SPEECH PROCESSING FOR MAKHRAJ RECOGNITION
(DESIGN ADAPTIVE FILTER FOR NOISE REMOVAL)

SITI NURMAISARAH BT ABDUL AZIZ

This thesis is submitted as partial fulfillment of the requirement for the award of the Bachelor of Electrical Engineering (Electronics)

Faculty of Electrical & Electronics Engineering
Universiti Malaysia Pahang

NOVEMBER, 2010

"I hereby acknowledge that the scope and quality of this thesis is qualified for the award of the Bachelor Degree of Electrical Engineering (Electronics)"

Signature : ______________________________

Name : NURUL WAHIDAH BT ARSHAD

Date : 29 NOVEMBER 2010

"All the trademark and copyrights use herein are property of their respective owner. References of information from other sources are quoted accordingly; otherwise the information presented in this report is solely work of the author."

Signature : ________________________________

Author : SITI NURMAISARAH BT ABDUL AZIZ

Date : 29 NOVEMBER 2010

ACKNOWLEDGEMENTS

First of all, I want to thanks to Allah for giving me this opportunity, the strength and the patience to complete my project successfully, after all the challenges and difficulties that I have face it.

Foremost, I would like to express my greatest gratitude to my supervisor Madam Nurul Wahidah Bt Arshad, who have guide and helped me a lot throughout this final year project. This appreciation is also dedicated to Mr. Mohd Zamri Bin Ibrahim, Madam Nurul Hazlina Bt Nordin, and Madam Rosyati Bt Hamid and all the FKEE staffs, those who are really generous and helpful.

I also would like to thanks to my parents, for supporting me mentally and physically not just during finishing this tasks but also during my whole studies in order to become a good Muslims.

Finally, I would like to take this opportunity to thank all my friends and colleagues who have given their support and help.

Hopefully, this final year project will not be the end of my journey in seeking for more knowledge to understand the meaning of life.

ABSTRACT

Speech Processing for *MAKHRAJ* Recognition is a topic that very useful in many applications and environments in our daily day to improve *MAKHRAJ* for Arabic alphabets. In this project, it needs to design Adaptive Filter for noise removal. There are 30 Arabic, أ until ي but for this project, only 7 Arabic will be used as samples, أ until خ. The speech processing will be used to obtain same waveform output from two different situations, road and cafeteria. Least Mean Square (LMS) Algorithm based on Adaptive Filter technique is used to remove noise. Filter Design Toolbox provides many adaptive filter design functions that use the LMS algorithms to search for the optimal solution to adaptive filter, including system identification and noise cancellation. The filtered data will be processed to match the standard pronunciations and it will be integrated with filter design process in MATLAB. As a result, the noise will be removing and produce same waveform signal.

ABSTRAK

Pemprosesan Suara untuk Pengakuan Makhraj adalah satu topik yang sangat berguna dalam pelbagai aplikasi dan persekitaran dalam kehidupan seharian kita untuk meningkatkan Makhraj untuk huruf Arab. Dalam projek ini, ia perlu untuk mereka Penapis Adaptif untuk menyingkirkan bunyi bising. Ada 30 huruf Arab, أ sampai ي tapi untuk projek ini, hanya 7 huruf Arab akan digunakan sebagai sampel, أ sampai خ. Pemprosesan suara akan digunakan untuk mendapatkan keluaran gelombang yang sama dari dua situasi yang berbeza, jalan raya dan kafetaria. Least Mean Square (LMS) Algoritma berdasarkan teknik Penapis Adaptif digunakan untuk menyingkirkan bunyi bising. Filter Design Toolbox mempunyai banyak fungsi mereka penapis adaptif yang menggunakan algoritma LMS untuk mencari penyelesaian optimum untuk menapis adaptif, termasuk pengenalan sistem dan penyingkiran bunyi. Data yang ditapis akan diproses untuk menyesuaikan dengan sebutan sebenar dan akan diintegrasikan dengan proses penapis desain di MATLAB. Akibatnya, bunyi bising akan disingkirkan dan menghasilkan isyarat gelombang yang sama.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

LMS	Least Mean Square
FIR	Finite Impulse Response
MSE	Mean Square Error
SNR	Signal Noise Ratio
NLMS	Normalized Least Mean Square
SSLMS	Sign-Sign Least Mean Square
SDLMS	Sign-Data Least Mean Square
SELMS	Sign-Error Least Mean Square
RAM	Random Access Memory

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

This project is about Speech Processing for *MAKHRAJ* Recognition by using Adaptive Filter. This filter is use to remove or filter the noise and it is more efficient method. The main purpose of this project is to remove the noise in *MAKHRAJ* recording. It is because the existing system cannot recognize the wanted alphabets because of the noise. As an example *"ha"*, with the disturbance from the noise, the system may recognize wrong alphabet like *"kho"*.

