processing part.

1.4.4. PERFORMANCE EVALUATION

Due to the complexity of speech assessment problem and rather vague notions of intelligibility and naturalness, no clear standard method of performance testing has been developed yet.

Nevertheless, a number of tests are carried out to test the performance of TTS systems. To mention some, the *Diagnostic Rhyme Test (DRT)*, the *Modified Rhyme Test (MRT)* and *Cluster Identification tests (consonant-vowel-consonant (CVC) or vowel-consonant-vowel (VCV))* for phoneme-level intelligibility; the *Semantically Unpredictable Sentence (SUS) test* for sentence-level intelligibility; and *Paired Comparison* and the *Mean Opinion Score (MOS)* test for both intelligibility and naturalness are some of the performance testing methods (Dutoit, 1997).

MRT and DRT basically follow the same principle. According to these testing schemes a listener is allowed to hear a sequence of isolated words and is made to select what he/she heard from a list of rhyming alternatives. The number of alternatives is usually six (Donovan, 1996). The alternatives differ by their initial consonant in DRT and by their initial or final consonants in the MRT (Dutoit, 1997).

The DRT and MRT tests have a weakness in that they test only initial and final phonemic transitions. Besides, the limited number of alternatives tends to propel listeners to modify their perception. But, the two tests also have an advantage in that they are easy to conduct because each of them requires only a less number of listeners (Dutoit, 1997).

It is also possible to modify these tests to have an *Open Rhyme Test (ORT)*. In this case, the user will not be given any alternative words. Rather he/she will be plainly asked of what he/she heard (Donovan, 1996).

The cluster identification test evaluates the intelligibility of sequences of one or more consonants (consonant clusters) and sequences of one or more vowels (vowel clusters). Most of the test words generated here are meaningless. An open response is expected from the listener, stating what he/she has heard (Dutoit, 1997).

The SUS test evaluates the sentence-level intelligibility of a TTS system. This test involves semantically unpredictable sentences. In this test, a number of sentences are generated for different sentence structures, without worrying about the semantics, and listeners are simply asked to write down what they hear. The sentences are generated by filling the raw sentence structures, filled by random selection of words out of predefined lists of possible candidates (Dutoit, 1997).

Testing the naturalness and overall quality of speech synthesized by a TTS system is one of the difficult tasks in TTS synthesis. One possibility to testing the naturalness of synthesized speech could be to present pairs of sentences each synthesized by two different systems and ask the listeners preference. The other alternative is to ask listeners describe their impression of the quality of the synthesized speech in terms of, say, labels ranging from 'unsatisfactory' to 'excellent' (Donovan, 1996).

In this thesis work, the latter (which is also called the MOS test) has been used to test the overall quality and intelligibility of speech synthesized by the prototype. This test rates a system for the sentence-level intelligibility and overall quality based on the opinion of listeners. Listeners are allowed to listen to a number of test sentences and will be asked to give their opinion about the overall quality of the speech.

The number of acoustic units extracted from the corpus is limited because it takes a lot of time to record a fully representative corpus and extract all the available CV-Syllables and diphones. According to the MRT and DRT tests, the user should be given alternative words that differ from the uttered one by the first or last consonant. Finding such alternatives, while being limited by the

number of acoustic units, is difficult and time taking. Hence, the ORT test had been preferred to MRT and DRT to test phoneme-based intelligibility in this thesis work.

1.5. SCOPE AND LIMITATIONS OF THE STUDY

First and most of all, the prototype to be developed considers and works for only the Amharic language (i.e. it is not multilingual). Additionally, only general intonation rules for two statement types (yes/no interrogative type statements and simple statements) are considered because building prosodic rules for all cases is another research by itself and needs the compilation of a rule dictionary. In general, Interrogative statements are uttered with rising intonation; and the reverse is true for simple statement.

One of the limitations in this thesis work is the unavailability of already made corpus data. Due to this, only a limited number of diphones and syllables are extracted from a limited number of recorded speech utterances. Later in the testing phase, only those sentences or words for which all the diphones or syllables are available could be uttered.

The unavailability of an already made data also limits the TTS system to use the voice of one speaker only, because it takes a longer time to consider the incorporation of other alternative speaker voices.

The unavailability of well suited, noise free lab to record the corpus data is the other limitation. The environmental condition of the recording lab has got its own impact on the quality of the final speech later at synthesis time. Thus, external noise incorporated into the data during the corpus recording session may result in the quality degradation of the final synthesized speech.

The other obvious constraint is time. Had it not been for the limited time available, it would have been possible to record more utterances and extract a relatively larger number of acoustic units. Besides, it would have been possible to incorporate more than one speaker voice and more intonation rules.

One other limitation that had been sucking away the already shortened time was the problem of power interruption that occurs every two days in a week in Addis Ababa town. This problem has inflicted its own impact on the timely submission of this thesis work.

1.6. ORGANIZATION OF THE THESIS

This paper is divided into five chapters. The first chapter gives a brief highlight of speech synthesis and TTS systems. In addition to this, statements of the problem and justification of the study, the general and specific objectives of the research, and the methods employed in doing the research are briefly outlined.

The next chapter (chapter two) presents the historical background of speech synthesis, highlights of the human speech production system, the Amharic sounds, and a brief introduction of acoustic phonetics. Signal analysis and representation as well as some core terms used in speech signal analysis are also introduced briefly in this chapter.

Chapter three discusses about waveform synthesis techniques available in the concatenative speech synthesis. In this chapter, major synthesis techniques are briefly introduced and a special emphasis is given to the PSOLA technique.

Chapter four discusses the experimentation part of the work as a major issue. Every implementation detail of the TTS system, the tests and results obtained are discussed in detail in this chapter.

Finally, Chapter five discusses the conclusions drawn from the results of the experiment and the recommendations.

CHAPTER TWO

FUNDAMENTALS OF SPEECH SYNTHESIS

2.1. INTRODUCTION

The development of TTS systems requires a knowledge of the phonology of the target language, as well as technical knowledge like signal processing. Most often speech synthesis researches are conducted with the collaboration of technical experts and linguists.

There are two basic aspects to speech synthesis. One is the physical process of producing the speech sounds. This is equivalent to making the machine vocalize. The other aspect involves telling the machine what to say, thereby teaching it to read symbols (Rodman, 1999). In this chapter a number of concepts about speech and speech synthesis are discussed.

The chapter starts with the discussion of historical background of speech synthesis. The various attempts made through time to produce speech artificially and the resulting historical advancements had been touched upon briefly. This chapter also tries to discuss about the human articulatory system and basics of the Amharic phonetics. The discussion covers a topic about the different sounds of Amharic, the places of articulation, the seven orders of Amharic symbols and the Amharic writing system.

The other aspect raised in this chapter is about speech signals and their digital representation. Some basic concepts and terms are discussed here. Lastly, basic components of Text-To-Speech systems, synthesis units, and various speech synthesis methods are discussed briefly.

Most of the ideas in sections 2.2 and 2.3.1 had been taken from Rodman (1999). Therefore, unless specified explicitly, all the ideas in the above mentioned sections are acknowledged to be from Rodman (1999).

2.2. HISTORICAL BACKGROUND

Speech science could be believed to start about 2,500 years ago. This conclusion is reached from what the famous Greek physician called Hippocrates had said. Hippocrates wrote in his writings:

> *"The voice is articulated by the lips and the tongue. Man speaks by means of the air which he inhales into his entire body and particularly into the body cavities. When the air is expelled through the empty space it produces a sound, because of the resonances in the skull. The tongue articulates by its strokes; it gathers the air in the throat and pushes it against the palate and the teeth, thereby giving the sound a definite shape. If the tongue would not articulate each time, by means of its strokes, man would not speak clearly and would only be able to produce a few simple sounds"* (Hippocrates, n.d. quoted by Rodman, 1999).

The first talking machine was built by a person called Christian Gottlieb Kratzenstein, at about the same time the United States of America was brought forth as a new nation. Kratzenstein's machine was composed of tube like acoustic resonators – each with its own shape – and each of which produced specific vowel sounds. The specificity resulted from the different shapes of the tubes. A reed forced to vibrate in a stream of air had been used to resonate the acoustic resonators. The machine had managed to accurately express vowel sounds.

Some time later, Wolfgang Von Kempelen succeeded in constructing a more elaborate sort of machine in Vienna. Unlike Kratzenstein's machine, this machine used bellows to produce an air

stream. Constricted passages controlled by the fingers had been used to produce consonant sounds; and a conical, hand controlled leather resonator had been used to produce vowel sounds.

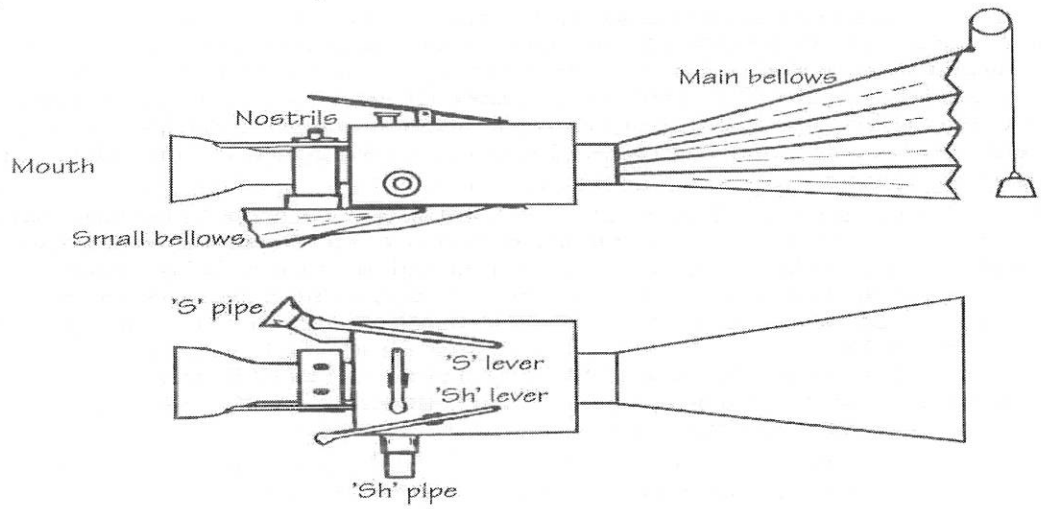


Figure 2.1: Kempelen's talking machine (taken from Dutiot, 1997).

Inspired by Von Kempelen's achievements, Alexander Graham Bell, in collaboration with his brother Melville, built the "talking head." The two brothers simulated the human articulatory system in their talking head. First, they made a cast of the human skull and then they inserted vocal parts made from wire, rubber, cotton and wood into it. The vocal cords as well as the larynx were simulated by rubber bands. Bellows were used to produce air stream, and a system of levers controlled the synthesizer.

A number of people have tried to improve Bell's model. However no one had been successful in showing progress. Actually one dares to say that Bell's model had been waiting for the electronic age to show progress, because no improvements had been made up until an engineer built the first synthesizer that works electrically in 1920. This machine consisted of an oscillator and two electrical resonators, and was able to produce vowels, nasals and a few words.

The *Voder* (later called the *Vocoder*), built in 1939, had been a big hit at its time. The voder records speech in an abridged format and later the recorded speech would be played back with some modifications. The voder needed an operator to manipulate keys to simulate the acoustic parameters of the speech to be synthesized. One problem with the voder was that it was not easy to manipulate and as a result it needed a trained operator.

So far, the synthesizers constructed needed human intervention. In other words, operators needed to get their hands dirty by setting up parameters mechanically through the use of levers and keys - as observed in the case of musical instruments. The construction of a hands-off device that accepts symbolic inputs and returns speech output had not been possible until the 1950s. Later, the invention of integrated circuits led to the possibility of producing high quality electronic speech synthesis systems.

All the attempts in the past have their own marks on the existence of the present day intelligent artificial speech synthesizers. Although the quality of speech synthesized by the currently available systems is adequate enough to be applicable in areas such as multimedia and telecommunications, one dares not say that the quality is high because there still exist threats, such as loss of naturalness, that still need to be addressed (Lemmetty, 1999).

2.3. ARTICULATORY PHONETICS

The science of phonetics studies the speech sounds of the human language. Articulatory phonetics is one branch of the science of phonetics that deals with the physical process of human speech production. The articulators, their position when creating a certain sound, places of articulation, categories of sounds of human speech, and the like are studied in the science of articulatory phonetics (Rodman, 1999).

2.3.1. THE HUMAN SPEECH PRODUCTION SYSTEM

The human speech production system is composed of organs ranging from the diaphragm and the lungs to the vocal and nasal cavities. In between there are a number of components like the tongue, hard palate, velum (soft palate), pharynx, larynx and the vocal folds (vocal cords).

The main energy source in the human speech production system is the lungs with the diaphragm. When speaking, the air from the lungs is forced through the vocal cords and the larynx to the three main cavities of the vocal tract; namely the pharynx, the oral cavity and the nasal cavity. The vocal cords tighten up and vibrate by the air from the lungs if voiced consonants and vowels are to be produced. On the other hand, the vocal cords relax and let the air from the lungs pass smoothly if the sound to be produced is an unvoiced one.

