The Islamic University Gaza

Deanery of Higher Studies

Faculty of Engineering

Computer Engineering Department

Sound Visualization for Deaf Assistance Using Mobile Computing

Submitted by:

Mahmoud S. Alhabbash

Supervised by:

Dr. Aiman Abu Samra

A Thesis Submitted in Partial Fulfillment of Requirements for the Degree of Master in Computer Engineering.

1435 هـ -2013 م

DEDICATION

To My Father, and Mother,

To My Wife,

To My Family, and Friends,

ACKNOWLEDGEMENT

All thanks are to Allah the almighty, who guide me to accomplish this work, so all praise is to Allah.

Also the completion of this work cannot done, without all people around me, especially my advisor Dr. Aiman Abu Samra, who guided me through this research; his patience and support leaded me to success through and until the completion of this thesis.

LIST OF CONTENTS

LIST OF ABBREVIATIONS

STFT	Short Time Fourier Transform
MFCC	Mel-Frequency Cepstral Coefficients
LPC	Linear Predictive Coding
CBIR	Content Base Image Retrieval
k-NN	K-Nearest Neighbor
DTW	Dynamic Time Warping
DCT	Discrete Cosine Transform
SDK	System Development tool Kit
GPU	Graphical Processing Unit
ASR	Automated Speech Recognition

LIST OF FIGURES

LIST OF TABLES

عرض الصوت لمساعدة الصم باستخدام حوسبة الهواتف المحمولة

محمود صبحي الهباش

ملخص الدراسة

تعرض هذه الدراسة طريقة جديدة لعرض الصوت بغية مساعدة الصم في التعرف على الأصوات المحيطة بهم. وذلك بعرض اهم مميزات الصوت المتغيرة بالتزامن مع عرض رموز الأصوات التي يتم التعرف عليها بطريقة سهلة للمستخدم. ولإمكانية عرض الأصوات بشكل فعال ، تم استخدام مميزات للصوت تدعى MFCC لكي تعرض المميزات المصنفة لخصائص الصوت بشكل جلي. وللتغلب على مشكلة تعدد ابعاد طريقة MFCC التي تساوي 39 بعدا تم عرض بعد واحد فقط يمثل هذه الابعاد التسع والثلاثون ألا وهي القيمة الناتجة عن مقارنة مميز مرجعي من نوع MFCC مع المميز الوارد. لهذا الغرض تم طرح طريقة جديدة لمقارنة هذا النوع من مميزات الأصوات تتغلب على المشاكل الموجودة في خوارزميات مقارنة الأصوات المعروفة , والتي تقوم بمقارنة كل بعد على حدة مما نتج عنه قرارات خاطئة في الحكم على تشابه بعض الأصوات.

ولجعل التطبيق اكثر قابلية للاستخدام تم ادراج خاصية التعرف على الأصوات , حيث أن كل شريحة زمنية من الصوت اخضعت لخوارزمية تصنيف من نوع K-NN يتم بواسطتها التعرف على الاحداث الصوتية السريعة وكل ثانية زمنية يتم تخزينها وإحالتها إلى خوارزمية تصنيف ثانية تدعى DTW مخصصة للتعرف على السلاسل الزمنية المتغيرة. كلا الخوارزميتين تعملان في نفس الوقت وتسلمان نتيجتهما إلى جزئية العرض.

هذا النظام مصمم كتطبيق يعمل بواسطة الهواتف الذكية التي تعمل بنظام Android وتمت برمجته باستخدام لغة Java لذلك كان هناك عدة اعتبارات متعلقة بتعقيدات الخوارزميات اخذت بالحسبان , حيث يتم العرض باستخدام وحدة عرض الرسومات GPU بالأجهزة الذكية التي تحوي هذه الوحدة بالتالي يمكننا ضمان سلاسة العرض وسرعته. إضافة لذلك تم تصميم هذا النظام بناءً على مقابلات أجريت مع خمسة اشخاص ممن لديهم إعاقة سمعية ,آخذين بعين الاعتبار طريقتهم المفضلة لعرض الصوت وفيما بعد تم اختبار النظام بواسطة نفس الأشخاص الخمسة وتم تقييمه بناء على طريقة تفاعلهم معه ونتج عن ذلك طريقة أسهل للتعامل مع عرض الصوت ومناسبة أكثر للأشخاص ذوي الخبرة القليلة بالتقنية الحديثة.

ABSTRACT

This thesis presents a new approach to the visualization of sound for deaf assistance that simultaneously illustrates important dynamic sound properties and the recognized sound icons in an easy readable view. .In order to visualize general sounds efficiently , the MFCC sound features was utilized to represent robust discriminant properties of the sound. The problem of visualizing MFCC vector that has 39 dimension was simplified by visualizing one-dimensional value, which is the result of comparing one reference MFCC vector with the input MFCC vector only. New similarity measure for MFCC feature vectors comparison was proposed that outperforms existing local similarity measures due to their problem of one to one attribute value calculation that leaded to incorrect similarity decisions.

Classification of input sound was performed and attached to the visualizing system to make the system more usable for users. Each time frame of sound is put under K-NN classification algorithm to detect short sound events. In addition, every one second the input sound is buffered and forwarded to Dynamic Time Warping (DTW) classification algorithm which is designed for dynamic time series classification. Both classifiers works in the same time and deliver their classification results to the visualization model.

The application of the system was implemented using Java programming language to work on smartphones that run Android OS, so many considerations related to the complexity of algorithms is taken into account. The system was implemented to utilize the capabilities of the smartphones GPU to guarantee the smoothness and fastness of the rendering. The system design was built based on interviews with five deaf persons taking into account their preferred visualizing system. In addition to that, the same deaf persons tested the system and the evaluation of the system is carried out based on their interaction with the system. Our approach yields more accessible illustrations of sound and more suitable for casual and little expert users.

CHAPTER 1

INTRODUCTION

This chapter introduces the problem that a deaf person faces in his daily life and explains how deaf can overcome these difficulties by his own or by various aid devices. The vision sense that deaf person had can be the best solution, where we explained how we could take advantage of it to build our proposed system. This chapter also explains the goals of this study and the methodology we followed to achieve the resulted system.

1.1 Sound awareness

People use sound mainly to gain awareness of the state of the world around them. For example, many everyday devices such as mobiles, doorbells, ovens, and telephones produce sound to make us aware of their states. At office, sounds of co-workers provide awareness of whether they are still working or one is alone in an office. Similarly, at street, one might hear the horn of cars and guess a passing car is becoming closer.

Hearing is a very important sensory function to human beings [1]. However, not all the humans have the sense of hearing. According to Palestinian Central Bureau of Statistics, more than 43617 people in Gaza and West Bank are deaf and 95% of them suffer from the illiteracy [2] as they need special equipment and learning criteria. Globally, the National Center for Health Statistics states that, more than thirty-seven million people in the United States have some form of hearing loss, approximately seventeen percent of the population [3]. While the largest percentage of persons with hearing loss are sixty-five or older, there are many persons lived all entire their lives with an inability to hear most sounds. One child in every thousand is born deaf or becomes deaf by the age of three [4].

1.2 How hearing impaired can experience sound

For centuries, many devices have been made to allow deaf persons to function normally in a hearing society. Today, deaf persons have a variety of “hearing aids”; including cochlear implants, and assistive listening devices and some other assistive tools as shown in figure (1.1). These devices do not improve hearing like corrective lens can improve vision; they simply make certain sounds louder or softer. The majority of “hearing aids” are small personal devices that manipulate sounds before they reach the person’s ear and amplify them to levels that are suitable to the wearer [3]. They are made in a variety of shapes and sizes, fitting inside the ear, around the ear, or as an external module connected to a headset. The common property of these devices that, each device has a microphone that receives sound and a processor that converts the sound into an electrical signal, and then amplifies or transforms the sound, into another form of acoustic properties more perceptible to the user but they are optimized for speech only[5].

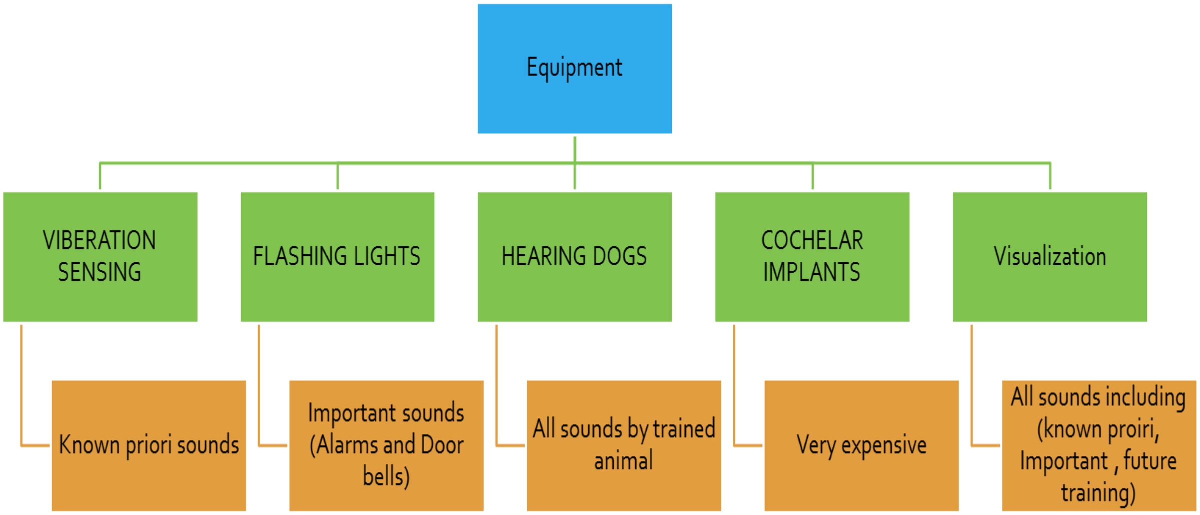
Equipment
VIBERATION SENSING
FLASHING LIGHTS
HEARING DOGS
COCHELAR IMPLANTS
Visualization
Known priori sounds
Important sounds (Alarms and Door bells)
All sounds by trained animal
Very expensive
All sounds including (known proiri, Important , future training)

Figure 1.1 : Equipment used often used to help hearing impaired

1.2.1 Assistive listening devices based on vibrations

There are many ways that a hearing-impaired person can experience sound, both with his or her own hearing faculties and with other senses. We often think only about how sound vibrations reach our ears, while these vibrations also reach the rest of our body. There parts of the human body other than the ear that can sense the vibration of sounds. Research has shown that hearing-impaired person can experience sounds similar to normal hearing people [6]. According to Dr. Dean Shibata [7], "*the perception of the musical vibrations by the deaf is likely every bit as real as the equivalent sounds, since they are ultimately processed in the same part of the brain*". Russ Palmer [8] made a study about the way that sound can create sympathetic vibrations in the body. He noted that lower frequencies can be felt in the feet, legs, and hips; mid-range frequencies can be felt in the stomach, chest, and arms; and higher frequencies can be felt in the fingers, head, and hair. Similarly, certain parts of our bodies react in certain ways with some specific frequencies. Most of these frequencies are in lower range of our hearing, with some lying below our range of hearing .This explains why sometimes we feel string vibrations from music in our lungs or in our eyes.

The use of vibration devices to experience sound (mainly musical sound) has been applied in several ways over the past few years. Several simple methods, such having hearing impaired individuals sit close to sources of vibrations or holding balloons, have been applied in the past , but in the recent years many advanced technologies are designed to enhance the experience of the sound vibration like never before. Russ Palmer designed a sound system that amplifies music vibrations through feet [8]. Similarly, Strike-A-Chord Company produces vibration chairs that amplify music and direct its vibrations to sections of chairs.

The vibration itself is not enough for hearing impaired person to experience sound, so other techniques were merged to produce better experience. In some deaf schools a specific room is made for listening to music where several 100-watt amplifiers are positioned to the face of deaf, sending vibrations through the floor and exciting the nerves of listeners. In addition to these amplifiers, there is an audio spectrum analyzer device that displays the properties of music as visual information (colors, lines, etc.).

1.2.2 Assistive listening devices based on vision

Vision can help a hearing-impaired individual extract meaning (or assign meaning) to sound events, if the sound visualizing describes the sound properly for hearing impaired. The rapid development of video technology has inspired many researches for sound expression on visual displays. Even without video, a hearing impaired individual can experience the sound. The movements of speaker lips, the facial expression, the rise and fall of drum sticks can offer a lot of information to someone who cannot hear. Also, the use of sign language can be extremely useful tool to those who cannot hear.

Implementing aid device for hearing impaired based on vision is very important and offers much useful information for him. The problem here arises from that hearing impaired individual already uses vision for his own aid not only for experiencing sounds. The designing of visual displays for sound must not encounter on disabling the human vision for things other than the sound visualization. Beside this design consideration, there are other many factors that are related to the hearing impaired individual must be taken in consideration.

Technology to represent sound visually has existed for several years. One of the most familiar examples of sound visualization software is the Microsoft Windows Media Player [9]. The software does not give a true representation for the music, but it responds visually to various aspects of music such as amplitude, rhythm, and tempo. The user can simply detect the meaning of the displayed image for some specific sounds.

1.3 Challenges in sound Visualization

Sound visualization research falls under the general area of sound processing and image/video processing [10]. Image and video processing are essential parts of sound visualization research as they can be used as textural representation of audio input. However, there are many aspects relevant to sound visualization. In [10] Zhang illustrated in Figure (1.2) the main research areas that are relevant to sound visualization. The thickness of the lines indicates the importance of each area to this research. Bi-directional arrows are indicative of reciprocal relationships.

Other related research areas include machine learning, computer vision, and data mining. For example, the similarity measures (which play essential rules in data mining) are needed for comparing different sound features. Sound visualization will benefit from any improvements in these related research areas.

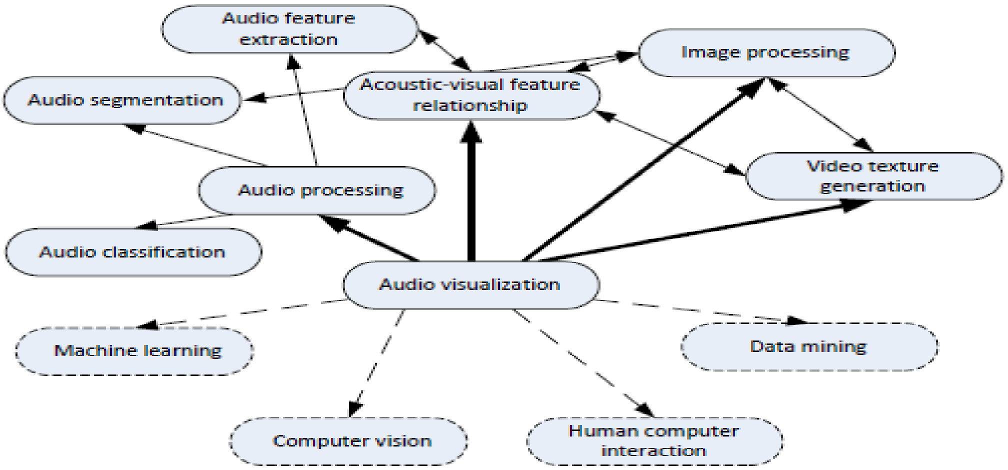

Figure 1.2 : Aspects of research areas related of audio visualization [10].

Sound visualization’s success depends on correct extraction of sound features and similarity measure between these features. The sound features can be used in the production of resultant images/videos generation and the similarity measure can be used for differentiation between them.

To summarize, the sound visualization research in this thesis explores the following research questions:

1. How can we construct an image or sequence of images that represent real time sounds simple for hearing impaired individual to understand?
2. What sound features can be used to accurately describe the heterogeneous sound environment?
3. How can we differentiate between sound features that are mapped to various sounds and represent these differences on the generated image sequence?
4. How can features of sound be mapped into the features of the generated images?
5. Is it possible to use or develop a sound classification method that classifies general sounds accurately enough to allow visualization of real time sound?

1.4 Approach

The approach that was used to solve the sound visualization for deaf starts from capturing sound from a portable device which can be carried easily by deaf individual , then extracting the most describing features for every time frame of sound , while applying suitable similarity measure between the extracted features in real time. The visualization is done based on the result of the applied similarity measure by assigning different colors to specific intervals of that result.

