VIDEO SEGMENTATION BASED ON AUDIO FEATURE EXTRACTION

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

BY

NERİMAN ATAR

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

FEBRUARY 2009

Approval of the thesis:

VIDEO SEGMENTATION BASED ON AUDIO FEATURE EXTRACTION

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

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

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

Prof. Dr. Gözde Bozdağı Akar
Supervisor, **Electrical and Electronics Engineering** _____

Examining Committee Members

Assoc. Prof. Dr. A. Aydın Alatan
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. Gözde Bozdağı Akar
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. Engin Tuncer
Electrical and Electronics Engineering Dept., METU _____

Prof. Dr. Adnan Yazıcı
Computer Engineering Dept., METU _____

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

Date: 12.02.2009

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

Name, Last name: NERİMAN ATAR

Signature :

ABSTRACT

VIDEO SEGMENTATION BASED ON AUDIO FEATURE EXTRACTION

Atar, Neriman

M.Sc., Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Gözde Bozdağı Akar

February 2009, 98 pages

In this study, an automatic video segmentation and classification system based on audio features has been presented. Video sequences are classified such as videos with “speech”, “music”, “crowd” and “silence”. The segments that do not belong to these regions are left as “unclassified”. For the silence segment detection, a simple threshold comparison method has been done on the short time energy feature of the embedded audio sequence. For the “speech”, “music” and “crowd” segment detection a multiclass classification scheme has been applied. For this purpose, three audio feature set have been formed, one of them is purely MPEG-7 audio features, other is the audio features that is used in [31] the last one is the combination of these two feature sets. For choosing the best feature a histogram comparison method has been used. Audio segmentation system was trained and tested with these feature sets. The evaluation results show that the Feature Set 3 that is the combination of other two feature sets gives better performance for the audio classification system. The output of the classification system is an XML file which contains MPEG-7 audio segment descriptors for the video sequence.

An application scenario is given by combining the audio segmentation results with visual analysis results for getting audio-visual video segments.

Keywords: Video Segmentation, Audio Segmentation, MPEG-7 audio-visual segmentation, Video indexing.

ÖZ

SES ÖZNİTELİK ÇIKARIMINA DAYALI VİDEO BÖLÜTLENMESİ

Atar, Neriman

Yüksek Lisans, Elektrik-Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Gözde Bozdağı Akar

Şubat 2009, 98 sayfa

Bu çalışmada, ses öznitelik çıkarımına dayalı otomatik bir video bölütleme sistemi sunulmuştur. Video dizileri "müzik", "konuşma", "sessizlik" ve "kalabalık" gibi belirli sınıfları içeren bölümlere ayrılmıştır. Bu sınıflara ait olmayan bölümler ise "sınıflandırılamayan" olarak nitelendirimiştir. "sessizlik" sınıfının belirlenmesi için kısa-zamanlı enerji özniteliği kullanılarak, eşik değer karşılaştırma uygulaması yapılmıştır. "müzik", "konuşma" ve "kalabalık" sınıflarınının belirlenmesi için ise çoklu-sınıf ayırma yöntemi kullanılmıştır. Bu amaçla, ses bilgisine ait bazı karakteriksel öznitelikler çıkarılmıştır. Çıkarılan öznitelikler üç gruptan oluşmaktadır: MPEG-7 işlevsel veri çatısında tanımlı olan alt seviyeli ses öznitelikleri, [31]' de kullanılan öznitelikler ve bu iki öznitelik grubunun bileşimi olan öznitelikler. Sınıflandırma için en uygun özniteliği seçmek için, özniteliklerin dağılımları incelenmiştir. Ses sınıflandırma işlemi her üç öznitelik grubu için öğrenme- sınıflandırma ve test aşamalarına tabi tutulmuştur. Test ve değerlendirme sonuçları, üçüncü gruptaki özniteliklerin en iyi sonucu verdiğini göstermektedir. Sistemin çıktısı, sınıflandırılmış video parçalarına ait ses kısımlarını içeren MPEG-7 tanımlayıcılarının olduğu bir XML dosyasıdır.

Çalışmanın son kısmında, elde edilen ses sınıflandırma sisteminin, görsel işlemlerle birleştirilerek işitsel-görsel video kısımlarının elde edildiği bir uygulama senaryosu verilmiştir.

Anahtar Kelimeler: Video Bölütleme, Ses Bölütleme, MPEG-7 Görsel-İşitsel bölütleme, Video İndeksleme.

To My Family

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my supervisor Prof. Dr. Gözde Bozdağı Akar for her understanding, patience and supervision throughout this thesis.

I would also like to thank to the committee members Assoc. Prof. Dr. A. Aydın Alatan, Prof. Dr. Engin Tuncer, Assoc. Prof. Dr. Tolga Çiloğlu and Prof. Dr. Adnan Yazıcı for reading and commenting on this thesis.

I would also like to express my appreciation to ASELSAN Inc. for providing tools and other facilities throughout this study.

Special thanks to all my friends and colleagues for their understanding and continuous support during my thesis.

It is a great pleasure to express my gratitude to Hamza Ergezer for his valuable comments, suggestions, understanding and patience during my thesis.

Finally, I would like to thank my family especially to my brother İsmail Cem Atar, for their love, trust, understanding and every kind of support not only throughout my thesis but also throughout my life.

This work has been partly supported by TÜBİTAK (Scientific and Technological Research Council of Turkey) under contract – EEEAG-106E012.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ASB	Audio Spectrum Basis
ASE	Audio Spectrum Envelope
ASF	Audio Spectrum Flatness
ASP	Audio Spectrum Projection
AV	Audio Visual
BN	Bayesian Networks
CD	Compact Disc
CPD	Conditional Probability Distributions
D	Descriptor
DCT	Discrete Cosine Transform
DDL	Description Definition Language
DFT	Discrete Fourier Transform
DTD	Document Type Definition
DS	Description Scheme
DVD	Digital Versatile Disc
FFT	Fast Fourier Transform
GM-VQ	Gaussian Model-Vector Quantization
HMM	Hidden Markov Model
KNN	K-Nearest Neighbor
LLD	Low-Level Descriptor
LSP	Line Spectrum Pair
MFCC	Mel Frequency Cepstral Coefficients
MPEG	Moving Picture Experts Group
OVA	One versus All
IBM	International Business Machines
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization

RGB	Red Green Blue
XML	Extensible Markup Language
ZCR	Zero Crossing Rate

CHAPTER 1

1 INTRODUCTION

Recently, there is a rapid increase in the amount of digital video in multimedia applications due to improvement in the technology. Digital videos are widely used in the areas such as TV domains, geographic information systems, monitoring systems, education domains, mobile phones. For these types of applications large video databases are created and stored. The ability to browse the stored video data or to retrieve the content of interest is becomes a fundamental requirement of any video archiving system. For example, a large amount of audiovisual material has been archived in television and film databases. If these data can be properly segmented and indexed, it will be easier to retrieve the desired video segments for the editing of a video clip. As the volume of the database becomes huge, manual segmentation and indexing became very hard to apply. Automatic segmentation and indexing through computer will be very useful. Thus, segmentation of a video into its constituent short pieces is fundamental functionality for video retrieval and management tasks.

The purpose of multimedia indexing is to accelerate the access to large multimedia data bases. In Figure 1, representation of a general multimedia indexing and retrieval system is given. The main components of systems are: *a feature extraction/ segmentation module* to extract the features and segmentation of incoming data and, a *multimedia indexing system* to form the extracted features and segments to a known structure for later use, *a storage module* to store the features of the incoming data and a *search engine* for browsing, searching and filtering process between the database and user. However if each organization use a different mechanism for the multimedia indexing, it will be difficult to access the data from out of the organization. So, a standardization process is required for automatic indexing of multimedia data. MPEG-7 helps in describing multimedia content in such a way that

multimedia information can be easily archived, accessed, located, navigated, searched and managed [7]. A detailed explanation of MPEG-7 standard is given in chapter 3.

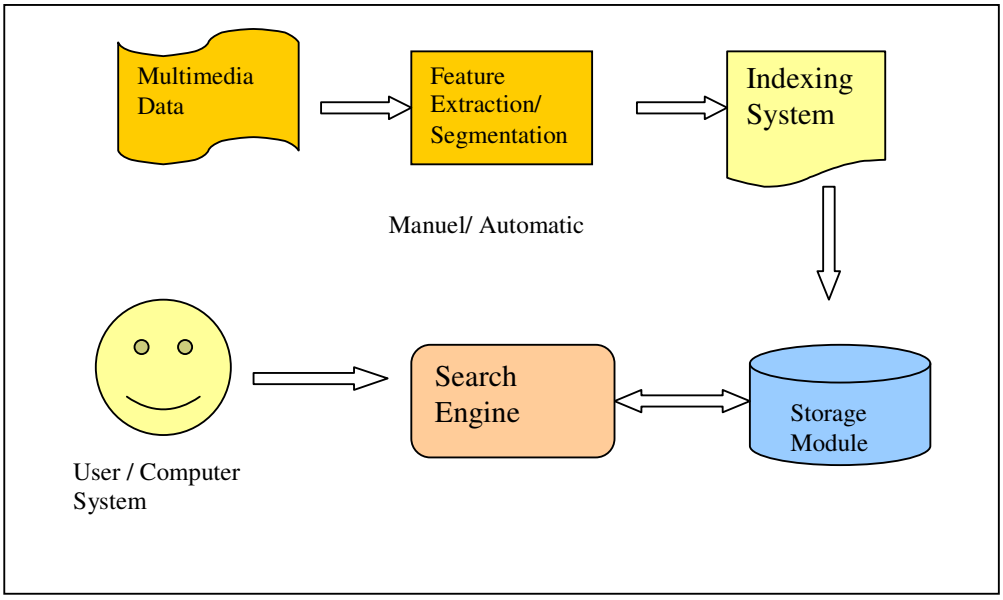

Figure 1. A multimedia indexing and retrieval system

The higher-level temporal structure is the predominant feature in the video sequences. Video is often described as having a four layer hierarchical structure, as shown in Figure 2. A shot can be considered as the basic unit of a video sequence. It can be defined as a sequence of frames captured by one camera in a single continuous action in time and space [21]. The shots are separated by shot boundaries. A scene is a semantic unit that formed by one or more adjoining shots where the shots each have similar content but are taken from different camera positions. Scene changes therefore demarcate changes in semantic context. Segmenting a video into its constituent scenes permits it to be accessed in terms of meaningful units. Temporal segmentation is the process of decomposing video streams into meaningful segments such as shots, scenes.

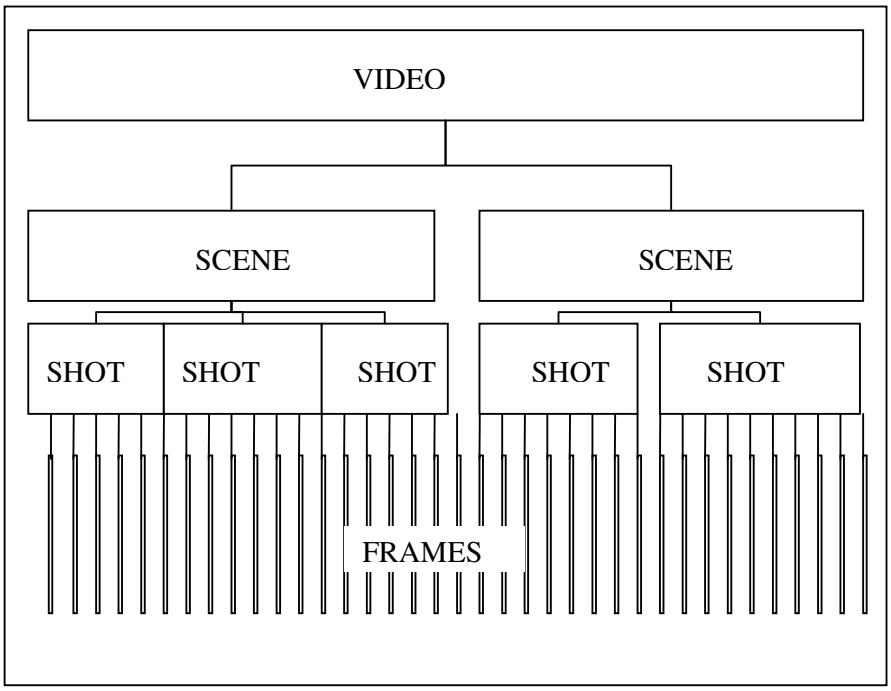

Figure 2. A Video Structure

For the temporal segmentation, algorithms are mostly focused on visual processing techniques such as color histogram differences, motion vectors, shape detection methods [12][19][21][22]. The researches on the audio information shows that audio can also be used for video segmentation [24] [25] [26] [27].

In the multimedia indexing and retrieval a lot of system has been developed. However, no system or technology has yet become widely pervasive. By the improvement in the audio-visual analysis techniques, better results are getting place. Most of the developed multimedia indexing and retrieval systems used visual features for indexing. Recently, the audio features are used in combination with the visual features in multimedia indexing and retrieval systems.

Most of the developed multimedia indexing and retrieval systems have used temporal video segmentation based on visual information and audio-visual information for the indexing purpose. Some of them are *VideoSTAR [3] , MUVIS* [4], *BilVMS* [1], *BilVideo* [2], *IMKA [4],[10].* A detailed survey on the on multimedia indexing and retrieving systems can be found in [47]

1.1 Scope of the Thesis

By being aware of the audio information importance for the video segmentation process, we propose a method for segmentation of a video by using only audio features in this thesis. The initial problem is segmentation of the audio into acoustically similar regions such as “silence”, “speech”, “music”, and “crowd” .For the silence segment detection a simple threshold comparison method on the short time energy of the signal is proposed. In the process for segmentation “speech”, “music”, “crowd”, “unclassified” regions, a pattern classification scheme has been adopted [31]. This pattern classification scheme is composed of two sections: a feature extraction and selection stage, training and a classification stage. For the feature selection and training process three audio classes that are manually labeled as “speech”, “music” and “crowd” have been formed. A series of audio features such as MPEG-7 low-level audio features, MFCC, Chroma, Spectral Roll-off, Zero Crossing Rate, Spectrogram features, have been extracted. A histogram comparison method [31] has been done on these feature sets for choosing the best feature for the segmentation process. Then, three feature sets is chosen: one of them is purely formed from MPEG-7 low-level audio features, other is the combination of some MPEG-7 low-level audio features with the features that are not belong to MPEG-7 audio framework and the last one is the audio features that are used in [31]. After feature selection process, a training and classification method based on One-versus-All classification scheme [17] in combination with the Bayesian Networks [40] and KNN classifier has been used [31]. Training and Classification stages are applied for both cases of feature sets. The feature set that gives better results for the audio

classification is used as the final feature set. The audio segmentation results are written in a XML file as the MPEG-7 audio segment descriptors.

An application scenario of this audio classification system is given as the combination of audio analysis with the visual segmentation results. For the visual segmentation, an automatic shot detection process is done by using IBM MPEG-7 Annotation Tool: *VideoAnnEx* [43] . The output of the IBM MPEG-7 Annotation Tool is an XML file which includes the shot boundary information of the video sequence with MPEG-7 metadata.

After getting the output of the Audio Segmentation and Video Shot Detection process, the results of two systems are combined to get more meaningful video segments. For the combination of the output of these systems two cases are considered: Audio Priority and Video Priority. The output of the audio-visual segmentation application is an XML file that contains MPEG-7 audio-visual segment descriptors for the annotated video sequence. The scheme of the implemented system for the application scenario is given in Figure 3

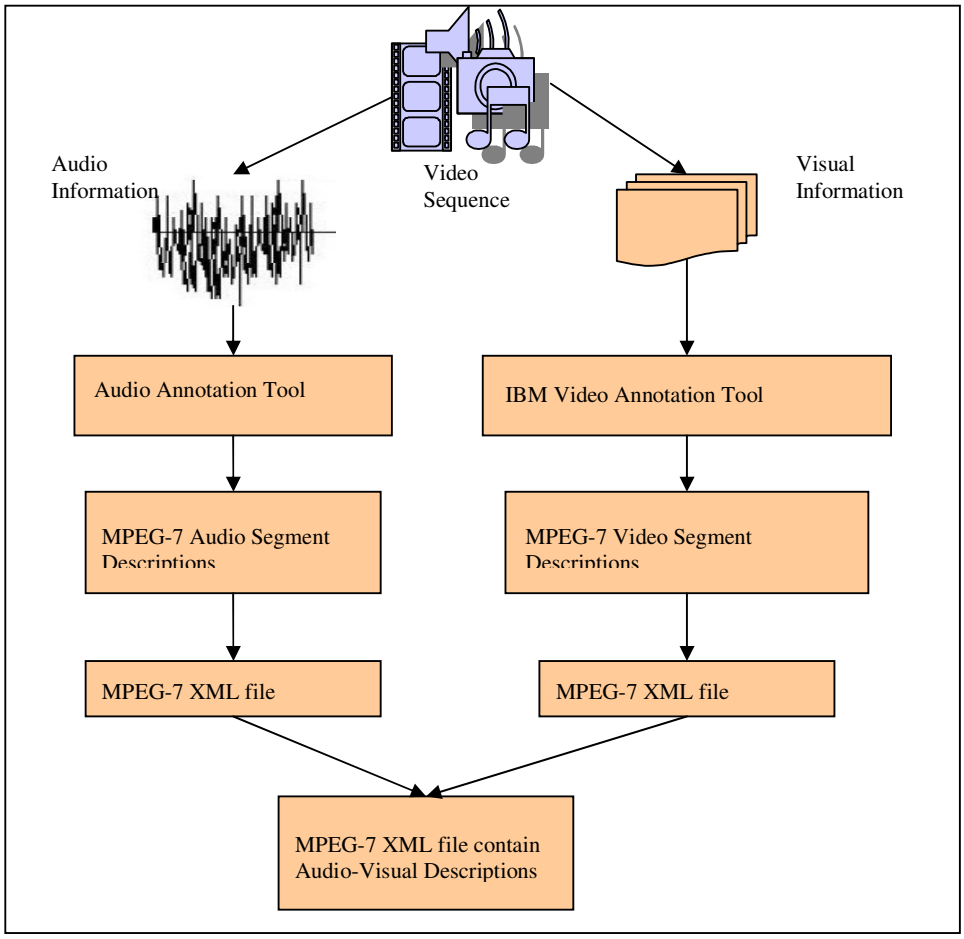

Figure 3. The Scheme of the video segmentation prototype based on Audio-Visual Processing

1.2 Outline of the Thesis

In Chapter 2, related works and broad overview for the audio segmentation, video segmentation and audio visual segmentation applications are presented.

In Chapter 3 MPEG-7, the Multimedia Content Description Interface is described. The Audio Framework of the MPEG-7 standard is summarized.

In Audio Analysis part, Chapter 4, the audio features that considered in this study are presented. After explaining the audio analysis, in Chapter 5 Audio Classification parts are described. The method for audio feature selection, implemented statistical classification techniques and the results of the audio segmentation system are described.

After audio analysis and audio segmentation parts, an application scenario for the combining audio segmentation results with a visual analysis is presented in Chapter 6. For the visual analysis IBM Annotation Tool is used for automatic shot detection process.

In Chapter 7 the conclusion and future works are presented.

CHAPTER 2

2 RELATED WORKS ON VIDEO SEGMENTATION

In this chapter the latest studies in the literature related with video segmentation applications based on visual processing algorithms, audio processing algorithms and combination of audio-visual processing algorithms are described.

2.1 Visual Processing Methods in Video Segmentation

As we mentioned before, a video sequence temporally segmented in to scenes, shots and frames. These temporally segments are used for multimedia indexing purpose. In the following paragraphs the studies for the shot detection and scene detection algorithms depend on only visual processing are presented.

2.1.1 Shot Detection

As we mention before, a shot is a group of frames that are taken from one camera in a single action. The frames of the shots are related each other, if a discontinuity occurs between the frames, a shot boundary is detected. There are many algorithms that perform the shot boundary detection. They are based on certain features extracted from video frames. These features can be different perspectives of the video content, such as color, objects, and motion, or be video stream attributes such as compression information. The basic shot detection algorithms in the literature can be listed as:

Pixel Differences: A frame of a digital video consists of a grid of pixels all with their own color, and the sequence of these colors determines how the given frame will appear [19]. The idea behind this is that gradual changes between corresponding pixels do not get counted as changing, so that only when a shot boundary is encountered if a substantial number of changes occurs.

Color Histogram: The main idea of this algorithm is that two frames having an unchanging background and unchanging objects which means they are in the same shot will show little difference in their respective histograms. A shot boundary is assumed, if the bin-wise difference between the two histograms of the two frames is above a threshold [12].