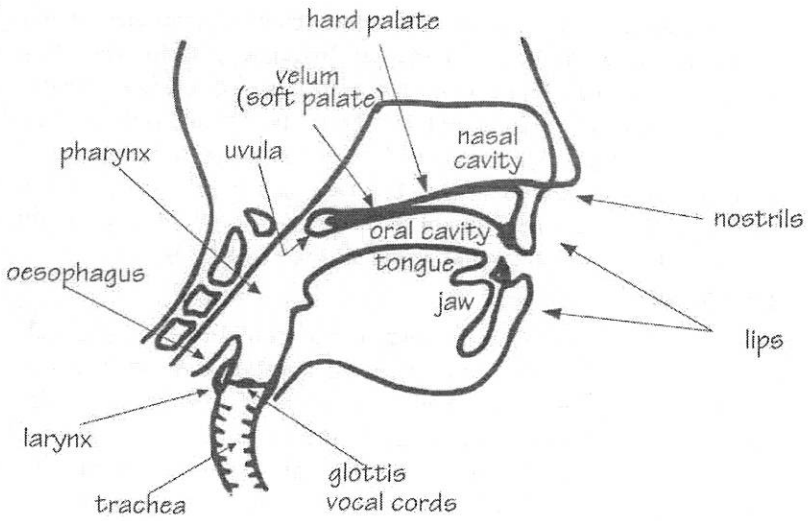


Figure 2.2: The human articulatory system (taken from Dutiot, 1997)

A v-shaped opening between the vocal cords, called the glottis, modulates the air flow by rapidly opening and closing. The rapid opening and closing of the glottis result in the production of different buzzing sounds, each with different fundamental frequencies.

The next station for the air passing out of the lungs is the so called soft palate or velum. The velum serves as a portal to the nasal cavity. When it is open, air can pass through both the oral and nasal cavities. Sounds in which air is allowed to pass through the nasal cavity are called nasal sounds. On the other hand, speech sounds that do not need the passage of air to the nasal cavity are called oral sounds.

Air passing through the nasal cavity won't encounter any further obstacles. But, this is not true for the air passing through the oral cavity. When air passes through the oral cavity, the tongue, the teeth and the lips are potential obstacles. Amongst the three, the tongue plays a chief role in the articulation of most sounds. It moves around the mouth as needed and may block or narrow the air flow passage. The lips close, open, or move against the teeth as necessary.

The vocal cords, the tongue, the velum and the lips are moveable parts of the human vocal tract. They work together to produce speech utterances. On the other hand, the teeth, the alveolar ridge (a small protrusion of the hard palate just behind the upper front teeth), the hard palate, and the velum play passive roles in the human speech production system. Note that the velum (soft palate) plays both active and passive roles.

2.3.2. THE AMHARIC PHONOLOGY

In most languages, two basic types of speech sound are of importance. These are consonants and vowels. Speech itself can be defined as a stream of vowel sounds with interfering consonants for that matter.

Amharic is one of the so called Semitic languages. Semitic languages are thought to descend from the Afro-Asiatic languages. Apart from Amharic, Hebrew, Arabic and Tigrinya are other common Semitic languages (McCarthy, n.d.). Just like any other language, Amharic incorporates

consonant and vowel sounds that make up the language. One thing to note about these sounds is that they are not separately treated because most Amharic symbols represent a combination of one consonant and one vowel.

2.3.2.1. CONSONANT SOUNDS

When producing consonants, articulators come close to each other and form obstructions. Sometimes the articulators even touch with each other. According to Rodman (1999), three major factors determine the sound of a given consonant. These are:

- ⇒ *The place of articulation*: - the location of the obstruction in the vocal tract.
- ⇒ *Manner of Articulation*: - the relative position and activity of articulators while forming the obstruction.
- ⇒ *Voicing*: - the state of the vocal cords.

In Amharic there are about 27 consonant sounds. These consonants are differentiated by the manner and places of articulation, and their voicing.

There could be a number of possibilities regarding the manner of articulation. One possibility is complete blockage. Such an articulation is observed in the articulation of the so called stop consonants (e.g. the initial sounds of the Amharic words '**በላ**' (/bella/) and '**ደረሰ**' (/derese/)). Air could also be blocked only in the mouth but pass through the nasal cavity. Consonants created in this manner are called nasal stops. Sounds represented by '**ም**' (/m/ in English) and '**ን**' (/n/ in English) are good examples of nasal stops.

Another possible manner of articulation is the case where two articulators are very close but not touching. Such an articulation manner creates partial obstruction and the resulting sounds are

called fricatives (e.g. sounds represented by 'ፍ' [equivalent to /f'/ in English] and 'ቭ' [equivalent to /v/ in English]).

The following table summarizes the place of articulation, manner of articulation and voicing of the Amharic consonants.

			ARTICULATION PLACE						
			Bilabial	*Labio-dental*	*Alveolar*	*Palatal*	*Velar*	*Labio-Velar*	*Glottal*
ARTICULATION MANNER	*Stops*	Voiced	ብ /b/		ድ /d/		ግ /g/	ኍ /g^{W}/	
		Voiceless	ፕ /p/		ት /t/		ክ /k/	/k^{W}/	ዕ (?)
		Ejective	ጵ /p'/		ጥ /t'/		ቅ /k'/	/k'^{W}/	
	Fricatives	Voiced		ቭ /v/	ዝ /z/	ዥ /ž/			
		Voiceless		ፍ /f/	ስ /s/	ሽ /š/			H/h/ኍ /h^{W}/
		Plosive			ጽ /s'/				
	Affricates	Voiced				ጅ /j/			
		Voiceless				ች /č/			
		Plosive				ጭ /c'/			
	Nasals			ም /m/	ን /n/	ኝ /ñ/			
	Liquids					ል /l/	P /r/		
	Glides			ው /w/			Ψ /y/		

Table 2.1: The Amharic consonants and their corresponding place and manner of articulation

2.3.2.2. VOWEL SOUNDS

Vowels are created whenever the air stream from the lungs is unobstructed. Besides, vowels are more prolonged than consonants. Vowels can be classified by the position of the tongue and the lips. The tongue and the lips produce different vowels by altering the shape of the vocal tract and

enabling the vibrating air produce sounds in which different frequencies are emphasized (Rodman, 1999).

The Amharic vowels are seven in number. These seven vowels can be categorized based on the position of the tongue. The tongue may be positioned high, mid, or low vertically and front, central, or back horizontally when producing vowels. The lips may also be rounded or unrounded. The following table summarizes the vowel sounds of the Amharic language.

	Front		**Central**		**Back**	
	Rounded	*Unrounded*	*Rounded*	*Unrounded*	*Rounded*	*Unrounded*
High		α | /i/		A /ï/	α /u/	
Mid		α? /ë/		/e/	∑ /o/	
Low				♦ /a/		

Table 2.2: The Amharic vowel sounds

The Amharic language consists of 33 basic symbols and a few other additional symbols. Each of these basic symbols has seven different orders. The orders are created by combination of each consonant with the seven different vowel sounds. The vowels and the consonants cannot be separated apart and treated alone (as stated previously). Thus, most Amharic symbols are consonant-vowel combinations (or CV-Syllables).

Consider the consonant sound ለ (/l/). By combining it with the seven vowels, the seven orders are created as follows.

	Orders						
	1st	*2nd*	*3rd*	*4th*	*5th*	*6th*	*7th*
Symbol	ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ
	/le/	/lu/	/li/	/la/	/lë/	/lï/	/lo/

Table 2.3: An example of seven orders of Amharic

Bender, *et al.* (1976) mentioned two basic limitations regarding Amharic.

- There is no mechanism of distinguishing simple sounds from geminate consonants. In English or some other languages, germination is denoted by double consonants. But this is not the case with Amharic because the consonants are not separable from the vowels. For instance, an Amharic word '**ገና**' can be pronounced as /gena/ or /genna/ depending on the context, and yet there is no indication at the symbolic level.

- Sometimes, only the consonant part of the sixth order needs to be taken. But Amharic lacks the indication of whether to take out the 'ï' sound or not. If the last symbol of a word is in its sixth order, then it can be taken as a rule that the vowel sound 'ï' should be omitted.

  For instance, in the Amharic word (**ስብ**), the last symbol is at its sixth order and only the consonant part of the last symbol is considered. Therefore, the transcription will be /sïb/, not /sïbï/. But the case is not as simple as this always. Sometimes, it would be appropriate to omit the vowel part of a sixth order symbol in the middle of a word, and other times it is not appropriate. Amharic lacks the ability to exactly indicate such situations.

2.4. ACOUSTIC PHONETICS

Contrary to articulatory phonetics, acoustic phonetics studies the physical properties of the sound waves of speech. Aspects of speech signal representation, analysis and processing are studied in the science of acoustic phonetics (Rodman, 1999).

2.4.1. SPEECH SIGNAL ANALYSIS

At the acoustic level, speech is nothing but a combination of different signals whose amplitude is changing with time. Waves of pressure variation oscillate relative to the surrounding medium (usually air), thereby creating sound.

2.4.1.1. SPECTROGRAPHIC ANALYSIS

Speech waveforms can be graphed as pressure variation, the y-axis representing the amplitude and the x-axis representing time. But to determine those phonetic properties that allow listeners to differentiate one sound from another, the different wave signals in the sample speech should be decomposed into individual waves. Then the amplitude and frequency of the individual waves can be examined. Such an approach towards the analysis of speech waveforms is called *Spectrographic analysis* (Ibid).

One thing to observe in spectrographic analysis is the formant frequencies. Formant frequencies, also called formants, are bands of high energy frequencies that occur in vowels. At the formant frequencies, the amplitude of the wave will be relatively higher. For instance, when pronouncing the vowel 'o' frequencies centered about 700 Hz, 1,120 Hz and 2,240 Hz have much higher amplitudes than the other frequencies (Rodman, 1999).

The formant frequencies are named as first formant (F1), second formant (F2), third formant (F3) and so on starting from the lowest one. For the above case, 700 Hz will be F1, 1,120 Hz will be F2 and 2,240 will be F3. The formant frequencies are nothing but harmonics (multiples) of a certain initial frequency called fundamental frequency (F0). For instance, the above given formants of 'o' are the 5^{th}, 8^{th} and 16^{th} harmonics respectively (Ibid).

The formant frequencies cannot be seen on a simple x-y graph. A better visual representation is the spectrogram, which is a plot of frequency on the y-axis versus time on the x-axis. The third dimension (amplitude) is represented by the degree of darkness of the plot. I.e. the darker a certain portion on the graph is, the higher the amplitude at that location; and this indicates that the particular frequency (frequency range) is a formant frequency. Below are the waveform and spectrogram of the vowel sound 'o'.

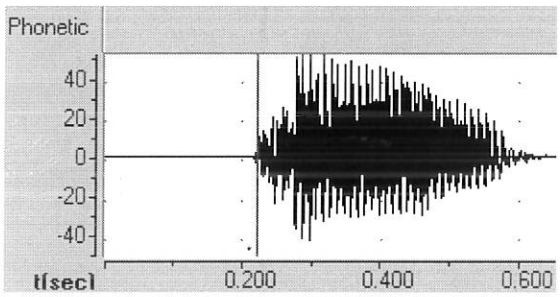


Figure 2.3: Waveform of the vowel sound 'o'

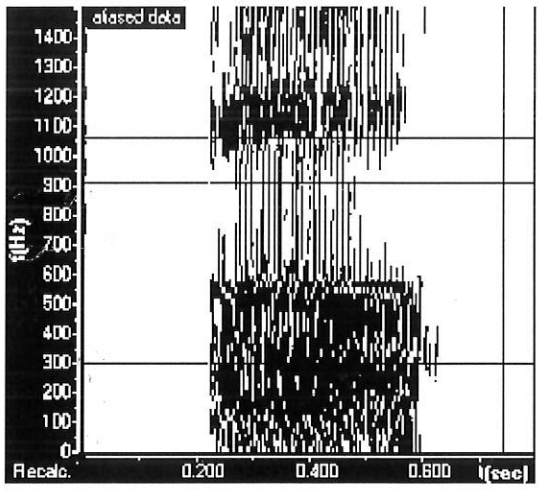


Figure 2.4: Spectrogram of the utterance of the sound 'o'

As can be seen from the figure above, the darker spots indicate the frequencies at which the speech signal will have higher energy (or amplitude) when generating the sound represented by 'o'.

2.4.1.2. CLASSIFICATION OF SOUNDS

Speech sounds are divided into two major types; voiced and unvoiced. Voiced sounds are produced with the vibration of the vocal cords. Voiced sounds are found to be periodic when plotted. Vowels and some consonants (called voiced consonants) are voiced sounds (Minghui, 2000).