1.4.1 Sound Features

Many different types of sound features have been proposed to describe sound coming from speech recognition community [11][12][13][14][15]. Researchers tried to taxonomy the features that can be used to describe sounds. Sound features can be categorized into basic groups [16]; temporal , energy , spectral , harmonic , and perceptual features . We will describe each category .

(a) Temporal shape features :

Can be computed from the wave form of the signal . Examples: attack-time . temporal increase/decrease , effective duration .

(b) Temporal features :

Like auto-correlation coefficient, zero crossing rate .

(c) Energy features :

Features representing various energy content of the signal . Examples: global energy , harmonic energy , noise energy .

(d) Spectral shape features :

Features resulted from applying Short Time Fourier Transform (STFT) to the signal. Examples: centroid , spread , skewness , kurtosis , slope , Mel-Frequency Cepstral Coefficients (MFCC) .

(e) Harmonic Features :

Features computed from the Sinusoidal Harmonic modeling of the signal . Examples : harmonic/noise ratio , harmonic deviation .

(f) Perceptual features :

Features computed using a model of the human ear process. Examples: relative specific loudness, sharpness, spread .

In this thesis we will focus on spectral shape features as it proved higher discrimination results than other features[16].

1.4.2 Similarity measures

There are many methods that can be used to compare and derive the differences between two vectors. They are grouped into main categories according to their functionality.

a) Local dissimilarity/distance measure

Similarity measures that compare the differences between two vectors according to the values of these vectors only. Usually, this type of similarity measures makes one to one attribute comparisons. Such as Euclidean [19], cosine[20],…,etc.

b) Statistical similarity measures

Similarity measures take into consideration the statistics parameters of the dataset to compare between two vectors from that dataset. Such as Kullback Leibler distance [21], and the Hotelling T2-Statistic distance [22].

The local similarity measures are more suitable to our proposed system because it is hard to have full dataset for all environmental sounds, so statistical measures will be biased according to the dataset.

1.5 Research overview

1.5.1 Objective

In our method, we focus on helping the deaf for experiencing surrounding sound by visualizing the sound. Due to the deaf lack of experience of sound generally, the visualization of sound rendering should be readably simple, real time runnable, and portable.

Due to the sensitivity of resulted application for deaf individual life, the application performance should be high and capable of processing many input sound classes in real time. Besides that, the accuracy of results that will be displayed on portable device screen should be high because there is no other way that the deaf can guess the correctness of the visualized result.

Our proposed system will pick the most suitable sound features, similarity measures, classifiers , rendering frame work to achieve the main goals of the system and this not an easy job at all!.

1.5.2 Methodology

Our method starts be making interviews with deaf persons living in different environments by considering their profession, capabilities, and ages. The interviews should give a whole overview about good visualization system behavior. In addition to that, the interviews should present some existing applications resulted from previous studies and collecting the deaf impressions about these applications and guessing their points of weakness and power.

As mentioned in approach section, many different sound features can be used for representing sound. We tries to pick the most discriminative sound features taking into consideration the computation power of the device where the system will be implemented. Since there are many studies evaluating these features, we will take advantage to pick the most suitable features rapidly without making exhaustive experiments on every previously proposed sound features.

Sound features are not the only obstacle that our methodology faces. Comparing between two sound features is also an obstacle; since sound features vectors have some special properties led us to propose new similarity measure for this purpose.

Rendering the visualized sound is the step that we have made based on the interviews with deaf after analyzing their preferred picture about the proposed system. The interview made us aware of some special needs that we did not take into consideration because we can experience sounds not like the deaf.

Finally, after combining the results of the whole previous steps we noticed that it is hard for the deaf to use our proposed system directly without continuous help, so we added recognition module to the system that classifies some prior known sounds to help the user. We made a database that contains multiple important classes of sound and trained our classifiers on it taking into consideration using some lightweight classifiers for real time classifications. We tend to update the training set of the classifiers and update the application when the users connect their smart phones to the internet.

1.5.3 Contribution

Our contribution to sound visualization was done in more than one direction to develop system that helps the deaf to experience sound, despite their loss of very important sensation, which is hearing.

- We used the advantage of deaf vision to handle sounds as visual signals instead of sound signals by our visualizing system, which simplifies this job by acquiring the sound from the surrounding environment and processing it to be displayed in simple way to understand.
- The system was not developed through straight easy steps, but required additional contribution to overcome the existing similarity measures shortcomings.
- Another contribution is the usage of smartphone capabilities to develop such system where performance optimization is required as so as taking into consideration the computational complexity of used algorithms.

The proposed system can be viewed as sound class free system that doesn’t depend on specific data set to handle specific sound class . It was designed to work with any type of sound.

1.5.4 Thesis Organization

In Chapter (2) we will review the related work for sound visualization problem, we will describe some of the work that was done in the field, focusing on the advantages /disadvantages of resulted application that resulted from those works and we will review the result for each work in order to develop our application.

Then we will overview some background theory in Chapter (3), explaining some basic steps for processing sound signals to extract sound features. We will explain the most well-known sound features in literature in detail explaining our point of view about their validity to be implemented in our system. After that, we will present most generally used local similarity measures definition in order to evaluate them in Chapter (5).

Chapter (4) describes our proposed system in detail. It explains the interview mechanism that we made with deaf participants and presents the summary of our and their view about the preferred visualization system. It describes every single step in building our visualization system , starting from the input of the sound, passing with feature extraction methods by explaining the implemented equation for each feature extraction method according to the theoretical presentation in Chapter (3). In addition, in Chapter (4), we describe the proposed similarity measure for comparing MFCC sound features effectively explaining the drawbacks of other similarity measures. Finally, the visualizing system framework was expressed in detail to be later evaluated.

Chapter (5) views the data set that was used to develop the proposed system and describes the system environment phases and procedures. In addition, it describes the results of each experiment and presents the overall evaluation of our proposed system.

Finally, Chapter (6) includes the conclusion of our research, which summarizes research remarks and notes about our research.

1.6 Conclusion

We introduced the problems of experiencing sounds by hearing impaired and presented our solution by building visualization system of sound. The sound visualization has many challenges related to audio processing, audio feature extraction, audio classification, and image processing. We introduced the methodology that we followed to accomplish our mission by introducing new similarity measure and new visualization criteria.

CHAPTER 2

RELATED WORKS

In this chapter, we present previous related works to our research. We've started with early devices that tried to display sound as any other signal. We advanced with more specified sound visualization systems. We presented the system that tried to display some specific sound classes for training or aiding people. Finally, the systems that proposed to aid deaf people are explained in details and analyzed from our point of view.

2.1 Sound display as general signal

The term "Sound visualization", also called "Audio visualization" has been defined by Nomura, Shiose, Kawakami, Katai and Yamanaka as reading sounds [23]. There are many researches that were done in sound visualization field. This chapter contains a review of existing approaches to sound visualization, we are going to focus on them starting from historical review of sound visualization methods and discussing in detail the drawbacks of the existing methods that led us to propose our sound visualizing method.

Various attempts have been made to visualize different kinds of sounds; some of them were prior to the computer invention. The first attempt at visualizing sound can be traced back to the development of the phonautograph [24] see Figure (2.1). A phonautograph is a device for converting sound into visible traces. Invented by Frenchman *Édouard-Léon Scott de Martinville* , it could be used to visually study and measure the amplitude envelopes and waveforms of speech and other sounds, or to determine the frequency of a given musical pitch by comparison with a simultaneously recorded reference frequency [25]. The images from a phonodiek (advanced phonautograph) illustrated that the differences of the sounds could be visually presented using different wave shapes. Similar to modern oscilloscopes, the phonautograph and modern oscilloscopes can display the wave shape along the time axis.

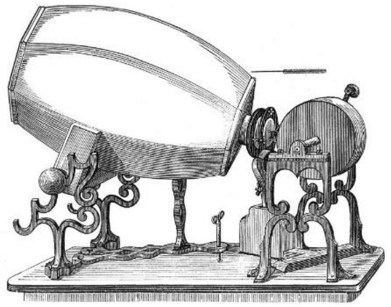

Figure 2.1: An early phonautograph (1859) [24].

2.2 Advanced sound visualization

Besides representing sound amplitudes as waves in the time domain using modern oscilloscope, sounds may be visualized using spectrograms.

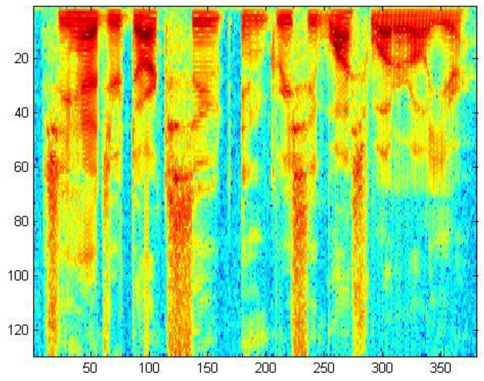

Figure 2.2 : Spectrograph for speech

The spectrograms are visual representation of the spectrum of frequencies in a sound. They represent an audio signal in the frequency domain, as shown in figure (2.2). The magnitudes of the windowed discrete-time Fourier transform are shown against the two orthogonal axes of time and Frequency. Experts may directly derive information from sound spectrographs such as bandwidth (wide band or narrow band), or even recognize certain words by reading their spectrographs [26].

The oscilloscope and spectrogram representations of sound are hard to understand by non-professional users, so more easier methods are required to represent the sound visually.

Tzanetakis and Cook proposed a method for visually representing audio files by *TimbreGrams* images [27], light and bright colors typically correspond to speech and singing, while purple and blue colors represent classical music segments see figure (2.3). They depended on the human color perception and the pattern recognition capabilities to extract timbral and temporal information from *TimbreGrams* .

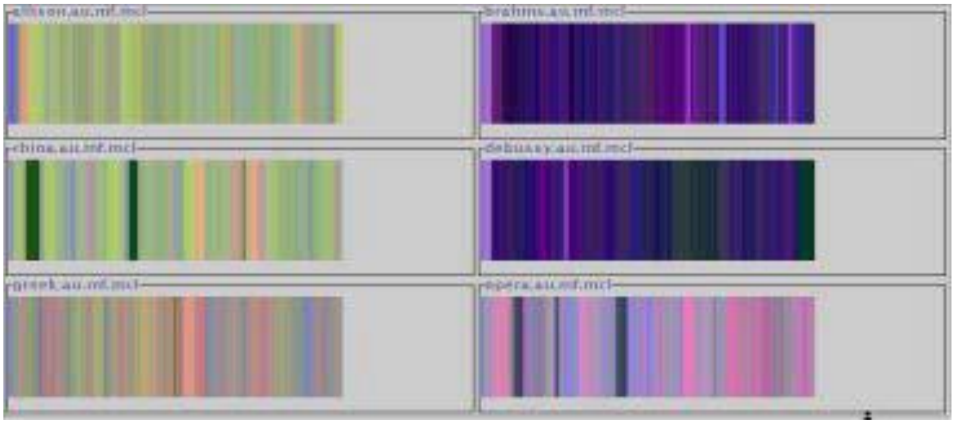

Figure 2.3: TimbreGrams for speech and classical music [27].

This approach can be used as an effective tool for speech/music discrimination as non-professional users can easily distinguish speech from classical music by the colors in the images. We mainly depend on this approach to build our visualizing system. However, this method does not meet our audio visualization requirements because it does not give enough information about the content of an input file especially for deaf person.

Margounakis and Karatsoris proposed similar work for music only [28], in which visualizing features are limited to a specific feeling they described as the "chromatic of music". A chromatic is adopted from color models and used to describe a user's feeling for a piece of music. The drawback of this visualizing system is that it cannot be used effectively on general audio input or by non-professional users.

Other visualizing approaches represented the sound by varying shapes as visual feature. One approach generated black-white image containing shapes representing sound features [29]. In this algorithm, any sound signal f(t) is transformed into a three-dimensional phase space $\Phi3(f)=(f(t),f'(t),f''(t))$. Simple shapes used to represent periodic signals, for example a sinusoid represented by a circle or an ellipse. Natural sounds that contain different sinusoidal signals have more complicated shapes. The images in Figure (2.4) represent the same note played on three different instruments. A modified version of this approach made the roughness of the curve representing the consonance of a chord [30].

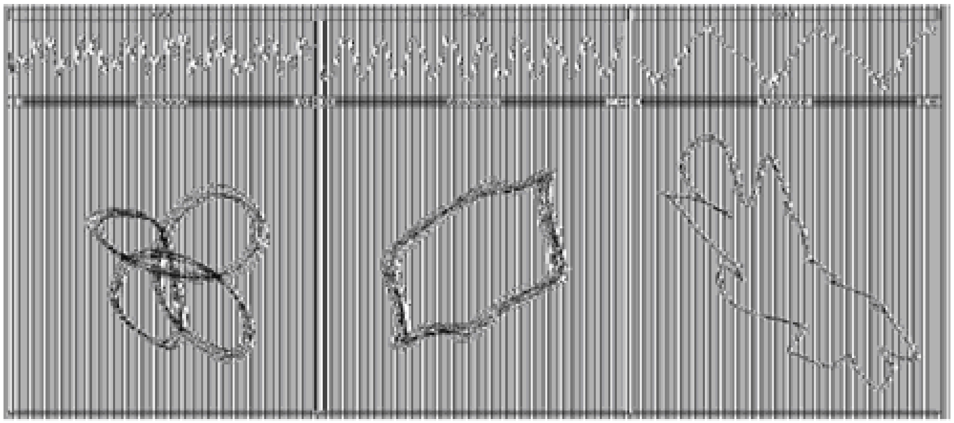

Figure 2.4: Visualization of a single note on various instruments in phase space [29].

Some audio visualization systems (especially music) aim to use the value similarity/dissimilarity between audio pieces to be an element in a matrix and represented as a pixel in the final. The audio pieces are parameterized into acoustic feature vectors where similarity/dissimilarity measures can be applied, so that similar repeating elements are visually distinct, allowing identification of structural and rhythmic characteristics. This approach firstly used by Foote’s visualization of music using self-similarity [31] produced a check-board image, intended to show the resemblance among the pieces of music input. This approach has been extended to structural analysis for indexing and thumbs nailing [32] [33].

More recently, Karahalios and Hart introduced a new method for visualizing the structure of music showing consonant intervals between notes and common chords [34]. All of these methods are made for knowledgeable users and are strictly designed to visualize musical input, for example songs often have repeating regions, from the resultant structure image. Viewers can find the repeating patterns which are important for music summarization. But they do not help in understanding the content beyond the structure.

Other works have also represented audio properties by other visual features in other applications, such as loudness by height of a sphere [35], reverberation by color [36], and pitch by light intensity [37].

2.3 Music visualizing

Music visualization is the most commonly studied topic in audio visualization [10]. Most research in audio visualization handles music input only. The audio features that best describe music are not necessarily the best for general audio input. Some audio features of music, such as tempo, are not likely suitable to general audio files so the approaches for visualizing music are not preferable for general audio visualization. Any system for the visualization general audio input must be able to visualize music as well as other sounds. Therefore, the audio feature that should be used for visualizing audio input must be able to describe all types of audio input.

Widespread availability of powerful and user-friendly personal computers and portable computing devices led to the development of music visualizers, which generate animated imagery based on music. The 1999 Windows Media Player application Visualizations created various designs as visual representations of any given music played through it. Such applications are now more developed in other digital media players like Winamp and iTunes. This produces a fluid, textured, rhythmic and animated video stream that is generally nonrepresentational .

Some music visualization research has resulted in methods that are used to represent or describe a piece of music [38] [39]. Others have developed ways in which music can be generated to represent given visual features [23] [28] [40].

Hiraga and Matsuda [39] categorized music visualization methods into two types, augmented score and performance visualization. Augmented score visualization method was intended to assist performers in learning a piece of music [41] [42], while performance visualization method was developed to assist musical performances [43] [44]. Hiraga and Watanabe [45] generated a system to illustrate any change in performance using a series of Chernoff faces that may be used in music training or practice.