Motion Detection: Motion is another characteristic of video that can be measured, although this can be useful for removing potential shot boundary cases rather than identifying them. This is because when panning and zooming are occurred most other video features change, making it difficult to distinguish whether a shot boundary has occurred.

Edge Detection*:* A video can be thought of as being composed of a series of objects, each of these objects having edges. This edge information can be captured for a sequence of frames, and used to determine whether edges remain in the shot from one frame to the next, or whether they disappear and are replaced by different edges.

Zhang et al. [19] compared pixel differences, statistical differences and several different histogram methods and found that the histogram methods were a good trade-off between accuracy and speed. In order to reduce the camera motion and noise effects, they applied additional step by using a 3x3 averaging filter before the comparison pixel comparison. Each pixel in a frame is replaced with the mean value of its nearest neighbors. They used a threshold that changes with the input sequence and good results were obtained, although the method was somewhat slow.

Liu et al. [20] used the motion vectors extracted as part of the region-based pixel difference computation to decide whether there is a large amount of camera or object motion in a shot.

Bhattacharjee et al. [21] used running histogram methods to detect the shot boundaries in the raw video streams and macro blocks for shot detection in MPEG-2 streams. In the raw video streams, they detected both straight cuts and dissolves of two frames at a time, by comparing their histograms. In the MPEG-2 streams each frame is splitted into blocks called macro blocks, each of which is coded in one of three modes: *intra mode* where each block is represented by its DCT coefficient; *predicted mode* where each block is represented by a motion vector and a set of DCT coefficient for the prediction-error and *block repetition mode* where each block is just copied from the same position in the previous frame [21]. The type of a block used gives information about how much new image data is introduced in a frame and how many blocks are predicted from another frame, both of which are useful in determining shot boundaries.

In [18], *Mai et al*. compared color histograms, chromatic scaling and their own algorithm based on edge detection. They aligned consecutive frames to reduce the effects of camera motion and compared the number and position of edges in the edge detected images. The percentage of edges that enter and exit between the two frames was computed. Shot boundaries were detected by looking for large edge change percentages. Dissolves and fades were identified by looking at the relative values of the entering and exiting edge percentages. They determined that their method was more accurate at detecting cuts than histograms and much less sensitive to motion than chromatic scaling.

Yuan et al [45] conducts a formal study of the shot detection problem in pattern recognition manner. They examined the three techniques in the perspective of pattern recognition: *the representation of visual content*, *the construction of continuity signal* and *the classification of continuity values.*

2.1.2 Scene Detection

A scene is a part of video that is formed by shots that are semantically related. Scene detection is a harder process than shot detection because it's semantic relation.

A method for detection of a specific type of scenes in which two people are talking to each other is proposed by *Forlines* [22]. In this work, scene detection is not maintained for indexing or summarization; they maintain scene detection technique for playback the video in TV. They applied a two step approach for scene detection. In the first step, they find chains of related shots within the video. In the second step, they combine these chains into scenes. For comparing the similarities in the shots and making chains they again used color histograms as they used for shot detection, only using a more relaxed threshold. They compare the first frame in a current shot with the last five frames of each of the previous five shots If a shot begins with a frame that is visually similar to the last five frames of a previous shot, then the shots are likely to be of the same person or object. A chain of shots is created whenever two or more shots are found to be visually similar.

In the second step they combine the shots that have grouped into chains in step one by using time relation between them. But not all the shots are included in the chains. These shots called orphans. An orphan is appended in the scene that covers the orphan.

Yeung et al [24] introduced not only a pioneering piece of scene segmentation work, but also a means to visualize a video's structure. They propose that visual similarity of shots alone may not be sufficient to differentiate the context and contents of individual shots as each shot is itself a distinct unit of time in the featured video presentation.

A general framework of clustering of video shots based on both visual similarities and temporal localities of the shots which it is called time-constrained clustering is proposed.

2.2 Audio Based Video Segmentation

The audio information within a video sequence gives information for partition of video in to meaningful segments. In this section the video segmentation algorithms that use only the audio information are presented.

J. Son, et al.[25], used speech recognition for segmenting a video sequence. Since the audio signal in the sound track is synchronized with image sequences in the video program, a speech signal in the sound track can be used to segment video into meaningful segments. This method requires a good speech recognition performance. To avoid difficulties in the recognition problem, they propose using speech recognition with closed caption. Closed caption is spoken portion of television show or video program that appears as text at bottom of screen when processed by a decoder installed in set.

A multi-class classification algorithm proposed by *Giannakopoulos et al.* [31] for audio segments recorded from movies, focusing on the detection of violent content. In order to classify the audio segments into six classes (three of them violent), Bayesian Networks have been used in combination with the One Versus All classification architecture. In this study I have used nearly same classification scheme but using different audio features and getting different classes. They do not combine the audio segmentation results with any visual processing technique for getting a video segmentation process as I do.

2.3 Audio-Visual Video Segmentation

By combining the audio information and visual information of a video sequence better results can be obtained for getting meaningful segments of video sequence.

Boreczky et al. [26], combined audio information with visual information for detecting the shot boundaries. For the visual features the histogram and motion vectors are used. In the audio processing parts they do not classify the audio in the parts (speech, silence etc.) or speech recognition. For reflecting the difference in audio types such as speech, silence etc. they calculate an audio distance measure. The audio distance measure is similar to the likelihood ratio measure. They compute audio distances based on sliding two-second intervals. A segmentation technique allows features to be combined within the HMM framework is used instead of a standard threshold segmentation types.

A specific type of scene detection method, by combining audio-visual analysis of a video sequence is presented by *Alatan* in [14]. The dialogue scene detection method in a TV sitcom or film is proposed by using the state transitions of a Hidden Markov Model (HMM). A face recognition process is done for detection a video segment that includes or not includes a human face for the visual analysis part. In the audio analysis part, audio information segmented acoustically similar regions as "speech", "music", "silence". In the audio analysis part signal energy, periodicity and zero crossing rate features are used. By combining the face detection results with the audio segmentation results a dialogue scene detection is proposed.

A video segmentation scheme is proposed by *Jiang et al* [27], in which audio and color information is integrated in video scene extraction. An audio segmentation scheme is developed to segment audio tracks into speech, music, environmental sound and silence segments. In the audio classification part firstly a classification is performed for speech, non-speech discrimination. KNN (K-Nearest Neighbor) classifier based on zero crossing rate and short time energy contour is used as a pre-classifier for speech and non-speech discrimination. Then, a GM-VQ (Gaussian Model-Vector Quantization) method based on line spectrum pair (LSP) divergence shape analysis is used to refine the classification result and make the final decision. Second, non-speech segments are further classified into music and environment sound based on the audio periodicity and other features. A two stage combination of visual and audio process has been adopted. At the first stage, shots of a video sequence are clustered based on audio analysis. Audio breaks are first detected in one-second interval. When a shot break and an audio break are detected simultaneously within the one-second interval, the boundary of the sequence of shots is marked as a potential scene boundary.

In [28], *Pye et al*. propose a method that combines audio segmentation with visual segmentation, for segmentation of digital audio/video recordings Firstly, an audio segmentation algorithm that partitions a soundtrack into manageably sized segments for speech recognition is proposed. For the audio segmentation part they use Gaussian Mixture Model for training and classifying the acoustically homogenous

segments such as speech, male speech, music and noise. For detecting the speaker/channel change or music onset, they estimate the local changes in acoustic characteristic frame by frame through the soundtrack after the Gaussian Mixture Model. Secondly, they present an algorithm for detecting camera shot-break locations in the video. The output of these two algorithms is combined to produce a semantically meaningful segmentation of audio/video content, appropriate for information retrieval. They report the success of the algorithms in the context of television news retrieval.

Lu, et al. [30] present a scene detection technique that measures the continuity of visual, aural, and textual (closed captioning) elements in a video, and labels a shot boundary as a scene boundary when these continuities drop.

CHAPTER 3

3 MPEG-7 MULTIMEDIA CONTENT DESCRIPTION INTERFACE

As previously mentioned, there is a large collection of digital video in multimedia applications and it is getting larger day by day due to improvement in the technology storage devices and compression techniques. Since all of these video databases come from different sources, a system to manage it gets importance. In this chapter, an international standard, named as MPEG-7 will be described. MPEG-7 parts are constructed by means of MPEG-7 Description Definition Language. This language uses a schema based on Extensible Markup Language (XML). So the XML, its importance and its parts will also be presented in this chapter.

3.1 Introduction to MPEG-7

MPEG-7 is an ISO/IEC standard developed by the Moving Picture Coding Experts Group (MPEG). The MPEG is in charge of the development of international standards for compression, decompression, processing, and coded representation of moving pictures, audio, and a combination of the two [11]. MPEG-1, MPEG-2 and MPEG-4 are also developed by MPEG group in order to standard the video and audio compression. MPEG-1 is developed for the storage and retrieval of moving pictures and audio on storage media such as mp3 and video CD players, DVD players, recorders. MPEG-2 standardize the digital TV sets, it's the timely response for the satellite broadcasting and cable television industries in their transition from analog to digital formats [11]. The MPEG-4 standard uses advanced compression algorithm. It codes content as objects and enables those objects to be manipulated individually or collectively on an audiovisual scene [11][7].

The MPEG-7 standard aims to satisfy the need by standardizing a framework for compact, efficient, and interoperable descriptions of audio-visual content [11]. It is a standard for describing the features of multimedia content in order to facilitate various multimedia searching and filtering applications [8]. But the scope of MPEG-7 does not include the automatic extraction of audio-visual descriptions/features or specifying the search engine that can make use of the description, it only standardizes the descriptions of the features.

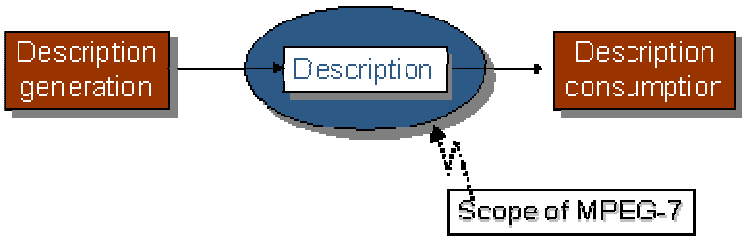

Figure 4. The Scope of MPEG-7 ([11])

3.2 Technical Overview of MPEG-7

MPEG-7 does not standardize the extraction of audio-visual features; it defines a multimedia library of methods and tools by standardizing the descriptions of the extracted audio-visual features.

Audiovisual data content that has MPEG-7 descriptions associated with it may include: still pictures, graphics, 3D models, audio, speech, video, and composition information about how these elements are combined in a multimedia presentation [11].

For this purpose the main standardization effort is divided in to four step: descriptors (Ds), description schemes (DSs), description definition language (DDL), and coded descriptions.

The Descriptors (Ds) is a representation of a feature that defines the syntax and semantics of the feature representation [11]. For example for a motion feature in a video, Camera Motion is a descriptor.

The Description schemes (DSs) specifies the structure and semantics of the relationships between its components, which may be both descriptors and description schemes [11]. For example Audio-Visual Segment is a DS which gives information about audio and visual information for the specified segment.

The Description Definition Language (DDL) is a language that specifies syntactic rules to express and combine Description Schemes and Descriptors description schemes [11].

Coded Description is a description that's been encoded to fulfill relevant requirements such as compression efficiency, error resilience, and random access. MPEG-7 Coded Description may be textual or binary, as there might be cases where a binary efficient representation of the description is not needed, and a textual representation would suffice [11].

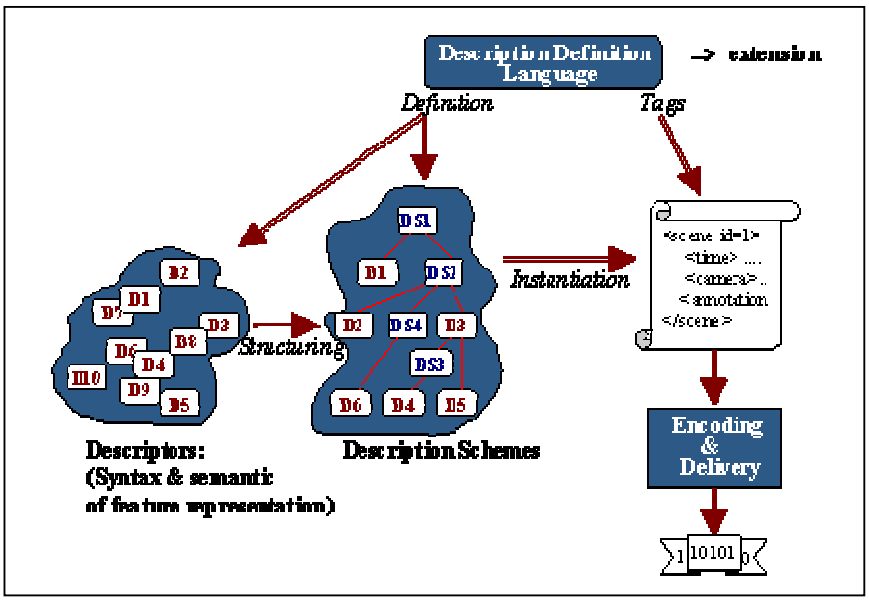

Figure 5. MPEG-7 main elements ([11])

In Figure 5 , relationships between different elements are shown.

The MPEG-7 Standard consists of the following parts [11] :

- *MPEG-7 Systems:* The tools needed for transmission and encoding of MPEG-7 descriptions
- *MPEG-7 Description Definition Language:* The language for defining the syntax of the MPEG-7 Description Tools and for defining new Description Schemes based on XML-Schema
- *MPEG-7 Visual*: The Description Tools dealing with (only) Visual descriptions.
- *MPEG-7 Audio:* The Description Tools dealing with (only) Audio descriptions.
- *MPEG-7 Multimedia Description Schemes:* The Description Tools dealing with generic features and multimedia descriptions.
- *MPEG-7 Reference Software:* A software implementation of relevant parts of the MPEG-7 Standard with normative status.
- *MPEG-7 Conformance Testing*: Guidelines and procedures for testing conformance of MPEG-7 implementations
- *MPEG-7 Extraction and use of descriptions*: Informative material (in the form of a Technical Report) about the extraction and use of some of the Description Tools.
- *MPEG-7 Profiles and levels:* Provides guidelines and standard profiles.
- *MPEG-7 Schema Definition*: Specifies the schema using the Description Definition Language

In this study, we mostly deal with MPEG-7 Audio descriptors, MPEG-7 Description Definition Language and MPEG-7 Multimedia Description Schemes (for modeling the Audio-Visual Segments). The dealt parts of MPEG-7 with in this study are described in detail in the next subsections.

3.3 Description Definition Language

The Description Definition Language (DDL) allows defining and extending descriptors and description schemes [11]. It also allows for the extension and modification of existing description schemes. MPEG-7 adopted XML (Extensible Markup Language) Schema Language as the MPEG-7 DDL. However because XML Schema language has been designed to use on the Web, not for audiovisual content, certain extensions have been necessary in order to satisfy all of the MPEG-7 DDL requirements.

3.3.1 XML Schema Overview

XML is a mark-up language for documents containing structured information [37]. XML is similar to HTML, but it contains user definable tags. XML Schemas include elements and their content, attributes and their values, cardinalities and data types. The definition of valid document structure is expressed in grammar known as DTD (Document Type Definition) [7]. XML Schemas provide a superset of the capabilities of DTDs.

XML documents are completely text-based. XML documents give users the ability of expressing very complex structural or hierarchical information in a single text based document.

3.3.2 MPEG-7 Audio Descriptors

MPEG-7 Audio provides structures for describing audio content of the multimedia data. An audio descriptor is an audio feature. The MPEG-7 audio framework includes a set of low-level Descriptors, for audio features that cut across many applications (e.g., spectral, parametric, and temporal features of a signal), and high-level Description Tools that are more specific to a set of applications [11].

In this study some low-level audio features of the MPEG-7 Audio Framework is used by the combination of the audio features that are not belong to the MPEG-7 Audio Framework such as MFCC are used. Since the high-level audio descriptors are not used, a detailed explanation of MPEG-7 low-level audio descriptors is given in the following sections whereas there is a rough explanation of the MPEG-7 high level descriptors. A detailed information about the high-level audio descriptors can be found in [11].

3.3.2.1 Low Level Audio Descriptors

There are seventeen temporal and spectral descriptors that may be used in a variety of applications in MPEG-7 Audio Framework. These descriptors are the low-level audio descriptors; they can be roughly divided into the six groups [11] :

- *Basic*
- *Basic Spectra l*
- *Signal Parameters*
- *Timbral Temporal*
- *Timbral Spectral*
- *Spectral Basis*

The Low-Level Audio Descriptors of MPEG-7 Audio Framework are shown in Figure 6.

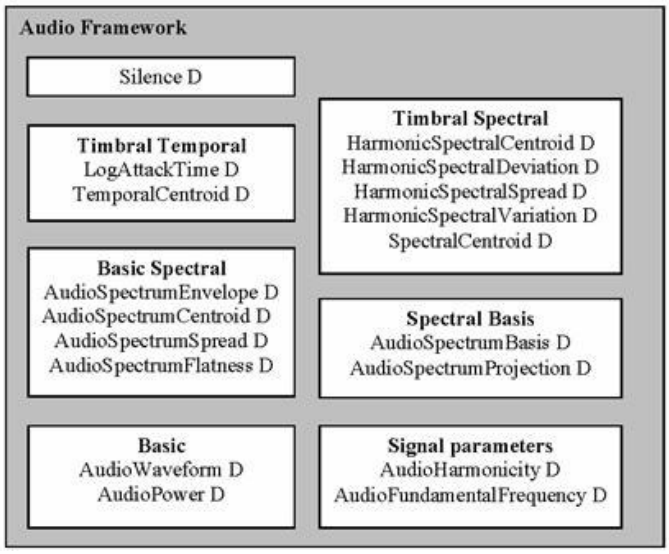

Figure 6.Overview of Audio Framework including low-level Descriptors ([11])

Basic : There are two descriptors in this class: *AudioWaveform Descriptor* and *AudioPower Descriptor*. The *AudioWaveform Descriptor* describes the maximum and minimum audio waveform envelope typically for display purposes. The *AudioPower Descriptor* describes the temporally-smoothed instantaneous power.

Basic Spectral: There are four descriptors in this group. They are derived from a single time-frequency analysis of an audio signal. *AudioSpectrumEnvelope Descriptor* which takes place in this group is a vector that describes the short-term power spectrum of an audio signal.

The *AudioSpectrumCentroid Descriptor* describes the center of gravity of the log-frequency power spectrum. This Descriptor is a description of the shape of the power spectrum, indicating whether the spectral content of a signal is dominated by high or low frequencies. The *AudioSpectrumSpread Descriptor* complements the previous Descriptor by describing the second moment of the log-frequency power spectrum, indicating whether the power spectrum is centered near the spectral centroid, or spread out over the spectrum. This may help distinguish between pure-tone and noise-like sounds.

The *AudioSpectrumFlatness Descriptor* describes the flatness properties of the spectrum of an audio signal for each of a number of frequency bands. When this vector indicates a high deviation from a flat spectral shape for a given band, it may signal the presence of tonal components.

Signal Parameters: This group contains two descriptors: *AudioHarmonicity Descriptor* and *AudioFundamentalFrequency Descriptor*. The *AudioHarmonicity Descriptor* represents the harmonicity of a signal, allowing distinction between sounds with a harmonic spectrum, sounds with an inharmonic spectrum and sounds with a non-harmonic spectrum. *AudioFundamentalFrequency descriptor* describes the fundamental frequency of an audio signal.

Spectral Basis: The Spectral Basis Descriptor consist of two Descriptors represent low-dimensional projections of a high-dimensional spectral space to aid compactness and recognition. The *AudioSpectrumBasis Descriptor* is a series of basis functions that are derived from the singular value decomposition of a normalized power spectrum. The *AudioSpectrumProjection Descriptor* is used together with the *AudioSpectrumBasis Descriptor*, and represents low-dimensional features of a spectrum after projection upon a reduced rank basis.

Together, the descriptors may be used to view and to represent compactly the independent subspaces of a spectrogram.

Timbral Temporal: Timbral Temporal Descriptors describe temporal characteristics of segments of sounds, and are especially useful for the description of musical timbre. There are two Timbral Temporal Descriptors: The *LogAttackTime Descriptor* and The *TemporalCentroid Descriptor*. *The LogAttackTime Descriptor* characterizes the "attack" of a sound, the time it takes for the signal to rise from silence to the maximum amplitude. This feature signifies the difference between a sudden and a smooth sound. *The TemporalCentroid Descriptor* also characterizes the signal envelope, representing where in time the energy of a signal is focused.