This project uses two inputs. The first input is the distorted signal, the *MAKHRAJ* recording without noise. The second input is the desired signal, the unfiltered noise. The filter works to eliminate the difference between the output signal and the desired signal and outputs the difference, which, in this case, is the clean *MAKHRAJ* recording. When start the simulation, we hear both noisy signal from environment and voice from human. Over time, the adaptive filter filters out the noise so we hear only the voice from human.

For this project, the application that use is noise or interference cancellation where the filter adapts in real-time to remove noise by keeping the error small. The term of filter is often used to describe a device in the form of piece of physical hardware or software that is applied to a set of noisy data in order to extract information about a prescribed quantity of interest.

And the technique that applied in this project is Least-Mean-Square (LMS) algorithm to remove noise because it is easy and stable but the only disadvantage is its weak convergence. Besides that, it enjoys less computational complexity because of the sign present in the algorithm and good filtering capability because of the normalized term. LMS algorithm also represents the simplest and most easily applied adaptive algorithms.

According to the MATLAB software, there is Adaptive Filter by using Least Mean Square (LMS) algorithms Toolbox that helps this project to train the network.

1.2 OBJECTIVE

The objectives of this project are to:

i. Remove noise from unknown system.

ii. Design the system based on Least Mean Square (LMS) technique on adaptive filter.

iii. Developed *MAKHRAJ* recognition software using Adaptive Filter.

1.3 SCOPE OF PROJECT

There are three scopes of this project:

i. To remove noise of the speech recognition that able to recognize in road environment and cafeteria environment.

ii. To remove noise from human voice that produces filtered speech *MAKHRAJ* recognition by using Least Mean Square (LMS) algorithm.

iii. To develop software that can remove noise by using MATLAB environment.

1.4 PROBLEM STATEMENT

In our daily life, speech recognition is very important in order to improve the quality of our speech but most of the people take it for granted especially Muslim. They prefer improve their English rather than *MAKHRAJ*.

For that reason, this project is proposed in order to create a system that can be improving their speech of *MAKHRAJ*. This system can easily recognize the *MAKHRAJ* of human voice in two different environments, cafeteria and road.

1.5 THESIS OUTLINE

The Speech Processing for *MAKHRAJ* Recognition final thesis is a combination of 5 chapters that contains and elaborates specific topics such as Introduction, Literature Review, Methodology, Result and Discussions and Conclusions and Recommendation that applied in this project.

Chapter 1 basically is an introduction of the project. In this chapter, the main idea about the background and objectives of the project will be discussed. The basic concept of the project will be focused in this chapter.

Chapter 2 is about literature review to review the critical points of current knowledge including substantive findings as well as theoretical and methodological contributions to a particular topic about this project.

Chapter 3 will be discussed more detail about the method that used to achieve an objective of this project. It wills shows and explain the flow chart that been used to write the coding, developing the process using the MATLAB.

Chapter 4 discusses all the results obtained and the limitation of the project. All discussions are concentrating on the result and performance of the speech recognizer.

Chapter 5 will be explained about the problem and the recommendation for this project.

CHAPTER 2

LITERATURE REVIEW

2.1 INTRODUCTION

MAKHRAJ is a set of range of organs in speech that will create variety of letter with its own character that is one of the vocalized forms of human communication. Each letter is created out of the phonetic combination of a limited set of vowel and consonant speech sound units that can be differentiate from others.

MAKHRAJ recognition is important to help in practicing the pronunciation the letters correctly. So, in this chapter, the basic knowledge and fundamental concept in creating the *MAKHRAJ* recognition will be discussed. This *MAKHRAJ* recognition project is using Adaptive Filter as a main processer.

2.2 SPEECH PROCESSING FOR MAKHRAJ RECOGNITION

Speech is the way of choice for humans to communicate. There are no special equipment required, no physical contact required, no visibility required, and can communicate while doing something else. Speech processing includes speech coding, speech synthesis, speech recognition, identity verification and enhancement.

Speech coding is to transmit or store a speech waveform using a few bits as possible while retaining high quality because to save bandwidth in telecoms applications and to reduce memory storage requirements.

Speech synthesis is to convert a text string onto speech waveform because for technology to communicate when a display would be inconvenient because too big, eyes busy, via phone, in the dark and moving around [1].