In some approaches, researchers concentrated more on music analysis than on visualization. For example, Hiraga and Matsuda visualized tempo change, dynamic changes and the articulation of music pieces with vertical lines, horizontal intervals and the height and width of bars [39]. For example, Politis et al. [28] argued that the song *"How you gonna see me now*" (*Alice Cooper*) is most similar to *"The trooper" (Iron Maiden)* and both belong to the category of Metal songs. They visualized these two songs by using chromatic graphs. It is also important to note that the goals for music visualization differ from our goal. For example, music visualization has been used to support music learning and analyze performance[45]. Mood was used instead of content and tags for a musical data mining interface [32]. Our proposed system has been tested using different type of music sound and provided visualization that clearly represent the difference between these types in simple memorable interface.

2.4 Speech visualization

Hailpern et al. [46] hypothesized that speech visualization techniques can be mainly used to support communication and to help autistic children develop speech skills by visualizing vocalization.

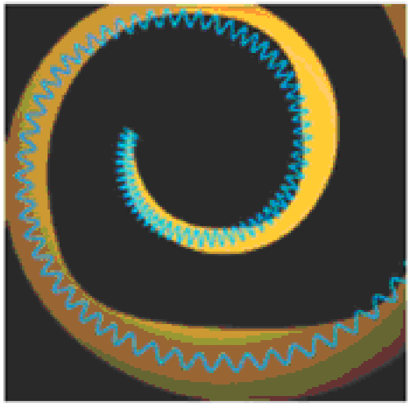

Figure 2.5 : Graphic matching in speech learning [47].

Karahalios and Bergstrom [47] visualized speech using simple graphical elements to identify the current speaker as shown in figure (2.5). Different colors were used to represent different speakers around a table and the thickness of the line, which forms a section of the circle, corresponds to the average amplitude of voice. Similar approaches can be found in [48] where Bergstrom and Karahalios visualized speakers in conversations around a table using a clock-like image as shown in figure (2.6).

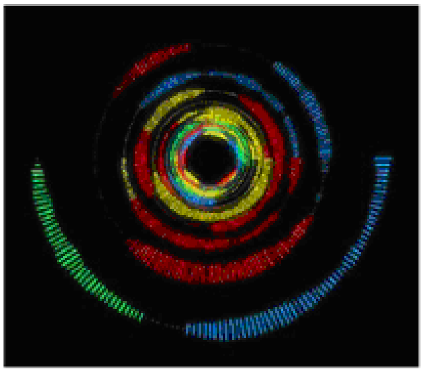

Figure 2.6: Clock visualization for conversation**:** [48].

Simunek [49] developed a system that animate human face based on phonemes that produce a kind of visualization for speech by animating lip movement. Bregler et al. created a video of a person mouthing words that he did not speak on image of still face [50]. Bregler's methods could be adapted to generating new video sequences that accurately represent the time sequence of sounds in a general audio file.

The methods of speech visualization may differ from our proposed system, except the approaches used in speech visualization and the uses of video represent commonalities.

2.5 General sound visualization

Audio visualization results are most commonly static 2D or 3D images. Smith and Williams [51], Chaudhary and Freed[52], Kaper et al. [36] and Hiraga et al. [32] have also constructed visualizations of audio files in three-dimensional space. Smith and Williams present a method for visualizing the audio properties of MIDI music by using color in 3D space [51]. The mapping function is defined by the musical characteristics and the piece of music is transformed into three-dimensional graphical views. While in [52] the time, amplitude and features of sounds are visualized in 3D as in Figure (2.7). Tones are represented by colored spheres and the pitch, and volume and timbre are the audio properties that define the spheres. Kunze and Taube also generated a 3D graphical tool that could be used by composers for writing music [53]. Hiraga et al. [32] produced 3D images in three properties; pitch, volume and tempo were represented by the height, diameter and color saturation of stacked cylinders.

Either 2D or 3D images can be employed to visualize sounds as long as there are enough visual features available to represent the selected audio features. The advantages of 3D over 2D become apparent when more than one sound is visualized at the same time.

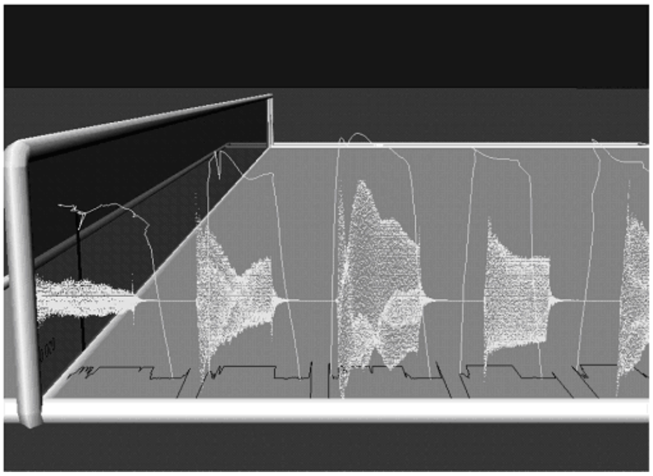

Figure 2.7: Sound visualizations in method [52].

2.6 Sound visualization for hearing impaired or deaf

Hearing impaired and deaf people requires special properties for visualizing sounds. They need simple and available techniques to get their attention and to connect their awareness of sound with the real properties of sounds. Many devices and equipment are available to help deaf and hearing impaired people, adapt to their environment, and function in society more efficiently. For example, smoke alarms, phone ringing, and alarm clocks can all be converted to vibrating mechanisms or flashing lights for notification.

Audio visualization for hearing impaired and deaf has been proposed in [54][55][56]. In [54] the authors analyzed the techniques used by deaf people for sound awareness; they made interviews with deaf and hearing impaired participants for designing an efficient method of sound visualization, and based on these interviews, they based their visualization system on drawing sounds as waves (circles). Based on these results, two sound displays have been presented. One is based on a spectrograph and the other is based on positional ripples. In the spectrograph scheme, height is mapped to pitch and color is mapped to intensity (red, yellow, green, blue, etc.). In the positional ripple prototype, the background displays an overhead map of the room. Sounds are depicted as rings, and the center of the rings denotes the position of the sound source in the room. As shown in Figure (2.8) the size of the rings represents the amplitude of the loudest pitch at a particular point in time. Each ring persists for three seconds before disappearing.

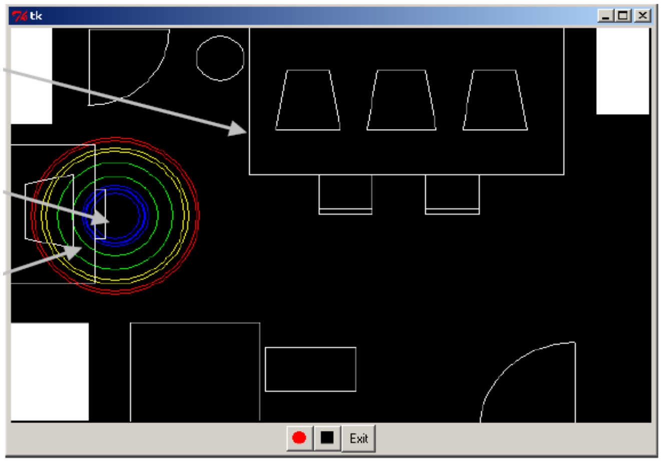

Figure 2.8: Speech Visualized by Positional Ripples[54].

This architecture however is impractical since it requires prior knowledge of the surrounding place (e.g. office); also it is expensive in terms of equipment setup (array of microphones placed at certain corners in the room) and is also not portable (bound to the workplace environment).

In [55], new models have been proposed, based on the proposed system in [54]. The authors proposed two models. The first model, based on single icon scheme, which displays recognized sounds as icons, located on the upper right corner of the user's computer monitor as shown in Figure (2.9). It was used throughout the analysis and was shown to give good results.
According to the survey performed in [55], all participants liked it because it identified each sound event.

The disadvantage of this method however is the actual need for prior knowledge of the type of sound to be detected which is very hard for a person who cannot hear well. The second model, the spectrograph with icon visualization, improves over the single icon model in that it combines the black and white spectrograph model in [54] with the single icon model. This mapping tends to associate a particular sound with its shape on the spectrograph [55].

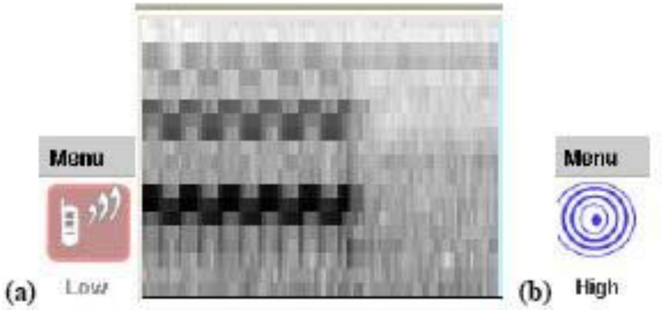

Figure 2.9 : Single icon model [55].

But this method also shares the disadvantages of the first method where the need for prior knowledge of the type of sound to be detected.

2.7 Conclusion

A more general and simpler method is needed because the existing research related to audio visualization provides no approach that can be used to visualize the content of audio in simple manner suitable for deaf person as well as depends on highly representative audio features.

Our proposed method presents a general method for audio visualization, which is more accessible than the existing methods and depends on robust sound features because the resulted video from the application is directly representative of sounds in the real world as well as highly discriminating between different types of sound.

CHAPTER 3

THEORETICAL BACKGROUND

In this chapter, we overview some background theory explaining some basic steps for processing sound signals to extract sound features. We explain the most well-known sound features in literature in detail explaining our point of view about their validity to be implemented in our system. After that, we present most generally used local similarity measures definition in order to evaluate them. In addition, we present the theory of most well-known classifiers suitable for building our system.

3.1 Sound signal analysis

Sound can be processed digitally by extracting its features. Luckily, the sound is a natural signal, thus the signal processing theories can be applied to sound for extracting its features. This section introduces the basic principles of feature extraction. Several most commonly used in literature analysis methods are considered, inspired from the speech recognition community. LPC and Cepstral models are presented, as well as techniques representing behavioral models of the human ear.

The feature extraction [57] is the most important part of recognition, classification, and visualizing systems. If the features have not good discrimination properties between sound classes, no classifier or visualizing system architecture will be efficient, as advanced as it could be. In practice, features always present some degree of overlap from one class to the other. Consequently, it is important to choose the features that are most robust for representing sound classes and immune for noise.

Ideally good features should have the following properties:

- They have to emphasize the difference between classes of the sound.
- They have to be immune to noise effect , preserving the class separability as can as possible.
- Their intra-class variance should be minimal, and their inter-class means well separated.

- The feature dimension number has to be limited but sufficient. Too large dimension leads to more complex systems architecture to handle these features. This fact is the well-known “Curse of dimensionality” [58].
- A high correlation between features should be avoided, as much as possible. Despite that there are situations where slight amount of correlation can be benefit, where one extracted feature supports the other.

We will present general well-known feature extraction models used commonly in speech recognition community.

3.2 Sound Features

3.2.1 LPC Features

In the speech processing community, the analysis of speech signal often involves the LPC (*Linear Prediction Coefficients)* model. This section will review the LPC model in abstract manner as there is large published literature, as [15] that explains in detail this model. This analysis model consists in a linear all-zero analysis filter to separate between two main components of the speech signal:

- The *excitation*, which represents the air pressure waveform resulting from vibration of the vocal cords that are excited by air expulsed from lungs, this excitation signal is the result of the LPC analysis filter where pulse period or *pitch* can be derived from the excitation signal (*pitch detection* [15]).
- The *spectral envelope* of the speech, which is produced by the shape of the vocal tract. The vocal tract is represented by the synthesis filter, whose input is the excitation. The spectral envelope information is held in the coefficients of the adaptive filter, the larger number of coefficients, the more precise the envelope.

The LPC filter coefficients are derived for each overlapping (1/3) hamming windowed frame of length (20 to 30 ms) of speech signal, using a short-term prediction of the speech samples.

In speech or speaker recognition problems, the number of filter taps is chosen between 8 and 12. The LPC coefficients features are well adapted to speech recognition and provide better results than the time frequency analysis [59]. On the other hand, the LPC does not perform well for general audio analysis, which is the study domain of our research.

3.2.2 Cepstral Coefficients

The cepstral transformation is an improved and more robust way to isolate the two components of speech [60]. The cepstrum $C(n)$ in equation (3.1) is defined by inverse Fourier transform of the amplitude logarithm of an input signal $x(k)$

$$C(n) = F^{-1}\{\log|x(k)|\} \tag{3.1}$$

Where F represents the Fourier transform operator. This Cepstral domain is useful in speech processing because, the excitation and vocal tract components are linearly combined. Since the speech is produced by convolving the excitation with the impulse response of the vocal tract synthesis filter (inverse LPC filter), the transforming into the frequency domain makes this convolution process turns to simple addition!. Then, the linear Fourier transform or its inverse can be taken once again, to analyze the “frequency” content of the log spectrum. This shows the advantages of the cepstral information, because for speech, the envelope shows slowly spectral properties. On the other hand, the excitation of voiced speech is made of important spectral variation (pitch and harmonics), so more easy discrimination between excitation and envelope components is then possible.

For computation reasons, only the first cepstral coefficients (15 or 20) are usually kept. The expression (3.2) illustrates how to calculate the first Nc cepstral coefficients $cc(i)$ can be derived at each time frame.

$$cc(i) = \sum_{n=0}^{N-1} X_n \cos\frac{\pi i n}{N} \text{ for } i = 1, \dots, N_c \tag{3.2}$$

Where N represents the frame length, and X_n represents the log-amplitudes corresponding to frequency n.

The resulting set of cepstral coefficients are usually smoothed using sinusoidal window. Equation (3.3) deemphasizes both the first coefficients that are sensitive to the spectral tilt , and last coefficients , more sensitive to noise.

$$cc'(n) = cc(n)\, w(n) \text{ and } w(n) = 1 + \frac{N_c}{2} \sin\left(\frac{\pi n}{N_c}\right) \quad (3.3)$$

Speech recognition systems based on cepstral features are said to be slightly better and more robust than LPC features[61]. *Delta Cepstral Coefficients* is used in calculating more robust cepstral features. Those coefficients can be computed as the difference between cepstral coefficients of the current frame and past frames, including temporal evolution information in the features [60].

For general sounds, cepstral features may be most useful as they are able to de-correlate slowly and fast varying components of the spectrum.

3.2.3 Features From Human Ear Model

3.2.3.1 Human ear frequency scale

Sound waves are transformed into mechanical vibrations in the outer ear, at the eardrum. The little bones of the middle ear convert those vibrations into liquid pressure variations in the inner ear that finally create the traveling waves of the *basilar membrane* of the cochlea [62]. The membrane moves according to the energy of the incoming sound [63]. Low frequencies result in movement at the beginning of hypothetically unrolled basilar membrane, and higher frequency response appear farther in the cochlea. Thus the human perceptual frequency scale will vary in response to low and higher frequencies making the perceptual frequency scale logarithmic in some regions and linear in others.

3.2.3.2 Mel Frequency Cepstral Coefficients (MFCC)

Stevens[17] measures the human perceptual frequency scale producing of what is called the *mel-frequency* scale or similarly *Bark* scale. The mapping between the linear frequency and the human scale seems to be linear up to 1 kHz, and gets logarithmic above this value.

It can be expressed as a function of one variable f which is the linear frequency in kHz[18] as in equation (3.4) which shows how to calculate the bark scale resulting the curve shown in Figure (3.1)

$$f_{bark} = 13\arctan(0.76f) + 3.5(\frac{f^2}{56.25}) \tag{3.4}$$

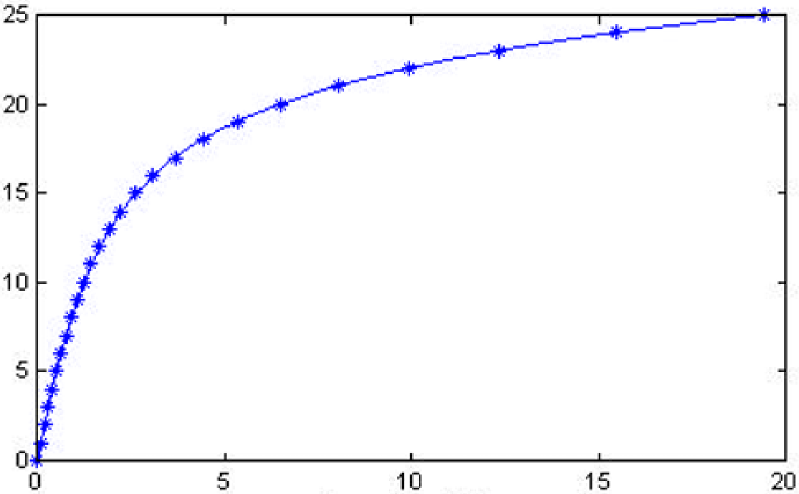

Figure 3.1: Linear frequency to Bark scale mapping

Generally the use of human ear models seems to bring an improvement on the performance of speech recognition systems especially in the presence of important background noise, where the Mel-Frequency Cespstral Coefficients (MFCC) is proved to be providing important performance improvements, compared to the linear cepstral coefficients[60]. The idea of the MFCC is to distribute the cepstral coefficients according to the critical bands , instead of the linear distribution. This is done by applying “critical band filters” to the current frame spectrum [61]. The complete spectrum is rebuilt by placing zeros at indexes that do not correspond to the critical band central frequencies.