Timbral Spectral: The five Timbral Spectral Descriptors are spectral features in a linear-frequency space especially applicable to the perception of musical timbre. The

SpectralCentroid Descriptor is the power-weighted average of the frequency of the bins in the linear power spectrum.

The four remaining timbral spectral Descriptors operate on the harmonic regularly-spaced components of signals. For this reason, the descriptors are computed in linear-frequency space. The *HarmonicSpectralCentroid Descriptor* is the amplitude-weighted mean of the harmonic peaks of the spectrum. It has a similar semantic to the other centroid Descriptors, but applies only to the harmonic (non-noise) parts of the musical tone. The *HarmonicSpectralDeviation Descriptor* indicates the spectral deviation of log-amplitude components from a global spectral envelope. The *HarmonicSpectralSpread Descriptor* describes the amplitude-weighted standard deviation of the harmonic peaks of the spectrum, normalized by the instantaneous *HarmonicSpectralCentroid*. The *HarmonicSpectralVariation Descriptor* is the normalized correlation between the amplitude of the harmonic peaks between two subsequent time-slices of the signal.

Silence Segment : The silence segment simply attaches the simple semantic of "silence" to an Audio Segment. Although it is extremely simple, it is a very effective descriptor. It may be used to aid further segmentation of the audio stream.

In this study following low level audio descriptors are used:

- Basic Spectral: *AudioSpectrumEnvelope* Descriptor, *AudioSpectrumFlatnessD* Descriptor and *AudioSpectrumSpread* Descriptor
- Signal Parameters: *AudioFundamentalFrequency* Descriptor
- Spectral Basis: *AudioSpectrumBasis* Descriptor and *AudioSpectrumProjection* Descriptor
- *AudioSpectrumBasis Descriptor*, *AudioFundamentalFrequeny Descriptor,* The detailed explanation of MPEG-7 low-level audio features (descriptors) is given in Chapter 4.

A detailed calculation process for these features is given in Chapter 4: Audio Analysis.

3.3.2.2 High-Level Audio Description Tools

MPEG-7 audio framework includes smaller set of audio features as compared to visual features. MPEG-7 Audio includes a set of specialized high-level tools that exchange some degree of generality for descriptive richness for this reason. The five sets of audio Description Tools that roughly correspond to application areas are integrated in the standard: *audio signature*, *musical instrument timbre, melody description*, *general sound recognition and indexing*, and *spoken content*. These high-level descriptors are obtained from the low-level audio descriptors. A detailed information about the high-level audio descriptors can be found in [11].

3.3.3 Audio-Visual Segment Modeling by Using MPEG-7 Segment Description Schemes

In this study a video segmentation process has been proposed. The output of the system is an XML file includes information about the audio visual segment descriptions.

The MPEG-7 segment entity tools describe spatiotemporal segments of generic types of multimedia content such as images, videos, audio sequences and audiovisual sequences [9].

The XML file, MPEG-7 description definition file begins with a root element that signifies whether the description is complete or partial. A complete description provides a complete, standalone description of AV (audio-visual) content for an application. On the other hand, a description unit carries only partial or incremental information that possibly adds to an existing description. In the case of a complete description, an MPEG-7 top-level element follows the root element.

The following segment entity tools are the main description schemes that are used in this application: AudioVisualSegment, AudioSegment, VisualSegment.

AudioVisualSegment: This description scheme describes a temporal interval of audiovisual data, which corresponds to both the audio and video in the same temporal interval [9].

AudioSegment: It represents a temporal interval of an audio sequence, a group of samples in the digital case [9].

VideoSegment: This description scheme describes the temporal interval of a sequence of frames [9].

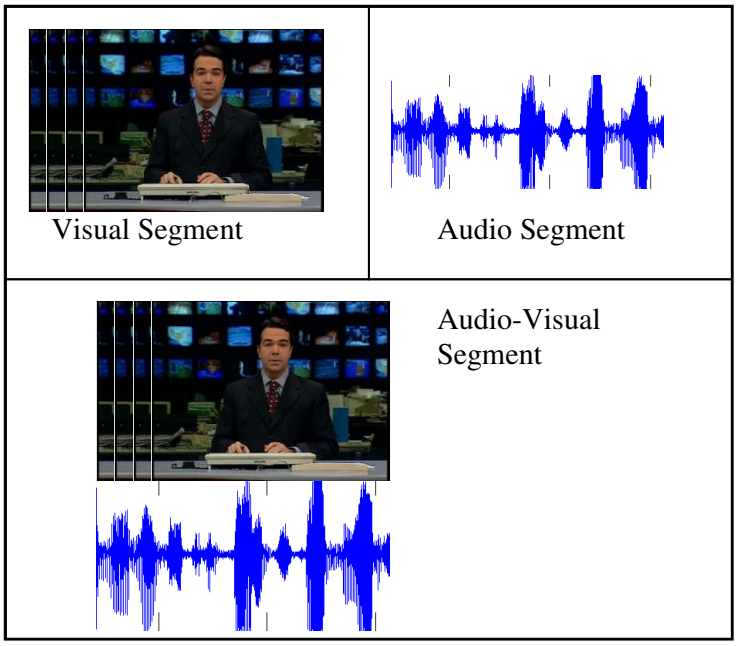

Figure 7. A visual segment, an audio segment and an audiovisual segment

MediaTime: MediaTime description scheme describes the time information in the media. It includes MediaTimePoint and MediaTimeDuration Descriptions to represent the start time of the segment and duration of the segment respectively for audio segments, video segments and audio visual segments.

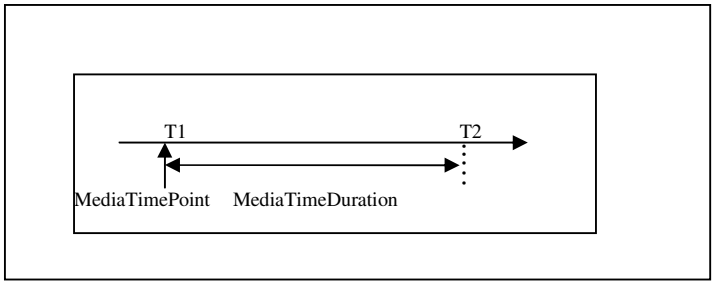

Figure 8. Overview of the MediaTime DSs (edited, [11])

TemporalDecomposition: A temporal segment may be a set of samples in an audio sequence, represented by an Audio Segment DS, a set of frames in a video sequence, represented by a Video Segment DS or a combination of both audio and visual information described by an Audio Visual Segment DS.

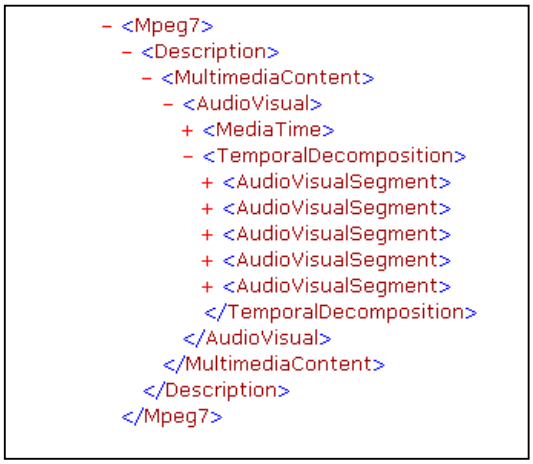

Figure 9. Usage of TemporalDecomposition and AudioVisualSegments

MediaSourceDecomposition: A description scheme used to decompose the segments to audio and video segments.

StillRegion: The StillRegion DS can describe a spatial segment or region of an image or a frame in a video [11]. StillRegion DS is decomposed to MediaTime DS in this application to represent the key frame time information in a VideoSegment DS.

TextAnnotation: MPEG-7 provides a number of different basic constructs for textual annotation. The most flexible text annotation construct is the data type for FreeTextAnnotation. FreeTextAnnotation allows the formation of an arbitrary string of text. In this application FreeTextAnnotation is used to represent the label of the classified segment of the audio segment.

Figure 10. Usage of MediaSourceDecomposition and AudioVisualSegments

CHAPTER 4

4 AUDIO ANALYSIS

A pattern classification scheme has been used for the audio classification task. As many other pattern classification tasks, audio classification is made up of two main sections: a signal processing section and a classification section. The signal processing part deals with the extraction of features from the audio signal. In this chapter signal processing part and audio features that are used in this study are explained. Audio features divided to parts: MPEG-7 low-level audio features and the audio features that are not included MPEG-7 standard.

4.1 Audio feature extraction

Feature extraction is the process of converting an audio signal into a sequence of feature vectors that carry the characteristic information about the signal. These vectors are used as basis for various types of audio analysis algorithms. It is typical for audio analysis algorithms to be based on features computed on a window basis. These window based features can be considered as short time description of the signal for that particular moment in time.

The performance of a set of features depends on the application. The design of descriptive features for a specific application is hence the main challenge in building audio classification systems. A wide range of audio features exist for classification tasks.

There are two main feature extraction methods: time-domain and frequency-domain feature extraction methods. The time-domain method takes waveform as a direct

input while the frequency domain carries out spectral transformation of the speech signal. Time analysis requires relatively little calculation but is limited to simple speech measures, whereas spectral analysis takes more effort but characterizes sounds more precisely.

4.2 Short Term Analysis of Audio Signal

The audio signals are not statistically stationary signals. Speech signals are considered to be slowly time varying signals. A speech signal over a period of say between 10 to 100 ms has a characteristic which is fairly stationary. Although music has a larger dynamic range than speech, like speech its characteristics over a short period of time remain stationary. So the first step in audio analysis is segmenting the continuous audio signal into short time frames to obtain statistically stationary signals. We must choose the frame length that each frame is not so long that significant signal variations are retained within a frame, but not so short that we lose spectral characteristics of the signal.

Usually overlapped frames structure is considered in speech analysis for reducing the loss of data because the discontinuity that formed by segmenting audio signals in to frames.

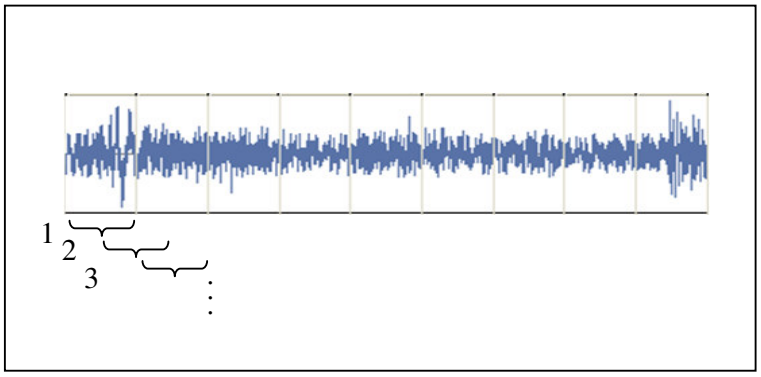

Figure 11. Overlapped Frame Structure of an Audio Signal

After segmentation process, at the edges of the frames, the signal may change very abruptly; a feature not present in the original signal can be present. A transform of such a segment reveals a curious oscillation in the spectrum, an artifact directly related to this sharp amplitude change. Better way to overcome this problem is to apply a window to the each frame of the signal. A window function is a function that is zero-valued outside of some chosen interval. In fact it is a shape the signal values within a frame so that the signal decays gracefully as it nears the edges.

There are several types of windows. It is desirable for a window function that has large bandwidth and the amplitude of the main lobe is large while the amplitude of the side lobes is small. It means that window suppresses the signals that are out of the interested bandwidth.

The bandwidth of the hamming window, as it can be seen in Figure 12, is larger than the bandwidth of a rectangular window of the same length. Moreover the hamming window results in a much higher attenuation outside the bandwidth when compared to the rectangular window. Because of the advantages of the hamming window over the rectangular window, hamming window is used in this study. However, in both cases, increasing the length of the window decreases the bandwidth.

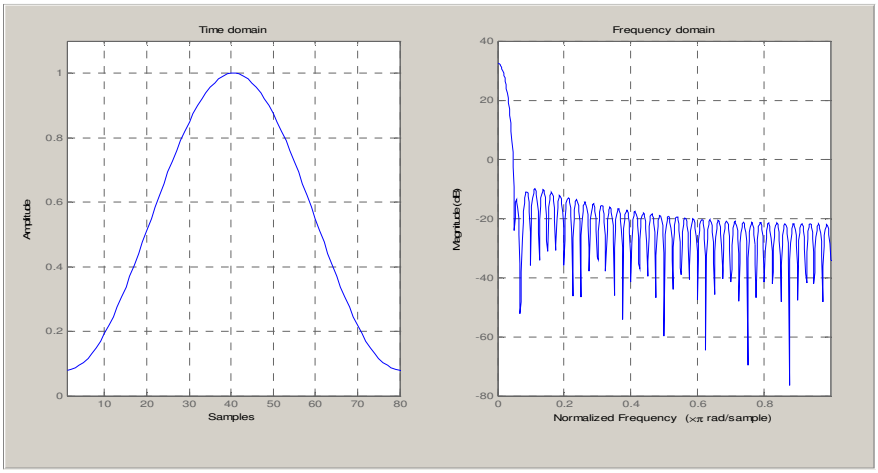

Figure 12. Characteristic of the Hamming window. Relative Side lobe attenuation :- 42.5 dB Main lobe width (3 dB)=0.0312

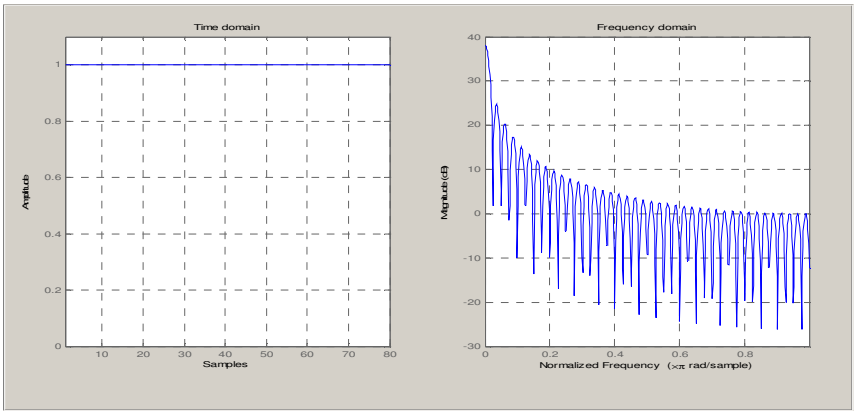

Figure 13. Characteristic of the Rectangular window. Relative Side lobe attenuation : -13.3 dB Main lobe width (3 dB)=0.0214

4.2.1 MPEG-7 Low-Level Audio Features

As we mentioned in Chapter 3, MPEG-7 Audio Framework provides some audio features to describe the content of the multimedia data. While each of these descriptors has been designed with specific purposes (power measures, spectral envelope estimation, spectral flatness measure...). In this study, AudioSpectrumEnvelope, AudioSpectrumFlatness, AudioSpectrumSpread AudioFundamentalFrequency, AudioSpectrumBasis and AudioSpectrumProjection descriptors are used. These MPEG-7 features are dimension-reduced, de-correlated spectral features.

4.2.1.1 AudioSpectrumEnvelope

Time- frequency analysis of audio signal is widely used in audio analysis applications. However the direct frequency spectrum of the signal is a high dimensional feature. *AudioSpectrumEnvelope* (ASE) descriptor represents the spectrum of the audio signal. To extract reduced-rank spectral features a log-frequency power spectrum was initially computed with a hamming window of length

20ms at 10ms intervals. Frequency channels were logarithmically spaced in ¼-octave bands spaced between 62.5Hz and 8 KHz.

4.2.1.2 AudioSpectrumFlatness

This feature describes the flatness properties of the spectrum of an audio signal for each of frequency bins. AudioSpectrumFlatness (ASF) allows the distinction between more tone-like and more noise-like signal quality. It is defined as the ratio of the geometric mean to the arithmetic mean of the power spectral density components. It is currently using for the Audio Identification task.

In MPEG-7 standard ASF is calculated on a logarithmic frequency scale also. Frequency channels are again logarithmically spaced in ¼-octave bands spaced between 62.5Hz and 8 KHz as in ASE calculation process.

After the short-term analysis of the audio signal, for each frame the power spectrum is calculated. Then the power spectrum is converted to logarithmic scale. The geometric mean and the arithmetic mean of the power spectral density are calculated. Finally; the ratio of these qualities gives the ASF feature. ASF is a series of vectors, NxM size, where N is number of frames; M is the number of frequency bands.

The calculation process of the ASF feature is given in Figure 14.

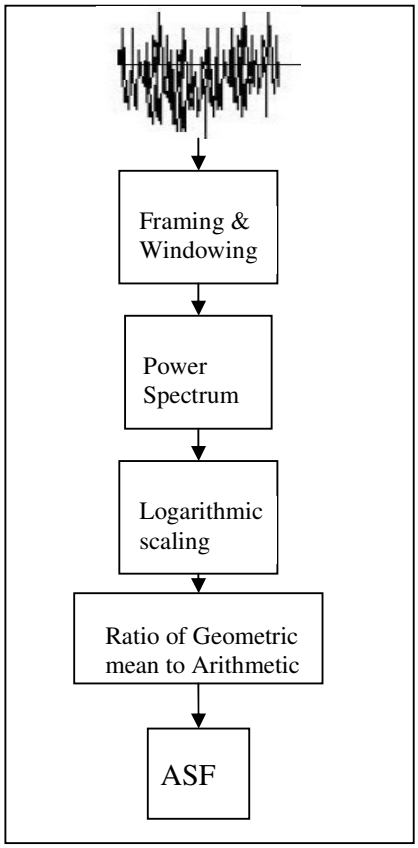

Figure 14. Calculation process of MPEG-7 ASF descriptor

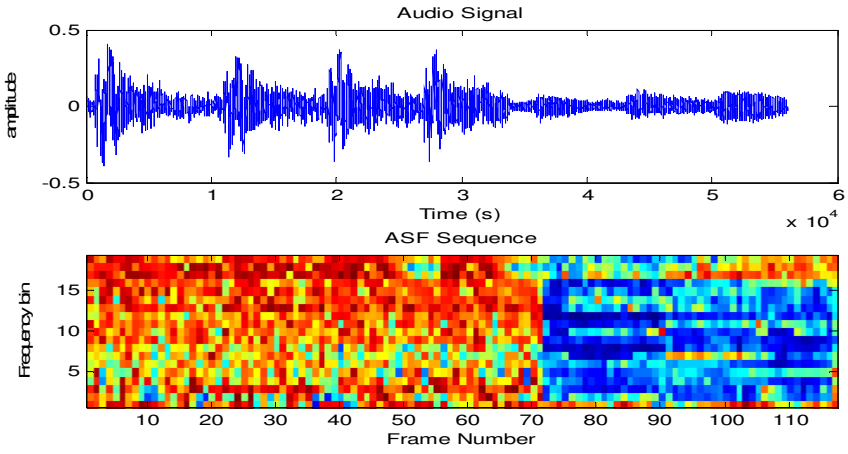

Figure 15. An audio signal and ASF plot

4.2.1.3 AudioSpectrumSpread

The AudioSpectrumSpread is a measure of the spectral shape. It is the spread of the spectrum around its mean value. It is the second moment of the log-frequency power spectrum, indicating whether the power spectrum is centered near the spectral centroid, or spread out over the spectrum.

4.2.1.4 AudioSpectrumBasis and AudioSpectrumProjection

AudioSpectrumBasis and AudioSpectrumProjection features based on the time-frequency (spectrogram) of the audio signal. These feature vectors are dimension-reduced spectral vectors represent low-dimensional projections of a high-dimensional spectral space.

The AudioSpectrumBasis Descriptor is a series of basis functions that are derived from the singular value decomposition (SVD) of a normalized power spectrum [49], [50]. The AudioSpectrumProjection Descriptor is used together with the AudioSpectrumBasis Descriptor, and represents low-dimensional features of a spectrum after projection upon a reduced rank basis.

The first step is calculating the power spectrum of the audio signal by using a logarithmic frequency scale. This is the *AudioSpectrumEnvelope* feature of the audio signal. The resulting data will be a series of vectors in MxN dimension where M is the number of frame; N is the number of frequency bin.