Unvoiced sounds, on the other hand, are random signal segments of speech. Unvoiced sounds are uttered without the vibration of the vocal cords. Most consonant sounds are known to be unvoiced. See the figure below.

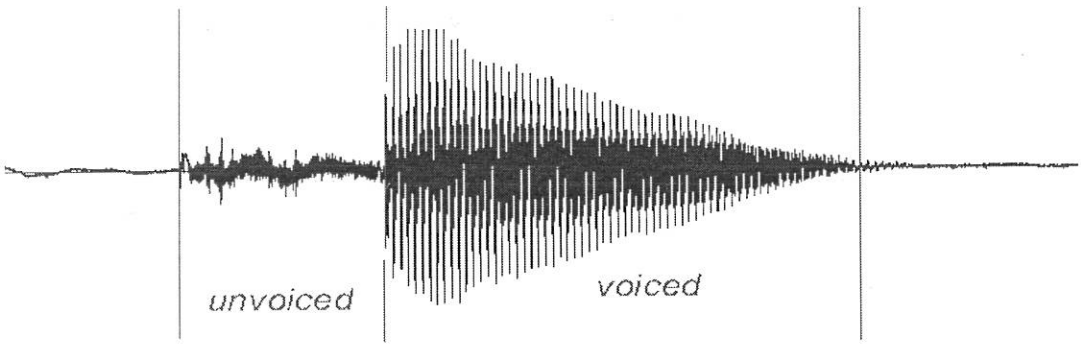


Figure 2.5: Voiced and unvoiced parts of speech

When a speech waveform is analyzed in the frequency domain, many frequency elements are observed. The first formant observed is the fundamental frequency and it is usually denoted as F0. The fundamental frequency is also referred to as pitch. The term pitch applies only to voiced

sounds. Because unvoiced speech has no periodicity, it won't have pitch. Pitch of a speech signal varies with time. The variation of pitch with time is expressed in terms of pitch contour.

While the frequency (or the number of oscillations within a second) determines the pitch of the sound, the size of pressure variations on the other hand determines the loudness (or intensity) of the sound (Minghui, 2000). Frequency is measured in hertz (Hz), which simply refers to the amount of wiggliness of the wave signal within a given time. Intensity or loudness, on the other hand, is measured in decibels (dB). This is a relative intensity, which is defined as 20 times the base 10 logarithm of the ratio of the pressure level of the sound in question, to a certain reference pressure level which usually corresponds to the faintest sound perceptible by normal ears (Rodman, 1999).

2.4.2. REPRESENTATION OF SPEECH IN THE COMPUTER

Sound, by its very nature, is an analog signal. In other words, the signal varies in smooth and continuous manner with time. On the other hand, the computer is digital, representing every data in terms of 0s and 1s. Therefore, in order for sound signals to be stored and processed within a computer, there should be a mechanism to convert the analog speech data into the corresponding digital form.

The microphone (considered as the ear of the computer) is a device which traps sound vibrations and converts them into an electrical signal. It consists of a diaphragm and a component which converts vibrations into electrical signal. Any sound whose frequency is within the range of operation of the microphone vibrates the diaphragm. These oscillations of the diaphragm are then converted to the equivalent electrical signals (Rodman, 1999).

The electrical signal from the microphone is still analog. That means the electrical signal is a smooth voltage variation through time. If one represents the signal on the x-y axis – the x-axis representing time and the y-axis representing the voltage (amplitude), the outcome will be a continuously wiggling line. This analog signal is then converted to the corresponding digital form by an *Analog-to-Digital (A to D)* converter.

Analog signals are continuous in that they have infinitely many points. But how can a computer represent infinitely many points digitally? The only option is to represent the infinitely many points by taking finitely many points (samples) and pray that the samples represent the signal well.

Each of the infinitely many samples is taken within equal intervals. This is called the *sampling rate.* For instance, if samples are taken with a sampling rate of 1000, then this indicates that 1000 representative points are taken from the speech signal within a second. In other words, a single sample will be taken each millisecond. The interval that elapses between each sample is called the *sampling period.* In the above case, 1ms will be the sampling period

But how can one know whether the sampling rate is sufficient or not? This can be known by reconstructing back the analog signal from the samples. If the results of the reconstruction are acceptable then the sampling rate can be considered as good. The choice of sampling rate is actually a trade-off between quality and storage space requirement. If the sampling rate is too high, the time and computer storage needed will be high, but the sound will be of high quality. On the contrary, if the sampling rate is too small, time and storage space will be saved but important information may be lost.

Under-sampling is not the only cause of error in the representation of analog signals. Precision is also another problem. Any computer, no matter how big or fast it is, cannot represent most numbers precisely. Rather, the computer approximates numbers in terms of bits. Using a fixed number of bits (0s and 1s), a range of numbers can be represented. The process of representing a range of numbers with a fixed number of digits is called *quantization*. A device that performs quantization is called a *quantizer*. The number of bits used for quantization is termed as the *resolution*. A quantizer of n bits is termed as an n-bit quantizer and it is capable of representing 2^n distinct values.

But the case of quantization is not that simple. As an example, assume that a wave has an amplitude value of 511 (meaning its range is between -511 and 511). Using a 10 bit quantizer, one can represent all numbers starting from -511 to 511 with a step size of 1. But the question is how can amplitude values like 211.13 be represented using the given 10-bit quantizer?

The only option is to represent it with the nearest integer (i.e. 211). But this somehow introduces quantization error. The sum of such quantization errors creates deviations when the signal is reconstructed later. Increasing the resolution of a quantizer may help in reducing the error, but as the resolution increases, more and more storage space is required. Because of this, most quantizers do not exceed a resolution of 16-bits in real life applications.

2.5. THE BASICS OF TEXT-TO-SPEECH (TTS) SYSTEMS

As described earlier, any TTS system takes in a text input and gives back speech utterances. In other words, TTS systems are developed for the purpose of reading out a given text. Any Text-To-Speech synthesis practically involves two major tasks. The first one is text analysis. In this

phase, an input text is transcribed into a phonetic equivalent or some other linguistic representation. This phase is also called high level synthesis (Lemmetty, 1999).

The second phase is the speech synthesis phase. Speech waveforms are produced from the phonetic data of the first phase and some prosodic information. This phase is known as low-level synthesis. A pictorial representation showing these two phases is given below:

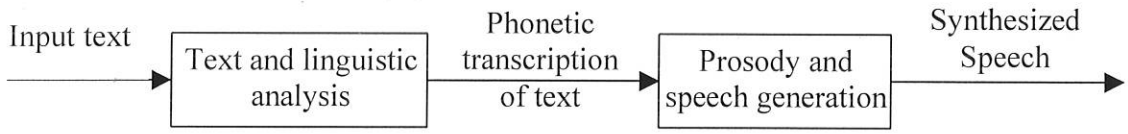


Figure 2.6: A Simple TTS System

Any TTS system contain components performing the above two phases. The high level synthesis is carried out by the *Natural Language Processing (NLP)* component of the TTS system. The low level synthesis on the other hand is carried out by the *Digital Signal Processing (DSP)* component of the TTS system (Lemmetty, 1999).

2.5.1. THE NLP COMPONENT

The natural language processing part of TTS synthesis is often the complex one of the two components. To begin with, numerals, abbreviations and acronyms all need some preprocessing techniques to convert them into the corresponding full words. Correct prosody and pronunciation analysis from written text is also a difficult task because written text does not contain explicit emotions that are expressed while speaking (Lemmetty, 1999).

The conversion of text into the corresponding linguistic representation (grapheme-to-phoneme conversion) may be simple or difficult depending on the specific language. If the written text corresponds to its pronunciation, as in the case of Amharic, then the grapheme-to-phoneme

conversion will be relatively easy. But if the written text doesn't correspond to the pronunciation (as in the case of some English and French words), the task will be more difficult (Minghui, 2000).

Problems regarding pronunciation often arise due to some words called homographs. Homographs are words which are spelled the same in different contexts but differ in pronunciation. For example, the English word "*lives*" is pronounced differently in the two sentences "*Three lives were lost*" and "*One lives to eat*" although its spelling is the same in both of them (Lemmetty, 1999).

In Amharic also, the same word may be pronounced differently depending on the context. Although the pronunciation doesn't change completely, as observed in some English words, consonants will sometimes be geminated and as a result a change in pronunciation occurs.

The other challenging task of the NLP module is to find correct intonation, stress and duration from written text. These features together are called prosodic or suprasegmental features. Simply speaking, prosody refers to variations in the pitch, duration and stress of speech. Nobody speaks without varying pitch under normal circumstances. Prosody gives additional information that the uttered words cannot give alone. As a result, prosody can even change the meaning of a sentence (Rodman, 1999).

Prosodic properties may also convey emotional or other connotations in addition to changing the meaning of a sentence or a word (Rodman, 1999). For example, the Amharic sentence "**አበበ መጣ**" becomes a statement when spoken with falling intonation. But the very same sentence becomes a question if spoken with rising intonation. This shows a change in meaning inflicted by

a change in intonation. On the other hand, the above Amharic sentence may be uttered with some emotional expression, like sadness or happiness.

Prosody can also be understood at different levels. At linguistic level, we know the tone, intonation stress of a speech. That is what prosody exhibits in linguistic level. Prosody is also perceived by human as pitch, loudness, length and strength. This is what we get at perceptual level. Prosody, if expressed at acoustic level, is actually fundamental frequency, duration, amplitude etc..., and that is what we can operate on in the synthesis process. The proper combination of these acoustic factors makes speech natural, expressive and active (Minghui, 2000).

The prosodic features of speech depend on a number of factors such as the meaning of the sentence and the speaker characteristics and emotions. But, since written text contains very little information about such things, and because some of the characteristics change dynamically during speech, it is very difficult to elicit prosodic features from written text (Lemmetty, 1999). The diagram below shows some of the factors contributing to prosodic features.

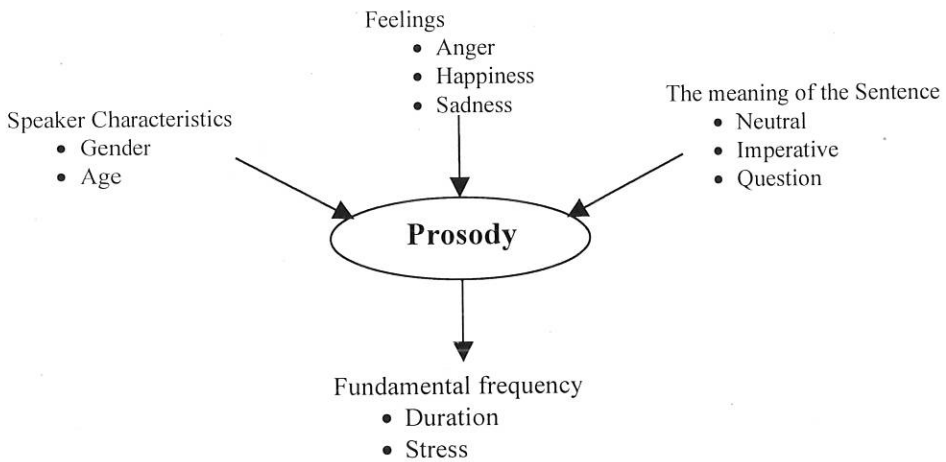


Figure 2.7: Factors contributing to prosodic features

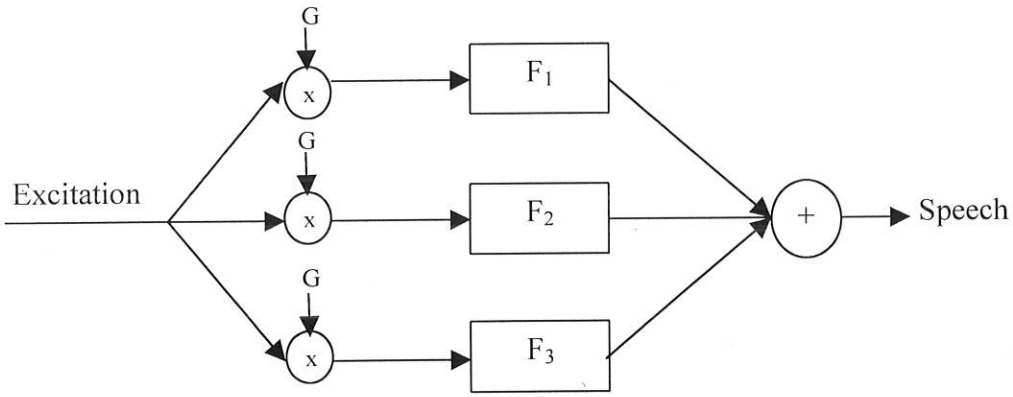


Figure 2.9: Parallel Formant Synthesizer

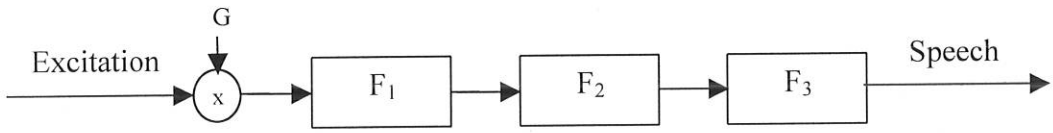


Figure 2.10: Cascade Formant Synthesizer

2.6.2.2. CONCATENATIVE SYNTHESIS

Concatenative synthesis is a method that is gaining much attention recently. Concatenative synthesis operates in such a manner that appropriate speech units are concatenated to construct the required speech. In this speech synthesis method, signal processing techniques should be applied to alter parameters like pitch and duration, and to smooth the discontinuity created by concatenation points (Minghui, 2000).