Speech recognition is the process of converting spoken input to text or sometimes referred to as speech-to-text. There are a few of the basic terms and concepts that are fundamental to speech recognition:

i. Utterance - When the user says something

   The speech recognition engine is "listening" for speech input. When the engine detects audio input (a lack of silence) the beginning of an utterance is signaled. Utterances are sent to the speech engine to be processed. If the user doesn't say anything, the engine returns what is known as a silence timeout that indicated there was no speech detected within the expected timeframe. An utterance can be a phrase or a sentence.

ii. Pronunciations

   One piece of information that the speech recognition engine uses to process a word is its pronunciation, which represents what the speech engine thinks a word should sound like. Words can have multiple pronunciations associated with them. For example, the word "*pa*" has at least two pronunciations in the transliterating foreign words in Arabic: "*pa*" in the Jawi script for "ڤ"and in Persian, Urdu, and Kurdish for "پ".

iii. Grammars

   Grammars define the domain, or context, within which the recognition engine works. The engine compares the current utterance against the words and phrases in the active grammars. If the user says something that is not in the grammar, the speech engine will not be able to decipher it correctly.

iv. Accuracy

The performance of a speech recognition system is measurable and perhaps the most widely used measurement is accuracy. Arguably the most important measurement of accuracy is whether the desired end result occurred. Measurement of recognition accuracy is whether the engine recognized the utterance exactly as spoken. This measure of recognition accuracy is expressed as a percentage and represents the number of utterances recognized correctly out of the total number of utterances spoken. It is a useful measurement when validating grammar design. For example, if the engine returned "*aaaliif*" when the user said "*alif*" this would be considered a recognition error. Based on the accuracy measurement, there must analyze the grammar to determine if there is anything that can do to improve accuracy. For instance, it might need to add "*aaliif*" as a valid word to grammar [2].

2.3 ADAPTIVE FILTER

Adaptive filtering involves the changing of filter parameters (coefficients) over time, to adapt to changing signal characteristics. There are four application of Adaptive Filter [3]:

i. System Identification - Using adaptive filters to identify the response of an unknown system such as a communications channel or a telephone line.
ii. Inverse System Identification - Using adaptive filters to develop a filter that has a response that is the inverse of an unknown system.
iii. Noise or Interference Cancellation - performing active noise cancellation where the filter adapts in real-time to remove noise by keeping the error small.
iv. Prediction - describes using adaptive filters to predict a signal's future values.

In noise cancellation, adaptive filters will remove noise from a signal in real time. Here, the desired signal, the one to clean up, combines noise and desired information. To remove the noise, feed a signal $n'(k)$ to the adaptive filter that represents noise that is correlated to the noise to remove from the desired signal.

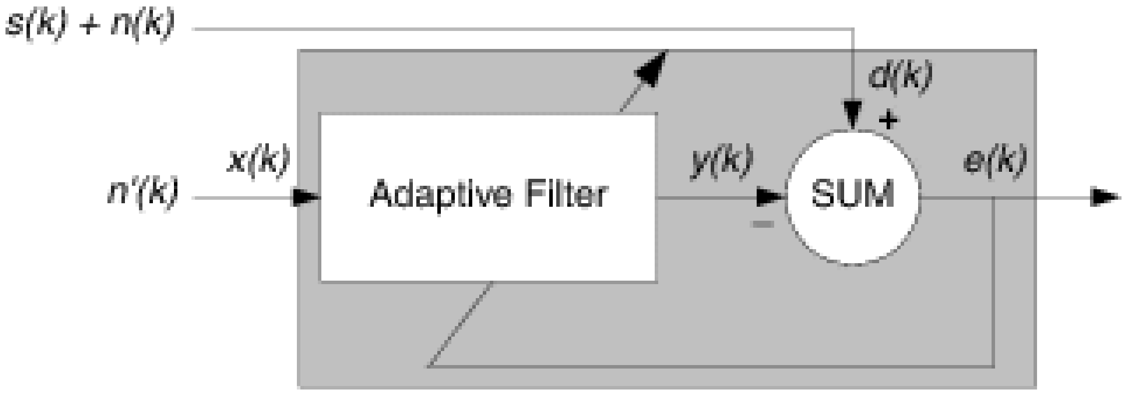

Figure 2.1: Using an Adaptive Filter to Remove Noise from an Unknown System.