For each spectral vector, x, in *AudioSpectrumEnvelope*, convert the power spectrum to a decibel scale:

$$z = 10\log_{10}(x) \qquad \text{(Eqn 4. 1)}$$

compute L2-norm of the each vector element:

$$r = \sqrt{\sum_{k=1}^{N} z_k^2} \qquad \text{(Eqn 4. 2)}$$

calculate new unit-norm spectral vector :

$$\tilde{X} = \frac{Z}{r} \qquad \text{(Eqn 4. 3)}$$

The size of the resulting spectral shape matrix is M x N where M is the number of time frames and N is the number of frequency bins.

The next step was to extract a subspace using the singular value decomposition. To yield a statistically independent basis we used a reduced-rank set of SVD basis functions. The SVD is a well-known technique for reducing the dimensionality of data while retaining maximum information content [49]. The SVD decomposes the spectrogram into a sum of vector outer products with vectors representing both the basis functions and the projected features. These basis functions and projection coefficients can be combined to form spectrogram.

$$\tilde{X} = USV^T \qquad \text{(Eqn 4. 4)}$$

where $\tilde{X}$ is factored into the matrix product of three matrices; the row basis U, the diagonal singular value matrix S and the transposed column basis functions V. Reduce the basis by retaining only the first K basis functions, i.e. the first K columns of V:

$$\overline{V_k} = [v_1, v_2, ..., v_k] \qquad \text{(Eqn 4. 5)}$$

k is chosen as 20 in this study. The resulted $\overline{V_k}$ matrix is the AudioSpectrumBasis feature.

The AudioSpectrumProjection feature is calculated as the projection of the AudioSpectrumBasis feature to the spectral shape. Then it is formulated as follow:

$$\tilde{Y} = \tilde{X}\overline{V_k} \qquad \text{(Eqn 4. 6)}$$

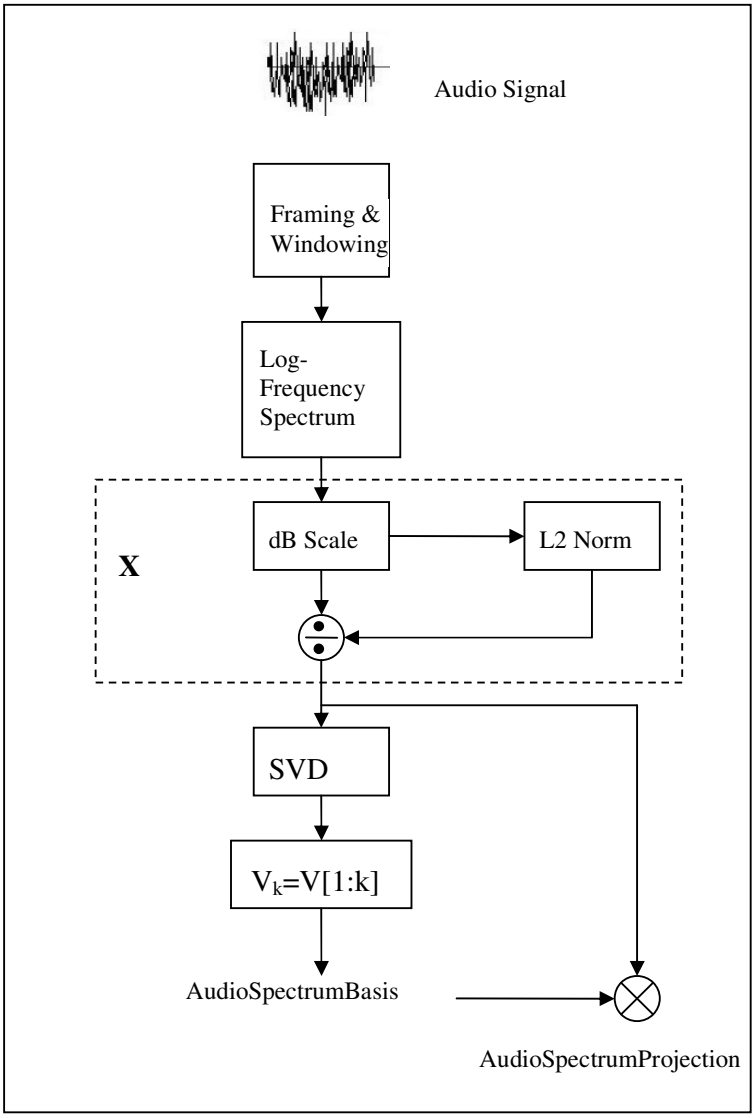

Figure 16. Calculation process of ASB and ASP descriptors (Edited [49])

4.2.1.5 AudioFundamentalFrequency

The AudioFundamentalFrequency feature describes the fundamental frequency of an audio signal. This feature can be useful for distinguish of voiced signals from the unvoiced signal (e.g., noise). The AudioFundamentalFrequency is calculated by the methods called “pitch estimation”.

There are different pitch estimation algorithms in the literature. There methods can be given for pitch estimation: Time Domain Methods, Frequency Domain Methods and Statistical methods. In this study time domain, autocorrelation method [35] is chosen for estimating the pitch contours.

The correlation between two waveforms is a measure of their similarity. The waveforms are compared at different time intervals, and their “sameness” is calculated at each interval. The autocorrelation function is defined as:

$$Rxx(\tau) = \sum_{t=1}^{N-\tau-1} x(t)x(t-\tau) \qquad \text{(Eqn 4. 7)}$$

The audio signal x(t) is being convolved with a time-lagged version of itself. To obtain a useful set of results, the autocorrelation function is computed over a range of lag values. The autocorrelation function is itself periodic. For periodic signals the function attains a maximum at sample lags of 0, +/-P, +/-2P, etc. where P is the period of the signal.

One major limitation of the autocorrelation function is that it can retain too much information present in the signal. In speech, numerous peaks present in the autocorrelation function are due to damped oscillations of the vocal tract response. If these peaks happen to be bigger than the peaks due to periodicity, the simple procedure of picking the largest peak to be the period will fail.

Therefore, the signal needs to be pre-processed in some way to make the periodicity more dominant while suppressing other features which may cause distracting peaks. To partially eliminate the effects of the higher formant structure on the autocorrelation function, most methods use a sharp cutoff low-pass filter with cutoff

around 900 Hz as used in this study. In addition to linear filtering to remove the formant structure, such pre-processing techniques are sometimes called spectrum flatteners are applied. Centre clipping technique for a spectrum flattener is used in this application [34].

Centre clipping works by clipping a certain percentage of the waveform. Let Amax be the maximum amplitude of the signal and CL is the clipping level. CL is a fixed percentage of Amax. Therefore, the output from the center clipper is as follows:

$$y(n) = \begin{cases} x(n) - CL & x(n) > CL \\ 0 & x(n) \le CL \end{cases} \qquad \text{(Eqn 4. 8)}$$

i.e., for samples below the clipping level, the output is zero, and for samples above the clipping level, the output is equal to the input minus the clipping level. In this study %68 clipping level is used. A sinusoidal signal and the output of the clipping operation are given in Figure 17.

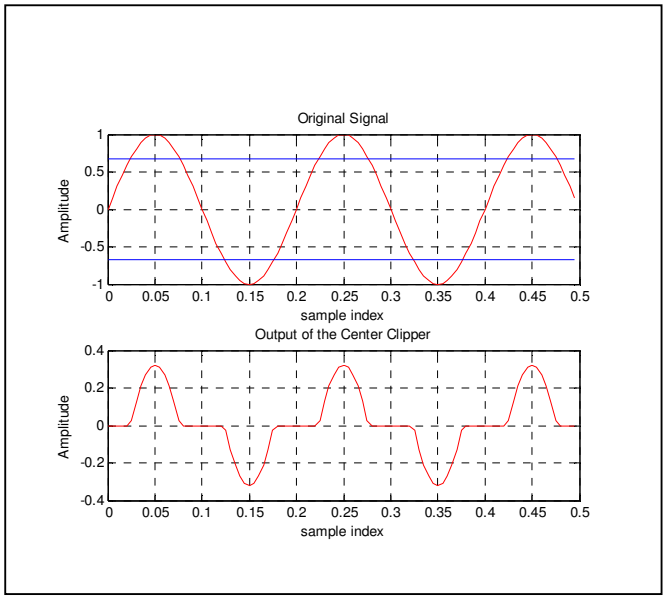

Figure 17. A Sinusoidal Signal and Clipped version of that signal

We can summarize the process for detection of the pitch contours as follow:

- Normalize the amplitude of the signal
- Determine the frame length as 30 ms, update rate 20 ms (10 ms overlap)
- For each frame
  - Apply linear filtering and clipping process
  - Calculate frame energy of the clipped waveform.

$$Energy = \left| Clipped \quad Signal \right|^2 \qquad \text{(Eqn 4. 9)}$$

  - Calculate the autocorrelation of the clipped waveform
  - Calculate the maximum amplitude of autocorrelation in the range of interested frequency (60 Hz is the lower frequency and 320 Hz is the upper frequency)

$$LF = floor\left| Fs / 320 \right| \qquad \text{(Eqn 4. 10)}$$

$$HF = floor\left| Fs / 60 \right| \qquad \text{(Eqn 4. 11)}$$

    where Fs is the sampling frequency

  - Calculate the fundamental frequency of the frame by using the index value of the maximum autocorrelation value.

$$F0 = Fs / indexOf \max imumValue \qquad \text{(Eqn 4. 12)}$$

  - Set a threshold value for the voiced and unvoiced component discrimination.

$$Threshold = 0.4 * Energy \qquad \text{(Eqn 4. 13)}$$

  - Decide the fundamental frequency of the frame by decision of the voiced/unvoiced value of the frame.

$$F0 = \begin{cases} \dfrac{Fs}{indexOf \max} & R\max > Threshold \quad \{voiced \quad segmeNt\} \\ 0 & R\max \le Threshold \quad \{unvoiced \quad segment\} \end{cases} \qquad \text{(Eqn 4. 14)}$$

In Figure 18, we see pitch contour plots for music, speech and crowd audio segments. Since the harmonic properties of the speech and music, the zero ratio of the pitch contours is small when comparing the crowd signal waveform.

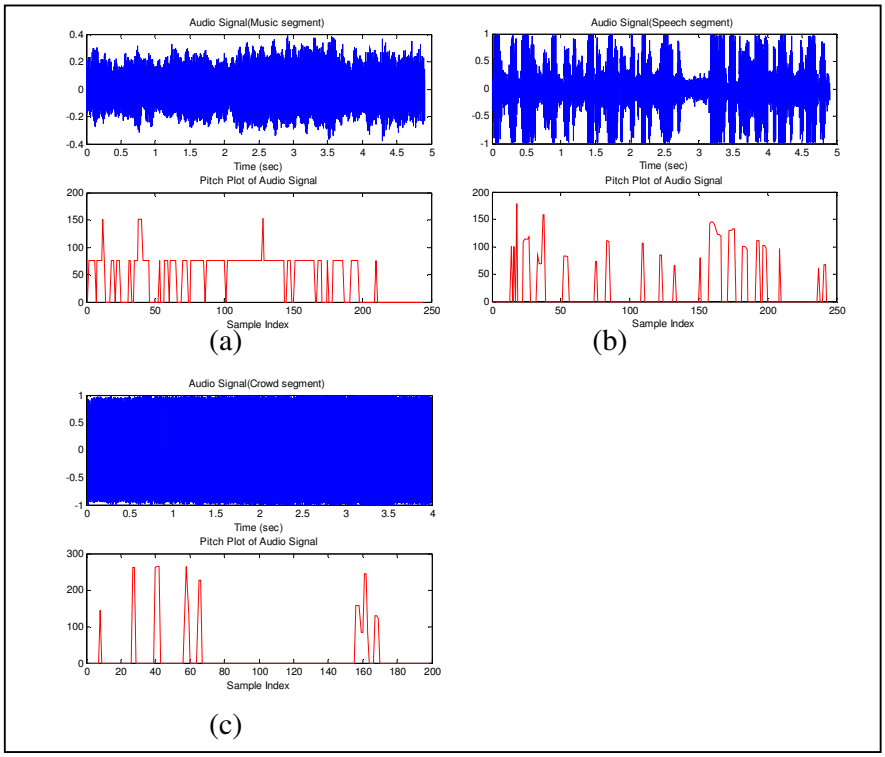

Figure 18. Pitch Contour Plots (a) Speech, (b) Music and (c) Crowd

4.2.2 Time Domain Audio Features

4.2.2.1 Short Time Energy

Short-Time energy function of an audio signal is defined as:

$$E_n = \frac{1}{N}\sum_{i=1}^{N-1}\left[x(m).w(n-m)\right]^2 \qquad \text{(Eqn 4. 15)}$$

N is the number of sample in the frame

This measurement can in a way distinguish between voiced and unvoiced speech segments, since unvoiced speech has significantly smaller short- time energy. It can be used as the measurement to distinguish audible sounds from silence when the SNR is high. Its change pattern over time may reveal the rhythm and periodicity properties of sound.

For the length of the window a practical choice is 10-20 msec that is 160-320 samples for sampling frequency 16 kHz. This way the window will include a suitable number of pitch periods so that the result will be neither too smooth, nor too detailed. In this study a hamming window in the length of 20 ms is chosen. Frame structure is chosen to be a non-overlapping framing structure. The audio waveform of a speech and silence segment and the temporal curve of its short-time energy function are shown in Figure 19.

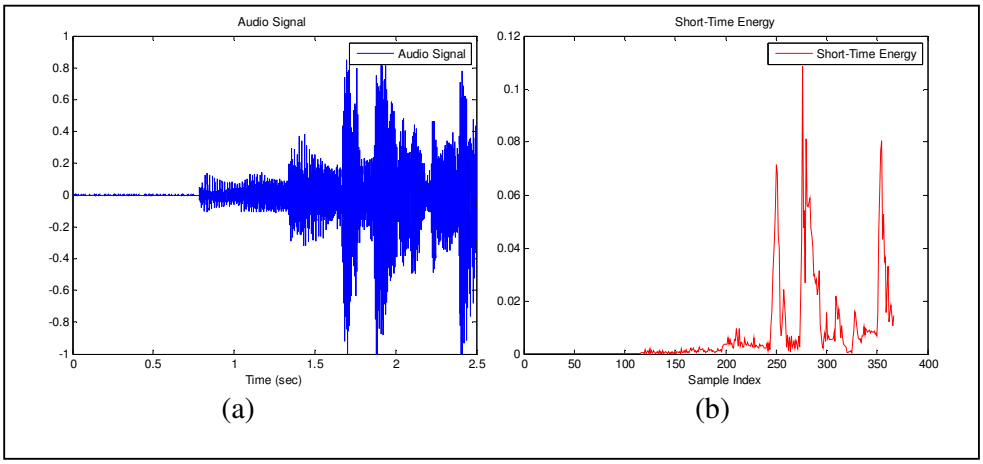

Figure 19. Short-time energy function for a speech and silence segment

4.2.2.2 Energy Entropy

The Energy Entropy feature is the measure of abrupt changes in the energy level of an audio signal. It is computed by further dividing each frame into K sub-windows of fixed duration [31]. For each sub-window the normalized energy σ^2 is calculated and divided by the whole frame's energy. Afterwards, the energy entropy is computed using the following equation:

$$H = \sum_{i=0}^{K-1} \sigma^2 . \log_2(\sigma^2) \qquad \text{(Eqn 4. 16)}$$

K was chosen to be 10.

4.2.2.3 Zero Crossing Rate

Zero crossing rate (ZCR) is the rate of sign-changes along a signal. It is defined as the number of time-domain zero-crossings within a frame, divided by the number of samples of that frame [32].

It is computed using the equation below:

$$ZRC = \frac{1}{N}\sum_{n=1}^{N-1}\frac{\left|\text{sgn}\{x(n)\} - \text{sgn}\{x(n-1)\}\right|}{2} \qquad \text{(Eqn 4. 17)}$$

where sgn(.) stands for the sign function, i.e.,

$$\text{sgn}\{x(n)\} = \begin{cases} 1 & x(n) \geq 0 \\ -1 & x(n) < 0 \end{cases} \qquad \text{(Eqn 4. 18)}$$

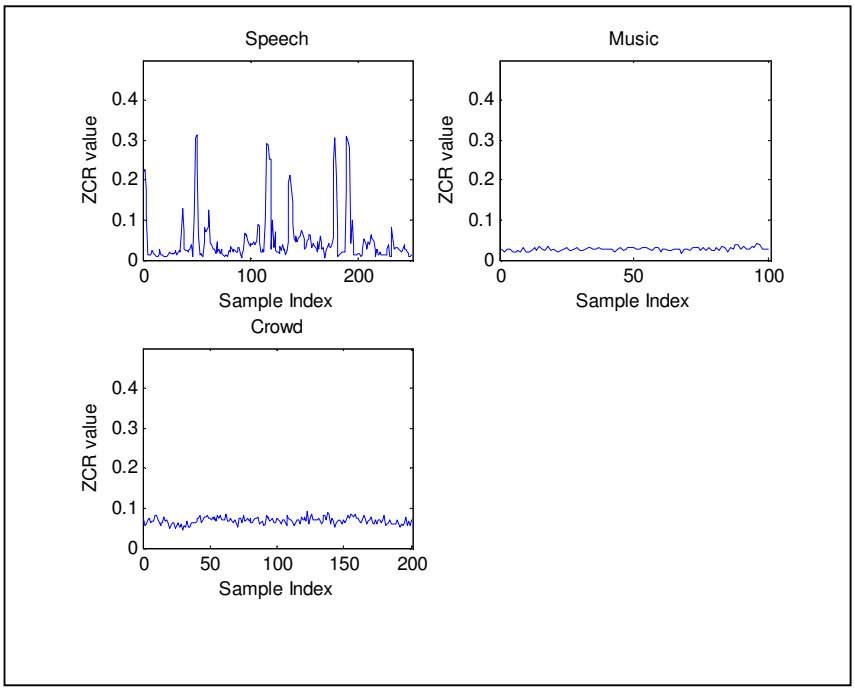

Figure 20. The ZCR plots for music, speech and crowd segments.

Zero crossing rate has been proven to be useful in characterizing different audio signals and has been popularly used in speech/music classification problems. The thought is that the ZCR should be directly related to the number of times the waveform repeats per unit time. The rate at which zero crossings happen is a simple measure of the frequency content of the signal. For narrow band signals, the average zero crossing rate gives a reasonable way to estimate the frequency content of the

signal. But for a broad band signal such as speech, it is much less accurate. However, by using a representation based on the short time average zero crossing rate, rough estimates of spectral properties can be obtained.

As shown in Figure 20 the speech ZCR curve has a large variance and a wide dynamic range of amplitude. Note also that the ZCR curve has a relatively low and stable baseline with high peaks above it.

The ZCR curve of music plotted Figure 20 in has a much lower variance and average amplitude, suggesting that the zero-crossing rate of music is normally much more stable during a certain period of time.

4.2.3 Frequency Domain Features

4.2.3.1 Spectrogram

However the MPEG-7 AudioSpectrumBasis and AudioSpectrumProjection features are based on the time-frequency analysis, their dimensions are reduced, and it results some data loss. So we have investigated spectrogram feature alone in this section for the classification purpose.

Spectrogram is a time-frequency representation of the speech signal [15] [16]. Spectrogram images are computed by concatenating spectra obtained from consecutive short time Fourier transforms (STFT). When speech signal is segmented such a small windows, each segment of the signal can be thought quasi-stationary, but since the time and frequency resolutions are inversely proportional, small analysis windows lead to poor localization of the frequency components and vice versa, as good resolution in both the time domain and the frequency domain in the same image is not possible. In practice, narrowband spectrograms are used for good frequency resolution while wideband spectrograms allow good temporal resolution of speech signals.

Given a signal x(t) and a window w(t), the short-time power spectrum at time t is [16]

$$S_X(t,w) = \left| \int_{-\infty}^{\infty} w(\tau) x(t+\tau) e^{-jw\tau} d\tau \right|^2 \qquad \text{(Eqn 4. 19)}$$

For each frame of the signal STFT is calculated. Each frame corresponds to a vertical line in the image; a measurement of magnitude versus frequency for a specific moment in time. This representation is called a spectrogram.

In this work, a Hamming window length was chosen to be 20 ms is used as it is typical in speech analysis. This length is short enough so that any single 20 ms frame will typically contain data from only one phoneme.

Spectrogram feature can be used for separating music, speech and crowd speech waveforms. In Figure 21 we see the spectrogram plots of speech, music and crowd waveforms. As we see, in crowd and music waveforms most of the energy distributes below the some frequencies, but in speech waveform energy is distributed a larger region. In music waveform, frequency of the signal remains same for a larger region of time whereas in speech and crowd waveform it changes more rapidly. In speech waveform deviation in frequency has larger dynamic range than music and crowd waveforms.