Concatenative synthesis is probably the easiest way to produce intelligible and natural sounding synthetic speech. But, despite its easiness, concatenative synthesis is usually limited to only one speaker and one voice. The other limitation of concatenative synthesis is that the method often requires more memory capacity than other methods (Lemmetty, 1999).

One of the most important aspects of concatenative synthesis is the proper choice of speech units (acoustic units). The speech units are small segments of speech that will be concatenated to form the desired speech utterance. The selection of an appropriate speech unit is usually a trade-off between naturalness and database size. The longer the units, the higher the naturalness and the lesser the number of concatenation pointes will be (i.e. less discontinuity is perceived). But the problem is that as the size of the speech unit gets larger, the number of units required will also increase, which implies that large storage space is required. This creates a problem in listing all possible units, as well as causes memory shortage because all the units need to be loaded into memory at runtime. Some of the frequently used speech units are:

- *Word*: - a word is probably the most natural unit for written text and some messaging systems that have a very limited vocabulary. It is easy to perform concatenation of words, for information like prosodic and co-articulation effects are already contained in the unit (i.e. the word) itself (Lemmetty, 1999).

  But the problem is that there is a big difference between a word spoken in isolation and in a continuous sentence. This makes the result a little bit unnatural and difficult to understand because of the pitch and formant discontinuities at the boundaries of the words. Such a problem can be treated by signal processing techniques (Donovan, 1996).

  Words spoken in isolation are often longer than words in sentences. Besides, the acoustic as well as phonetic realizations of a given word can vary depending on the context. Therefore, it is recommended to have multiple versions of a word, spoken in different contexts, if a high level of naturalness is to be achieved (Ibid).

Another problem in using words as acoustic unit is their number being so large, making it difficult to handle them. In a given language, there are hundreds of thousands of words. This makes it very difficult to address all of them as speech units in the speech unit database. This makes words unsuitable for an unrestricted TTS system (Lemmetty, 1999).

- *Syllable*: - Syllables are considerably smaller than words in size. However, the size of the speech unit database will still be too large for TTS systems. Syllables may preserve co-articulation effects like words do. But unlike words, co-articulation between syllable units may not be weak, and as a result smoothing across unit boundaries will not be that easy (Donovan, 1996).

  The number of syllables in a language is very large. This creates significant problems in recording and storage. Additionally, the above mentioned co-articulation problem needs to be handled anyway. Therefore, it is sensible to look for other units which are less numerous and easy to concatenate (Ibid).

- *Demisyllable*: - Demisyllables refer to the initial and final halves of syllables. Demisyllables are considerably smaller in number than syllables. For example, about 1,000 demisyllables are needed to construct the 10,000 syllables of English (Donovan, 1996). Demisyllables take into account co-articulation effects and allophonic variations due to the separation of initial and final consonant clusters (Lemmetty, 1999).

  The problem with demisyllables is that the co-articulation between syllables can still be problematic. Another problem regarding demisyllables is that it is often hard to define the exact number of demisyllables in a language. Hence, if a TTS system is developed based on demisyllables, it most likely would not synthesize all possible words (Lemmetty, 1999).

In general, some of the problems encountered in concatenative synthesis, when compared to other synthesis methods are:

- Distortion from discontinuities at concatenation points. Such distortions can be reduced by using steady state units like diphones, or by using signal smoothing methods.

- Memory requirements are often higher than that of the other speech synthesis methods. This memory requirement increases when longer concatenation units (which means larger unit database) are used.

- Data collection and labeling is often time consuming in concatenative synthesis (Lemmetty, 1999).

CHAPTER THREE

SPEECH WAVEFORM SYNTHESIS ALGORITHMS

3.1. INTRODUCTION

Concatenative synthesis is one of the frequently used speech synthesis methods now-a-days. When compared to other synthesis methods such as formant synthesis and articulatory synthesis, it is relatively easy. But the case is not as easy as merely concatenating prerecorded acoustic units. Rather there are a number of signal processing techniques applied on the speech units before and after concatenation, to ensure a better quality output.

Donovan (1996) points out two basic reasons for applying signal processing techniques in concatenative synthesis. These are:

- Signal processing techniques must be applied to the synthesis speech units so as to change their fundamental frequencies and durations to those required in the synthetic speech.

- Signal processing techniques must also be applied to smooth away spectral concatenation discontinuities between units.

The coming sections discuss about the commonly used waveform synthesis techniques in concatenative synthesis. Since this thesis work uses the PSOLA technique an emphasis is given to the part discussing about PSOLA.

3.2. PITCH SYNCHRONOUS OVERLAP ADD (PSOLA) METHOD

Among the recently developed TTS synthesis techniques, the PSOLA technique has drawn considerable attention because of its segmental and suprasegmental efficiency and simplicity.

The main idea behind the algorithm is that it is possible to perform pitch and duration modifications directly on continuous waveforms, without using any parametric model (Dutoit, 1997).

The Pitch Synchronous Overlap Add (PSOLA) technique was first introduced by France telecom at CNET. The PSOLA technique does not synthesize speech signals by itself. Rather, it manipulates prerecorded speech segments and enables the altering of pitch and duration of speech segments. The algorithm has got a number of varieties but all work essentially the same way (Donovan, 1996).

According to the PSOLA algorithm, the natural speech is first divided into a number of short-term (ST) signals. This is done by using a windowing function. Whenever analyzing a signal, the assumption is that the signal has a constant frequency. But for this assumption to be true, the signal should first be divided into a number of small portions whose frequency is constant or nearly constant. The process of segmenting a signal into very small portions is called *windowing* (Cassidy, 2002).

A windowing function segments a given speech signal into a number of smaller overlapping units. Windowing can be seen as multiplying a signal by a window which is zero everywhere except for the region of interest, where it is one. Since we pretend that our signals are infinite, we can discard all of the resulting zeros and concentrate on just the windowed portion of the signal (Ibid).

The above explained window, which multiplies the portion of interest with one and everything else with zero, is known as a rectangular window because of its shape. One problem with this

kind of window is the abrupt change at the edge. This can cause distortion in the signal being analyzed (Cassidy, 2002).

To reduce this distortion there are smoother window functions like Hamming and Hanning. These windows are zero at the edges and rise gradually to be one in the middle. When one of these windows is used, the edges of the signal are de-emphasized and as a result, the edge effects are reduced (Ibid). See the diagram below.

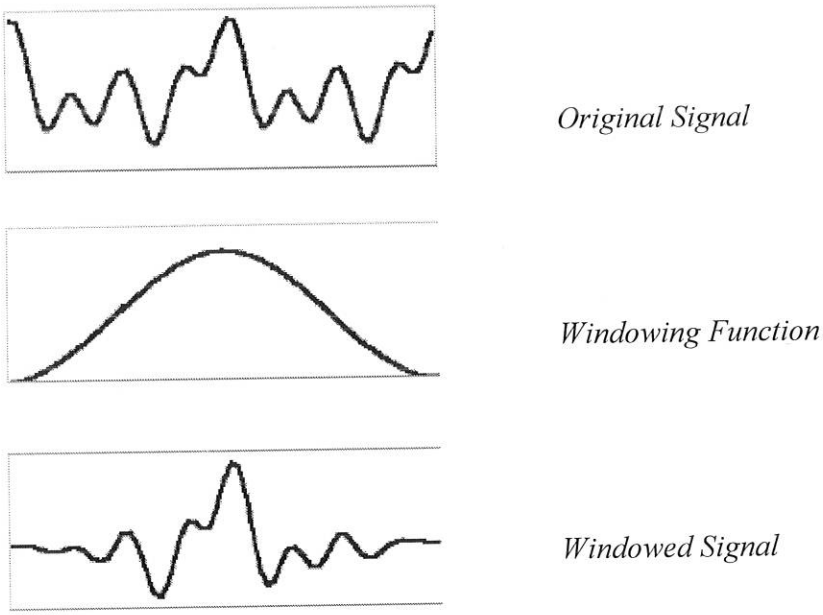
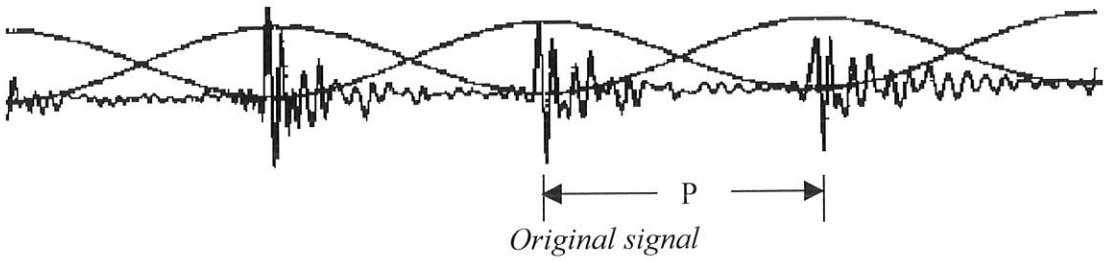


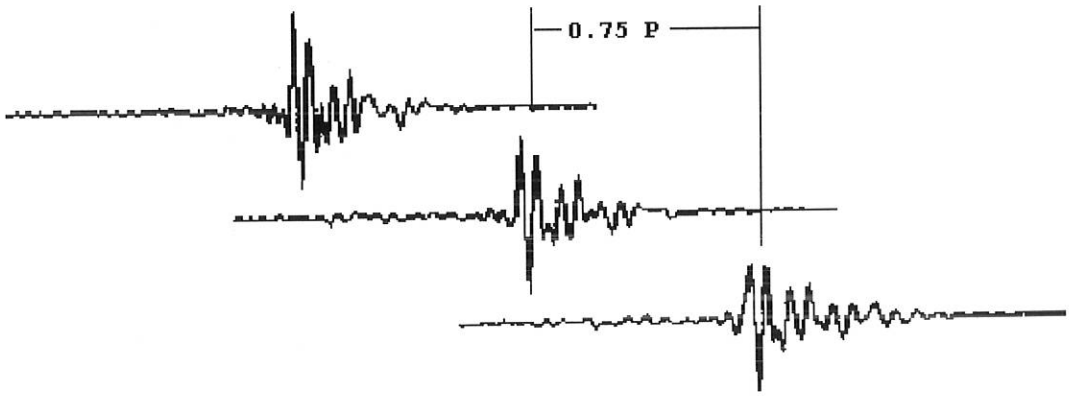
Figure 3.1: Windowing a signal

The length of the ST signals depends on the pitch period. Pitch and duration modifications are applied on the ST signals. For instance, the pitch is raised or lowered by varying the distance between the ST signals. On the other hand, duration is handled by repeating or deleting ST signals as necessary (Pasanen, 2001).

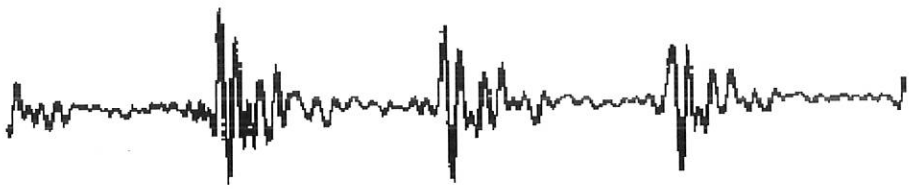
The final step in the PSOLA algorithm is to recombine (concatenate) the ST signals. Recombination is accomplished by using an overlap-add technique. See the figure below.



Original signal



ST-Segments of the original signal



Synthesized signal

Figure 3.2: Waveform Synthesis

As can be seen from the above figure, first the original signal was segmented into ST-Signals using a windowing function. Originally, the signal had a pitch period of P. At the time of overlap-add, this pitch period was reduced by a factor of 0.25 (i.e. the pitch period P became 0.75 P). Hence the pitch of the synthesized speech will increase. Pitch can also be decreased by increasing the pitch period in the synthesized speech. Decreasing and increasing the pitch period can easily be done by adjusting the amount of overlap (Türk, 2000).

most computationally efficient version than the other variants such as *Frequency-Domain PSOLA* (FD-PSOLA). The coming sub section discusses about the TD-PSOLA algorithm in detail.

3.3.1. TD-PSOLA

The TD-PSOLA algorithm works exactly as described above. The algorithm has got two main components. One is the analysis part, which analyzes speech and stores the analysis result into a database. The other component is the synthesis part which synthesizes speech utterances using the appropriate data from the speech database (Minghui, 2000).