So long as the input noise to the filter remains correlated to the unwanted noise accompanying the desired signal, the adaptive filter adjusts its coefficients to reduce the value of the difference between output signal, *y(k)* and desired signal, *d(k)*, removing the noise and resulting in a clean signal in estimation error, *e(k)*. Notice that in this application, the error signal actually converges to the input data signal, rather than converging to zero [4]. On this basis of this measure, the adaptive filter will change its coefficients in an attempt to reduce the error. The coefficient update relation is a function of the error signal squared and is given by

$$h_{n+1}[i] = h_n[i] + \frac{\mu}{2}\left(-\left(\frac{\delta}{\delta h_n[i]}((|e|)^2)\right)\right)$$

The term inside the parentheses represents the gradient of the squared-error with respect to the ith coefficient. The gradient is a vector pointing in the direction of the change in filter coefficients that will cause the greatest increase in the error signal. Because the goal is to minimize the error, however, Equation 1 updates the filter coefficients in the direction opposite the gradient; that is why the gradient term is negated. The constant, μ is a step-size, which controls the amount of gradient information used to update each coefficient. After repeatedly adjusting each coefficient in the direction opposite to the gradient of the error, the adaptive filter should converge;

that is, the difference between the unknown and adaptive systems should get smaller and smaller. To express the gradient decent coefficient update equation in a more usable manner, we can rewrite the derivative of the squared-error term as

$$\frac{\delta}{\delta h[i]}((|e|)^2) = 2\frac{\delta}{\delta h[i]}(e) \tag{2.1}$$

$$= 2\frac{\delta}{\delta h[i]}(d-y)e \tag{2.2}$$

$$= \left(2\frac{\delta}{\delta h[i]}\left(d - \sum_{\mathrm{i=0}}^{\mathrm{N-1}}(\mathrm{h[i]x[n-i]})\right)\right)e$$

$$\frac{\delta}{\delta h[i]}((|e|)^2) = 2(-(x[n-i]))e \tag{2.3}$$

which in turn gives us the final LMS coefficient update,

$$h_{n+1}[i] = h_n[i] + \mu ex[n-i] \tag{2.4}$$

The step-size, μ directly affects how quickly the adaptive filter will converge toward the unknown system. If μ is very small, then the coefficients change only a small amount at each update, and the filter converges slowly. With a larger step-size, more gradient information is included in each update, and the filter converges more quickly; however, when the step-size is too large, the coefficients may change too quickly and the filter will diverge. (It is possible in some cases to determine analytically the largest value of μ ensuring convergence.) [5]

The objects use various algorithms to determine the weights for the filter coefficients of the adapting filter. While the algorithms differ in their detail implementations, the LMS and RLS share a common operational approach to minimize the error between the filter output and the desired signal [6].

2.4 LEAST-MEAN-SQUARE (LMS) BASED

The least-mean-square (LMS) algorithm is a linear adaptive filtering algorithm that consists of two basic processes:

i. A filtering process, which involves computing the output of a transversal filtering produced by a set of tap inputs and generating an estimation error by comparing this output to a desired response.
ii. An adaptive filtering, which involves the automatic adjustment of the tap weights of the filter in accordance with the estimation error.

The combination of these two processes working together constitutes a feedback loop around the LMS algorithm. First, have a transversal filter (adaptive filter) that responsible for performing the filtering process. Second, have a mechanism (unknown system) for performing the adaptive control process on the tap weights of the transversal filter [7]. The filter calculates the filter weights, or coefficients that produce the least mean squares of the error between the output signal and the desired signal (minimize the error).

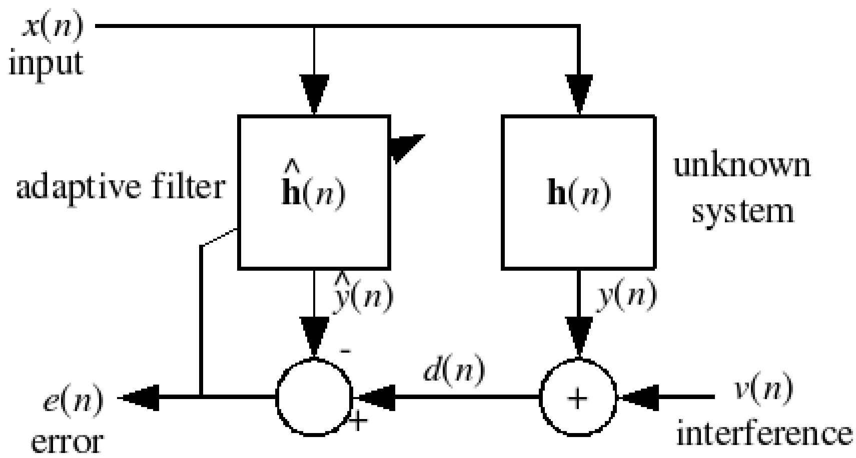

Figure 2.2 : Least-Mean-Square Implementation