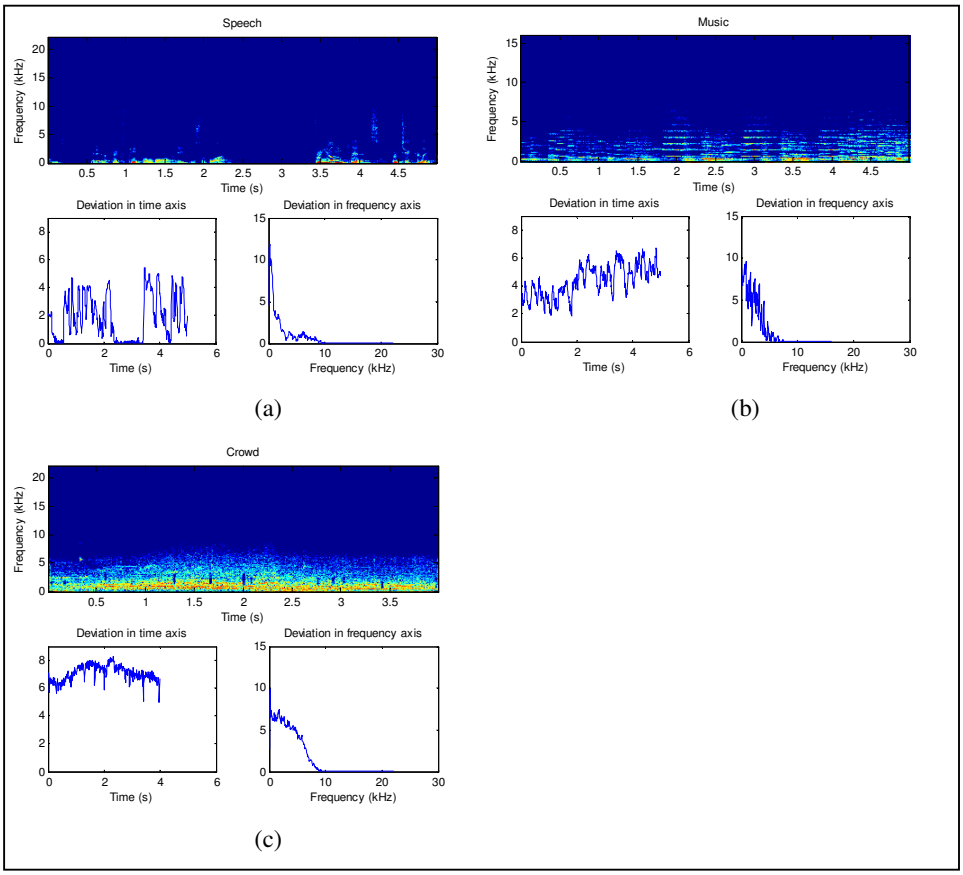

Figure 21. Spectrograms of a speech, music and crowd waveforms.

4.2.3.2 Spectral Roll-off

Spectral roll-off is measure of spectral shape. It is defined as the frequency bin $m_c^R(i)$ bellow which c% of the magnitude distribution of the DFT coefficients is concentrated [31],[44]. This relationship is given as following formula:

$$\sum_{m=0}^{m_c^R(i)} m.\left|X_i(m)\right| = \frac{c}{100}\sum_{m=0}^{N-1} m.\left|X_i(m)\right| \qquad \text{(Eqn 4. 20)}$$

This feature is a measure of skewness of the spectral shape. In the current work, we define c to be equal to 80.

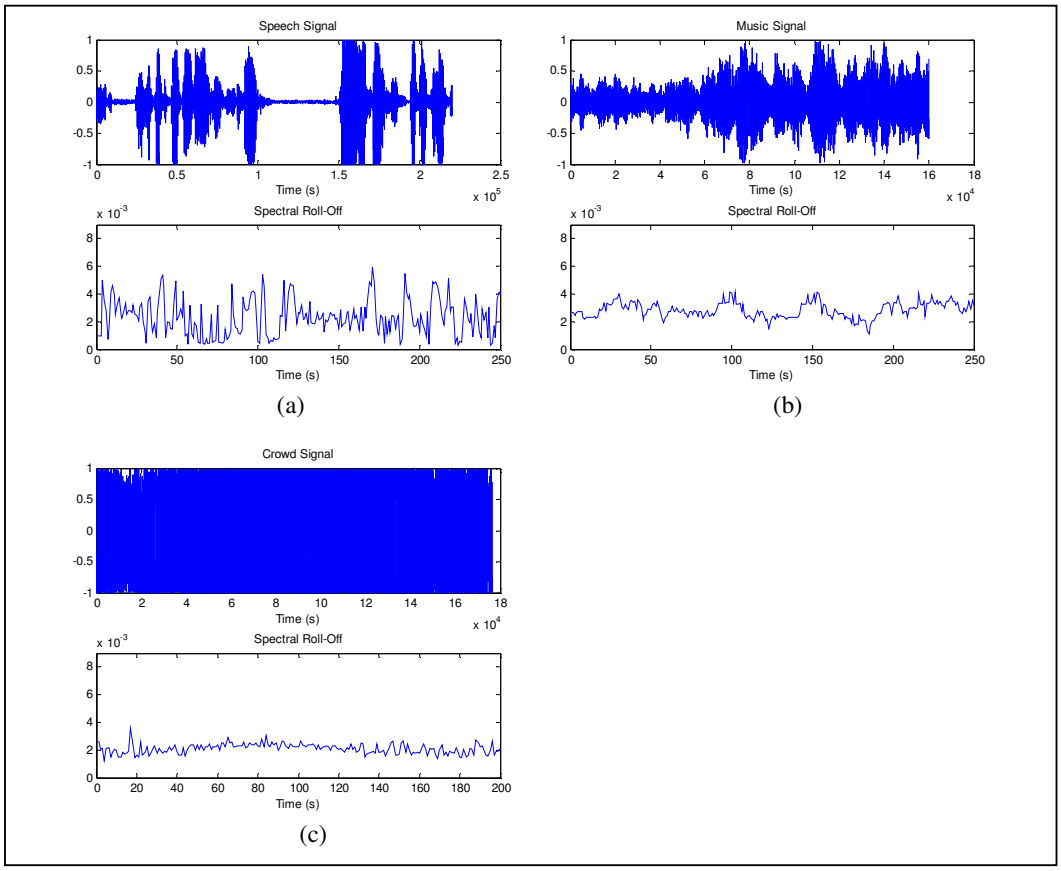

Figure 22. Audio waveforms and the Spectral Roll-Off plots. (a) Speech (b) Music (c) crowd

In Figure 22 we see the Spectral Roll-Off plots for the speech, music and crowd segments. We see that the plots have different shapes which can easily noticed. This feature can be used for classification of speech, music and crowd.

4.2.3.3 Short Time Fundamental Frequency Features

The pitch of an audio signal can be defined as the tonal height of a sound object, e.g. a musical tone or the human voice. It is a perceptual feature, relevant only in the context of a human listening to that signal. It is an inherent property of periodic

signals and also describes the naturalness of speech. Pitch represents the perceived frequency of a sound.

Most of the sounds consist of a series of major frequency components including the fundamental frequency and those which are integer multiples of the fundamental one, which are called harmonics. This type of sounds called harmonic sounds, such as speech, or a sound that produced by a musical instrument. The perception of pitch depends on these harmonic frequencies.

The speech signal is a harmonic and non-harmonic mixed sound, since voiced components are harmonic while unvoiced components are non-harmonic. Most environmental sounds are non-harmonic, such as the sounds of applause, footstep, and explosion. So estimating the pitch features can be useful in classification of audio signals.

The autocorrelation pitch detection method has been adopted to calculate the pitch contours of the audio signals in this study (see. Section 4.2.6). More attributes of the pitch feature is used in this study. In the following paragraphs the Cyclic Attribute of the Pitch; the Chroma feature is explained.

4.2.3.3.1 Cyclic Attribution of Pitch

In the early 1960s, Shepard [33] suggested that two dimensions are necessary to represent the perceptual structure of pitch, rather than one dimension. The perceptual structure of pitch presented as a helix rather than one dimension. The terms tone height and chroma characterizes the vertical and angular dimensions, respectively.

Tone height describes the general increase in the pitch of a sound as its frequency increases. Chroma, on the other hand, is cyclic attribute of the helix in nature with octave periodicity. Under this formulation two tones separated by an integral number of octaves share the same value of chroma.

In Figure 23 an illustration of this helix with its two dimensions is represented. In this representation, as the pitch of a musical note increases, say from C1 to C2, its locus moves along the helix, rotating cyclically through all of the pitch classes before it returns to the initial pitch class (C) one cycle above the starting point [1].

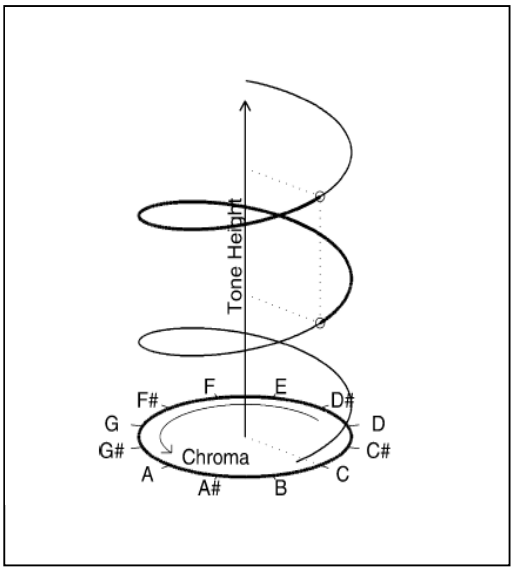

Figure 23. Illustration of Shepard's helix of pitch perception. The vertical dimension is tone height, while the angular dimension is chroma [1].

According to Shepard's results [33], the relationship between the perceived pitch, p, of a signal with values of chroma, c, and tone height, h, as

$$p = 2^{h+c} \qquad \text{(Eqn 4. 21)}$$

For this decomposition to be unique, it is sufficient for c∈[0,1] and h∈Z. Linear changes in c result in logarithmic changes in the fundamental frequency associated with the pitch. The interval of c is divided into 12 equal parts to obtain 12 equal-tempered chromatic scale as the analogous of traditional pitch classes for Western music. In fact, the terms pitch class and octave number are used as analogous to

Shepard's chroma and tone height [1]. So the interval of chroma is divided in to 12 equal regions.

In Section 4.2.3.3, we see that detection of signal pitch is similar to detect the fundamental frequency of the signal. In 1980, Patterson generalized Shepard's results to frequency [1]. So the equation which gives the relation between the pitch, tone height and chroma can be written as follow [1]:

$$f = 2^{h+c} \qquad \text{(Eqn 4. 22)}$$

where we again restrict c∈[0,1] and h∈Z [7]. Alternately, we can calculate chroma from a given frequency using

$$c = \log_2 f - \lfloor \log_2 f \rfloor \qquad \text{(Eqn 4. 23)}$$

where $\lfloor . \rfloor$ denotes the greatest integer function. Chroma is simply the fractional part of the base-2 logarithm of frequency. Similar to ideas of pitch, certain frequencies under this system share the same chroma class if and only if they are mapped to the same value of c. Thus, 200, 400, and 800 Hz all share the same chroma class as 100 Hz, but 300 Hz does not [1].

A signal analysis technique which gives the time-chroma relation of the signal can be defined, named as chromagram. This chromagram is based on the log magnitude spectrogram of the signal. It is the analogous of the standard time-frequency relation, spectrogram, of the signal. The chromagram is given as:

$$S(t,c) = S(t,f); \quad \forall \quad f = 2^{h+c} \qquad \text{(Eqn 4. 24)}$$

where S(t,f) is the time-frequency relation, spectrogram of the signal and c∈[0,1] and h∈Z.

In chromogram, the spectral energy in a signal, as measured by the spectrogram, is assigned to a chroma band. The energy in each chroma band forms a set of chroma

features. In this study, 12 chroma bands are chosen, as the analogies for the 12 traditional pitch classes for Western music [1].

The process for the calculation of chromagram feature vectors is given as follow:

- Segment the audio signal into frames
- Calculate the logarithmic amplitude of the DFT for the frame.
    - DFT length is chosen as the equal to the first power of 2 greater than or equal to the length of the frame
- The elements of the chromagram feature vector for the t^{th} frame v_t are calculated using the following equation [1].

$$V_{t,k} = \sum_{n \in S_k} \frac{F_t(n)}{N_k}, \quad k \in \{0...11\} \qquad \text{(Eqn 4. 25)}$$

Where each $S_k \in Z$ defines a subset of the discrete frequency space for each pitch class and N_k is the number of elements in S_k.

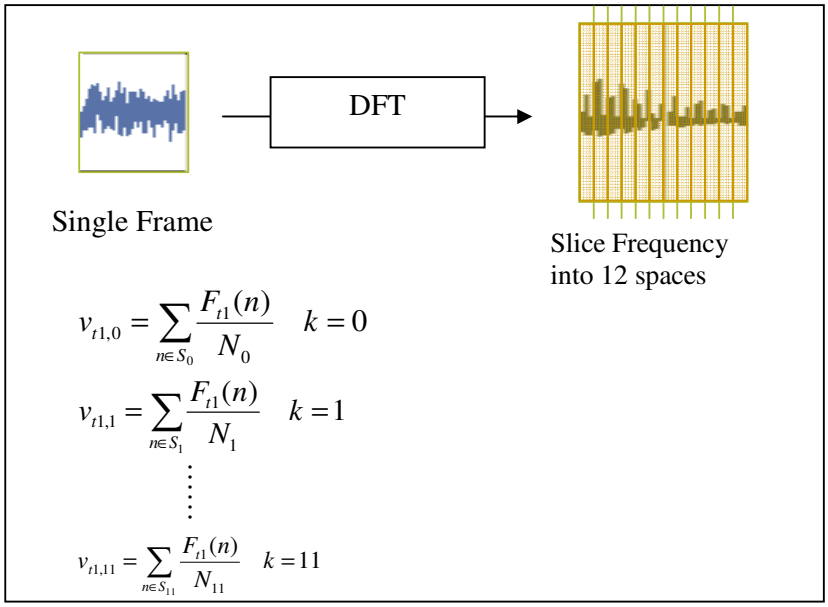

Figure 24. Calculation process of the Cromagram Feature Vectors

For the chromagram feature vector we calculate deviation between successive frames for each chromagram feature element as used in [31]. In this paper it is suggested that in music segments there is at least one chromagram element with low deviation for a short period of time while in speech segments, the deviation of each chroma element is high. To compute chroma-based feature, a sort-term window of 20 msecs has been adopted, while the minimum deviation of the chroma coefficients was computed for every 10 frames, i.e. a mid-term window of 200 msecs was used. By calculating the histogram distribution of this feature for “speech”, “music”, and “crowd”, we decided to use the standard deviation and the ratio of maximum value to standard deviation statistics as the final feature (see Section 5.1).

In Figure 25 we see the chromagram plots for the speech and music segments. As we see in music segment, chroma elements do not change so much whereas in the speech segment, all of the chroma elements changes.

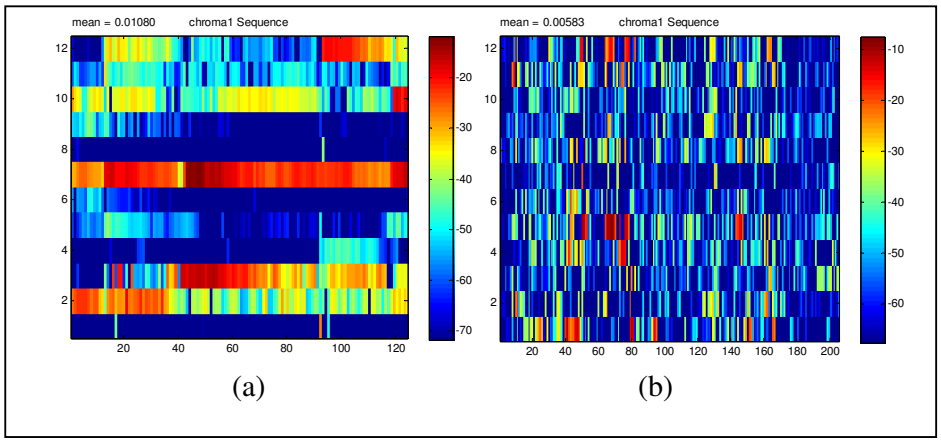

Figure 25. Chromogram Plots (a) Music (b) Speech

4.2.3.4 Mel-frequency Cepstral Coefficients

Mel Frequency Cepstral Coefficients (MFCC) are short-term spectral features of the audio signal. MFCCs are widely used for speech recognition, and Logan in [36] has

shown that they are also justified for music analysis. So we can use MFCC features for in music/ speech classification problem.

A schema that represents the calculation process of the MFCC is given in Figure 26. The detailed explanation of each step is given in the following paragraphs.

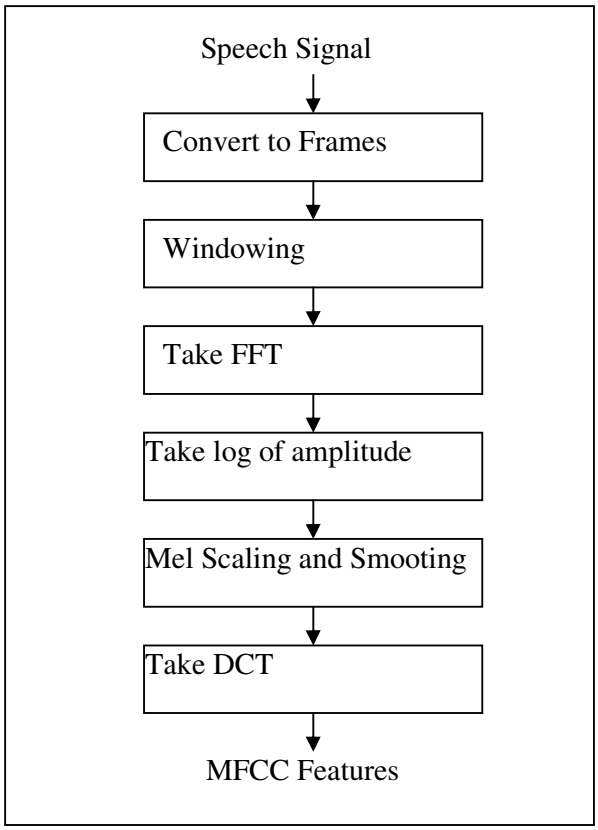

Figure 26. The calculation process of the MFCC features

As we mentioned in section 4.2 speech signal is not a statistically stationary signal. The first step is segmenting the continuous audio signal into short time overlapping frames to obtain statistically stationary signals. Frame length is chosen as 20 ms which is enough that signal variations are retained within a frame and each frame overlaps by 10 ms. In the windowing step a hamming window is used.

The next step is the process of converting each frame of N samples from the time domain to the frequency domain. Here we will take the Discrete Fourier Transform of each frame. We use the FFT algorithm, which is computationally efficient, to implement the DFT. As the amplitude of the spectrum is much more important than the phase, we will retain only the amplitude spectrum.

The next step is the transformation of the real frequency scale to the mel frequency scale. A mel is a unit of measure of perceived pitch or frequency of a tone. The mel-frequency is based on the nonlinear human perception of the frequencies of audio signals. The mel-frequency scale is a linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz.

$$mel = \begin{cases} f & f < 1000\,Hz \\ 2595 * \log(1 + f/700) & f > 1000\,Hz \end{cases} \qquad \text{(Eqn 4. 26)}$$

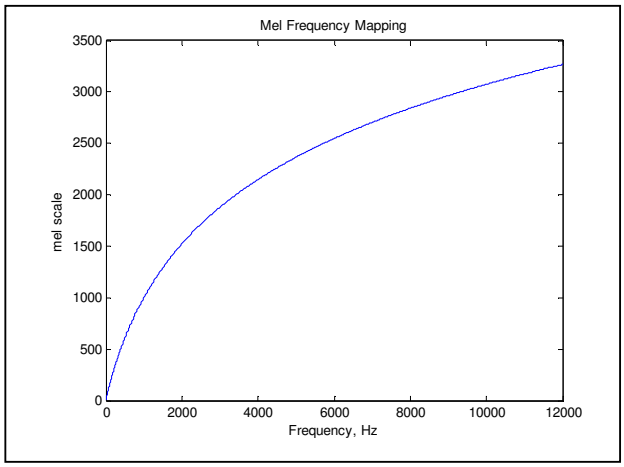

Figure 27. Mel Frequency Mapping

A mel-spaced filter bank showing the above characteristics is used for mel frequency transformation. In this study the filter bank used for the computation of the MFCCs consists of 40 triangular bandpass filters, with bandwidth and spacing determined by

a constant mel-frequency interval. The adopted filter bank covers the frequency range 0−8KHz, suggesting a sampling rate of 16KHz.