3.3 Similarity measures

Any visualizing system cannot do its job well without depending on similarity differentiating between sounds, either by using similarity measures or probability classifiers. In this section we will present the most common similarity measures that are found in the open literature. *xi*, *yi* are the feature values of the reference feature vector and the test feature vector.

3.3.1 Minkowski Distance (L2)

Minkowski distance [32] is one of the most popular similarity measures used in literature , and its defined as **:**

$$D_0 = \ (\sum_i (x_i - y_i)^k)^{1/k} \tag{3.5}$$

Where k is representing multiple forms of Minkowski distances

3.3.2 Euclidean Distance (L2)

One of the commonest distance measures in literature is the *Euclidean* distance [31]. It corresponds to the Minkowski-form for k =2, and is defined as:

$$D_1 = \ \sqrt{\sum_i (x_i - y_i)^2} \tag{3.6}$$

3.3.3 Manhattan Distance (or City Block Distance)

The *Manhattan* distance [64] corresponds to Minkowski-form for k =1. It requires less computation than the Euclidean distance and many other distances, and is defined as:

$$D_2 = |x_i - y_i| \tag{3.7}$$

3.3.4 Canberra Distance

Canberra distance [65] is very popular in CBIR applications. It has the advantage of a relatively low computational complexity and high retrieval efficiency.

$$D_3 = \sum_i \frac{|x_i - y_i|}{|x_i + y_i|} \tag{3.8}$$

3.3.5 Jeffrey Divergence Distance

The *Jeffrey divergence* distance[66] is defined as:

$$D_4 = \sum_i \left[\ x_i \ log\left(\frac{x_i}{m_i}\right) \ + y_i \ log\left(\frac{y_i}{m_i}\right)\ \right] \tag{3.9}$$

Where $m_i = \ \frac{x_i + \ y_i}{2}$

3.3.6 Bray Curtis Distance

Bray Curtis distance [67] is quite similar to Canberra metric. It is defined as:

$$D_5 = \frac{\sum_i |x_i - y_i|}{\sum_i |x_i + y_i|} \tag{3.10}$$

3.3.7 Angular Separation Distance (cosine similarity)

Angular Separation distance [68] is defined as:

$$D_6 = 1 - \frac{\sum_i x_i . y_i}{\sqrt{\sum_i x_i^2 \; \sum_i y_i^2}} \tag{3.11}$$

3.3.8 Chord Distance

Chord distance [69] measures the distance between the points where vectors cross a unit sphere. It is defined as:

$$D_7 = \sqrt{2 - 2 \frac{\sum_i x_i . y_i}{\sqrt{\sum_i x_i^2 \; \sum_i y_i^2}}} \tag{3.12}$$

3.3.9 Non-Correlation

The *non-correlation* metric[70] is defined as:

$$D_8 = 1 - \frac{\sum_i |x_i - \overline{x_i}| . |y_i - \overline{y_i}|}{\sqrt{\sum_i (x_i - \overline{x_i})^2 \; \sum_i (y_i - \overline{y_i})^2}} \tag{3.13}$$

3.3.10 Matusita Distance

The *Matusita* [69] distance is defined as:

$$D_9 = \sqrt{\sum_i (\sqrt{x_i} - \sqrt{y_i})^2} \tag{3.14}$$

3.2.11 Wave – Hedges

The *Wave-Hedges* metric [69] is defined as:

$$D_{10} = \sum_i 1 - \frac{\min(x_i, y_i)}{\max(x_i, y_i)} \tag{3.15}$$

3.2.12 Weighted Euclidean Distance

The *Weighted Euclidean Distance* [71] has the same form as Euclidean but the square difference of the two distributions for every *i* is multiplied by a weight *wi* depending on the value of distribution x.

$$D_{11} = \sqrt{\sum_i w_i \, (x_i - y_i)^2} \tag{3.16}$$

Where $w_i = x_i$ if $x_i \neq 0$, and $w_i = 1$ otherwise.

There are many other weighted distances similar to Weighted Euclidean Distance but the distributions for every i is multiplied by a weight w_i depending on the value of distribution x , like weighted Manhattan distance.

$$D_{12} = |m_x - m_y| \tag{3.17}$$

Where $m_x = \sum_i x_i \, p(x_i)$ and $m_y = \sum_i y_i \, p(y_i)$.

3.4 Classification

3.4.1 Which classifier

There are large several classification techniques following different approaches. Statistical methods such as Bayes classification and Gaussian Mixture Models try to estimate the probability density function of the underlying data [72]. Another group of classifiers is learning algorithms that employ artificial intelligence techniques. There are supervised learning methods such as Support Vector Machines, neural networks, and non-supervised techniques such as Self-Organizing Maps [73]. Beside parametric techniques (e.g. Support Vector Machines), there are non-parametric techniques such as the k-Nearest Neighbor (k-NN). Another special classifier used for time series signal to measure sequence called Dynamic Time Warping classifier (DTW).

Two classifiers were selected for classification to be implemented in our system which are the K-NN and DTW. Because the simplest way to classify feature vectors is the nearest neighbor rule (k-NN) which suits for the real time application. In addition to that, since sounds are sequence time frames and have perceptual properties they required to employ the well-known DTW.

3.4.2 K-Nearest Neighbor

K-Nearest Neighbor (K-NN) is a popular non-parametric classier [74]. Like other non-parametric techniques K-NN operates on the data directly. Therefore, it cannot measure sound sequence similarities.

The 1-NN (NN) algorithm assigns a new vector x to the class label s of the nearest training vector x_i, as shown in equation (3.18)

$$s = \arg\min_i \|\boldsymbol{X} - \boldsymbol{X_i}\| \qquad , 1 < i < N \tag{3.18}$$

Similarity in nearest neighbor classification can be measured by any similarity (distance) measure. Frequently, Euclidean distance is used for K-NN.

The K-NN algorithm with K > 1 considers more than just the nearest neighbor for classification. K denotes the number of nearest neighbors of a new feature vector x that are considered for classification. From these K vectors, k_j vectors belong to class ωj, with $\sum_j^c k_j = K$ where c is the number of classes. Vector x is assigned to class s with the greatest number of representatives in the set of K neighbors as it can be shown in equation (3.19).

$$s = \arg\max_j \; k_j, 1 < i < c \tag{3.19}$$

Hence, memory and computation costs grow linearly with the size of the training set $(O(N))$. This computational complexity fits perfectly for our real time application.

3.4.3 Dynamic Time Warping

DTW is a dynamic programming technique that measures the similarity and finds the minimum-distance warping path between two time series [75]. Given two time series A and B, of length m and n, respectively,

$$A = [a_1, a_2, \dots, a_m]$$
$$B = [b_1, b_2, \dots, b_n]$$

the distance of a_i and b_j is denoted as in equation (3.20) as

$dist(i,j) = \left|a_i - b_j\right|$ if $1 \le i \le m, 1 \le j \le n.$ (3.20)

A 2D cost matrix D of size m by n is constructed, where $D(i,j)$ represents the minimum distance between two partial series

$$\acute{A} = [a_1, a_2, \dots, a_i]$$
$$\acute{B} = [b_1, b_2, \dots, b_j]$$

D is initialized as $D(0,0) = 0$, infinity otherwise , and then D is filled from $D(1,1)$ to $D(m,n)$ using equation (3.21) with

$D(i,j) = dist(i,j) + \min[D(i-1,j-1), D(i-1,j), D(i,j-1)]$
$if\ 1 \le i \le m, 1 \le j \le n$ (3.21)

The algorithm increments i and j until the cost matrix is filled such that $D(m,n)$ is the minimum distance between series A and B. Since a single member in one series can map to multiple successive members in the other series, the two series can -be of different lengths.

3.4 Conclusion

In this chapter, we overviewed some background theory explaining some basic steps for processing sound signals to extract sound features. We explain the most well-known sound features in literature in detail explaining our point of view about their validity to be implemented in our system. After that, we present most generally used local similarity measures definition in order to evaluate them. In addition, we present the theory of most well-known classifiers suitable for building our system.

CHAPTER 4

PROPOSED SYSTEM

In this chapter we are going to describe our proposed system in detail and describe every single step in building our visualization system , starting from the procedure of gathering design properties by interviewing group of the deaf. The behavior of the proposed system is explained starting from the sound input, passing with feature extraction methods by explaining the implemented equation for each feature extraction method according to the theoretical presentation in Chapter (3). Also we are going to describe the proposed similarity measure for comparing MFCC sound features effectively explaining the drawbacks of other similarity measures. Finally, the visualizing system framework will be expressed in detail to be later evaluated.

4.1 Gathering design requirements

4.1.1 Interviews

The properties of good sound visualization system must answer the following questions;

- What sounds are important to people who are deaf?
- What display size is preferred (e.g. mobile, PC monitor, or large wall screen)?
- What information about sounds is important (e.g. sound classes, location, or characteristics like volume and pitch)?
- How the person who is deaf can be aware with the visualizing system?

These questions should be answered by a sample of deaf people. In the next section, we present the extracted results from interviews made with people who are deaf according to previous questions. These questions will be discussed in detail as subsections in this chapter to describing our visualizing system requirements. We meant by these surveys to produce a small set of scenarios, not to enclose the list of all sounds of interest.

The initial data for the deaf participants was gathered by interviewing five of the deaf persons. The participants were chosen in different ages and jobs see Table (5.1).

Table 4.1: Interview participants

Deaf participant	Work place	Job	Age
1	Dar Elarkam Library	Printing & photocopying documents	24
2	Atfaluna Society for Deaf Childern [76]	Student (grade 9)	14
3	Future Society for Deaf Adults	Worker at craftworks Club	31
4	Atfaluna Society Restaurant	Waiter	28
5	House	Housewife	33

We can notice the differences of deaf ages and jobs in Table (4.1) . These differences in the sample participants enhance our understanding of the participants needs. With this understanding, we were able to answer the first question which is "*What sounds are important to people who are deaf?"* that would help us to learn about the sounds which the participants wanted to be aware .

4.1.2 What sounds are important to people who are deaf?

Our interviews helped us to learn about sounds that participants wanted to be aware of sound as follows;

- **The activity and presence of others**:

Participants who we interviewed, mentioned that they hope to have awareness of any soft sounds like music and colleagues speech when they are alone.

- **Highly dynamic environment sounds***:*

  In environments where needs change frequently especially in work, it is hard for a deaf person who works in a mixed working environment with hearing co-workers to maintain awareness of sounds alone. For example , in a library where visitors need to be served as soon as they arrive. One participant mentioned how waiting for a visitor can be very difficult. He would have to visually check every few minutes because he could not hear a door knock.

- **Sound from home environment**:

  In the home environment, the sounds of the appliances such as microwave ovens, and kettles can be considered as a set of sounds which the deaf would like to be aware. This is particularly important from the own view of the deaf because those sounds are designed to notify users of important state changes.

Our participants confirmed that sound awareness was important in work, home, and mobile settings involving social interactions (*e.g.* presence of co-workers or children playing in another room), safety (*e.g.* sudden car horns , and fire alarms), and many other situations.

4.1.3 What display size is preferred (mobile phone , PC monitor, or large wall screen)?

Based on the participants views about the display size (e.g. mobile, PC monitor, or large wall screen), participants preferred smaller displays in all locations such using a mobile phone or using part of a PC screen. In other words, most of the participants pointed their desire toward a small display which they can use it whenever and wherever they wanted . So, they considered using mobile phone is more practical tool than the other techniques like PC monitor or large wall screen. However, some of them suggested using glasses technique in further researches . Also, two participants referred to a new display that showed a single icon (for recognized sounds) and rings (for unrecognized sounds) .

During our interviews with the participants, they focused our attention towards several issues. One of the participants wanted a way to look at a history of identified sounds. One participant commented "I wouldn’t want to be looking at the monitor all the time, but it would work if it has a history component". Another wanted to minimize background noises and commented "I don’t really care about hearing the other environmental noises".

4.1.4 What information about sounds is important (e.g. sound classes, location, or characteristics like volume and pitch)?

We seek to answer questions about many design issues ranging from place of use to type of information displayed. For example, is sound location more useful than sound volume and pitch?.

We explored how information about sound characteristics such as volume, pitch, and location affected deaf distraction and ability to identify sounds**.** Participants felt that the displays should allow them to ‘look & know’ or ‘figure out the sound’. Participants liked the location information because it gave them even more information with which to identify sounds.

Participants tended to prefer displays that showed location or identity of sound over volume and pitch alone, although participants thought all features would be useful in identifying unknown sounds. Functionally, we found that participants wanted mechanisms to:

- Identify what sound occurred, with or without computer recognition,
- view a history of displayed sounds,
- determine the accuracy of displayed information.

4.1.5 How the person who is deaf can be aware with the visualizing system?

Four of the participants preferred visual designs that were easy to use and have less distraction, over designs with more detailed information or single type of notification.

One participant wanted the displays to show every sound that was made, including co-workers coughing or sneezing. Each individual will have varying preferences about the types of sounds of which they wish to be aware. Ideally, the tool would be flexible enough to allow deaf to be aware of any type of sound.

We have discovered two main guidelines for an ambient sound visualization display:

- **The display should identify or help the user identify sounds:**
  Because sound identification was the major goal, sound recognition received a positive feedback. However, due to limitations of sound recognition technology, recognition for all sounds is improbable. Deaf were willing to interpret sounds themselves, as long as the system can provide information that would help them do so. Our evaluations showed that each type of sound information is limited by itself, but a combination of sound recognition, location, volume, and frequency might improve sound identification in more situations.
- **The display should allow users to choose which sounds to show and filter out the rest:**
  Ambient sounds are present all around us all the time. However, different sounds are important to a person depending on his/her context. deaf need the ability to choose which sounds should be displayed. The display might have performed better if background noises had been filtered out.

The interview results provided us with an understanding of participants visual design preferences and functional requirements. Visually, participants preferred designs that were easy to interpret. Displays using mobile phones were preferred because participants could easily understand what sound occurred in a glance. Participants criticized displays they thought would be overly distracting, like Rings. More complex displays like Ambient Visualization were criticized for being difficult to understand.

4.2 Sound input

The sound is sampled from the portable device microphone at 44100 samples per second, 16 bit per sample and mono. The sampling rate can be lowered according to the device computation power , for example 24000 sample per second to guarantee not missing some environmental sounds that has frequencies higher than 10 KHz. Further sound processing requires framing the sound to be processed in real time and due to memory limitations of the computing devices. The frames must be overlapped and windowed so the transaction between frames is smoothened to prevent the distortions. The choice of the frame length N of sample must be done carefully, considering temporal properties of the audio data, and to speed up the detection at further stages of the process. Therefore, choosing a precise frame size cannot fit every acquiring audio system depending on sampling rate and computation power of the system. The time duration of each frame is about 20~30 ms that can support the assumption of the stationary of the audio signal within the frame. If the frame duration time is too long, we cannot get the time-varying characteristics of the audio signals , and consumes much memory resources which are valuable for further processing steps, especially when converting to other domains on behalf of signal processing techniques . On the other hand, if the frame duration time is too short, then we cannot extract valid acoustic features and causes too much detail that would needlessly appear in the signal power bins. Thus, some tradeoff is needed, and a good solution was found using hamming window of size N=1024 samples with overlap of 50% at sampling rate of 44100 sample/second, which approximately corresponds to 23.2ms of sound input.