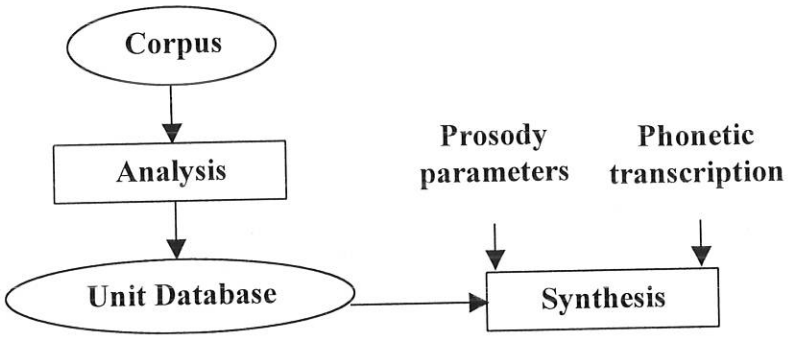


Figure 3.4: Components of the TD-PSOLA method

3.3.1.1. THE ANALYSIS PART

The analysis part of the TD-PSOLA algorithm has three basic steps.

- *Voiced / Unvoiced detection*: - Once the necessary acoustic units are extracted from the speech corpus, the next task will be to identify the voiced and unvoiced part of each acoustic unit. This is necessary because pitch modifications are carried out on the voiced part of the units. Voiced / unvoiced detection is carried out through the use of time domain parameters such as the *Root Mean Square (RMS)* and *Zero Crossing Rate (ZCR)* (Cassidy, 2002).

RMS is a measure of the energy in speech signals. The RMS of successive signal windows gives the measure of change in amplitude through time. Voiced signals have high RMS values while unvoiced ones have low RMS values (Cassidy, 2002).

The ZCR, on the other hand, measures the number of times the signal crosses the zero line per unit of time. This parameter is also often used to detect between voiced and unvoiced signals. Voiced signals have low ZCR values while unvoiced signals have high ZCR values (Ibid).

- *Pitch marking*: - the next task is to mark the location of pitch pulses. Pitch pulses are the locations at the energy peak of the ST signals. They are often found at the positive or negative extremes of every pitch cycle (Goncharoff & Gries, 1998). Finding the pitch marks is necessary because the pitch marks serve as a reference point for the overlap between ST-Signals (Chen & Kao, 2001).
- *Storage*: - the last task is to store the data into the database. The speech units, the pitch marks, and the voiced/unvoiced detection data should be stored so that they could be used by loading into memory at synthesis time.

3.3.1.2. THE SYNTHESIS PART

The synthesis part of the TD-PSOLA takes the data stored in the speech database and then reconstructs speech by concatenating the appropriate speech units. There are three basic steps in the synthesis part:

- *Appropriate data selection*: - the first task in the synthesis part is to select the appropriate data units to be concatenated. The selection can be done by looking at the result of the grapheme-to-phoneme conversion.

- *Synthesis of unvoiced part*: - the unvoiced part of the speech can be synthesized by simply copying it from the database.
- *Synthesis of the voiced part*: - synthesis of the voiced part involves overlapping the windowed signals with the proper displacement and adding them. The displacement is determined by the target pitch. If the displacement offset is high then the pitch will be low, and vice versa (Minghui, 2000).

Although the TD-PSOLA technique produces good quality speech utterances, it has also got some limitations. One of the limitations is the problem faced when increasing significantly the duration of unvoiced sounds. In such cases a local periodicity may result and the unvoiced part will be perceived as tonal noise as a result. Of course it is possible to overcome this problem by reversing the time axis of the repeated ST signal. But this solution works only for purely unvoiced sounds and does not work for voiced fricatives (Minghui, 2000).

The other problem with the TD-PSOLA algorithm is the lack of smoothness at the concatenation points. Unless the speech units are prepared and selected carefully, the discontinuities at the concatenation points would be perceived (Donovan, 1996).

Some of the limitations of TD-PSOLA can be easily overcome by using other variants like the FD-PSOLA.

3.3.2. OTHER VARIANTS OF PSOLA

The simplest one being TD-PSOLA, the PSOLA technique has also got other versions. The FD-PSOLA, LP-PSOLA and MBR-PSOLA are some to mention.

The FD-PSOLA algorithm is a variant of the PSOLA algorithm which operates at frequency domain. The algorithm theoretically gives better result than the TD-PSOLA algorithm. However, the FD-PSOLA algorithm has a drawback in that it requires high computational power (Türk, 2000).

The LP-PSOLA (Linear Predictive PSOLA) algorithm, on the other hand, is a combination of PSOLA with parametric synthesis techniques. In this approach prosodic modifications are carried out using TD-PSOLA on the excitation of all pole synthesis filters rather than on direct speech (Dutoit, 1997).

Both FD-PSOLA and LP-PSOLA have and advantage over TD-PSOLA in that they provide independent control over the spectral envelope of the synthesis signal (Lemmetty, 1999).

The MBR-PSOLA (Multi-Band Re-synthesis PSOLA) technique was first proposed by Dutoit. This technique involves the modification of acoustic speech units using a computationally expensive Multi-Band-Excited (MBE) analysis-synthesis procedure, so as to make the acoustic unit inventory more suitable for the TD-PSOLA algorithm (Minghui, 2000).

Specifically, all segments are re-synthesized to have the same constant pitch, with the new pitch-marks. This avoids the problem of trying to locate the pitch-marks in the segment inventory, and reduces the discontinuity problems, which can otherwise arise when concatenating spectral similar segments of speech with very different pitches (Ibid).

3.4. OTHER WAVEFORM SYNTHESIS TECHNIQUES

3.4.1. LINEAR PREDICTIVE CODING (LPC)

Like formant synthesis, the LPC method is a source-filter method of speech synthesis. The source-filter model is a model which assumes excitation signal source is independent from the vocal tract. But this is approximately true because resonance of the vocal tract may affect the vibration pattern of the vocal cords by building up pressure waves (Donovan, 1996).

The LPC method has been used extensively in concatenative systems, since it enables the rapid coding of concatenation units. The basic theory behind the LPC is the assumption that the current speech sample y(n) can be predicted as a linear combination of the previous P samples of speech, plus a small error term e(n). See the following diagram:

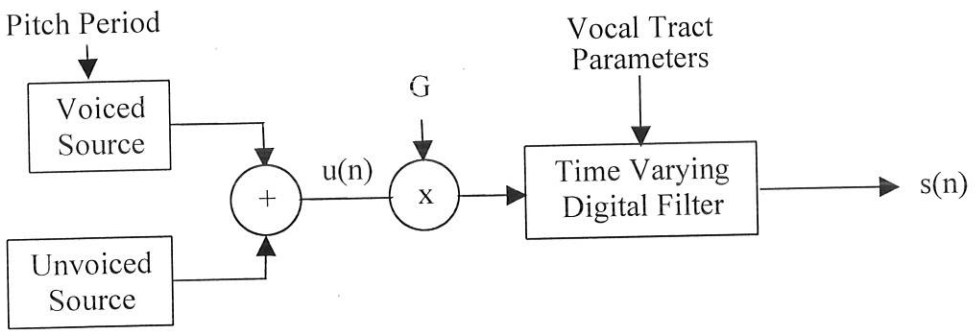


Figure 3.5: Speech generated with the LPC method

According to the LPC method, the current speech sample s(n) can be expressed in terms of the past p speech samples. In other words, s(n) is expressed as a linear prediction or order p on a sequence {s(0), s(1), …, s(N-1)}representing N speech samples. A set of prediction coefficients $\{a_1, a_2, \ldots, a_p\}$ are searched to verify the following

$$s(n) = \sum_{i=1}^{p} -a_i.s(n-i) \qquad \text{For n = p, p+1, ..., N-1} \quad \text{.....(3.1)}$$

Introducing a prediction error factor f(n) implicitly into the above equation leads to:

$$s(n) = \sum_{i=1}^{p} -a_i.s(n-i) + f(n) \qquad \text{For n = p, p+1, ..., N-1} \quad \text{.....(3.2)}$$

The values of the coefficients a_1, a_2, ..., a_p are chosen in such a way that the value of $\sum_{n=p}^{N-1} f(n)^2$ is minimum (Dutoit, 1997).

The LPC method does not produce the formant frequencies correctly, especially when speech is re-synthesized at a different fundamental frequency to that of the original. Because of this, speech synthesized by the LPC method is far from perfect. The most noticeable effect in the resulting speech is the existence of a buzzing sound. Most of these problems could be solved by using another variant of the LPC method called *multi-pulse linear prediction* (Donovan, 1996).

3.4.2. HMM (HIDDEN MARKOV MODEL) BASED SYNTHESIS

Synthesis systems can be made in such a way that they can be trained. This makes it easy to create synthesis systems for new voices or even new languages. HMM based speech synthesis is a fully corpus based method, and therefore, it is a trainable method. New voice can be easily got by providing new speech data. This is a promising method in this sense (Minghui, 2000).

3.4.3. HNM (HARMONIC PULSE NOISE MODEL)

HNM is also one of the concatenation synthesis techniques. This method is reported to give a higher synthesized speech quality than PSOLA. HNM is based on a pitch-synchronous harmonic plus noise representation of the speech signal (Minghui, 2000).

HNM analysis consists of two basic steps. The first step of the HNM analysis consists of estimating pitch and maximum voiced frequency based on pitch detection. The second step of the HNM analysis consists of estimating a continuous spectral and phase envelope per voiced frame. At synthesis time, the HNM frames are concatenated and the prosody of units is altered accordingly (Ibid).

CHAPTER FOUR

EXPERMENTATION

4.1. INTRODUCTION

This chapter discusses the experiment and the steps undergone in doing it starting from data preparation. In this experiment, diphones and syllables have been used as acoustic units to build the prototype. The syllable units are of CV-Syllable type, which are commonly used in the Amharic writing system.

Excluding the data preparation part, two basic modules have been developed in designing the prototype. One is the grapheme-to-phoneme converter, which converts input Amharic texts into their equivalent phonetic transcriptions. The other one is the speech waveform synthesis module, which incorporates tasks like pitch & duration modification.

In addition to these two basic modules, there have also been additional tasks like data recording and segmentation. These tasks are categorized as data preparation. This chapter discusses in detail these mentioned tasks, starting from the data preparation; and then the actual experimentation and the obtained results are discussed.

4.2. THE EXPERIMENT

4.2.1. DATA PREPARATION

The data preparation step is the first step in this experiment. This is probably true for the development of other TTS synthesis systems too. The basic data required for concatenative TTS

systems is a collection of speech segments (acoustic data units). These speech units are extracted from prerecorded speech utterances.

Four basic steps are involved in the data preparation process.

4.2.1.1. ACOUSTIC UNIT INVENTORY DESIGN

In the acoustic unit inventory design process the acoustic units that are to be used in the experiment are selected. Those selected units are later to be extracted from prerecorded speech utterances and stored in a database so as to use them in the concatenation synthesis process later.

The Amharic language has got about 28 consonants, and seven vowels which come together with each of the consonants. The combination of a consonant with each of the seven vowels gives the seven orders (Laine, 1998). Totally, there are about 196 (28 X 7) basic CV-syllables in the Amharic language. The number of symbols representing these CV-syllables is somewhat higher than 196 because some CV-syllables have more than one representative symbol in the Amharic writing system.

On the other hand, there are about 39 phonemes in the Amharic language (kinfe, 2002). By taking the combination of each phoneme with the others, there will be about 1521 (39 X 39) diphones. Of course this figure is a little bit exaggerated because there are some combinations that do not exist in the Amharic language. For instance, there are no vowel-vowel (V-V) combinations in the Amharic language (Laine, 1998). But, even omitting those non-existent diphones for this specific language, the number of diphones will still be difficult to handle.

Therefore, having a database of all the diphones and syllables is out of question, mainly because of the time limitation. Therefore, those CV-syllables and diphones that are frequently used in the

Amharic language have been considered in this experiment. Given a frequently used consonant sound, all the seven orders of that sound have been included in the acoustic unit inventory.

For instance, the sound /m/ is one of the frequently used sounds in the Amharic language (Baye, 1997). Therefore, all the seven orders of /m/ (i.e. 'መ' /me/, 'ሙ' /mu/, 'ሚ' /mi/, 'ማ' /ma/, 'ሜ' /më/, 'ም' /mï/, 'ሞ' /mo/) are considered.

The sixth order of a given Amharic CV-Syllable is read in two different ways depending on its location in the word and the context. One way is reading it with the vowel sound /ï/ and the other is omitting the vowel sound. The latter is often true when the sixth order symbol comes at the end of a word. For the sixth order CV-syllables though, two acoustic units are stored. For instance, for the sixth order CV-syllable 'ም', /mï/ and /m/ are stored separately.

When it comes to selecting diphones, again it is not possible to consider all combinations of a given phoneme with the others. For instance, the CV-syllable 'መ' (/me/) is constructed from the phonemes /m/ and /e/. However, it is not possible to consider all combinations of /m/ or /e/ with the other phonemes.