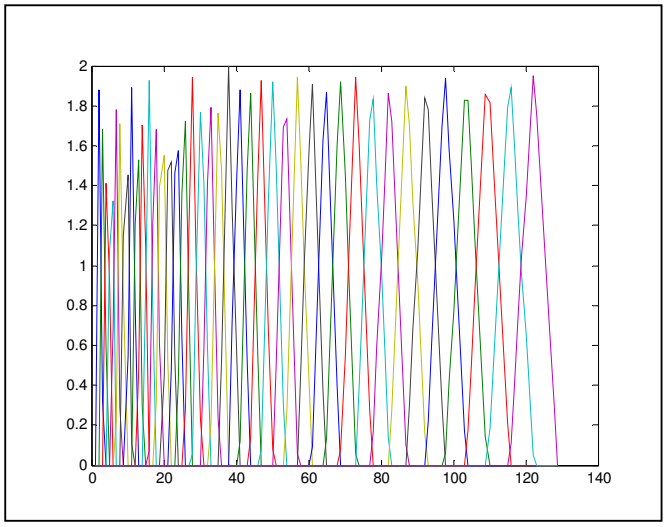

Figure 28. Frequency response of a mel-spaced filterbank

After the mel frequency conversion, the last step is converting the log mel spectrum back to the time domain and the result is the Mel Frequency Cepstral Coefficients. The components of the mel spectral vectors calculated are highly correlated. In order to reduce the number of the parameters, some transform, which decorrelates their components, is applied to these vectors. Theoretically, the Karhunen-Loeve (KL) works well for this purpose. However, the KL is very complex and since the DCT can still end in good results, the DCT is frequently used. After the DCT, 13 cepstral features are obtained for each frame.

In this study the mean value of the MFCC vectors, and the mean and max values of the second cepstral coefficients are used. The second cepstral coefficient is labeled as MFCC2.

CHAPTER 5

5 AUDIO TRAINING AND CLASSIFICATON STAGE

As we mention in Chapter 4, audio classification is made up of two main sections: a signal processing part and a classification part. A training stage is required before applying the classification section. In this Chapter Audio training and classification part explained in detail.

In the classification section a One-Versus-All classification scheme has been adopted as in [31].

5.1 Audio Feature Selection

In audio processing system the performance of a set of features depends on the application as we mentioned in Section 4.1. The design of descriptive features for a specific application is hence the main challenge in building audio classification systems. A wide range of audio features exist for classification tasks. The features used in this study are described in Chapter 4. In fact more audio features were examined but the not all of them has been used in this study.

A simple class separability measure based on feature histograms is used for measuring the ability of each feature to be used for classifying the given classes [31]. The statistics of the features that have more discriminative/different histograms for the classes have been chosen. Almost 4000 audio samples have been extracted and manually labeled as “music”, “speech” and “crowd”. The duration of those audio segments varies from 2 to 5 seconds. The Trecvid 2001 [51] dataset, MPEG-7 Content Set (1998) [52] some sports records (basketball, football…etc. videos), TV

show videos are used. The video files have been decomposed into audio and video files firstly. The Trecvid 2001 dataset is downloaded from the Open Video Project [46]. Then, audio samples have been manually labeled and collected in the directories named as “Speech”, “Music”, “Crowd”. For manually labeling the audio files Audio Annotation and Classification Tool (see Section 5.4) was used. By using Matlab, the features were calculated in a two-step way:

Step 1: The audio signal was broken into short-term frames. The length and the overlap structure of the frames based on the specified feature. For an example, as explained in the Section 4.1. for the Spectral Roll-off feature a non-overlapping frame structure has been used and the frame length has been chosen as 20 ms. For each frame, the specified feature has been calculated. This step leads to a feature sequence for the whole audio signal.

Step 2: For the feature sequence calculated in Step 1, 6 statistics are calculated. For example, for the Spectral Roll-off sequence, the standard deviation, mean, max, min, median, zero ratio values are calculated. This step leads to 6 single statistic values (for feature sequence). Those 6 values are the final feature values that characterize the input audio signal for the specified feature.

For the each audio in the 3 directories (“Speech”, “Music”, “Crowd”) Step 1 and Step 2 are calculated. At the end of these steps, 3 vectors which are in 6xN, 6xM, 6xC dimensions are obtained. N, M and C are the number of audio sample in the “Speech”, “Music”, “Crowd” directories respectively. The histograms of the six statistics for the specified features are calculated and plotted on the same plot for each run. By analyzing the histogram plots, the most separable statistic is chosen as the final statistic for the specified feature. For an example, the histogram plots for the second cepstral coefficient of the MFCC for six statistics (std, mean, max, median, min, mean exceed ratio) are given in Figure 29

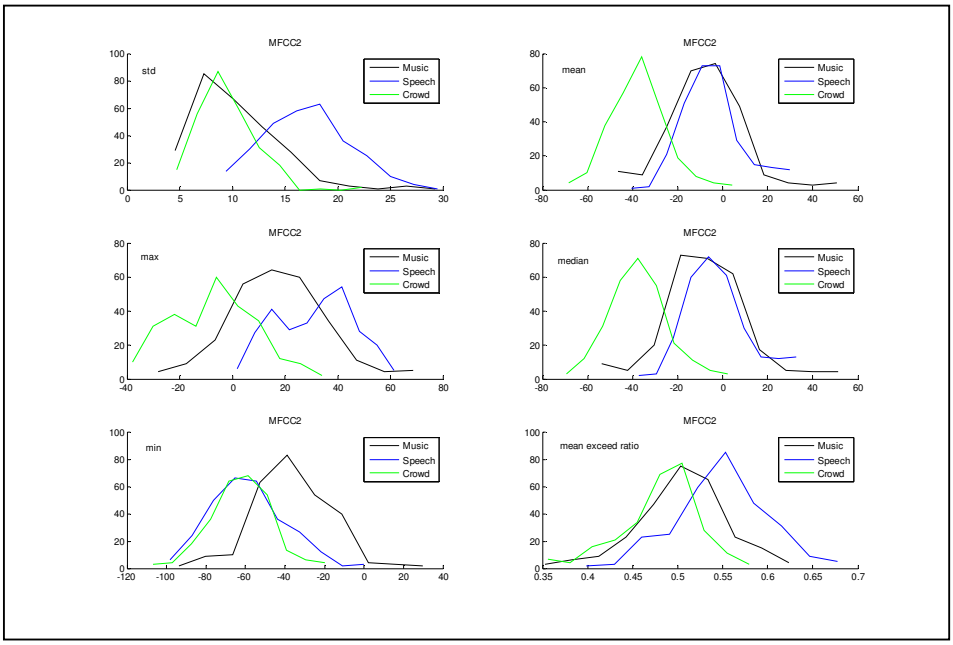

Figure 29. Histogram plots of the six statistic for the MFCC feature

These steps were applied for all the audio feature types. Then the features and their final statistic were chosen. Then three feature sets were formed. One of them includes the features that are belong MPEG-7 standard, other is the features that are used in [31] and the last one is the combination of some MPEG-7 features and other features that used in [31].

In forming Feature Set 3, we choose the features from both sets which give maximum separability for the 'speech', 'music' and 'crowd' classes using the histogram figures. The most separable statistic is also chosen as the final statistic for the specified feature. The list of the features and their statistics are given in Table 1, Table 2 and Table 3. The meanings of the statistical values are given in Table 4

Table 1 Feature set 1: MPEG-7 features

	Feature	Statistic
1	AudioSpectrumSpreadD	max
2	AudioSpectrumSpreadD	std
3	AudioSpectrumFlatnessD	mean
4	AudioSpectrumFlatnessD	median
5	AudioSpectrumEnvelopeD	Std by mean
6	AudioSpectrumBasisD	Max by std
7	AudioSpectrumBasisD	mean
8	AudioSpectrumProjectionD	Max by std
9	AudioSpectrumProjectionD	median
10	AudioFundamentalFrequency	Zero Rate
11	AudioFundamentalFrequency	Max by std
12	AudioFundamentalFrequency	std

Table 2 Feature set 2: audio features and their statistics used in [31].

	Feature	Statistic
1	Spectrogram	Std
2	Chroma 1	Mean
3	Chroma 2	Median
4	Energy Entropy	Max
5	MFCC 2	Std
6	MFCC 1	Max
7	ZCR	Mean
8	Specral Roll-Off	Median
9	Pitch Feature	Zero Rate
10	MFCC 1	Max by mean
11	Spectrogram	Max
12	MFCC 3	median

Table 3 Feature set 3: MPEG-7 features and other features

	Feature	Statistic
1	AudioSpectrumSpreadD	max
2	AudioSpectrumFlatnessD	mean
3	Spectral Roll-Off	Std by max
4	Zero Crossing Rate	Mean exceed ratio
5	Chroma	Max by std
6	Chroma	std
7	MFCC	mean
8	MFCC2	max
9	MFCC2	mean
10	Spectogram	mean
11	Spectogram	Std by mean
12	AudioFundamentalFrequency	zeroRate

Table 4 The list of statistic and their explanation

Feature	Statistic
Max	Maximum value of the feature set
Mean	Mean value of the feature set
Median	Median value of the feature set
Std	Standard deviation value of the feature set
Min	Minimum value of the feature set
Zero Rate	Zero values rate of the feature set
Mean exceed ratio	The rate of the number of values that larger the mean value to the number of feature set
Std by max	The ratio of standard deviation to the maximum value
Std by min	The ratio of standard deviation to the minimum value
Std by median	The ratio of standard deviation to the median value
Std by mean	The ratio of standard deviation to the mean value
Max by std	The ratio of the maximum value to the standard deviation
Max by mean	The ratio of the maximum value to the mean value

5.2 Classification Stage

In this study, two classification schemes have been applied. For distinguish the silence segments from the non-silence segments only the short time energy feature has been used. A simple threshold comparison method has been applied on the short time energy feature.

As a second classification scheme, a multiclass classification scheme based on Bayesian Networks and KNN classifier has been adopted to segment audio data acoustically similar regions such as “speech”, “music”, “crowd” segments. The segments that are not belong each of these segments have been left as unclassified regions. To achieve multiclass classification, “One-vs-All” (OVA) classification scheme has been applied.

In the current work, Bayesian Networks (BNs) were used for building binary classifiers. The detailed explanation of the Silence Segment Detection, Bayesian Networks and KNN is given in the following paragraphs.

5.2.1 Silence Segment Detection

In the many research it has been shown that the silence segments have smaller energy than the non-silence segments. In the silence segments detection process only the short time energy feature has been used. A simple classification scheme based on setting a suitable threshold has been applied for silence segment detection. If the short-time energy function is continuously lower than a certain threshold for 2 seconds than a silence segment is detected.

For choosing the threshold value, I have modified the threshold value that is used for the Voice Activity Detectors to detect the speech and noise parts of the signal [52]. Following algorithm is performed

- Calculate the short time energy of the signal

- Determine an initial segment that is assumed can be silence segment. (0.1 second)
- Calculate the mean value of the initial segment.
- Find the maximum value of the signal energy.
- Calculate threshold 1 as:

  I1=0.03*(maximum energy of the signal-mean of initial segment)+ mean of initial segment
- Calculate threshold 2 as

  I2= 4* mean of initial segment
- Then choose the minimum of I1 and I2 as the final threshold value.

  Threshold= min (I1,I2)

5.2.2 K-Nearest Neighbor Classification

The K- nearest neighbor classifier is an example of a non parametric classifier [38]. The basic algorithm in such classifiers is simple. For each input feature vector to be classified, a search is made to find the location of the K nearest training examples, and then assign the input to the class having the largest members in this location. Euclidean distance is commonly used as the metric to measure neighborhood. For the special case of K=1 we will obtain the nearest neighbor classifier, which simply assigns the input feature vector to the same class as that of the nearest training vector. The Euclidean distance between feature vectors $X = \{x_1, x_2, ..., x_n\}$ and $Y = \{y_1, y_2, ..., y_n\}$ is given by:

$$d = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \qquad \text{(Eqn 5. 1)}$$

The discriminant function which is used to distinguish the classes can be defined as

$$p_i = \frac{k_i}{k} \qquad \text{(Eqn 5. 2)}$$

The class that has the maximum value of the discriminant function is the winner class.

The KNN algorithm is very simple yet rather powerful, and used in many applications. However, there are things that need to be considered when KNN classifiers are used. The Euclidean distance measure is typically used in the KNN algorithm. In some cases, use of this metric might result in an undesirable outcome. For instance, in cases where several feature sets (where one feature set has relatively large values) are used as a combined input to a KNN classifier, the KNN will be biased by the larger values. This leads to a very poor performance. A possible method for avoiding this problem would be to normalise the feature sets.

In Figure 30, an example of a three class classification task is shown. The aim is to use the KNN classifier for finding the class of an unknown feature x. As it can be seen in the figure, of the closest neighbors (K=5 neighbors) four belong to class B and only one belongs to class C and hence x is assigned to class B.

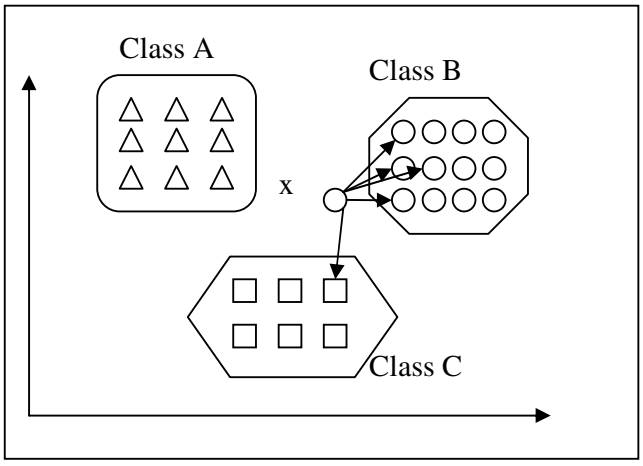

Figure 30. The K nearest neighborhood rule (K=5)

Some of the disadvantages of the K nearest neighbor classifiers are:

- Need the entire feature vectors of all training data when a new vector feature is to be classified and hence large storage requirements.
- The classification time is longer when compared to some other classifiers.

The K nearest neighbor classifiers have some qualities that are important such as

- It requires no training and this is helpful especially when a new training data is added.
- Uses local information and hence can learn complex functions without needing to represent them explicitly.

5.2.3 Bayesian Networks

A bayesian network is a graphical model for probabilistic relationships among a set of variables [39] [41]. The joint probability distribution of a set of n variables, $\{X_1, \ldots, X_n\}$ is encoded as a directed acyclic graph and a set of conditional probability distributions (CPDs). Each node corresponds to a variable, and the CPD associated

with it gives the probability of each state of the variable given every possible combination of states of its parents. The set of parents of X_i, denoted C_i, is the set of nodes with an arc to X_i in the graph. The structure of the network encodes the assertion that each node is conditionally independent of its non-descendants given its parents. The joint distribution of the variables is thus given by

$$P(X_1,...,X_n) = \prod_{i=1}^{n} P(X_i|C_i) \qquad \text{(Eqn 5. 3)}$$

In the case of discrete random variables, the simplest form of CPD is a conditional probability table, but this requires space exponential in the number of parents of the variable.

In Naïve Bayesian Network scheme it is assumed that all the attributes of the networks are conditionally independent. Naive Bayes models can be viewed as Bayesian networks in which each random variable in the network, X_i has C as the sole parent and C has no parents. Under these assumptions the joint distributions of the attributes can be written as:

$$P(C, X_1,..., X_n) = P(C)\prod_{i=1}^{n} P(X_i|C_i) \qquad \text{(Eqn 5. 4)}$$

The conditional distributions, $P(X_i|C_i)$ is estimated through a Gaussian Density function in the case where X_i is a continuous variable. For the discreate variable case, $P(X_i|C_i)$ is estimated as the relative freq of samples having value X_i as i-th attribute in class C_i.

When the variable C is observed in the training data, naive Bayes can be used for classification, by assigning test example ($X_1, \ldots, X_n$) to the class C with highest $P(C|X_1, \ldots, X_n)$.

For the calculation of $P(C|X_1, \ldots, X_n)$ we can use the Bayes' theorem as in the following :

Given a *training data C, posterior probability of the test data X*, P(C|X) follows the Bayes theorem as:

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} = \frac{\prod_{i=1}^{n} P(X_i|C)P(C)}{P(X)} = \frac{P(C)\prod_{i=1}^{n} P(X_i|C)}{P(X)} \qquad \text{(Eqn 5. 5)}$$

P(X) can be ignored, since it is the same for all classes, and does not affect the relative values of their probabilities. So *P(C|X)* can be written as:

$$P(C|X) = P(C)\prod_{i=1}^{n} P(X_i|C) \qquad \text{(Eqn 5. 6)}$$

5.2.4 Multiclass Classification Process

For segmenting the audio file acoustically similar regions such as, "speech", "music", "crowd" segments a multiclass classification; "One-vs-All" (OVA) classification scheme has been used. This multiclass classification task [17] is based on decomposing the N-class classification problem into N binary sub-problems. N binary classifiers are trained to distinguish the examples in a single class from the examples in the remaining classes. For example, in this audio classification system, one of the single binary classifiers is trained to distinguish a music signal for non-music signals. When a new classification is desired for a data, the N classifiers are run, and the classifier which output is the largest value is chosen.

A Bayesian Network Classifier is a classification scheme which uses Bayesian Networks to combine the output of N simple classifiers [40]. For composing the OVA structure, the output of the each simple classifier fed as an input to the Bayesian Networks. The decision via Bayes' rule has been made for the output of the simple classifiers.

For obtaining the binary input information of the Bayesian Classifiers, KNN classifiers have been used as proposed in [31]. The output of the each KNN classifier has been fed as an input for the Bayesian Network. Our method is separated from the proposed method in [31] such a way that we have selected different audio features, we use Naïve Bayesian Networks. In our method output of the classification system

has there classes, “speech”, “music” and “crowd” and an additional class “unclassified” that includes the samples that are not belong to other classes.

As a first step, the 12 feature values vi, i = 1. . . 12 described in Chapter 4, have been grouped into three 4D separate feature vectors:

$$V(1) = [v1, v2, v3, v12] \quad \text{(Eqn 5. 7)}$$

$$V(2) = [v4, v5, v6, v7] \quad \text{(Eqn 5. 8)}$$

$$V(3) = [v8, v9, v10, v11] \quad \text{(Eqn 5. 9)}$$

This grouping was applied randomly. Afterwards, for each one of the 3 binary sub-problems, three k-Nearest Neighbor classifiers have been trained on the respective feature space. In particular, each KNN classifier KNN_i^j; $i = 1,2,3$ *and* $j = 1,2,3$ has been trained to distinguish between class i and all i' (not i), given the feature vector V^j. This leads to three binary decisions for each binary classification problem. Thus, a 3x3 matrix R has been defined as follow:

$$R_{i,j} = \begin{cases} 1, & \text{if the sample was classified in class } i, \text{ given the feature vector } V^j \\ 0, & \text{if the sample was classified in class not } i, \text{ given the feature vector } V^j \end{cases} \quad \text{(Eqn 5. 10)}$$

Let us consider the following result matrix

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

For example, the fact that $R_{1,1} = 1$, indicates that KNN_1^1 (i.e. the KNN classifier of the first binary sub-problem that functions on the feature space of the V^1 feature vectors) decided that the input sample is speech. The first row of the R matrix contains the output of KNN classifier for the “speech” class, second row for the “music” and third row for the “crowd”. The other two KNN classifiers of the same binary sub-problem decided that the input sample is non-speech. The emerging subject here is to decide to which class the input sample will be classified, according

to R. In order to classify the input sample to a specific class, the KNN binary decisions of each subproblem (i.e. the rows of matrix R) are fed as input to a separate BN, which produces a probabilistic measure for each class. In this study, the BN shown in Figure 31, has been used as a scheme for combining the decisions of the KNN individual classifiers as it used in [31]. Nodes $R_{i,1}$, $R_{i,2}$ and $R_{i,3}$ correspond to the binary decisions of the KNN individual classifiers for the i-th binary subproblem and are called attributes of the BN, while node Y_i is the output node and corresponds to the true binary label.