4.3 Feature extraction

The extraction of the best parametric representation of acoustic signals is an important phase in our system since it affects the visualizing and recognition behaviors in the next phases. The most well-known state of art-feature extraction methods are MFCC and LPC; by considering their popularity in sound recognition systems [77]. The widespread use of the MFCCs is due to its low computational complexity and better performance for most ASR systems [10-14]. MFCCs is used for speech data in most cases but it can be generalized for environment sound as in [10]. The characteristics of MFCCs that made it preferable for our system is that it has lower computational complexity than many other algorithms $(n\log(n))$ [78], its discrimination rate (as we will see in Chapter(5)) , and for its simplicity in implementation .

Seven computational steps for generating MFCC vectors are summarized in figure (4.1) and expressed as following;

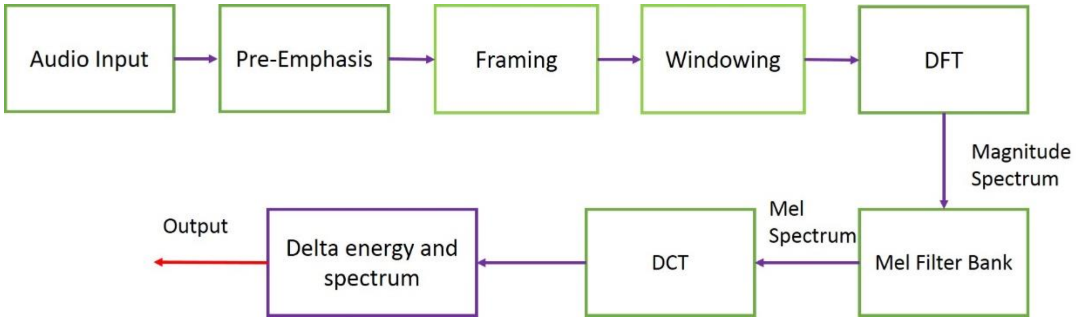

Figure 4.1: MFCC block diagram

1. Pre-emphasis

The emphasis filter we used is shown in equation (4.1) derived from the equation in [79], which makes 95% of any one sample is presumed to originate from the previous sample

$$Y[n] = X[n] - 0.95X[n-1] \tag{4.1}$$

Where $Y[n]$ is the emphasized frame , $X[n]$ is the input frame , and n is the sample number

2. Framing

As mentioned in the previous section the frame length is chosen to be 1024 sample at sampling rate of 44100Hz , which approximately corresponds to 23ms of sound duration time .

3. Windowing

For constructing the hamming window form the form in [80] we used the parameter shown in equation (4.2)

$$W(n) = 0.54 - 0.46\, cos\left[\frac{2\pi n}{N-1}\right] , 0 \leq n \leq N-1 \qquad (4.2)$$

The result of windowing signal is shown in equation (4.3)

$$Y(n) = X(n) \times W(n) \qquad (4.3)$$

Where

$Y(n)$ is the resulted windowed signal

$X(n)$ is the input signal

$W(n)$ is the Hamming window

4. Fast Fourier Transform

Each frame can be considered a stationary signal. The windowed frame can be transformed into the frequency domain by using Fast Fourier Transform [81] according to equation (4.4).

$$S[k] = \sum_{n=0}^{2N-1} s[n] e^{-j\frac{2\pi kn}{2N}} \qquad (4.4)$$

Where

$s(n)$ is the windowed frame , N is the frame length , and k is the corresponding frequency bin and the power spectrum is produced by applying absolute value for $S[k]$.

5. Mel Filter Bank Processing

The power spectrum is passed through triangular shaped Mel filters with M filters (m = 1, 2, …, M), M usually ranges from 24 to 40 and we used M=24 according to the equation (4.5) [82].

$$H_m[k] = \begin{Bmatrix} 0\,, if\ k < f[m-1] \\ \dfrac{2(k - f[m-1])}{(f[m+1] - f[m-1])(f[m] - f[m-1])}, if\ f[m-1]\ \le k\ \le f[m] \\ \dfrac{2(f[m+1] - k)}{(f[m+1] - f[m-1])(f[m] - f[m])}, if\ f[m]\ \le k\ \le f[m+1] \\ 0\,, if\ k > f[m+1] \end{Bmatrix} \qquad (4.5)$$

In addition, we used equation (4.6) to compute the Mel scale [18] for given frequency in Hz:

$$F(Mel) = [2595 * log_{10}[1 + f] * 700\ ] \qquad (4.6)$$

6. Discrete Cosine Transform

Discrete Cosine Transform (DCT) [83][84] is applied to the logarithm of the filter-bank coefficients, to find the MFCCs vector parameters [82] as in equation (4.7)

$$ci = \sqrt{\frac{2}{N}}\, \textstyle\sum_{j=1}^{N} \log\left(e_j\right) \cos\left(\frac{\pi i}{N}\ (j + 0.5)\right) \qquad i = 1, \dots, D \qquad (4.7)$$

Where e_j represents the energy output of the $j - th$ triangular filter, N number of filters, and D is the length of the feature vector . Because the first DCT coefficients correspond to low-frequency components of the transformed information. The low-frequency components usually contain the important transformed information. The later coefficients contain the less-important data information [82], so we used D to be 12.

7. **Delta Energy and Delta Spectrum**

We used 13 delta or velocity features, and another 13 double delta or acceleration features are added to 13 (12 cepstrum and 1 energy) features to produce 39 MFCCs . The overall complexity of the system is $(n \log(n))$.

4.4 Distance measure for MFCCs feature vector

4.4.1 Similarity measures and MFCC

In Chapter (3), we investigated popular similarity measures, which are often used in clustering, recognition, and information retrieval algorithms. In this following three sections we will explain the main reasons that made the mentioned distance measures in Chapter (3) fail in measuring the similarity between MFCCs and propose new distance measure suitable for not only MFCCs , but also for many vectors often used to describe sound , image ,and natural language features . These features vectors are in fact Discrete Cosine Transform (DCT) coefficients, which have some properties that made our proposed similarity measures superior on conventional distance measures. Thus, we can generalize the proposed similarity measures for measuring the similarity between DCT vectors.

4.4.2 Discrete Cosine Transform

The Discrete Cosine Transform (DCT) is a Fourier-like transform. While the Fourier Transform represents a signal as the sums of sines and cosines, the DCT expresses a signal (a set of numbers) in terms of a sum of cosine functions with different frequencies. The advantage of DCT is that the energy of the original data may be concentrated in only a few low frequency components of DCT depending on the correlation in the data, so it is used in large scale for feature vectors to reduce the dimensionality of these vectors.

We will express the general equation for DCT to extract the properties of DCT coefficients, hence the MFCCs coefficients.

Given a set A of N values; M= { m_0 , m_1 ,..., m_{N-1}} , one dimensional discrete cosine transform coefficients [83] are given by equation (4.8),

$$c(i) = \alpha(i)\sqrt{\frac{2}{N}}\sum_{j=0}^{N-1} m_j \cos\left(\frac{\pi}{N}\ (j+0.5)i\right) \qquad i = 0, \dots, N-1 \qquad (4.8)$$

Where $\alpha(i) = \begin{cases} \frac{1}{\sqrt{2}} & , i = 0 \\ 1 \, , & i > 0 \end{cases}$

As can be seen from equation (4.8), the substitution of $= 0$, $c(0) = \alpha(0) \sum_{j=0}^{N-1} m_j$, which is the sample mean of the set A .In literature , it is called the DC coefficient of the transform and the other coefficients are called the AC coefficients .

If we ignored m_j and $\alpha(i)$ in (4.8) , then ;for every value $i = 0,1, \dots, N-1$ transform coefficients correspond to a certain waveform. The first waveform renders a constant value, whereas all other waveforms ($i = 1, 2, \dots, N-1$) produce a cosine function at increasing frequencies. The output of the transform for each i is the convolution of the input signal with the corresponding waveform. This is explained if we plotted $\sum_{j=0}^{N-1} \cos\left(\frac{\pi}{N}\ (j+0.5)i\right)$ for $N = 1000$ and varying values of i , then the corresponding waveforms are shown in figure (4.2). These waveforms are called *cosine basis functions.* These waveforms are orthogonal and independent, that is, none of the basis functions can be represented as a combination of other basis functions [84].

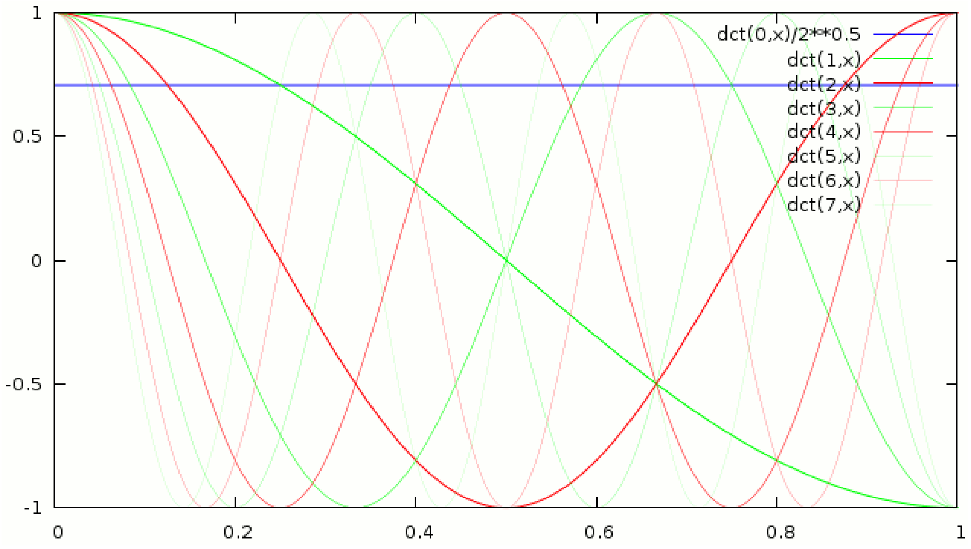

Figure 4.2 : 1D DCT basis function

The mentioned properties of DCT coefficients will be reflected on MFCC as its coefficients are DCT coefficients. We will discuss the impact of changing the values of DCT coefficients either by the value coefficient itself or by changing its place in the MFCC feature vector to extract the main properties of suitable distance measure.

4.4.3 Properties of MFCCs

The steps for generating MFCCs were shown in section (4.2). The final step for generating basic MFCCs (step 6) shows that MFCCs are DCT coefficients, so we will use the terms DCT coefficients and MFCCs alternatively in this section .There are two important properties for DCT/MFCC coefficients making them hard to be measured in conventional similarity measures, which are:

1. Shuffling variant

As discussed in section (DCT) we determined that none of DCT coefficients can be made as combination of other coefficients , so any reordering or permuting of MFFC coefficients makes them describing another set of data and hence, describing another sound. This property clarify the inadequacy of distances D1-D11 (Defined in Chapter (3)) on MFCCs by explaining the *“shuffling invariance*”[85] property . A distance measure between two MFCCs is “shuffling invariant” if and only if the distance does not change when levels { m_0 , m_1 ,…, m_{D-1}} in the MFFCs are permuted or reordered. Distances D_1 – D_{11} have the “*shuffling invariance”* property, because the definition of distances D_1 – D_{11} shows that they are sum of individual distances of each attribute of the feature vector, and due to the commutative law, the distances do not change when the attributes are reordered.

The reordering of some DCT coefficients has big impact on the shape of the transformed data. The shape of waveform corresponds with the largest absolute coefficient value will dominate the shape of the transformed data and the next waveform corresponds to the next largest absolute value of the coefficients will dominate, but with less influence and so on.

We can deduce that any distance measure should take into account the places of the absolute values of DCT coefficients in order. In other words, the suitable similarity measure should compare the places of most dominant coefficients to compute the similarity between MFCC vectors.

2. Different impacts of attributes

The independence property of DCT coefficients made it hard to compare coefficients with different order by value because each coefficient has different impacts on the transformed data. In other words, the right comparison by values should be done between one coefficient and another if they take the same order in the DCT vector. As shown in the previous section, the first DCT coefficients correspond to low-frequency components of the transformed information. The low-frequency components usually contain the important transformed information. The later coefficients contain the less-important data information [83]. This forces us to design a similarity measure to take into account the importance of each coefficient by assigning a weight for every coefficient.

4.4.4 The proposed similarity measure

In this subsection, we introduce efficient algorithm for measuring the similarity between two vectors of MFCC taking into account “shuffling property” and the different impact of the coefficients of the MFCC vector.

Algorithm (1) handles the shuffling property of the MFCCs, as it differentiates between two MFCC vectors regarding to the places of the dominating coefficients. The following pseudo-code shows the steps for calculating such similarity.

```
Purpose : to measure the distance between two vectors of MFCC
Input : MFCC vector A, MFCC vector B of length N
Output: distanc between A, B
Procedure:
1 Create two vectors Ai, Bi with the same length of A,B to store
  the indices of the elements in A,B
2 distance=0
3 Sort the elements of both A,B descending with corresponding
  indices in Ai, Bi
4 For i=0 to N-1
    5 distance += $wi|Ai(i)-Bi(i)|$, wi is the corresponding weight of
      each attribute
6 Return (distance)
```

Algorithm (1) : The proposed distance measure

Consider MFCCs vectors A, B, C, D, E, and F with length N = 8 as follows:

A= (4, 4, 4, 4, 4, 4, 4, 4) , B = (8, 4 , 7 , 4, 1 , 3 , 2, 9) , C = (8, 4, 4, 1, 7, 3, 2, 9), D= (4, 4, 4, 4, 4, 3, 4, 4) , E= (4, 3, 4, 4, 4, 4, 4, 4) , and F= (4, 4, 4, 4, 4, 4, 4, 3)

Algorithm (1) measures the similarity between A and B; D(A,B)=14 and between A and C , D(A,C)= 18, while all conventional distances deals with D(A,B) and D(A,C) as same distance. If we considered the distance D(F,D) , D(F,E) we can notice that D is more similar to F than E , and the proposed distance confirms that with D(F,D)=4 and D(F,E)=12. Algorithm (1) measures the similarity between A and B as shown in Table (4.2):

Table 4.2: The distance calculation using the proposed distance measure

A	4	4	4	4	4	4	4	4		…	(1)
B	8	4	7	4	1	3	2	9		…	(2)
Sorted A	4	4	4	4	4	4	4	4		…	(3)
Ai	1	2	3	4	5	6	7	8		…	(4)
Sorted B	9	8	7	4	4	3	2	1		…	(5)
Bi	8	1	3	2	4	6	7	5		…	(6)
|Ai-Bi|	7	1	0	2	1	0	0	3	⇨ 14	…	(7)

Algorithm (1) concentrates on the order and the places of the MFCC coefficients only, it does not consider the difference between the coefficients in the same place in the MFCC vector as the other conventional methods in Chapter (3) do. The values of the coefficients have important effect on distance calculation between the MFCC vectors, so Algorithm(1) needs to be combined with one of the conventional methods to get the most efficiency. However, the combination operation may affect the proposed distance measure results, so some separation is needed to not fall in the problems of conventional methods again. Thus, we propose new separation method while keeping the effect of the proposed distance measure highest. That can be accomplished by calculating the farthest distance between MFCC vectors when using one of the conventional methods and then adding it to the result of the proposed distance after raising it to the least nearest power of ten.

We used Manhattan distance [64] as it completes the idea of the proposed distance measure of calculating the required steps of converting one MFCC vector into another. Finally, equation (4.9) summarizes the modified proposed distance.

$$P = M_D + 10^m * A1_D \tag{4.9}$$

Where M_D is the Manhattan distance, $A1_D$ is the distance calculated from alogrithm1, and m is the next power of then after the maximum value of M_D.

4.5 Visualization system

4.5.1 Visualization system framework

In this section, we present our proposed visualizing system based on the interviews that were held in Chapter (3). Figure (4.3) illustrates the overall system separated in modules (Audio processing module , similarity measure module , classification module , and visualizing module).