Therefore, only the combination of the most frequently used sounds with the seven vowels is considered in this work. In other words, all consonant-consonant combinations are omitted even though they exist in the Amharic language. For a given consonant sound, there are about 14 (2 X 7) diphones according to this convention. The number is doubled to 14 because the order of the phonemes matters. As an example, /m/ and /e/ can be coupled as /me/ and /em/ to form two different diphones.

The syllables and diphones are extracted from prerecorded word utterances. The Syllables and diphones constituting the acoustic unit inventory and the corresponding words from which they are extracted are given in Appendix A.

4.2.1.2. RECORDING THE CORPUS

Once the acoustic unit inventory is designed, the next task is to record the corpus data, from which the acoustic units (CV-syllables and Diphones) are extracted. Recording is carried out using Praat. Only the voice of one speaker has been recorded because of the time limitation.

4.2.1.3. ACOUSTIC UNIT EXTRACTION

Once the corpus is recorded, the next stage is to extract the acoustic units from the corpus. The extraction can be performed using any signal visualization tool. For this experiment, Praat has been used to obtain the waveform of the recorded words and subsequently extract the acoustic units manually. The extraction process takes a long time because it is often difficult to identify the boundaries of u-nits.

4.2.1.4. ACOUSTIC UNIT NORMALIZATION

Once the acoustic units are extracted, the final step to this end is to normalize the extracted units. Due to a number of factors, the different acoustic units have different energy (amplitude). To mention some, speaker's instability and the variation in the distance of the microphone are some of the causal factors for the variation in amplitude of the acoustic units.

Such amplitude variations must be corrected so as to avoid disasters like voice quality degradation that would be inflicted later at the synthesis time. The normalization tool packaged with Praat has been used to scale the amplitude of the acoustic units within some range.

All the above steps, excluding the first one, have been done using the Praat software. The software gives capabilities of recording a speech utterance; viewing the spectrograph and waveform of a speech; extracting, cutting, copying and deleting parts of the speech utterance; normalizing waveforms; and many others. The following figure shows the spectrograph and waveform segmentation of the Amharic word '**ተመለሰ**' (/temelese/) using Praat.

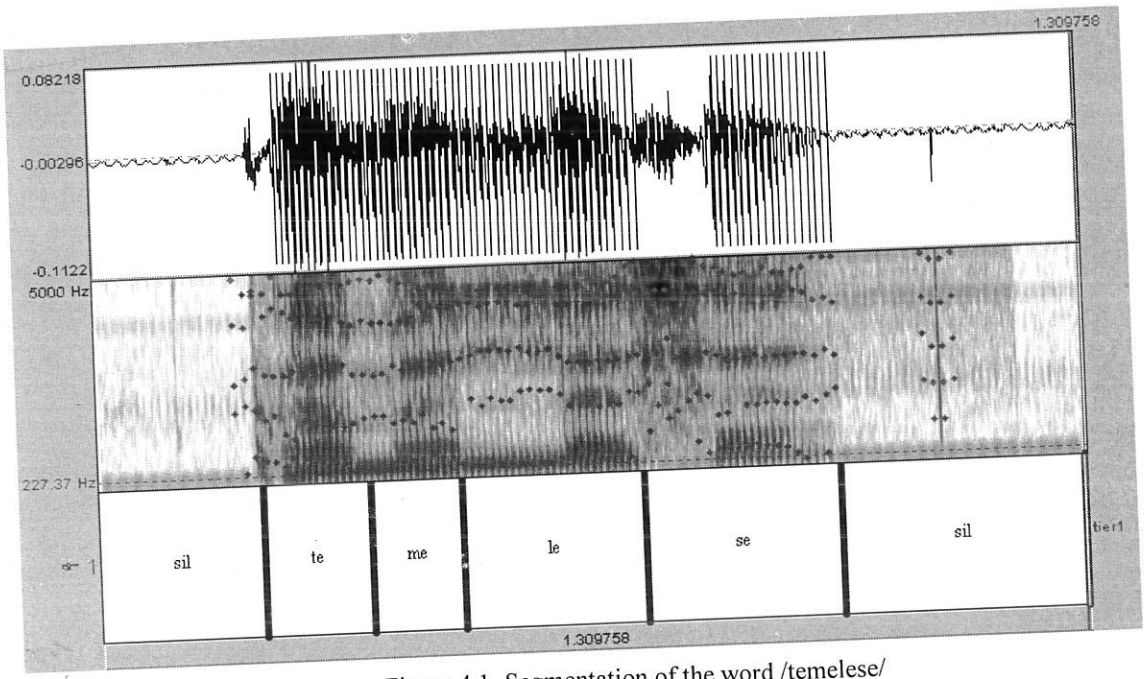


Figure 4.1: Segmentation of the word /temelese/

Once the above four steps are accomplished successfully, then the acoustic units are stored in the speech database and would be readily available at synthesis time.

4.2.2. ADDITIONAL TASKS

Right after the above data preparation steps, there are other additional steps (two in number to be precise) that are worth considering. These two steps involve the extraction of some useful data (voiced/unvoiced detection data and pitch mark location data) from the acoustic speech units.

Of course these two steps can be performed at the time of waveform synthesis only for the currently selected acoustic units. But this will definitely add to the computational overhead and may make the system rather slow. Therefore it is preferred to store such information in a database and read them from memory at run time.

4.2.2.1. PITCH MARK IDENTIFICATION

The pitch mark identification step can also be considered as part of the data preparation step. Pitch marks are locations at the energy peaks of ST-signals. The location of pitch marks is later used in the detection of voiced and unvoiced parts of an acoustic unit, as well as in the application of the TD-PSOLA algorithm.

A MATLAB script has been used to extract the location of pitch marks of each acoustic unit in the acoustic unit inventory. The script takes as an input a column vector representation of the acoustic units and the sampling rate. The output is the location of the pitch marks in the form of a row vector. The algorithm for pitch mark identification is given below:

4.2.2.2. VOICED/UNVOICED (V/UV) DETECTION

Once the pitch marks are identified, the next step is to identify the voiced and unvoiced parts of a given acoustic unit. This is done over the sequence of pitch mark locations and the final result is a sequence of zeros and ones, zeros indicating unvoiced and ones indicating voiced sounds.

The voiced/unvoiced data for CV-Syllables often starts with a stream of 0s and at some point it changes to a stream of 1s. This is logical because CV-Syllables start with consonants, which are often unvoiced, and end with the vowels, which are voiced.

Both the pitch mark and v/uv data are stored in simple text files. When the TTS program starts, all these data and the acoustic units will be loaded onto memory. A structure (also called record in Delphi) has been defined to hold a descriptive name of each acoustic unit, and the numeric data of the waveform of each acoustic unit with the corresponding pitch mark and the v/uv data.

The speech data, pitch mark data and the v/uv data are all in the form of one dimensional array. The name is obviously used for matching and selecting the appropriate acoustic units at run time, based on the result of the grapheme-to-phoneme transcription.

The flowchart for generating v/uv data is given below:

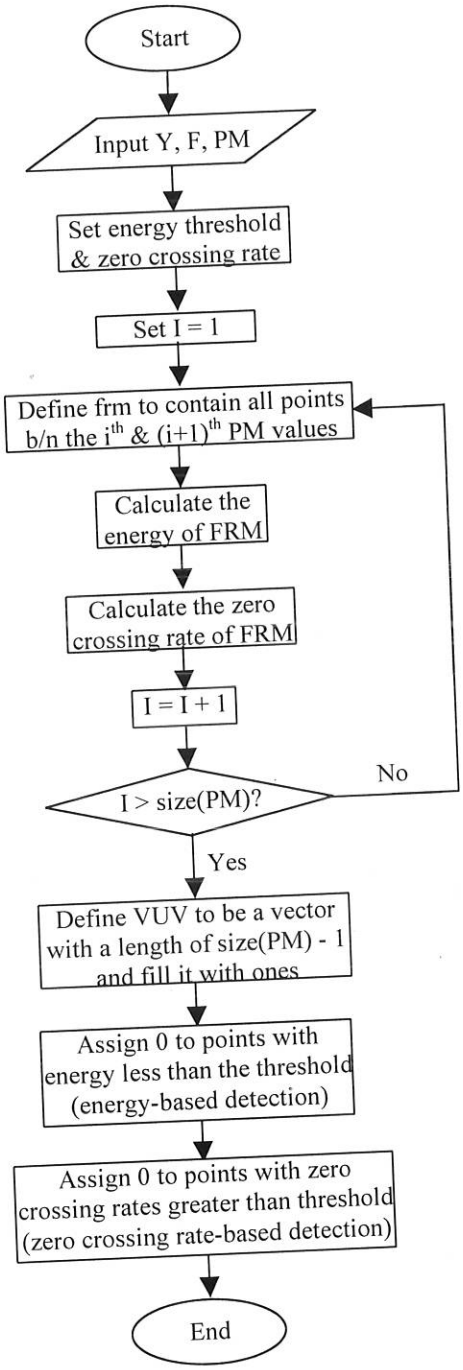


Figure 4.2: Flowchart for V/UV detection

4.2.3. GRAPHEME-TO-PHONEME TRANSCRIPTION

As has been mentioned previously, the grapheme-to-phoneme transcription module is one of the two basic components (modules). The other one is the speech waveform synthesis module. The grapheme-to-phoneme transcription process involves the transcription of the input Amharic symbols to the corresponding sequence of phonetic symbols that represent the acoustic units to be used.

For this experiment two types of transcription have been implemented because two acoustic unit types had been used (diphones and syllables).

4.2.3.1. SYLLABLE TRANSCRIPTION

In this transcription process, the Amharic symbols are transcribed to the corresponding symbols representing the sound of each CV-Syllable. To illustrate this, consider the Amharic word '**ተመለሰ**' (/temelese/). At the time of Syllable based synthesis this word will be transcribed as /te/-/me/-/le/-/se/. A hyphen separates subsequent syllable representatives. Based on this transcription, the syllable sounds 'te', 'me', 'le' and 'se' will be selected from the acoustic unit inventory.

The transcription in syllable based synthesis looks straight forward and simple. But this is not true in cases like gemination. Gemination refers to the lengthening of a consonant when produced. The Amharic word '**ገና**' can be read as /gena/ or /genna/ depending on the context. The Amharic writing system lacks the ability to differentiate between geminated and non geminated consonants. Hence there is no simple way of knowing whether a consonant is read in its geminated form or not, if it has one.

But this problem can be solved by using some special character after a consonant to show that it is geminated. In this experiment, an apostrophe (') has been used after CV-syllables to show that the consonant part is geminated. Therefore, the previous word **ገና** is written as **ገና'** to show gemination. Any time the transcriber finds the apostrophe symbol, it doubles the consonant part of the CV-syllable before the apostrophe.

The other problem faced during syllable based transcription is due to presence of a sequence of sixth order CV-syllables. As mentioned previously, the sixth order CV-syllables can be read in two forms: one with the vowel sound /ï/ and the other without the vowel sound /ï/. The transcription becomes relatively simple when a sixth order symbol comes at the end of a word. When this is the case the /ï/ sound will simply be omitted.

But the problem arises when a cluster of sixth order CV-syllables come in the middle of a word. In such cases there is no straight forward way of determining which one of the sixth order CV-syllables is read with the vowel sound and which one is not. In this thesis work, a set of simple rules have been used to solve this problem. But these rules do not work for all cases.

4.2.3.2. DIPHONE TRANSCRIPTION

In diphone based synthesis, the input text is transcribed into a string of diphones that represent the diphone sounds. The transcription is a straight forward one. For instance, the previous word '**ተመለሰ**' (/temelese/) is transcribed into diphones as

/sil-t/-/te/-/em/-/me/-/el/-/le/-/es/-/se/-/e-sil/

The 'sil' part at the beginning and the end of the transcription indicates silence.

The above transcription can easily be obtained by using the syllable transcription as an input. Once the syllable transcription is obtained, the remaining task is to add the silence element at the beginning and the end, and then to take the phonemes two by two with a step factor of one, starting from the silence element.

Consider the following example.

Sentence	አበበ ምግብ በላ
Phonetic form	**/abebe mïgïb bela/**
syllable transcription	**/a//be//be//sil//mï//gï//b//sil//be//la/**
Diphone transcription	**/sil-a//a-b//b-e//e-b//b-e//e-sil//sil-m//m-ï//ï-g//g-ï//ï-b//b-sil//sil-b//b-e//e-l//l-a//a-sil/**

The grapheme-to-phoneme transcription module is coded using the Delphi programming language. The module uses a look-up dictionary to identify the proper phonetic equivalents of the Amharic symbols. Subsequently, the module removes /ï/ from sixth order symbols, if there are any, based on the above mentioned set of rules.

The module then looks for any geminated consonant. As mentioned before, the apostrophe symbol is used after an Amharic symbol to show that it is geminated. The moment the module finds an apostrophe symbol, it doubles the consonant part of the CV-syllable before the apostrophe.