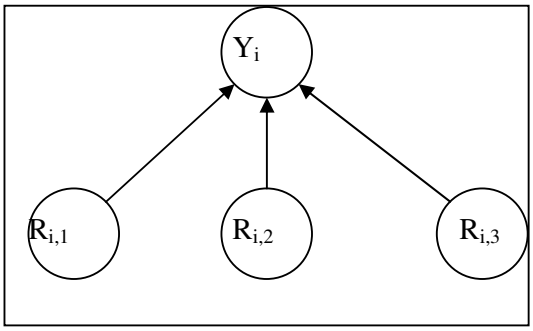

Figure 31. BN structure [31]

5.3 Training of the Audio data

In this stage, for training the binary sub-classifiers, 6 datasets have been created 3 of them contains data that are labeled as “speech”, “music” and “crowd”, other 3 of them contains data that labeled as “not-speech”, “not-music”, “not-crowd”. The “not” classes includes dataset other two classes and data that are not belong to all three classes labeled as “unclassified”. For example “not-music” class includes the data that belong to “speech”, “music” and “unclassified” audio samples. After the datasets have been created, the audio features and their statistical values that are described in Section 5.1, are calculated as the training data for the individual KNN classifiers. As we mentioned in Section 5.1, almost 4000 audio samples have been extracted and manually labeled as “music”, “speech” and “crowd”. Nearly %30 of each data sample was used as training data for individual KNN classifiers.

After populating the KNN classifiers, %55 of the audio samples in the datasets are used for training the Bayesian Networks. Bayesian Networks are trained via the validation of the individual KNN classifiers as described in Section 5.3. The remaining of the datasets are used for testing the proposed audio annotation system

5.4 Audio Annotation and Classification GUI

Audio Annotation and Classification tool has the capability of manual classification besides the automatic classification of the audio data. For the training parts, audio files are manually labeled by using this tool. Tool also has the capability for examining a set of audio features. A snapshot of the implemented audio annotation and classification tool is showed in Figure 32.

This Audio Annotation and Classification GUI has the following abilities:

- It allows user open an MPEG file or WAV file for the segmentation.
- The information about the file as sampling rate, duration, and number of audio channel is given in “File Info” part.
- It allows user to play and plot the audio file (audio of the MPEG video or wav file) in the desired segments. User can define the duration of the desired segment.
- In the “Labeling” part, user can manually label the current segment. The manually labeled segments are written in a XML file and a MATLAB “.mat” file for later usage by using the “Update Annotation File” button.
- When a manual annotation is applied, the percentage of the labeled data and the class information is shown in the “Class Distribution” part.
- Tool allows user to examine the audio features. The audio features for the selected part can be plotted and the desired statistical value can be calculated. Annotated segments can be saved as different “WAV” files.

- Tool gives the opportunity of the classifying current interested segment by automatically by using the proposed classifying method as described previous sections.

- Tool also can classify the whole audio file independently to the current segment. The result of the classified audio file can be shown in "classification" part. Different colors are used to represent the different audio segments.

The means of the colors are:

  - silence - black
  - music - purple
  - crowd – red
  - speech- blue
  - Unclassified- Yellow

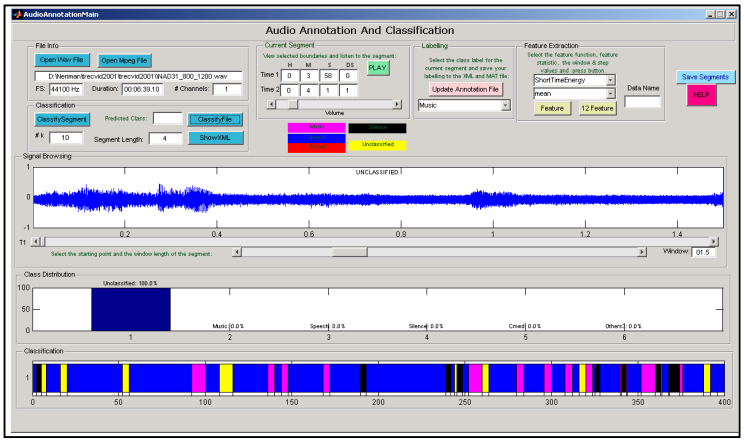

Figure 32 Audio Annotation and Classification GUI

The classification results also can be viewed in XML format.

5.5 Experimental Results

In the validation process for the proposed audio classification data 2 stages are constructed. For the system that includes the multiclass classification method, a hold-out validation has been used. In holdout validation, observations are chosen randomly from the initial sample to form the validation data, and the remaining observations are retained as the training data. In the silence segment detection part, there is no training stage; a simple threshold comparison method has been used in this stage. So, for testing the silence segment detection method, we have used data that includes silence parts and other audio types such as, music, speech, and crowd. The performance of the silence segment detection method is tested by using these data.

In order to test the multiclass classification system that segments audio in to acoustically similar regions; “music”, “speech”, “crowd”, the data that remains after training the system have been used for validation.

Both of the feature sets the validation process has been done. The normalized results for holdout validation system are given in Table 5, Table 6 and Table 7 for the Feature Set 1, Feature Set 2 and Feature Set 3 respectively. In these tables, the element in row 2, column 2 is the percentage of the music data that was indeed classified as music, whereas the element in row 2, column 1 is the percentage of music segments that were classified as speech segments.

Table 5 Normalized Validation results with Feature Set 1

True Class	Classified			
	Speech	Music	Crowd	Unclassified
Speech	%87	%9,5	%0	%3,5
Music	%15,1	%46,2	%29,3	%9,4
Crowd	%5,8	%16,7	%53	%24,5

Table 6 Normalized Validation results with Feature Set 2

True Class	Classified			
	Speech	Music	Crowd	Unclassified
Speech	%82	%10.5	%2.5	%5
Music	%6.6	%54.7	%7.6	%31.1
Crowd	%2.2	%12.1	%79.1	%6.6

Table 7 Normalized Validation results with Feature Set 3

True Class	Classified			
	Speech	Music	Crowd	Unclassified
Speech	%91,5	%6,5	%1	%1
Music	%9,4	%67	%15,1	%8,5
Crowd	%4,9	%22,5	%68,6	%4

The diagonals of Table 5, Table 6 and Table 7 are also the recall R_i of the multi-classification results for each Feature Set, i.e. the proportion of data with true class label i, that were correctly classified in that class. On the other hand, the precision of each class Pri, i = 1. . . 4 (i.e the proportion of data classified in class i, whose true class label is indeed i) is defined as:

$$\mathrm{Pr}_i = \frac{C_{i,i}}{\sum_{j=1}^{4} C_{ji}} \qquad \text{(Eqn 5.11)}$$

where C shows the normalized validation results table.

The recall and precision values of each class are presented in Table 8, Table 9 and Table 10 for the Feature Set 1, Feature Set 2 and Feature Set 3, respectively. The overall classification accuracy (i.e. the percentage of the data that were correctly

classified) of the proposed method is % 62 for the Feature Set 1, % 72 for the Feature Set 2 and % 76 for the Feature Set 3. These results show that MPEG-7 audio features that are considered in this study are not enough for audio classification purpose. The MFCC, Spectrogram, Chroma, ZCR and Spectral Roll-Off features gives more information for the audio classification purpose. In fact, these results are not surprising since the MPEG-7 features are calculated on a logarithmic frequency scale and their dimensions are reduced. So, some information loss can take place.

The AudioSpectrumFlatness feature of MPEG-7 can be thought as the dimension reduced analogue of the MFCC feature. The MFCC features are based on a mel frequency scale, that is based on the nonlinear human perception of the frequencies of audio signals. The results show that, MFCC feature is an effective feature for the audio classification purpose.

The MPEG-7 AudioSpectrumBasis and AudioSpectrumProjection features can be compared with the Spectrogram feature used in the Feature Set 2 and Feature Set 3 since they all are based on the time-frequency analysis. The test results show that using Spectrogram directly, a better classification system can be achieved. This is because of the dimension reduction done in MPEG-7 feature sets.

The chroma feature (cyclic attribute of fundamental frequency) gives a good separation result for the music-speech classification.

Table 8 Recall and Precision per Class for Feature Set 1

	Speech	Music	Crowd
Recall	%87	%46,2	%53
Precision	%80,6	%63,8	%64,38

Table 9 Recall and Precision per Class for Feature Set 2

	Speech	Music	Crowd
Recall	%82	%54.7	%79.1
Precision	%90.3	% 70.8	%88.7

Table 10 Recall and Precision per Class for Feature Set 3

	Speech	Music	Crowd
Recall	%91,5	%67	%68,6
Precision	%86,48	%69	%81

For the silence detection process, we have tested the signals that contain silence segments and purely contain no silence segment. We have used 4 file, total duration of the files are 91 minutes. The files that we have used for the silence region test are “NAD30.mpg”, “NAD31.mpg”, “NAD32.mpg” and “lanc.mpg” that first three of them belongs to TRECVID 2001 dataset, and the last one belongs to MPEG-7 Content Set (1998) [52].

The test results show that proposed method gives good results for the silence detection. The proposed method finds all the silence segment regions that have 2 seconds or greater length. However, if the length of the silence segment region is not multiple of 2 seconds, then it can miss some portion of the silence segment.

CHAPTER 6

6 APPLICATION SCENARIO

The audio analysis part can be combined with the visual analysis results to get more meaningful segments. In this chapter, we present an application scenario that combines the shot detection results with the audio segmentation results. For the visual analysis part, IBM Annotation Tool is used for getting automatically detected shot boundaries. In the following paragraphs IBM Annotation Tool is described.

6.1 IBM Annotation Tool

The IBM MPEG-7 Annotation Tool, *VideoAnnEx,* was developed by IBM [43]. It annotates video sequences with MPEG-7 metadata. IBM MPEG-7 annotation tool allows users semi-automatically annotate video sequence with semantic descriptions [42].

IBM MPEG-7 Annotation Tool performs the shot segmentation process automatically. The Shot Segmentation process is based on the multiple timescale differencing of the color histogram [42]. In the I-frames of video sequence RGB color histogram method is used while in the P-frame of the video sequence motion histogram method is used [42]. The annotated video shot segments are described in MPEG-7 descriptions and are stored in XML file. The Video Shot intervals and Video Shot key frames for all annotated video shot are given in the output XML file. The tool gives opportunities insert a key word in to annotated video shots, but in this study we used the automated case only.

The general system components of the IBM MPEG-7 Annotation Tool are shown in Figure 33. The input of the *VideoAnnEx* is a MPEG-1 video file. When you load the MPEG-1 video file, it starts shot detection automatically. User can save or load shot segmentation information by using MPEG-7 XML file. An example of MPEG-7 shot segmentation file is shown Figure 34.

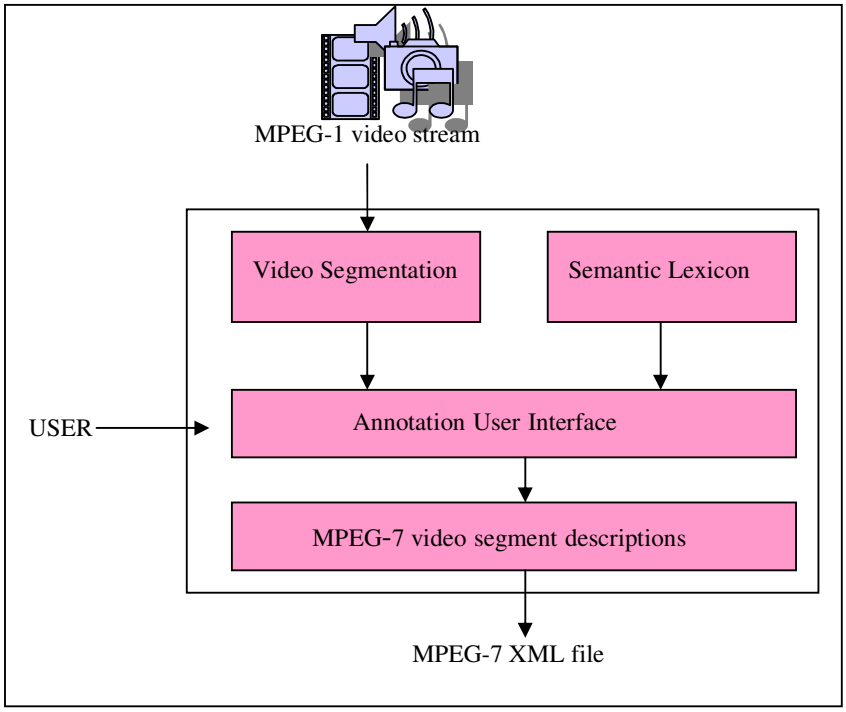

Figure 33 Major components of IBM MPEG-7 Annotation Tool (Modified [42])

Both < TemporalDecomposition > and < MediaTime > compose < VideoSegment >. The element < TemporalDecomposition > is composed of a <MediaTime> element again, that includes the time point information for the keyframe of that video segment (shot). The element < mediaTime > specifies the starting time point and the time duration of the shot.

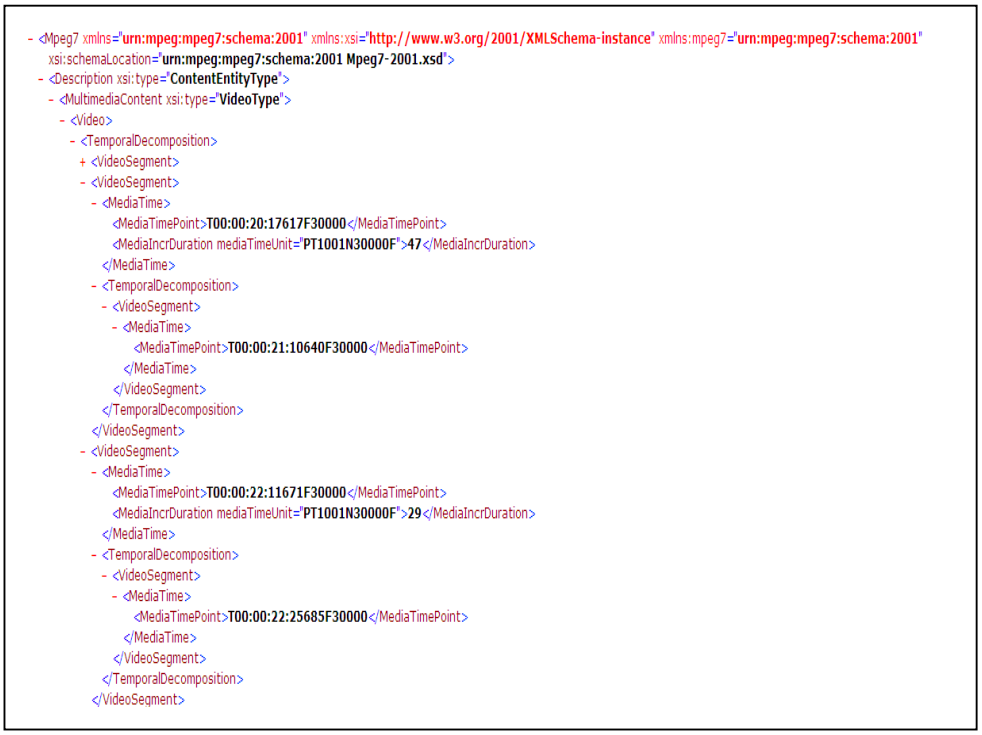

Figure 34. An example of XML output file

A snapshot of *VideoAnnEx* is shown in Figure 35. In this example a news video was used. The frames of the shot that two people are talking in a studio are shown.

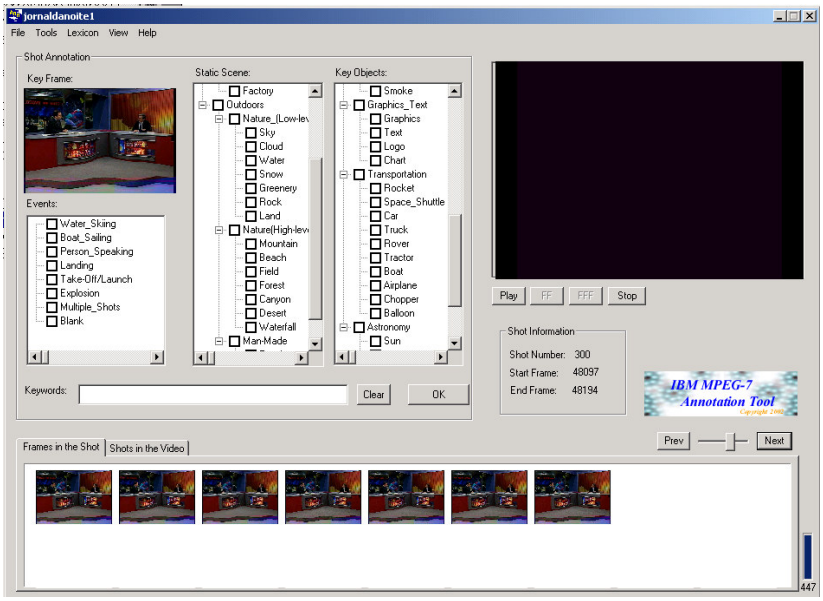

Figure 35. A Snapshots of IBM Mpeg-7 Annotation Tool

6.2 Combining Audio-Visual Analysis

In order to get audio-visual segments for the video sequence we have combined the audio segments and video segment in two ways: Audio Priority Type, Video Priority Type. In both combination methods we have the annotated descriptions that are associated with every audio-visual segment, pulled out and stored as MPEG-7 descriptions in an XML file.

The XML file begins with a root element that signifies whether the description is complete or partial. The following segment entity tools are the main description schemes that are used in this study: *AudioVisualSegment, AudioSegment, VisualSegment.*

The element <AudioVisual> indicates the whole video sequence. It consists of the <MediaTime> element that gives time information of the whole video sequence and <TemporalDecomposion> element that includes the temporal information of the

video sequence. The <TemporalDecomposion> element consists of the all <AudioVisualSegment> descriptions of the video sequence. The difference in Audio Priority Type and Video Priority Type begins in here. The duration of the <AudioVisualSegment> is equal the duration of audio segment in the audio priority type. All the <AudioVisualSegment> elements contain one <AudioSegment> element and one or more <VideoSegment> elements that their time interval remains in the time interval of the <AudioSegment> elements. An example of the output XML file in the case of Audio Priority part is given in Figure 36.

The <AudioSegment> and <VideoSegment> elements are the smallest region descriptor elements. The <AudioSegment> element includes the duration of the audio segment in <MediaTime> element and the label of the segment in the <TextAnnotation > elements. The <VideoSegment> element includes the duration of the audio segment in <MediaTime> element and the time information of the keyframe for the video segment in the <StillRegion > elements.

Figure 36. An example of XML output file for the Audio Priority Type

In the Video Priority part, the duration of the <AudioVisualSegment> is equal the duration of <VideoSegment>. All the <AudioVisualSegment> elements contain one <VideoSegment> element and one or more <AudioSegment> elements that their time interval remains in the time interval of the <VideoSegment> elements. An example output XML file in the case Video Priority part is given in Figure 37.

Figure 37. An example of XML output file for the Video Priority Type

duration of the audio visual segment is equal the duration of video segments. The audio segments that are in the duration of the video segment are written via each includes its own audio label and audio duration.

6.3 Results and Discussion

In this chapter the results of the fusion of the audio classification system and video classification system are discussed.

For the video lanc1.mpg, (supplied by the Lancaster University for the MPEG-7 Call for Test Material [52]) the results for the fusion of the audio segmentation part and the video segmentation part examined. A snapshot of the Audio Annotation and Segmentation tool is shown in Figure 38.

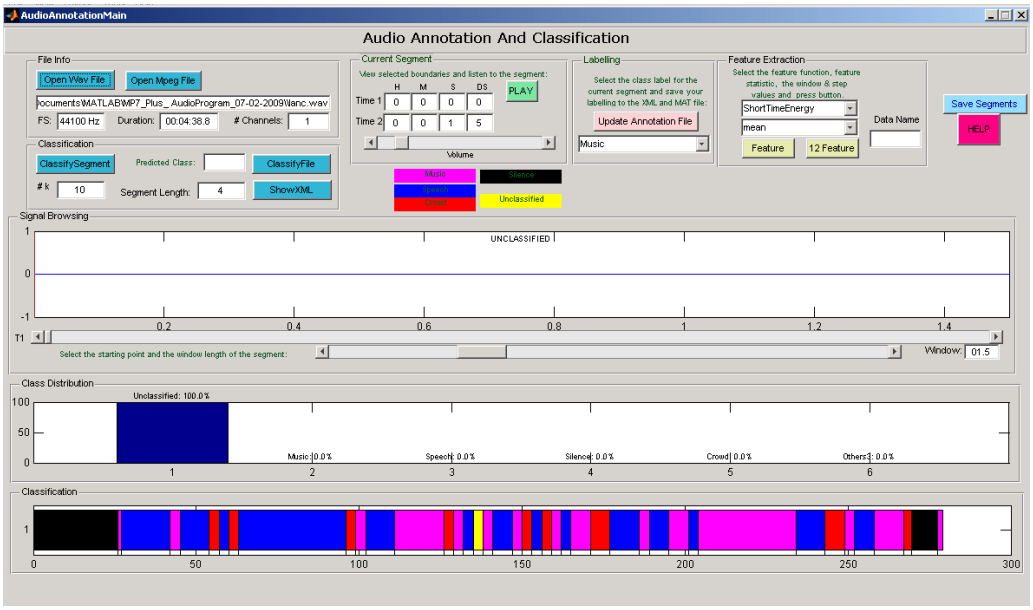

Figure 38. Snapshot of Audio Annotation and Classification Tool for the lanc.mpg video.