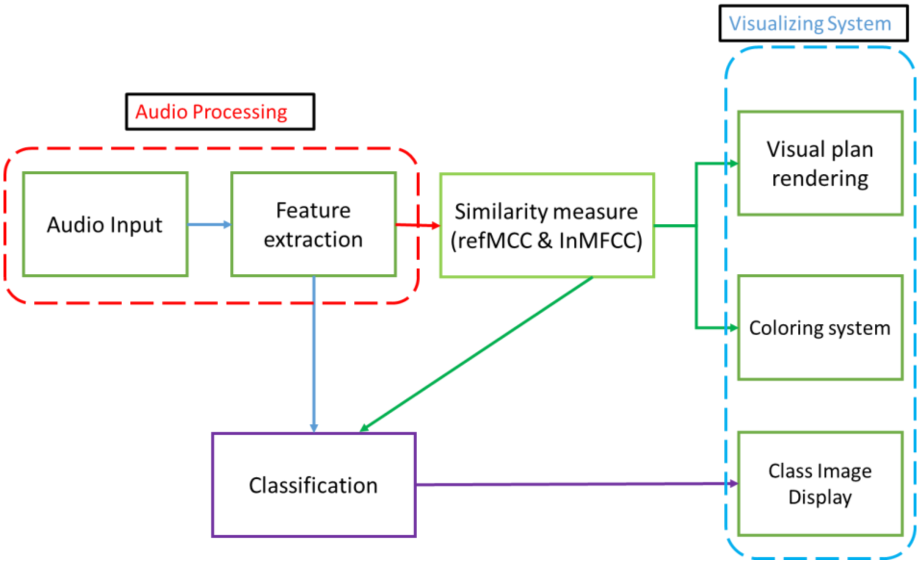

Figure 4.3: The overall system design

The proposed visualizing system mainly transforms the sound feature into colors in real time and visualizes them in 3D manner. A 2D plane with color represents a sound is drawn and its height varies according to the power level of the sound signal. In order to keep the user able to memorize the previous sounds, the previous sound levels and colors are kept on the drawing canvas like a sliding window of time with period of three second past, see figure (4.4). The visualizing is drawn line by line from upper left to the lower right. Each line is one pixel wide and represents the analyzing one frame of sound input in real time.

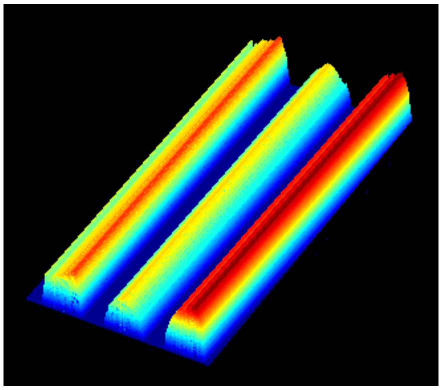

Figure 4.4: Sound visualizing early steps

The next two sections explain in detail the procedure of producing the visualizing system.

4.5.2 Color plan

As discussed in Chapter (3), the deaf person is highly attracted to colors. For that reason, we designed the system to visualize the sounds by mainly coloring them with wide range of color spectrum. Because environment contains wide range of sounds that cannot be grouped into finite number of sound sets, we intended to color similar sounds not by specified flat color, but with ranges of colors. The user can differentiate between similar and dissimilar sounds easily.

The coloring procedure is done by calculating distance between the MFCC vectors of input sound with a MFCC vector of reference sound by using the proposed distance measure in the previous section. The overall distance is normalized according to equation (4.10) to range of [0, 1].

$$Normalized\ distance = \frac{D_{R,I}}{\max(D_{R,I}) - \min(D_{R,I})} \tag{4.10}$$

Where $D_{R,I}$ is the proposed distance between MFCC vector R and MFCC vector of input sound.

The normalized distance is mapped to the value of the color map shown in figure (4.5) .This figure represents the jet color map in MATLAB [86] which is used by the research community in visualizing data. The used color map contains wide range of colors suitable for differing between sounds easily.

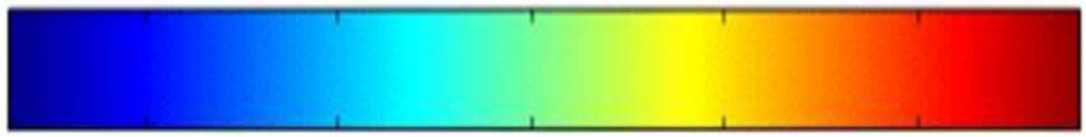

Figure 4.5 : MATLAB color map, jet

4.5.3 Sound power level

To keep the user of the system aware with the changes of the environmental sounds interactively, we need to connect the visualizing system with the loudness of the sound signal. The loudness of audio signals is the most prominent feature according to human aural perception [15].To define volume quantitatively we used equation (4.11) to calculate the loudness of the sound, which is the sum of absolute samples within each frame.

$$SL = \sum_{i=0}^{N-1} |x_i| \tag{4.11}$$

4.6 Classification

Our proposed system tends to simplify the sound visualizing operation as much as possible. The deaf person is naturally not aware of the surrounding sounds and cannot relate the visualized color sound with the real sound. However, the natural intelligence of the human can by practice connect the visualized sound with real sound by slight helping by other humans, direction of the sound or by automated recognition system that can help the deaf user. We tend to use recognition to increase the usability of the system by training the system for well-known sounds to be recognized by simple recognition method. Name or icon of the recognized sound is displayed on the screen for the user besides the corresponding visualized sound. The deaf person can connect the other visualized sounds by real life experience or by training our classifier with the new sound.

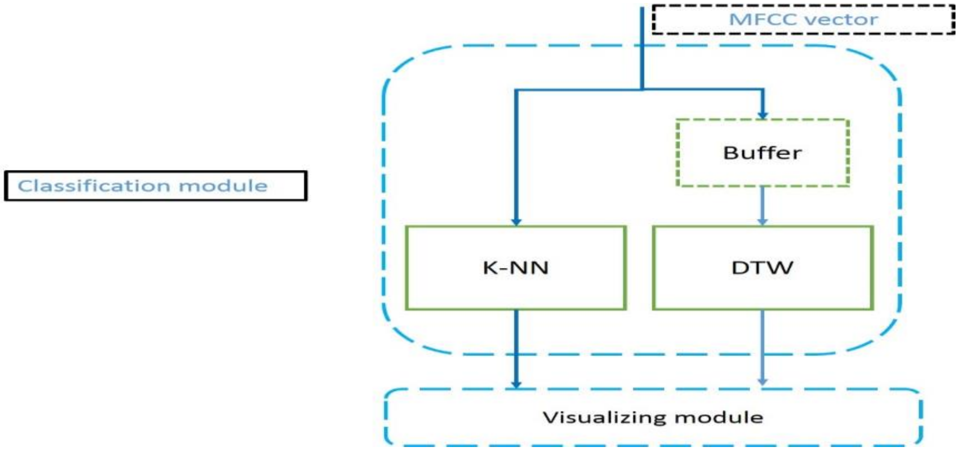

Figure 4.6: Classification module

Figure (4.6) illustrates in detail the classification module , where two classifiers are running in same time by implementing Java threads. The result of the classification is delivered to the visualization model to be displayed as icons on the display. Every prior known class has an icon and classifier model to be used in real time classification.

4.7 Implementation

The performance of the algorithms was tested using MATLAB since it support many useful toolboxes to implement and analyze many similarity measures, classification algorithms , and data visualization color maps. We decided for the MATLAB environment because it provides a comfortable interface for sound processing and a large number of basic sound algorithms.

The development of the system required to be implemented on a smartphone according to the preferred visualization device of the interviewed deaf. The smartphone application was developed in Java for Android. Android is an operating system and software platform for smartphones. We used Android v.2.3 (API-10) "Gingerbread" to implement the application , but the application can work on smartphones the runs Android version is v.4.x (API-15) "Ice Cream Sandwich" with slight deployment modifications. Eclipse IDE v.3.8.0 with Android Development Toolkit was used during development.

To produce real-time visualizations, Canvas and OpenGL framework were used within the Android platform. Threads were used to run the two classification algorithms in the same time.

4.8 Conclusion

We presented the steps of building our proposed system. We discussed the interview results with deaf participants. The behavior of the proposed system, starting from the processing details of sound input was explained. After that , the extraction of sound features and the proposed similarity measure were discussed in detail. Finally, we illustrated the overall system architecture and presented the implantation tools that we used in building our proposed system .

CHAPTER 5

EXPERIMENTS AND RESULTS

In this chapter, the performed experiments were described and discussed. Different groups of sound features, similarity measures, and classifiers where tested and compared in order to choose the best of each group to build the proposed visualizing system.

The proposed system was done through two phases of experiments. The first phase was done for choosing the best sound feature vector, similarity measure, and the best classifier for using them in the proposed system, where dataset of sound samples was built for experiments. The second phase was built on smart phone for testing with deaf individuals to produce the most suitable visual interface for sound based on the interviews that were made earlier.

5.1 Data set

Five types of environmental sounds, namely door bells, cars, speech, crowd, explosions are chosen for experimentations. Sounds of explosions are chosen to represent most sever situation where deaf individual must be notifie0d accurately. In the other hand, sounds of cars and crowds show many similarities on the technical and perceptual level. This makes the chosen classes suitable for measuring the quality of features, similarity measures, and the classification algorithms.

We built a database of sound samples by collecting the preferred samples from well-known datasets [87] [88] [89]. The dataset contains 430 (80 door bells, 100 cars, 130 speech, 70 crowds, and 50 explosions) samples. All signals in the database have a 16 bit resolution and are sampled at 44100 Hz mono channel. In this way, all possible sound spectrum components can be introduced for experimentation purpose. This point is very important for environmental sounds, because some sounds show an important energy content in the highest frequencies, like glass breaks for example. The samples duration is fixed of four seconds but have different loudness levels. Each sound sample is assigned to exactly one of the five classes.

5.2 System environment

5.2.1 Algorithm choosing phase

The system that was used during this phase is MATLAB program version 7.8.0.347 (R2009a). We used a platform of Intel Core i5 with 4 GB RAM during the experiments.

The goal of this framework is to choose the most suitable sound features, similarity measure, and classifier for the proposed system to be implemented on smartphone. Thus, the challenges arise when considering the smart phone computation power and real time performance with complex algorithms.

5.2.1.1 Distance measures algorithms

All the distance measures $D_1 - D_{11}$ that were mentioned in Chapter (3) with the proposed distance measure are evaluated to choose the most discriminative distance measure. The importance of this step is that the most important part of the visualizing system, which is sound colors, depends on the distance between reference sound and the input sound.

The evaluated distance measures are considered local distance measure, so the evaluation criteria we used is the recognition rate of a classifier that uses local distance measure for classification. We used K-NN classifier for classifying every time frame in real time and Dynamic Time Wrapping DTW for classifying input sounds with long duration to preserve the perceptual properties of the sound.

Figure (5.1) shows the recognition rate for the K-NN classifier and DTW using the mentioned distances. We can notice the benefit of the proposed distance measure for increasing recognition rate for both classifiers more than any other distance measure.

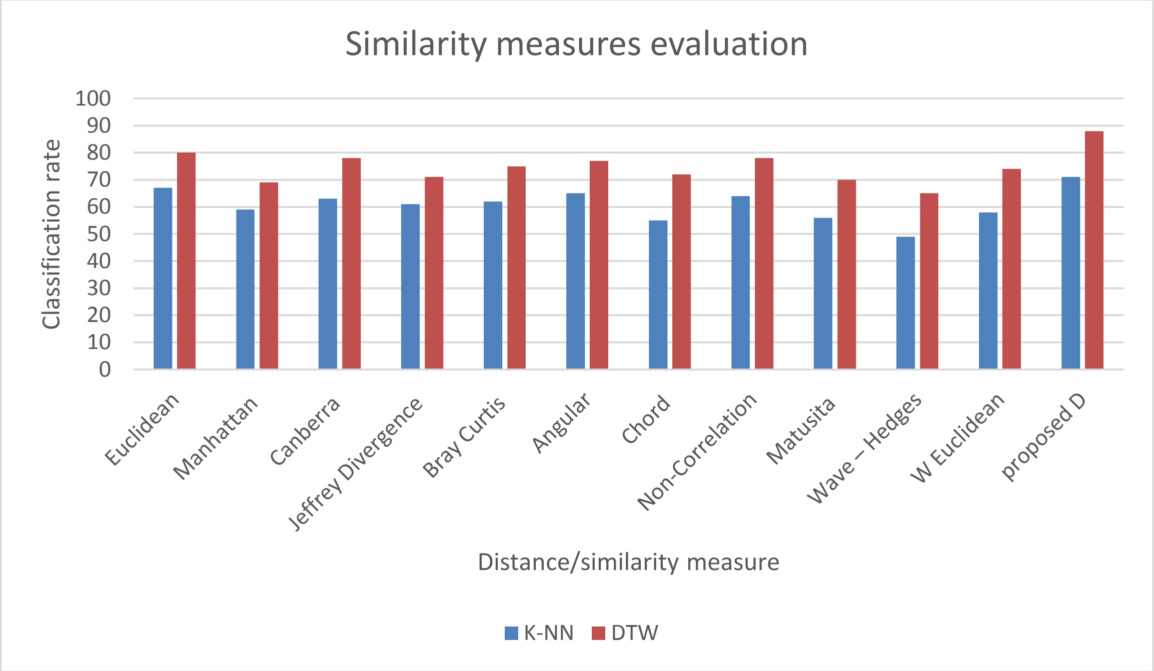

Figure 5.1: Similarity measures evaluation using k-NN and DTW

5.2.1.2 Feature extraction algorithms

By considering the popularity of sound features we will compare the most well-known sound feature in literature. The MFCC and LPC sound features were compared to be one chosen feature in our proposed system. The comparison will be done using popular classification algorithms to avoid classifier dependence false decisions as shown in Table (5.1) and Table (5.2).

Table 5.1: Recognition rate for LPC features

LPC	K-NN	DTW
	Recognition rate %	Recognition rate %
Cars	45	65
Crowd	41	40
speech	75	82
explosions	73	69
Door bells	72	79

Table 5.2: Recognition rate for MFCC features

	K-NN	DTW
	Recognition rate %	Recognition rate %
Cars	50	71
Crowd	49	52
Speech	78	86
explosions	79	72
Door bells	75	82

We notice that MFCC has beaten the LPC features in most of the sound classes. This result is not surprising as most previous research noticed the robustness of MFCCs for environmental and speech sound. In addition the computational complexity and the obtained results, MFCC will be applied in our visualization system.

5.2.1.3 Classification algorithms

The simplicity of K-NN and DTW in addition to their dependence on local distance measures made they suitable to be tested in our proposed system. In addition to that, K-NN uses short time frames for classification, while DTW uses sound duration more than one second for classification. The merging of these two classifiers make us avoid missing short sound events and classifying sounds that require longer time to be detected

The results for this step can collected from the results of the previous two section and illustrated in figure (5.2). As expected the K-NN with short time frames has recognition rates with classes that happens in short time like explosions, while the recognition rate increases when using DTW with other classes like speech and door bells.

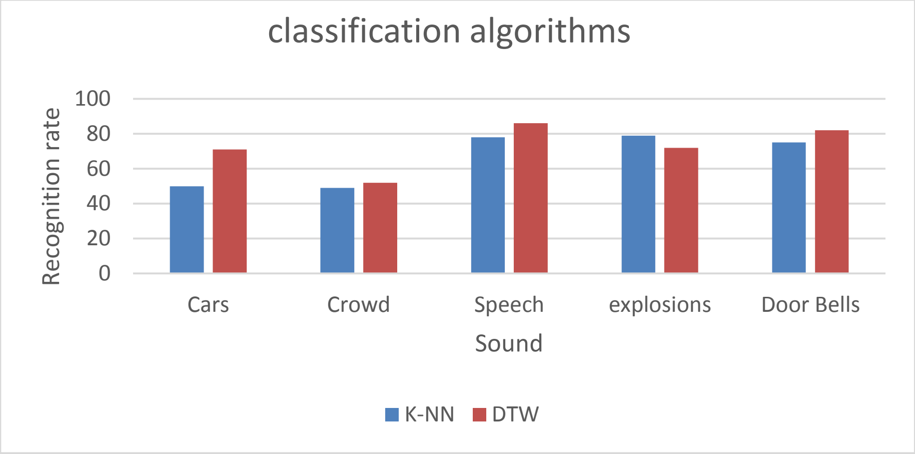

Figure 5.2: Recognition rate with multiple classes

5.2.2 System implementation phase

The overall system was implemented on smart phone of model number Galaxy Ace 2 made by SAMSUNG co , which has a Dual-core 800 MHz , 4 GB storage, 768 MB RAM , and runs Android operating system.