Once the Syllable based transcription of a word or a sentence is obtained, it is easy to derive the corresponding diphone based transcription. The same principle mentioned previously is used in obtaining diphone based transcriptions from syllable based transcriptions.

4.2.4. SPEECH WAVEFORM SYNTHESIS

Once a given input text is transcribed, the next step is to synthesize the speech waveform by concatenating the appropriate acoustic units. The waveform synthesis part has got two basic parts. The first and the easiest is the selection of appropriate units from the acoustic unit database. The other one involves concatenating the selected acoustic speech units with the application of the proper pitch and duration modifications.

The first thing that the system does when allowed to run is to load all the acoustic speech units and the other data, like the pitch mark data, into the RAM. This exploits the memory available, especially if the size of the acoustic inventory is large. But the movement of the data from the hard disk to the memory is very important in making the system faster, especially at times of searching.

The acoustic inventory elements (acoustic units) are stored in wave file format on the hard disk. A MATLAB script has been used to read the wave data to memory. Given the name of the wave file, the MATLAB wave reading script returns a column vector representation of the wave file. The vector elements represent amplitude values of the wave file at specific times.

On the other hand, the pitch mark and v/uv data are already written in text files in vector form, and therefore reading them into memory is straight forward.

4.2.4.1. ACOUSTIC UNIT SELECTION

The acoustic unit selection process is a simple and straightforward task, given that the speech data and the phonetic transcription of a text are available. The moment the transcription is

delivered by the transcriber, a search algorithm starts searching for the appropriate acoustic units. This is done by simply comparing the name of each acoustic unit with the transcription.

Let's take the same example we have seen before to illustrate this. For the Amharic word 'ተመለሰ' (/temelese/). The syllable based and diphone based transcriptions are /t-e//m-e//l-e//s-e/ and /sil-t//t-e//e-m//m-e//e-l//l-e//e-s//s-e//e-sil/ respectively. Therefore, in case of syllable based synthesis, acoustic units with the name of 'te', 'me', 'le' and 'se' are searched and selected. The same principle could be applied to select diphones.

Once the appropriate items have been selected, the remaining is to apply signal processing technique on the selected units and concatenate them.

4.2.4.2. PITCH AND DURATION MODIFICATION

The pitch and duration modification step is the last step involved before concatenating the selected units. The TD-PSOLA technique has been used to modify pitch and duration. The module is coded in MATLAB. Given a vector representation of an acoustic unit and its corresponding pitch mark and v/uv data, the module modifies the signal to have the desired duration and pitch.

In this experiment, only two general cases are considered: the case of simple statements and the case of yes/no questions. Considering all available cases is impossible here because it needs a thorough investigation of prosodic features of the Amharic language.

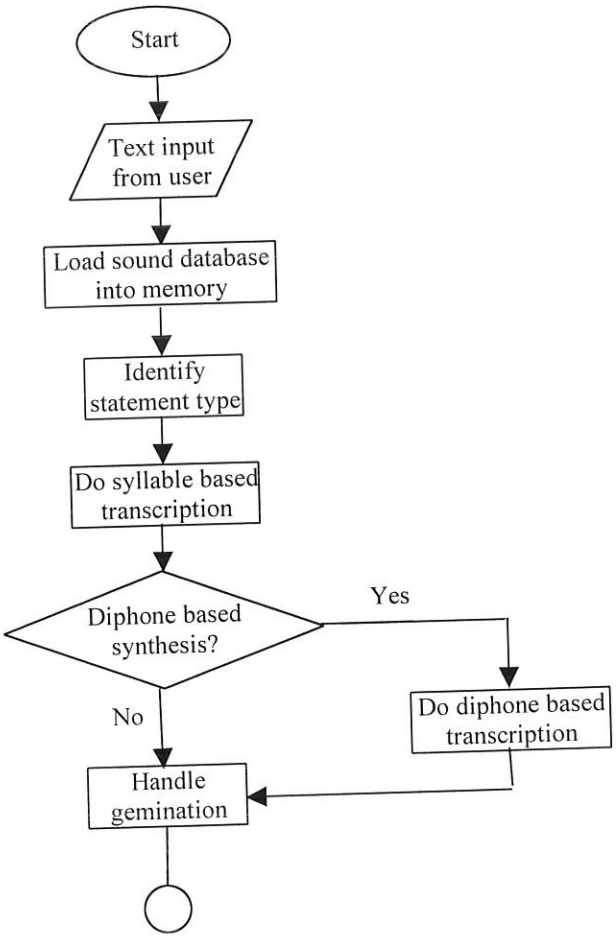
In general, statement type utterances are spoken with falling intonation while yes/no question types are spoken with rising intonation. Statements are recognized by the punctuation mark they have at the end. If a statement is found to be an interrogative one, then the pitch increases

gradually. On the other hand, if the statement is a simple statement, then its pitch gradually decreases.

4.2.4.2. CONCATENATION

The last step in the synthesis part is the concatenation of the synthesized units. The processed units are concatenated while they are in their vector representation using MATLAB. Then the resulting speech is written into a temporary file and played.

Below is a flowchart briefly showing the overall process.



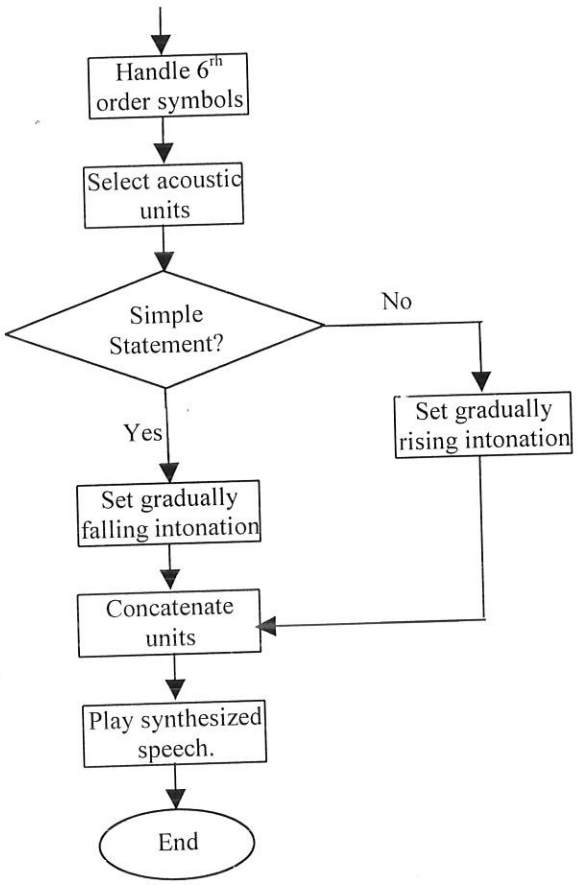


Figure 4.3: A flowchart showing briefly the TTS synthesis

4.2.5. TTS SYSTEM EVALUATION

As described before, two tests (ORT and MOS) had been conducted to test the performance of the developed prototype. For each of the tests two different test sets had been prepared. For the ORT test, twenty words which can be uttered using the available acoustic data had been prepared. For the MOS test, seven sentences had been prepared.

In the ORT test case, six users (three for diphone based synthesis and three for syllable based synthesis) were selected to simply listen to the words and say them back. The results obtained from the tests are given below:

Words	Person 1	Person 2	Person 3
አበበ /abebe/	X	√	√
ከመረ /kemmere/	√	√	√
በላ /bella/	√	√	√
አመነ /ammene/	√	√	√
አደረ /addere/	√	X	√
ወደደ /weddede/	√	√	√
ከረመ /kerreme/	√	√	√
ወረደ /werrede/	X	√	√
ረዳ /redda/	√	√	X
ወሰደ /wesede/	√	√	√
ጋረደ /garrede/	√	√	√
አሰበ /assebe/	√	√	√
ጉማሬ /gumarrë/	√	√	√
ገደለ /geddele/	√	√	√
በረበረ /berebbere/	X	√	√
ሰበሰበ /sebessebe/	√	√	√
ደረሰ /derese/	√	√	√
ተመላለሰ /temelallese/	√	√	√
ደመሰሰ /demessese/	√	√	√
ለጋስ /leggas/	X	X	X
ተገረመ /tegerreme/	√	√	√
አበበ /abebe/	√	√	√
ከመረ /kemmere/	√	√	√
በላ /bella/	√	√	√
አመነ /ammene/	√	√	√
አደረ /addere/	√	√	√
ወደደ /weddede/	√	√	√
ከረመ /kerreme/	X	√	√
ወረደ /werrede/	X	X	√
ረዳ /redda/	√	√	√

Table 4.1: Result table of ORT test for diphone based synthesis.

Words	Person 1	Person 2	Person 3
አበበ /abebe/	√	√	X
ከመረ /kemmere/	√	√	√
በላ /bella/	√	√	√
አመነ /ammene/	√	√	X
አደረ /addere/	X	√	√
ወደደ /weddede/	√	√	X
ከረመ /kerreme/	√	√	√
ወረደ /werrede/	X	√	X
ረዳ /redda/	X	X	√
ወሰደ /wesede/	X	√	√
ጋረደ /garrede/	X	√	X
አሰበ /assebe/	X	√	√
ጉማሬ /gumarrë/	√	√	√
ገደለ /geddele/	√	√	√
በረበረ /berebbere/	√	√	X
ሰበሰበ /sebessebe/	X	√	√
ደረሰ /derese/	√	√	√
ተመላለሰ /temelallese/	√	√	√
ደመሰሰ /demessese/	√	√	√
ለጋስ /leggas/	√	X	X
ተገረመ /tegerreme/	√	√	√
አየለ /ayele/	√	√	√
ተወዳደረ /tewedaddere/	√	√	√
ታገሰ /taggese/	√	√	√
ግሩም /gïrum/	√	√	√
ግመል /gïmel/	√	√	√
ነጋ /nega/	X	√	X
ተገበረ /tegebbere/	X	X	X
ትርጉ /tïrïta/	√	X	√
እናት /ïnnat/	√	√	√

Table 4.2: Result table of ORT test for syllable based synthesis.

Using the ORT test, an average of 88% of the words had been recognized correctly by listeners in case of diphone based synthesis. On the other hand, an average of 75% of the test words had been recognized correctly by listeners in case of syllable based synthesis.

The results of the MOS tests are given below:

Sentence	Person 1	Person 2	Person 3
አበበ በሶ በላ	2	4	2
ወደቤት ተመለሰ	1	3	2
አራት ኪሎ አካባቢ ደረሰ	2	2	2
ነገ ይመለሳሉ?	3	2	1
ሰባት ሰአት ሞላ?	2	2	2
አለሙ መኪና ነዳ	3	2	2
በሰላም ደረስክ?	2	3	3

Table 4.3: Result table of MOS test for syllable based synthesis.

Sentence	Person 1	Person 2	Person 3
አበበ በሶ በላ	1	1	1
ወደቤት ተመለሰ	1	2	2
አራት ኪሎ አካባቢ ደረሰ	0	1	1
ነገ ይመለሳሉ?	1	1	2
ሰባት ሰአት ሞላ?	1	1	1
አለሙ መኪና ነዳ	1	1	0
በሰላም ደረስክ?	1	1	1

Table 4.3: Result table of MOS test for diphone based synthesis.

Key: **0 – Poor**
1 – Fair
2 – Good
3 – Very Good
4 – Excellent

4.2.6. ANALYSIS OF RESULTS

In the ORT test, the words synthesized using diphones are more intelligible than the words synthesized using syllables. One of the problems observed with the syllable based synthesis is the problem of germination. Most words with geminated symbols had not been uttered correctly by the system. The recorded corpus itself doesn't have geminated data. Germination is handled by doubling the consonant part of the geminated symbol.

But extracting the consonant part is very hard because it needs the identification of the exact boundaries. As a result the symbol to be geminated may not be geminated correctly. Another problem that has been observed in both diphone based and syllable based syntheses was the problem to utter words with sixth order endings.

As mentioned previously, the vowel part of a sixth order symbol would be omitted if the sixth order symbol comes at the end of a word. This means that a sixth order symbol at the end of a word is represented only by its consonant sound. But the problem with such an event is that it is difficult to extract out consonant parts of a CV-Syllable from the speech corpus, as described in the previous paragraph, and as a result the synthesized word utterance may not be intelligible.

A better way to handle the problem of germination is to incorporate the geminated form of each acoustic unit in the acoustic inventory.

Although it is not as much as the syllable based synthesis, the gemination problem is observed in diphone based synthesis too. In case of diphone based transcription, however, the consonant part of the geminated symbol has a chance of being repeated more than once. This is easily illustrated using the following example.

Word	Diphone Transcription	Syllable Transcription
አበበ /Abbebe/	/sil-a//a-b//b//b-e//e-b//b-e//e-sil/	/a//bbe//be/

As can be seen from the above example, the consonant part of the middle symbol (i.e. 'b') is repeated three times in the diphone based transcription and two times in the syllable based transcription. The more times a consonant sound is repeated, the more its duration would be and as a result, the more geminated it would be.