The silence segments are almost same, for the real video file. At the time 00:03:24 a music segment begins, its duration is 30 second. This portion of the XML file, for the audio annotation is shown in Figure 39. The output of the IBM Annotation Tool for the same time intervals is given in Figure 40. In these files we see that while audio annotation tool classify this segment as a one region, IBM annotation tool

divides this section into 5 regions (shot 48-49-50-51-52). A snapshot of the IBM annotation tool with shot information of this video is shown in Figure 41. After the fusion of two results for the audio priority type, we get one audio-visual segment that includes one audio segment and five video segments.

```
- <AudioSegment>
  - <TextAnnotation>
      <FreeTextAnnotation>Speech</FreeTextAnnotation>
    </TextAnnotation>
  - <MediaTime>
      <MediaTimePoint>201.0</MediaTimePoint>
      <MediaDuration>3.0</MediaDuration>
    </MediaTime>
  </AudioSegment>
- <AudioSegment>
  - <TextAnnotation>
      <FreeTextAnnotation>Music</FreeTextAnnotation>
    </TextAnnotation>
  - <MediaTime>
      <MediaTimePoint>204.0</MediaTimePoint>
      <MediaDuration>30.0</MediaDuration>
    </MediaTime>
  </AudioSegment>
- <AudioSegment>
  - <TextAnnotation>
      <FreeTextAnnotation>Speech</FreeTextAnnotation>
    </TextAnnotation>
  - <MediaTime>
      <MediaTimePoint>234.0</MediaTimePoint>
      <MediaDuration>9.0</MediaDuration>
    </MediaTime>
  </AudioSegment>
```

Figure 39. XML output of the Audio Annotation and Segmentation Tool for lanc.mpg video.

Figure 40. XML output of the IBM Annotation Tool for lanc.mpg video[*]

[*] The output of the IBM annotaion tool is converted in the means of time formant to better comparasion with the output of the Audio Annotation and Classificaiton tool.

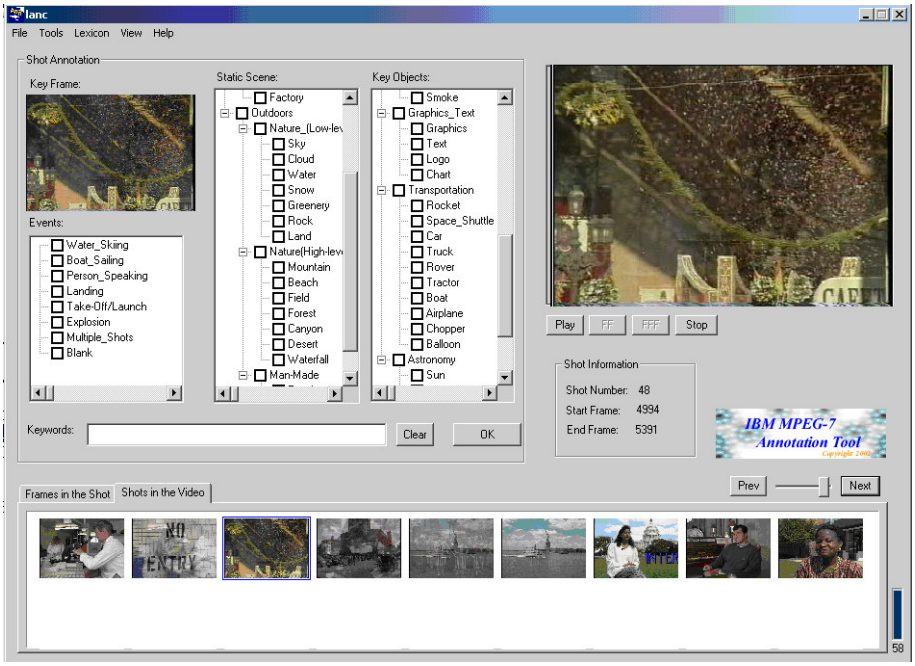

Figure 41. A snapshot of IBM Annotation Tool for lanc.mpg video.

```
- <AudioVisualSegment>
  - <MediaTime>
      <MediaTimePoint>204.0</MediaTimePoint>
      <MediaDuration>30.0</MediaDuration>
    </MediaTime>
  - <MediaSourceDecomposition>
    + <AudioSegment>
    + <VideoSegment>
    + <VideoSegment>
    + <VideoSegment>
    + <VideoSegment>
    + <VideoSegment>
    </MediaSourceDecomposition>
  </AudioVisualSegment>
```

Figure 42. XML file of the fused information.

CHAPTER 7

7 CONCLUSION AND FUTURE WORK

7.1 Conclusion

In this thesis we propose a video segmentation algorithm based on audio segmentation by using audio features of the video sequence.

In the Audio Segmentation we proposed two methods for the segmentation of the audio sequence into “speech”, “music”, “crowd”, “silence” and “unclassified” regions. The first method is detection of the silence segments. For this purpose a simple threshold comparison method by using the short time energy feature of the audio signal is used. The test results show that short-time energy feature is a good separator for the silence segments from the non-silence segments. But for the signals that contain silence segments, it can miss some portions since our search region is 2 seconds. If the duration of a silence segment is not the multiple of 2 seconds, then it can miss some portion of the silence segment.

In the second stage a multi-class audio classification system for segmenting audio into “speech”, “music” and “crowd” regions are used. The regions that are not belong to this regions are left as unclassified. In total, five audio classes were adopted in the audio classification system.

The feature vectors were formed from the various feature sets. A simple histogram comparison method was applied on the feature vectors. Best features were chosen via the histogram comparison results for the each class. Three feature set were formed: Feature Set 1 was purely formed from the MPEG-7 low level audio descriptors, Feature Set 2 is the audio features that are used in [31] and the last one

Feature Set 3 is the combination of Feature 1 and Feature Set 2 . However the Feature Set 2 includes the features that are formed as the combination of some MPEG-7 features and other audio features described in Chapter 4, we formed another feature set, Feature Set 3. The features and statistics of these features are chosen by analyzing the histogram plots. The most separable statistic is chosen as the final statistic for the specified feature.

The proposed scheme was tested using the audio samples that remain after the training stage. The audio classification system was tested for the three feature sets. This results show that for this type audio classification scheme the audio features that are not belong to MPEG-7 standard are very helpful for audio classification scheme. The MPEG-7 standard features are not enough, by using the other features their performance can be improved. In fact, these results are not surprising since the dimensions of the MPEG-7 features are reduced. So, some information loss can take place.

The MPEG-7 AudioSpectrumBasis and AudioSpectrumProjection features can be compared with the Spectrogram feature used in the Feature Set 2 and Feature Set 3 since they all are based on the time-frequency analysis. The test results show that using Spectrogram directly, a better classification system can be achieved. This is because of the dimension reduction done in MPEG-7 feature sets.

The AudioSpectrumFlatness feature of MPEG-7 can be thought as the dimension reduced analogue of the MFCC feature. The MFCC features are based on a mel frequency scale, that is based on the nonlinear human perception of the frequencies of audio signals. The results show that, MFCC feature is an effective feature for the audio classification purpose.

The differences between the MPEG-7 features and MFCC and Spectrogram can be summarized as follow:

- The Mel-frequency scale is used for MFCC has been shown to be better than the logarithmic scale for audio classification.

- Spectrogram feature use a linear frequency scale, it includes more information from the logarithmic frequency scale.

- The dimension of the MPEG-7 feature ASP, represents low-dimensional features of a spectrum after projection upon a reduced rank basis. However, the dimension of the spectrogram feature is not reduced.

- During training, the extraction of the MPEG-7 audio features requires more memory than extraction of MFCC and Spectrogram.

The test results and histogram comparison methods shows that the chroma feature (cyclic attribute of fundamental frequency) gives a good separation results for the music-speech classification.

The Spectrall Roll-off and ZCR features also give better results for audio classification.

The output of system is an XML file which contains MPEG-7 descriptors for audio segments such as duration of the segment, label of the segment. After the audio segmentation process, an application scenario is given by combining the results of the audio segmented files with the shot boundary information. By this combination operation we get more meaningful segment boundaries. Two methods were used for the combination operation. In the audio priority type we combine or split the video shot boundaries, in the video priority type we combine or split the audio segments. The audio priority method gives more meaningful segments from the video priority type.

The suggested system can be used in video indexing systems, for serving the queries like:

- “Bring the video segments where a group of people are present” that is the audio-visual segments where it contains “crowd” label for the audio segment.

- “Bring the video segments where a person is talking” that is the audio-visual segments where it contains “speech” label for the audio segment.

However this system is run on the video sequences and used for video segmentation, system can also be used for only audio signals for the audio indexing and retrieving systems too.

7.2 Future Work

In the audio segmentation, we chose the length of search region for the segmentation. As a future work, by using a powerful feature, segment boundaries of the audio files can be detected automatically. New feature sets could be examined and used for the audio classification part, in order to achieve an improved performance of the audio classification task.

On the other hand, more classes could be added in the classification problem, in order to have a more detailed.

A speaker change algorithm can be used after extracting the speech segments. It will be helpful especially for the news videos to detect the anchorperson speech or a different speech is taken place.

Finally, the audio classification system could be combined with visual cues for increased video segmentation process.

REFERENCES

[1] E. Esen, Y. Yasaroglu, O. Onur, M. Soysal, S. Tekinalp, and A.A. Alatan, "A Mpeg-7 compliant Video Management System: BilVMS", *WIAMIS 2003, London, UK.*

[2] Özgür Ulusoy, Uğur Güdükbay, Mehmet Emin Dönderler, Ediz Şaykol and Cemil Alper "BilVideo Video Database Management System" *30th VLDB Conference, Toronto, Canada, 2004*

[3] Rune Hjelsvold, Roger Midstraum, and Olav Sandsta "Searching and Browsing a Shared Video Database" *Norwegian Institute of Technology*

[4] MUVIS project http://muvis.cs.tut.fi/ last visited January 2009

[5] Digital Video and Multimedia Group (DVMM). http://www.ee.columbia.edu/ln/dvmm/newHome.htm last visited January 2009

[6] M. A. Bartsch and G. H. Wakefield, "Audio thumbnailing of popular music using chroma-based representations," *IEEE Trans. Multimedia, vol. 7, no. 1, pp. 96–104, Feb. 2005.*

[7] Sonera Medialab: MPEG-7 White Paper, *http://www.medialab.sonera.fi/workspace/MPEG7WhitePaper.pdf, TeliaSonera Finland.; last visited January 2009*

[8] A. B. Benitez, S. Paek, S.-F. Chang, Q. Huang, A. Puri, C.-S. Li, J. R. Smith, L. D. Bergman, C. Judice, "Object-Based Multimedia Description Schemes and Applications for MPEG-7", *Image Communications Journal, Special Issue on MPEG-7, 2000.*

[9] Ana B. Benitez, J. M. Martinez, Hawley Rising and Philippe Salembier "Description of a Single Multimedia Document" in "Introduction to MPEG-7: Multimedia Content Description Language", B.S. Manjunath, P. Salembier, T. Sikora (eds.), John Wiley & Sons, Ltd., pp. 111-138, 2002.

[10] Ana B. Benitez, Shih-Fu Chang, John R. Smith. IMKA: A Multimedia Organization System Combining Perceptual and Semantic Knowledge. In ACM Multimedia, Ottawa, Cananda, October 2001

[11] J. M. Martnez. *"MPEG-7 Overview [version 10]"*. ISO/IEC JTC1/SC29/WG11 N6828 (October 2004). http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm; *last visited January 2009*

[12] Rowe L A Boreczky J S. "Comparison of video shot boundary detection techniques. Proceedings" - *SPIE The International Society for Optical Engineering, Storage and Retrieval for Still Image and Video Databases IV(issue 2670):170-179, 1996*

[13] Thomas F. Quatieri "Discrete- Time Speech Signal Processing Principles and Practice" Massachusetts Institute of Technology Lincoln Laboratory, (Prentice-Hall signal processing series)

[14] Alatan, A.A., "Automatic Multi-Modal Dialogue Scene Indexing". IEEE International Conference on Image Processing , (2001), On page(s): 374-377 vol.3

[15] F. Plante, G. Meyer, and W. A. Ainsworth "Improvement of Speech Spectrogram Accuracy by the Method of Reassignment" IEEE Transactions On Speech And Audio Processing, vol. 6, no. 3, May 1998

[16] Michael D. Riley, "Speech Time-Frequency Representations" Copyright at 1989 by Kluwer Academic Publishers

[17] Ryan Rifkin, Aldebaro Klautau ”In Defense of One-Vs-All Classifcation”, in The Journal of Machine Learning Research, Volume 5, Pages: 101 -141, December 2004

[18] Mai K Zabih R, Miller J. “A feature-based algorithm for detecting and classifying scene breaks.” ACM Multimedia International Conference, (issue 3):189-200, 1995.

[19] Smoliar S W Zhang H J, Kankanhalli “A. Automatic partitioning of full-motion video.” ACM Multimedia Systems, volume 1(issue 1):10-28, 1993.

[20] Zhu Liu, David Gibbon, Eric Zavesky, Behzad Shahraray, Patrick Haffner “A Fast, Comprehensive Shot Boundary Determination System” Multimedia and Expo, 2007 IEEE International Conference on Volume , Issue , 2-5 July 2007 Page(s):1487 - 1490

[21] Bhattacharjee S K Cabedo X U. “Shot detection tools in digital video” Non-linear model based image analysis, pages 231-238, 1998.

[22] Clifton Forlines, “Content Aware Video Presentation on High-Resolution Displays” Mitsubishi Electric Research Labs, AVI'08, 28-30 May , 2008, Napoli, Italy

[23] Zuzana Cernekova, Ioannis Pitas, “Information Theory-Based Shot Cut/Fade Detection and Video Summarization” Recommended by Associate Editor R. Lienhart., published in November 3, 2004.

[24] Yeung, M. M. and Yeo, B. 1996. “Time-Constrained Clustering for Segmentation of Video into Story Unites”. In Proceedings of the international Conference on Pattern Recognition (ICPR '96) Volume Iii-Volume 7276 - Volume 7276 (August 25 - 29, 1996). ICPR. IEEE Computer Society, Washington, DC, 375.

[25] Jongmok Son, Jinwoong Kim, Kyungok Kang, Keunsung Bae, “Application of Speech Recognition with Closed Caption for Content-Based Video Segmentation”

The Broadcasting Technology Department , Electronics and Telecommunications Research Institute in Korea

[26] John S. Boreczky and Lynn D. Wilcox, "A Hidden Markov Model Framework for Video Segmentation Using Audio and Image Features," Int. Conf. on Acoustics, Seech, and Signal Processing, vol. & , pp. 3741-3744, 1998.

[27] Hao Jiang, Tong Lin and Hongjiang Zhang "Video Segmentation With The Support Of Audio Segmentation And Classification", Microsoft Research, China

[28] David Pye, Nicholas J. Hollinghurst, Timothy J. Mills and Kenneth R. Wood "Audio-Visual Segmentation for Content-Based Retrieval" In ICSLP-1998, paper 0517.

[29] U. Iurgel, R. Meermeier, S. Eickeler, G. Rigoll "New Approaches To Audio-Visual Segmentation Of TV News For Automatic Topic Retrieval" in IEEE Int. Conference on Acoustics, Speech, and Signal Processing (ICASSP), Salt Lake City, in 2001

[30] Lu, X., Ma, Y.-F., Zhang, H.-J. and Wu, L., "An Integrated Correlation Measure for Semantic Video Segmentation." In Proceedings of IEEE International Conference on Multimedia and Expo – ICME '02, (Lausanne, Switzerland, 2002), 57- 60.

[31] Theodoros Giannakopoulos, Aggelos Pikrakis and Sergios Theodoridis, "A Multi-Class Audio Classification Method With Respect To Violent Content In Movies Using Bayesian Networks" 2007 IEEE International Workshop on Multimedia Signal Processing, Chania, Crete, Greece, October 1-3, 2007

[32] Fabien Gouyon, François Pachet, Olivier Delerue "On The Use Of Zero-Crossing Rate For An Application Of Classification Of Percussive Sounds" Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00), Verona, Italy, December 7-9, 2000

[33] R. N. Shepard, “Circularity in judgements of relative pitch,” J. Acoust. Soc. Amer., vol. 36, pp. 2346–2353, 1964.

[34] L.R.Rabiner,”On the use of autocorrelation analysis for pitch detection” IEEE Trans. Acoust., Speech & Signal Process., vol.ASSP-25, no.l,pp.24-33, Feb.1977.

[35] M. M. Sondhi “New Methods of Pitch Extraction” IEEE Trans. Audio Electroacoust. (Special Issue on Speech Communication and Processing-Part II, AU-16:262-266, June 1968.

[36] Logan, B. & Chu, S., "Music summarization using key phrases" in Proc.ICASSP, 2000.

[37] XML.COM Web Site, “A Technical Introduction to XML”, http://www.xml.com/pub/a/98/10/guide0.html?page=1, last visited January 2009

[38] Richard O. Duda, Peter E. Hart, David G. Stork, “Pattern Classification”, Second Edition, Publisher: Wiley-Interscience, Location: New York

[39] D. Heckerman, “A Tutorial on Learning With Bayesian Networks”, Microsoft Research, MSR-TR-95-06, Mar. 1995

[40] A. Garg, V. Pavlovic and T.S. Huang, “Bayesian Networks as Ensemble of Classifiers”, Proceedings of the IEEE International Conference on Pattern Recognition, pp. 779-784, Quebec City, Canada, August 2002.

[41] Domingos, P., & Pazzani, M. (1997). “On the optimality of the simple Bayesian classifier under zero-one loss.” Machine Learning, 29, 103–130.

[42] C.-Y. Lin, B. Tseng and J. Smith, “VideoAnnEx: IBM MPEG-7 Annotation Tool for Multimedia Indexing and Concept Learning,” Proc. of ICME, Baltimore, Jul. 2003.

[43] C.-Y. Lin, B. L. Tseng and J. R. Smith, "IBM MPEG-7Annotation Tool," IBM Alphaworks, http://alphaworks.ibm.com/tech/videoannex, last visited January 2009

[44] Noris Mohd Norowi, Shyamala Doraisamy, Rahmita Wirza "Factors Affecting Automatic Genre Classification: An Investigation Incorporating Non-Western Musical Forms", ISMIR 2005

[45] Jinhui Yuan, Huiyi Wang, Lan Xiao, Wujie Zheng, Jianmin Li, Fuzong Lin, and Bo Zhang "A Formal Study of Shot Boundary Detection" , 2007, IEEE Transactions On Circuits And Systems For Video Technology, Vol. 17, No. 2,

[46] Open Video Project, http://www.open-video.org last visited January 2009

[47] "Network of Excellence in Content-Based Semantic Scene Analysis and Information Retrieval" Information Society Technologies (Ist) Programme , Project Report

[48] R. J. Qian, M. I. Sezan and K. E. Matthews, "A Robust Real-Time Face Tracking Algorithm", *Proc. International Conference on Image Processing, pp. 131-135, 1998.*

[49] Casey, M. A.: General sound similarity and soundRecognition Tools, in Introduction to MPEG-7: Willey, April 2002M.

[50] M. Casey, "Reduced-rank spectra and entropic priors as consistent and reliable cues for general sound recognition," *Proceeding of the Workshop 011 Consisretit & Reliable Acoustic Cues for Sound Analysis, 2001*

[51] TRECVID, TREC Video Retrieval Evaluation, http://www-nlpir.nist.gov/projects/trecvid/ , *last visited January 2009*

[52] MPEG Requirements Group. "Licensing agreement for MPEG-7 content set". Doc. ISO/MPEG N2466. MPEG Atlantic City Meeting. 1998.

[53] Steven F. Boll, "Suppression of Acoustic Noise in Speech Using Spectral Subtraction" *IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSp-27, No.2 April 1979*