Android is a free open source operating system for mobile devices, running on a Linux kernel, and owned by Google. Android provides various applications written in Java programming language. This operating system includes a set of core libraries [90] that provides most of the functionalities available in the core libraries of the Java programming language. In order to develop Android applications, developers use the Android System Development tool Kit (SDK). It provides all the necessary tools to write, compile and run an Android application with or without a connected mobile device, as the emulator emulates an Android mobile phone. Once the latter is installed, it is easy and simple to use it with Eclipse IDE.

For fast video rendering of the visualized sound we used OpenGL ES 2 framework on Android [91], which uses the phone's GPU and provides simple API to call the native interfaces implemented inside.

5.3 Visualization

The visualization is drawn on a rectangular canvas with adaptive size to fit on any android device’s display. It consists of two parts; the first part is the 3D colored visualization of the acquired sound while the second part series of images displaying the symbols of recognized sounds as shown in figure (5.3). The visualized sound flows from left to right as so as the additional icons that appear if the classifiers recognize any sound.

Since the daily sounds are countless we focused on the classes in section (5.1) to be presented in the following figures. In addition to that, the mentioned classes were trained with the used classifiers so we can present the full visualization system.

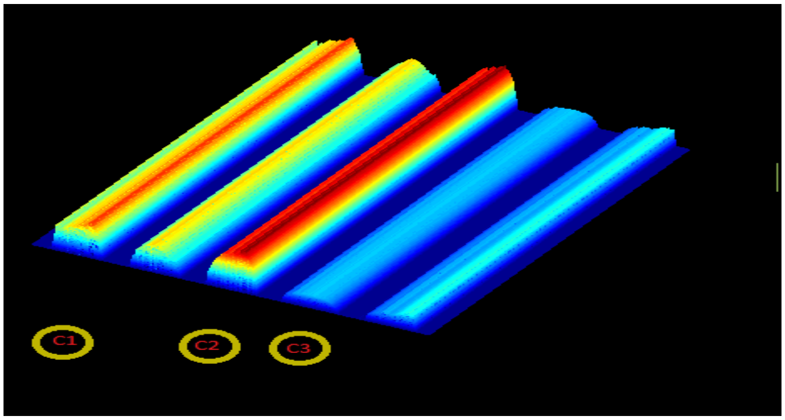

Figure 5.3: General view of the visualization

5.3.1 Speech

Figure (5.4) shows the visualizations of a number of different voiced Arabic vowels (ا , و , ي). Since vowels shows some constant representation of the sound during the voice, we can notice clearly the visualized sound even if it cannot recognized by the classifiers.

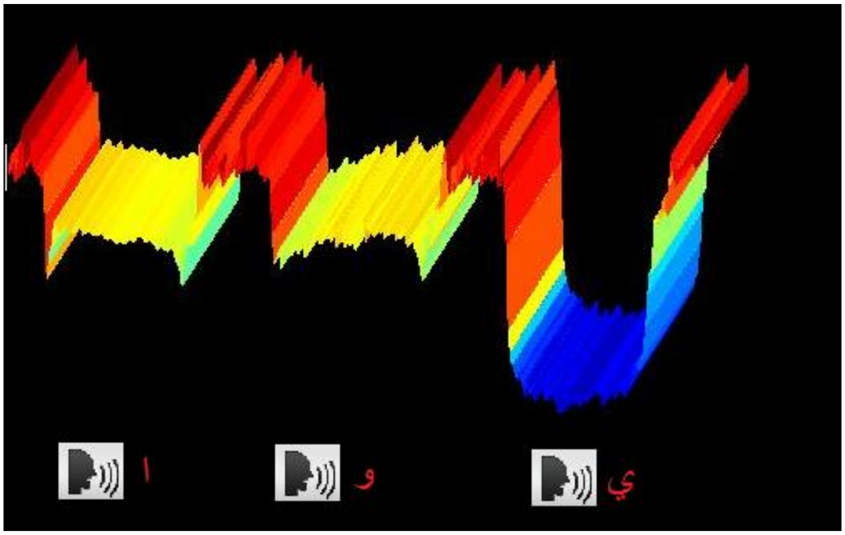

Figure 5.4: The visualized Arabic vowels

The reference sound used for the similarity measure and hence for the visualization system, is picked from the third vowel, so we can notice that the third vowel has the lowest height in 3D mesh.

The additional notices from figure (5.4) is that the three different vowels show related colors (except the third, because we used the reference from one of its frames) because they belong to speech class. The third vowel shows blue color during the visualization for expressing our point of view only hence we use a reference sound represents silence in the real time application.

5.3.2 Door bells

Figure (5.5) shows the visualizing result for doorbell sound. The interesting thing about this visualized doorbell is that it displays icon of doorbell in yellow (warning color) above bird icon. This happened because in fact the doorbell is designed to output bird sounds. Since one of the classifiers, detect that this sound is likely to be a bird sound and the other for doorbell sound. The visualizing system displays the icon of both classes.

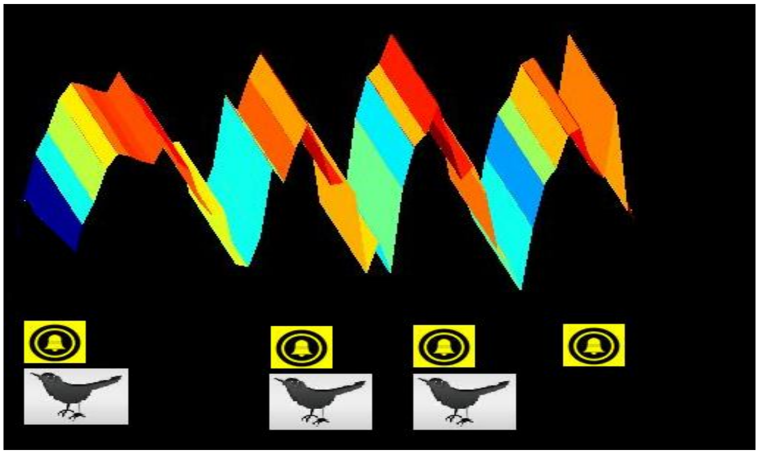

Figure 5.5: Visualized door bell tone

5.3.3 Explosions

This class represents the most severe case among all other classes. The mobile phone makes a vibration besides the visualization. The explosions includes gunshots, heavy falling mass and real known explosions.

Figure (5.6) shows the visualization of explosion sound. The visualizing system showed the explosion icon with vibration on the test smart phone.

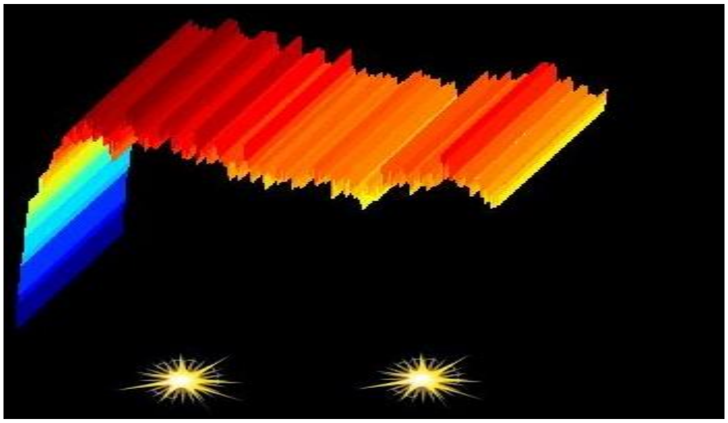

Figure 5.6 : Explosion sound visualization

5.4 Evaluation tests

5.4.1 Training phase

The evaluation of the proposed system was done to investigate readability of the visualization system and durability of the learning effect. Because deaf person is not aware of the sound he/she cannot evaluate the system directly without training. We can consider our visualization system successful if the trainers of the system can give high correct answer rate about if they understood the meaning of the visualized sound and if the response time duration was fast. The five deaf persons that involved in our interviews engaged in our evaluation tests.

In the training phase, a pair of recognizable sounds by our implemented classifiers and visualized sound is presented at the same time and one by one, for the learning users. Next , the confirmation of the learning procedure is done by simple tests with visual pattern only and the test trainees are asked for the meaning of the visualized pattern .After the correct response rate to be obtained by the confirming test had reached a sufficiently high score (more than 90%) new other sounds were appended to the list and the training continued likewise. Finally, when the confirming test using 50 recognizable sounds showed good results, the training session was closed, the learning time was recorded, and the real test session was started. Figure (5.7) shows the correct answer rate of each user during the learning sessions.

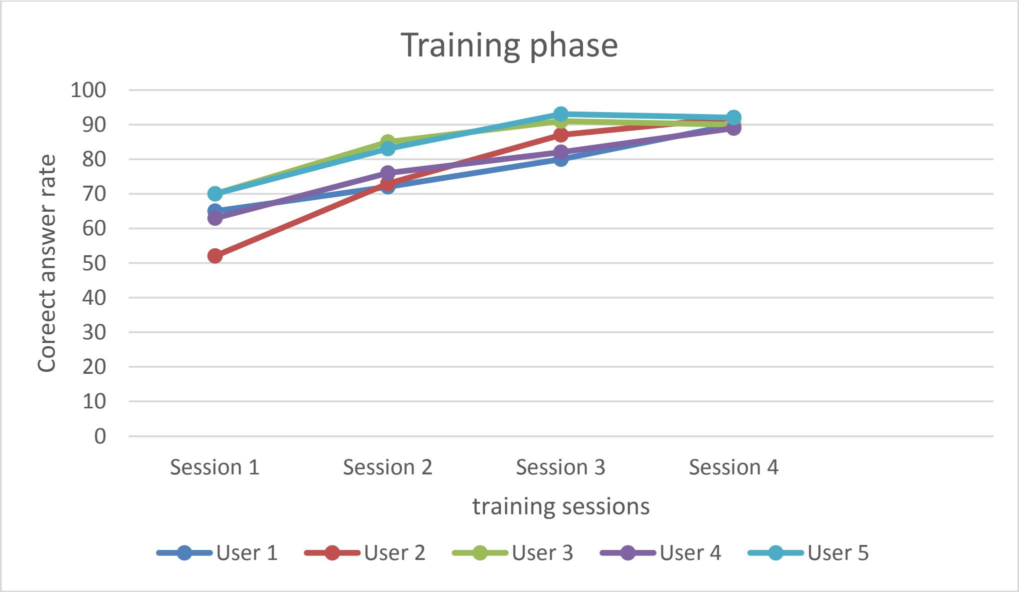

Figure 5.7: Learning sessions for users

5.4.2 Testing phase

The testing phase is somehow similar to the training phase but without any outer help of the trainers. New additional not recognizable sounds were played for the visualizing system , and the trainees are asked for every sound about its class if it was recognizable by them.

For every sound, the answers of the trainees are collected with their response time for every answer is recorded for analysis. Due to the deaf inability to analyze sound from previous experience, the tests were made repeatedly and the results only picked in the last two sessions and only for correct answer rate of 90% and above.

Figure (5.8) shows the average duration of correct answers curve for the testing users. As we can note, the users at first find some difficulties for giving correct answers with the new sounds during session 1. In the next sessions , the users shows improvements in response time . The interesting notice about the final results is that the response time reached several few seconds this indicates that they can use the program in real time with little difficulties

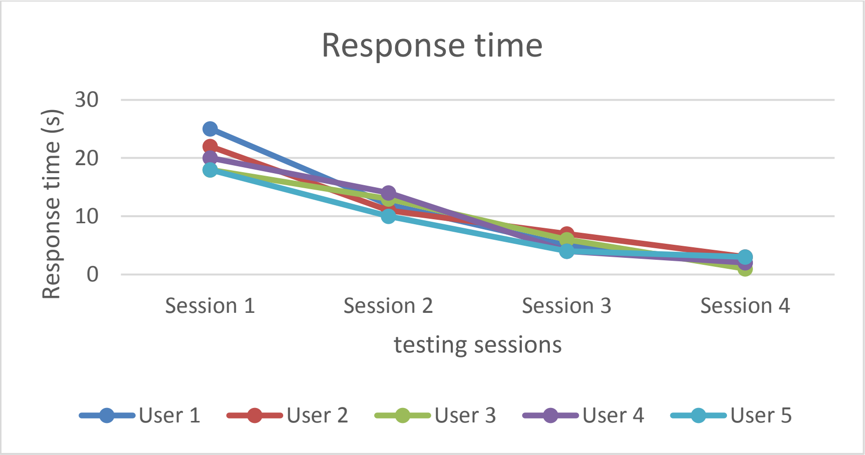

Figure 5.8: Testing sessions for users and their response time

5.4.3 Comparison with other visualization systems

In Chapter (4) we analyzed the interviews with some deaf persons and built our proposed system to avoid the disadvantages of other systems. Consequently, the comparison with other visualization systems is inadequate in the experiments phase. Besides to that, the implementation of other visualization systems is not an easy job.

5.5 Conclusion

The results that led us to present our proposed system were illustrated in this chapter. The extracted results came from two phases of experiments. The first phase was done, firstly by constructing a new data set and then by evaluating the suggested algorithms for building our system. Based on the results that depicted from the first phase, the second phase of testing included the evaluation of the constructed system with real users.

CHAPTER 6

CONCLUSION

6.1 Summary and conclusion remarks

We proposed a new visualization system to help deaf person to experience surrounding sounds. This system depends on vision sense of deaf to understand the sound visualization. Technically, the system depends on extracting robust sound features and comparing them with reference sound feature for using the comparison result for visualizing the sound in 3D curve with different colors. The building of the system involved in using feature extraction, similarity measures, classification, and rendering frameworks.

The proposed system could not use the traditional similarity measures for getting suitable visualization result, so a new similarity measure was proposed for handling the robust MFCC sound features in order to get robust visualization system. The proposed similarity measure suitable for many feature vectors that uses DCT in its final steps, where is used for dimensionality reduction and data compaction.

The sound feature that was used for representing sound is MFCC by evaluating many sound features and picking the highest recognition rate feature vector. Since, there is wide range of feature vectors proposed previously, our evaluation done on the most well-known features in open literature.

The classification algorithms that suited our proposed system are K-NN and DTW , considering there computational complexity , recognition rate ,and sound features special properties. K-NN classifier mainly used for short time frames to classify fast sound events so reducing miss rate. The DTW classifier used to fit the perceptual properties of sound since, there is many classes of sound can be detected in dynamic time series.

We formed sound database from other three databases to get different sound classes that fits the resulted application-working environment. We used our database for evaluating many sound features, similarity measures , and classification algorithms .

The visualization system renders the frames of sound as 3D dynamic mesh changing over time to give the user real time feeling with sound. The dynamic color and height of the visualized sound can be read easily by little experienced user. The color range is wide to represent various sound changes with time. The display presents 3 seconds of sound time, so the user can memories some past sound events easily. Classification results shown during sound visualization to give the user more self-experience of the system.

6.2 Recommendations and future work

The visualization system depends on many variables to be enhanced

- The sound features could be enhanced if we used more representative sound features than MFCC , since too many research were made to enhance MFCC and other sound feature that we recommend to test.
- The similarity measure can be enhanced if developed to be more robust to noise.
- The classification algorithms are simple so the classification rates reduced with some sound classes.
- We tend to add sound separation module to the system to enhance visualization as so as the classification rate .
- The proposed system is developed using Android operating system. It will be better if we programed it on many other smart phone operating systems.
- The main future work we need to add is the visualizing of sound direction , since our system doesn't support. This will enhance the usability of the system .

REFERENCES

[1] Yang, X. Wang, K. and Shamma, S. (1992). Auditory Representation of Acoust Signals. Transactions on Information Theory , IEEE, vol. 38, no. 2, pp.824-839.

[2] Palestinian Central Bureau of Statistics.(2012). [Online] Available at: <http://www.pcbs.gov.ps/default.aspx> [Accessed 10 June 2013].

[3] Adams, J.W. and Rohring, P., 2004. Handbook to Service the Deaf and hard of hearing: A bridge to accessibility, (2004), San Diego, CA: Elsevier Academic Press, p.31.

[4] Scheetz, N. (2004). Psychosocial Aspects of Deafness: Pearson Education Incorporated.

[5] Pratt, R. (1993). Music Therapy and Music Education for the Handicapped: Developments and Limitations in Practice and Research, 3rd ed. :MMB Music Incorporated.