The other problem that results in incomprehensible utterances in both diphone based and syllable based syntheses is pitch mismatch. If the acoustic units have different pitches then the concatenation result will somewhat be of less quality, showing pitch variation from one acoustic unit to another. This problem can be minimized by recruiting professional speakers to record the corpus with constant pitch (Dutoit, 1997)

From the MOS test result, it is apparent that syllable based synthesis gives a better overall quality. This is because of the fact that discontinuity problem is observed more in diphone based synthesis than syllable based synthesis. Since the concatenation points for diphone based synthesis are larger in number, the perceivable discontinuity increases correspondingly. The discontinuity is easily perceived by the listeners and most listeners do not seem to like perceiving discontinuities.

The MOS test result is also affected by the inability to utter some words intelligibly in both diphone based and syllable based syntheses.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.1. CONCLUSION

This thesis work is an attempt to see into the possibility of developing a TTS system for the Amharic language, by taking into consideration intonation and duration changes. The TD-PSOLA technique has been used to modify pitch and duration of speech segments and subsequently synthesize the desired speech utterance.

As can be seen from the ORT result, the intelligibility of the utterances is much better in case of diphone based synthesis. Those words with geminated sounds are often the ones that create problem in case of both syllable based and diphone based synthesis. Germination is handled by doubling the consonant part of the geminated Amharic symbol. But extracting out the consonant part of CV-Syllables is difficult because it is not easy to detect the boundary correctly. Even if the exact boundaries are identified, increasing the length of the consonant sound may end up with a noise.

The problem of germination could best be handled by recording the geminated version of the syllables as a stand alone unit. This may add some burden to the memory of the system, but would give a better result.

On the other hand, the diphone based synthesis seems a little better than syllable based synthesis in that more words have been recognized correctly by the listeners. But diphone based synthesis suffers from the problem of perceivable discontinuity. The discontinuity is caused due to the boundary mismatches of the subsequent diphones.

5.2. RECOMMENDATION

This thesis work is an attempt to develop a TTS system for Amharic language, considering the possibility of handling prosodic effects using the TD-PSOLA. From what has been observed during the test session, the quality of the recorded corpus has a direct impact on the final results. Recording of corpus data should be carried out in a noise free laboratory.

The other impact of the recording session on the final result is the problem of pitch mismatch. The acoustic units are extracted from different word/phrase utterances. If there is a high variation in pitch of these utterances, then a resulting pitch variation will be imposed on the acoustic units, which in turn causes a pitch mismatch in the synthesized speech. Therefore, care should be taken in recording the data with more or less the same pitch. It is also good to record the whole corpus at one time because recording the corpus at different times may impose variations of pitch and amplitude (energy) between the corpus members.

The major quality problem observed, especially in the case of diphone based synthesis, was the problem of discontinuity. This problem arises partly due to the inability to exactly identify the steady state regions of the acoustic units. This has been observed especially when trying to extract out sixth order consonants of the Amharic language from the corpus data. Additionally, the TD-PSOLA algorithm, being a time-domain synthesis technique, lacks a means of adapting the spectral envelops of concatenated segments to one another.

Such a problem of spectral envelop mismatch can be handled by using the other variants of PSOLA technique, like FD-PSOLA, MBR-PSOLA, LP-PSOLA, which can operate in the frequency domain and give a better quality than the TD-PSOLA. It is also possible to use other

high quality techniques like the HNM which are believed to give high quality than the PSOLA technique.

Last but not least, care should be taken in selecting the corpus member words or phrases because sometimes a given word may produce something different from the actually needed sound because of the nature of the word in the language. Hence, it is advisable to record two or three alternatives for a single acoustic unit.

REFERENCES

Bender, L. M. *et al.* (1976). *The Ethiopian Writing System: The Languages of Ethiopia.* Ed. Bender, M. *et al.*, London: Oxford University Press

Black, Alan W. & Lenzo, Kevin A. (2003). *Building Synthetic Voices.* Available From: http://festvox.org/festvox/festvox_toc.html

Carlson, Rolf (1993). *Models of Speech Synthesis.* Irvine, California: National Academy of Sciences. Available From: http://www.speech.kth.se/~rolf/papers/wwnasqpsr.pdf

Chen, Jao-Hung & Kao, Yung-An (2001). *Pitch Marking Based on An Adaptable Filter and A Pick-Valley Estimation Method.* Computational Linguistics Society of R.O.C. Available From: http://www.google.com/url?sa=U&start=2&q=http://rocling.iis.sinica.edu.tw /CLCLP /Vol6-2/paper3.pdf&e=42

Donovan, Robert E. (1996). *Trainable Speech Synthesis.* England: Cambridge University Engineering Department. Available From: http://citeseer.nj.nec.com/donovan96trainable.html

Dutoit Thierry (1997). *An Introduction To Text-To-Speech Synthesis.* Dordrecht: Kluwer Academic Publishers.

Goncharoff, Vladimir & Gries, Patrick (1998). *An Algorithm for Accurately Marking Pitch Pulses in Speech Signals.* USA: University of Illinois at Chicago. Department of Electrical Engineering and Computer Science.

Available From: http://www.ece.uic.edu/~goncharo/vgpg98.pdf

Kinfe Tadesse (2002). *Sub-Word Based Amharic Word Recognition. An Experiment Using Hidden Markov Model.* M.Sc. Thesis. Addis Ababa University.

Laine Brehane (1998). *Text To Speech Synthesis of the Amharic Language.* M.Sc.Thesis. Addis Ababa University.

Lemmetty, Sami (1999). *Review of Speech Synthesis Technology.* Masters Thesis . Helsinki University of Technology.

Available From: *http://www.acoustics.hut.fi/~slemmett/dippa/index.html*

Long, Brian (2001). *Speech Synthesis and Speech Recognition.*

Available From: *http://www.blong.com*

Minghui, Dong. (2000). *Speech Synthesis Techniques.* Singapore: National University of Singapore, School of Computing.

Pasanen, Annti (2001). *Speech Synthesis.*

Available From: http://www.cs.tut.fi/sgn/arg/synteesi/pasanen.pdf

Rodman, Robert D. (1999). *Computer Speech Technology.* London: Artech House, Inc.

Syrdal, Ann et. al. (n.d). *TD-PSOLA Versus Harmonic Noise Pulse Model in Diphone Based Speech Synthesis.* Florham Park: AT&T Labs-Research.

Available From: http://www.research.att.com/projects/tts/papers/1998_ICASSP/

Türk, Oytun (2002). *Objective Tests on Spectral Envelop Estimation Methods for FD-PSOLA Based Pitch Scale Modification.* Istanbul, Turkey: SesTek Inc. Research Department.

Available From: http://www.sestek.com.tr/voice_conversion/docs/

Türk, Oytun (2003). *New Methods For Voice Conversion.* Boğaziçi University.

Available From: http://www.sestek.com.tr/voice_conversion/docs/nmvc_2003.pdf

ባዬ ይማም "ፊደል እንደገና" " *የኢትዮጵያ ቋንቋዎችና የሥነ ጽሑፍ መጽሔት*" ቁጥር 7" (1 – 32) $ 1997

APPENDICES

APPENDIX A: CORPUS WORDS AND THE CORRESPONDING ACOUSTIC UNITS

Word	Syllables	Diphones	Word	Syllables	Diphones
መርከብ	me, ke	me, ke, er	ላዲስ		Is
መኸራ	mu, ra	mu, uk, ra	ሊማት	lë	lë, ëm
ሚሚ	mi	mi, im	መልመል		Ul
ማስተማር	ma	ma, as, em, ar	አበበ	Be	be, ab, eb
ሜንጦ	më,	më, ën	ቡራኬ	Bu, kë	bu, kë,ur, ak
ምላስ	mï, la	mï, ïl, la	ቢራቢሮ	bi, ro	bi, ir, ro
ምስጋና	ga, na	ga, an, na,ïs	እመቤት	ï	Ï
ሞረድ	mo, re	mo, or, re	ቦርሳ	bo, sa	bo, sa
ተማሪ	te, ri	te, ri	ሩር	ru	ru
ተለያየ	le	le, el	ገበሬ	rë	rë
ቱባ	tu, ba	tu, ub, ba	ክር	kï	kï
ቲማቲም	ti	ti	ርጉም	rï	rï
ታኒካ	ta, ni, ka	ta, ni, ik, ka	ስጋ	sï	sï
ቴምብር	të, bï	të, bï, ïr	ሱባኤ	su	su
ትርጉም	tï, gu	tï, gu, um	ካልሲ	si	si
ቶሎ	to, lo	to, lo, ol	መነኩሴ	së	us, së
ነገሰ	ge, se, ne	ge, se, ne, eg, es	ሶረኔ	so	
ኑግ	nu	nu, ug	ጊንጥ	gi	gi, in
ኒቆዲሞስ	ni	od, os	አሮጌ	gë	gë, og
ናፍቆት		ot	ጎበዝ	go	go, ob
አመኔታ	në	am, në	ደባል	de	de
ንግርት	nï, gï	nï, ïg, gï	ዲዳ	da	id, da
ኖራ	no	no	ድብ	dï	dï, ïb
ለካ		ek	ሁዳዴ	dë	dë, ad, ud
አለባልታ		lu, al	ዶማ	do	do, om
ሊጋባ	li	li, ig			

Vowel	Example
ë	ረ
u	አደት
i	ኢላማ
a	አገኘ
ë	ኤደን
ï	እመቤት
o	አሮሞ

APPENDIX B. THE AMHARIC CHARACTER SET

Adapted from: Bender et al. (1976)

Order							Labialised				
1st	2nd	3rd	4th	5th	6th	7th					
ሀ	ሁ	ሂ	ሃ	ሄ	ህ	ሆ					
ለ	ሉ	ሊ	ላ	ሌ	ል	ሎ			ሏ		
ሐ	ሑ	ሒ	ሓ	ሔ	ሕ	ሖ					
መ	ሙ	ሚ	ማ	ሜ	ም	ሞ			ሟ		
ሠ	ሡ	ሢ	ሣ	ሤ	ሥ	ሦ					
ረ	ሩ	ሪ	ራ	ሬ	ር	ሮ			ሯ		
ሰ	ሱ	ሲ	ሳ	ሴ	ስ	ሶ			ሷ		
ሸ	ሹ	ሺ	ሻ	ሼ	ሽ	ሾ					
ቀ	ቁ	ቂ	ቃ	ቄ	ቅ	ቆ	ቈ	ቍ	ቋ	ቌ	ቍ
በ	ቡ	ቢ	ባ	ቤ	ብ	ቦ			ቧ		
ተ	ቱ	ቲ	ታ	ቴ	ት	ቶ			ቷ		
ቸ	ቹ	ቺ	ቻ	ቼ	ች	ቾ					
ኀ	ኁ	ኂ	ኃ	ኄ	ኅ	ኆ	ኈ	ኍ	ኋ	ኌ	ኍ
ነ	ኑ	ኒ	ና	ኔ	ን	ኖ			ኗ		
ኘ	ኙ	ኚ	ኛ	ኜ	ኝ	ኞ					
አ	ኡ	ኢ	ኣ	ኤ	እ	ኦ					
ወ	ዉ	ዊ	ዋ	ዌ	ው	ዎ					
ዐ	ዑ	ዒ	ዓ	ዔ	ዕ	ዖ					
ከ	ኩ	ኪ	ካ	ኬ	ክ	ኮ	ኰ	ኵ	ኳ	ኴ	ኵ
ኸ	ኹ	ኺ	ኻ	ኼ	ኽ	ኾ					
ዘ	ዙ	ዚ	ዛ	ዜ	ዝ	ዞ			ዟ		
ዠ	ዡ	ዢ	ዣ	ዤ	ዥ	ዦ			ዧ		
የ	ዩ	ዪ	ያ	ዬ	ይ	ዮ					
ገ	ጉ	ጊ	ጋ	ጌ	ግ	ጎ	ጐ	ጕ	ጓ	ጔ	ጕ
ደ	ዱ	ዲ	ዳ	ዴ	ድ	ዶ			ዷ		
ጀ	ጁ	ጂ	ጃ	ጄ	ጅ	ጆ					
ጠ	ጡ	ጢ	ጣ	ጤ	ጥ	ጦ			ጧ		
ጨ	ጩ	ጪ	ጫ	ጬ	ጭ	ጮ			ጯ		
ጸ	ጹ	ጺ	ጻ	ጼ	ጽ	ጾ			ጿ		
ፀ	ፁ	ፂ	ፃ	ፄ	ፅ	ፆ					
ፈ	ፉ	ፊ	ፋ	ፌ	ፍ	ፎ					
ፐ	ፑ	ፒ	ፓ	ፔ	ፕ	ፖ					
ቨ	ቩ	ቪ	ቫ	ቬ	ቭ	ቮ					