[6] University of Washington. (2001). Brains Deaf People Rewire to 'Hear' Music, Science Daily. [Online] Available at: <http://www.sciencedaily.com/releases/2001/11/011128035455.htm> [Accessed 28 November 2012].

[7] Shibata, D. (2001).The Radiological Society of North America 87th Scientific Assembly and Annual Meeting. November 25-30. Chicago, USA.

[8] Palmer, R. (1997). Feeling the Music Philosophy. Third Nordic Conference of Music Therapy. June 12-15. Finland.

[7] Russ Palmer, Feeling the Music Philosophy, Third Nordic Conference of Music Therapy in Jyväskylä, Finland 12-15 June, 1997, p 5

[9] Microsoft Windows Media Player. (2013).[Online] Available: <http://windows.microsoft.com/en-us/windows/windows-media-player > [Accessed 28 January 2013].

[10] Zhang, X. (2009). Audio Segmentation, Classification and Visualization. Ph.D. thesis, Auckland University of Technology, New Zealand.

[11] Foote, J. (1997). Content-base Retrieval of music and Audio Multimedia storage and Archiving Systems II, Multimedia Storage and Archiving Systems II. In Proc. of SPIE, vol. 3229, pp. 138-147.

[12] Scheire, E. and Saleny, M. (1997). Construction and evaluation of robust multifeature speech/music discriminator. IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 2, pp. 1331 – 1334.

[13] Borwn, J. (1998). Music Instrument identification using autocorrelation coefficients. Proceedings International Symposium on Musical Acoustics ISMA 1998, Leavenworth, Washington, pp. 291-295.

[14] Matin, K and Kim, Y. (1998). Instrument Identification: a pattern recognition approach 136th Meet, USA

[15] Rabiner, L. and Jung, B. (1993). Fundamentals of speech recognition. Prentice-Hall,USA.

[16] GeoFroy, P. (2004). A large set of audio features for sound description (similarity and classification), Iracm project ,France.

[17] Stevens, S. Volkmann ,J. and Newman, E. (1937). A scale for the measurement of the psychological magnitude pitch. Journal of the Acoustical Society of America. vol. 8 , no. 3. pp.445-490.

[18] Zwicker, E. (1961). Subdivision of the audible frequency range into critical bands. The Journal of the Acoustical Society of America, vol. 33, no. 2, pp. 248-248.

[19] Foote, J. (1999). Visualizing Music and Audio Using Self-similarity. Conf. Multimedia, 7th ACM Int (Part 1), Orlando, pp. 77–80.

[20] Ong, B. (2005). Towards Automatic Music Structural Analysis: Identifying Characteristic Within-Song Excerpts in Popular Music. Masters Degree Thesis, Computer Science and Digital Communication Department, Universitat Pompeu Fabra, Spain.

[21] Siegler, M. Jain, U. Raj, B. and Stern, R. (1997). Automatic Segmentation, Classification and Clustering of Broadcast News Audio. DARPA Speech Recognition Workshop, USA, pp. 97-99.

[22] Zhou, B. and Hansen, J. (2000). Unsupervised Audio Stream Segmentation and Clustering via the Bayesian Information Criterion. Proc. ISCLP'00, China, vol.3, pp. 714-717.

[23] Nomura, S. Shiose, T. Kawakami, H. Katai, and Yamanaka, K. (2004). Sound Visualization Process in Virtual 3D Space: the Human Auditory Perception Analysis by Ecological Psychology Approach. 8th Asia Pacific Symp. Intelligent and Evolutionary Systems, Australia, pp. 137-149.

[24] Jones, R. (1999). Sound Visualization and Analysis in the Pre-Electronic Era. [Online] Available at: <http://people.seas.harvard.edu/~jones/cscie129/images/snd_vis/snd_vis.html> [Accessed 21 October 2013].

[25] Berliner, E. (1888). The Gramophone: Etching the Human Voice. Journal of the Franklin Institute, vol. 125, no. 6, pp.425-447.

[26] Hagiwara, R. (2009). How to Read a Spectrogram, [Online] Available at: <http://home.cc.umanitoba.ca/~robh/howto.html> [Accessed 22 May 2013].

[27] Tzanetakis, G. and Cook, P. 2000. Audio Information Retrieval (AIR) Tools. MA Thesis, Princeton University, USA.

[28] Politis, D. Margounakis, D. and Karatsoris, M. (2008). Image to Sound Transforms and Sound to Image Visualizations based on the Chromaticism of Music. 7th WSEAS Int. Conf. Artificial Intelligence, Knowledge Eng. and Databases, UK, pp. 309-317

[29] Gerhard, D. (1999). Audio Visualization in Phase Space. In Bridges: Mathematical Connections in Art, Music and Science. pp. 137-144.

[30] Sobieczky, F. (1996). Visualization of Roughness in Musical Consonance. In Proc. IEEE Visualization, San Francisco, pp. 355-357.

[31] Hiraga, R. Watanabe F. and Fujishiro, I. (2002). Music Learning Through Visualization. In Proc. WEDELMUSIC'02, Germany, pp. 101- 108.

[32] Hiraga, R. Mizaki, R. and Fujishiro, I. 2002. Performance Visualization: A New Challenge to Music Through Visualization," in Proc. 10th ACM Int. Conf. Multimedia, Juan-les-Pins, France, , p. 239-242

[33] Cooper, M. and Foote, J. (2002). Automatic Music Summarization via Similarity Analysis. In Proc. ISMIR'02, France, pp. 81-85.

[34] Bergstrom, T. Karahalios, K. and Hart, J. (2007) . Isochords: Visualizing Structure in Music. ACM Int. Conf. Graphics Interface, Canada, vol. 234, pp. 297-304.

[35] Ferguson, S. Moere A. and Cabrera, D. (2005). Seeing Sound: Real-time Sound Visualization in Visual Feedback Loops used for Training Musicians. Conf. Information Visualization, 9th Int. UK, pp. 97-102.

[36] Kaper, H. Wiebel, E. and Tipei, S. (1999). Data Sonification and Sound Visualization. Journal of Computing in Science and Engineering, vol. 1, no. 4, pp. 48-58.

[37] Giannakis, K. and Smith, M. (2001).Imaging Sound Scapes: Identifying Cognitive Associations between Auditory and Visual Dimensions in Musical Imagery. Netherlands, pp. 161-179.

[38] Politis, D. Margounakis, D. and Mokos, K.(2004). Visualizing the Chromatic Index of Music. In Proc. WEDELMUSIC'04, Spain, pp. 102-109.

[39] Hiraga, R. and Matsuda, N. (2004). Visualization of Music Performance as an Aid to Listener's Comprehension. Italy, pp. 103-106.

[40] Lubar, K. (2004). Color Intervals: Applying Concepts of Musical Consonance and Dissonance to Color. Leonardo, vol. 37, no. 2, pp. 127-132.

[41] Oppenheim, D. (1992). Compositional Tools for Adding Expression to Music. In Proc. ICMC'92, USA, pp. 223-226.

[42] Watanabe, F. Hiraga, R. and Fujishiro, I. (2003). Brass: Visualizing Scores for Assisting Music Learning. In Proc. ICMC'03, Singapore, pp. 107-114.

[43] Dixon, S. Goebl, W. and Widmer, G. (2002). The Performance Worm: Real Time Visualization of Expression Based on Langner's Tempo-Laudness Animation. In Proc. ICMC'02, Sweden, pp. 361-364.

[44] Hiraga, R. Igarashi, S. and Matsuura, Y. (1996) . Visualized Music Expression in an Object-oriented Environment. In Proc. ICMC'96, China, pp. 483-486.

[45] Hiraga, R. Watanabe F. and Fujishiro, I. (2002). Music Learning Through Visualization. In Proc. WEDELMUSIC'02, Germany, pp. 101- 108.

[46] Hailpern, J. Karahalios, K. Halle, J. DeThorne, L. and Coletto, K. (2008). Visualizations: Speech, Language & Autistic Spectrum Disorder. In Proc. CHI'08, Italy, pp. 3591-3596.

[47] Karahalios, K. and Bergstrom, T. (2006). Visualizing Audio in Group Table Conversation. Workshop Horizontal Interactive Human- Computer Systems. In Proc. 1st IEEE Int., Australia, pp. 131-134.

[48] Bergstrom, T. and Karahalios, K. (2007). Seeing More: Visualizing Audio Cues. In Proc. INTERACT'07, vol. 4663, pp. 29-42.

[49] Simunek, M. (2001) . Visualization of Talking Human Head. In Proc. CESCG'01, Slovakia, pp. 30-33.

[50] C. Bregler, M. Covell and M. Slaney, "Video Rewrite: Driving Visual Speech with Audio," in Proc. ACM SIGGRAPH, Los Angeles, CA, USA, 1997, p. 353- 360.

[51] Smith, S. and Williams, G. (1997). A Visualization of Music. In Proc. 8th Conf. Visualization, Phoenix, USA, pp. 499-503.

[52] Chaudhary, A. (1998). OpenSoundEdit: An Interactive Visualization and Editing Framework for Timbral Resources. Masters Degree Thesis, Computer Science, University of California, Berkeley, CA.

[53] Kunze, T. and Taube, H. (1996). A Structured Event Editor - Visualizing Compositional Data in Common Music. In Proc. ICMC'96, pp. 63-66.

[54] Wai-ling, F. Mankoff, J. James A. (2003). From Data to Display: the Design and Evaluation of a Peripheral Sound Display for the Deaf. In Proc. of CHI, p. 8.

[55] Matthews, T. Fong, J. and Mankoff, J. (2005). Visualizing Non-Speech Sounds for the Deaf. In Proc. of ACM SIGACCESS on Computers and Accessibility (ASSETS). Baltimore, pp. 52-59.

[56] Yeo, W. and Berger, J. (2006). A New Approach to Image Sonification, Sound Visualization: Sound Analysis And Synthesis. In Proc. of the International Computer Music Conference, New Orleans, pp.34-50.

[57] Kil, D.H and Shin, F.B. (1996). Pattern Recognition and Prediction with Applications to Signal Characterization. AIP Press, New York.

[58] Courveur, C. (1997). Environmental Sound Recognition : a Statistical Approach. PhD thesis, Faculte Polytechnique de Mons, Belgium.

[59] Dufaux, A.(2001). Detection and Recognition of Impulsive Sound Signals. PhD thesis, University of Neuchatel, Switzerland.

[60] Deller, J.R Proakis , J.G and Hansen, J.H.(1993). Discrete-Time Processing of Speech Signals. Prentice Hall, USA.

[61] Davis, S.B. Mermelstein, P. (1980). Comparison of Parametric Representation for Monosyllabic Word Recognition in Conituously Spoken Senteces. IEEE Transactions on Acoustics, Speech and Signal Processing, vol. 28, no.4, pp. 357-366.

[62] Yang, X. Wang, K. Shamma, S.A. (1992). Auditoy Representation of Acoustic Signals. IEEE Transactions on Information Theory, vol.38 , no 2, pp. 824-839.

[63] Fletcher, H. (1940). Audotiry Patterns. *Rev. Mod. Phys.*, vol. 12, pp. 47-65.

[64] Deza, E. & Deza, M. Marie. 2009 . Encyclopedia of Distances. pp.94-236.

[65] Johnson, R. and Wichern, D. (1998). Applied multivariate Statistical Analysis. Prentice Hall, USA, pp. 226-235.

[66] Deza, E. & Deza, M. Marie. (2012). Encyclopedia of Distances. The 1st edition, Elsevier, p.223

[67] Bray, J. and Curtis, J. (1957). An ordination of upland forest communities of southern Wisconsin. Journal of Ecological Monographs, vol.231 , no. 34, pp. 325-349.

[68] Foote, J. (2000).Automatic Audio Segmentation Using a Measure of Audio Novelty. In Proc. ICME'00, vol.1, pp. 452-455.

[69] Deza, E. & Deza, M. Marie. (2009). Encyclopedia of Distances. The 2^{nd} edition, Elsevier, p.236

[70] Székely, G. Rizzo, M.L. and Bakirov, N.K. (1992). Measuring and testing independence by correlation of distances. Journal of Annals of Statistics, vol. 34, pp. 2769–2794.

[71] Lu, L. Wang, M. and. Zhang, H.-J. (2004). Repeating Pattern Discovery and Structure Analysis from Acoustic Music Data. In Proc. MIR'04, USA, pp. 275-282.

[72] Jain, A. Murty, M. and Flynn. P. (1999). Data clustering: a review. ACM Computer Survey, vol. 31, pp.264–323.

[73] Kohonen, T. (1997). Self-organizing maps. Inc., Secaucus, NJ, Prentice-Hall, USA

[74] Cover, T. and Hart. P. (1967). Nearest neighbor pattern classifications. IEEE transaction on information theory, vol.13, pp.21–27.

[75] Rabiner, L. and Juang, B. H. (1993). Fundamentals of speech recognition. Prentice-Hall, USA.

[76] Atfaluna Society for Deaf Children. (2013). [Online] Available at: <http://www.atfaluna.net/ar/> [Accessed 3 May 2013].

[77] Temko, A. (2007). Acoustic Event Detection and Classification. PhD thesis, University Politecnica De Catalunya, Spain.

[78] Nirjon, S. Dickerson, R. Stankovic, J. Shen, G. and Jiang, X. (2013). SMFCC: Exploiting Sparseness in Speech for Fast Acoustic Feature Extraction on Mobile Devices. a Feasibility Study. In Proc. of Hot Mobile, p.456.

[79] Roger Jang, J.-S. (2013). Audio Signal Processing and Recognition. National Taiwan University .[Online] Available at : <http://mirlab.org/jang/books/audioSignalProcessing> [Accessed 3 September 2013].

[80] Becchetti, C. and Ricotti, L. (1999). Speech Recognition. John Wiley and Sons, England.

[81] Duhamel, P. and Vetterli, M. (1990). Fast Fourier transforms A tutorial review and a state of the art. Journal of Signal Processing, vol.19, pp. 259–299.

[82] Huang, X. Acero, A. and Hon, H. (2001) .Spoken language processing A guide to theory, algorithm, and system development. Prentice-Hall, USA.

[83] Discrete Cosine Transform Tutorial. (2013). [Online] Available at: <http://www.haberdar.org/Discrete-Cosine-Transform-Tutorial.htm> [Accessed 22 May 2013].

[84] Strang, G.(1999).The Discrete Cosine Transform. SIAM Review, vol. 41, pp. 135-147.

[85] Ahmed, M. and Natarajanm, T. (1974). Discrete Cosine Transform. IEEE Transactions on Computers, vol. 67, pp.90-100.

[86] Colormap .(2013).[Online] Available at: <http://www.mathworks.com/help/matlab/ref/colormap.html> [Accessed 5 November 2013].

[87] DeWolf Sound effect Database .(2013).[Online] available at: < www.dewolfe.co.uk > [Accessed 5 January 2013].

[88] EFX Guns Library. (2013).[Online] available at: <www.efx-sound.com> [Accessed January 2013].

[89] Sound Spaces Environmental Sound Library.(2013).[Online] available at: <http://sounds.bl.uk/environment/soundscapes> [Accessed 5 June 2013].

[90] Saha, A. (2008). Developer's First Look at Android: Linux for You. In Proc. of Hot Mobile, pp. 48-50.

[91] Android Frame Wrok Samples. (2013). [Online] available at: <http://developer.android.com/resources/samples/ApiDemos/src/com/example/android/apis/graphics/index.html> [Accessed 13 June 2013].

APPENDIX 1

Questionnaire

This questionnaire reflects the study main questions which make the base on designing the sound application of the study.

My dear deaf, this paper shows four items , please tick Yes/ No for each one according your own opinion.

No.	Items	Yes	No
1.	**What sounds are preferred for you?** ▪ The activity and presence of others. ▪ Sound from home environment . ▪ Highly dynamic environment sounds. ▪ Life critical sounds.		
2.	**What display size is <u>the most</u> preferred for you?** ▪ Mobile phone. ▪ PC monitor. ▪ large wall screen.		
3.	**What information about sounds do you think is important?** ▪ Display that shows sound classes. ▪ Display that shows sound location. ▪ Display that shows sound characteristics. ▪ Display that shows view a history of displayed sounds.		
4.	**Which way do you prefer to be aware with the visualizing system ?** ▪ Display that shows every sound that was made. ▪ Display that allows users to choose which sounds to show and filter out the rest.		