International Journal of Electronics and Electical Engineering

Volume 1 | Issue 1 Article 6

July 2012

Audio Retrieval Using Multiple Feature Vectors

Vaishali Nandedkar
Computer Dept., JSPM , Pune , India, Vaishu111@yahoo.com

Follow this and additional works at: https://www.interscience.in/ijeee

 Part of the Power and Energy Commons

Recommended Citation

Nandedkar, Vaishali (2012) "Audio Retrieval Using Multiple Feature Vectors," *International Journal of Electronics and Electical Engineering*: Vol. 1 : Iss. 1 , Article 6.
DOI: 10.47893/IJEEE.2012.1005
Available at: https://www.interscience.in/ijeee/vol1/iss1/6

This Article is brought to you for free and open access by the Interscience Journals at Interscience Research Network. It has been accepted for inclusion in International Journal of Electronics and Electical Engineering by an authorized editor of Interscience Research Network. For more information, please contact sritampatnaik@gmail.com.

Audio Retrieval Using Multiple Feature Vectors

Vaishali Nandedkar
Computer Dept., JSPM , Pune , India
E-mail- Vaishu111@yahoo.com

Abstract— ***Content Based Audio Retrieval system is very helpful to facilitate users to find the target audio materials. Audio signals are classified into speech, music, several types of environmental sounds and silence based on audio content analysis. The extracted audio features include temporal curves of the average zero-crossing rate, the spectral Centroid, the spectral flux, as well as spectral roll-off of these curves. In this dissertation we have used the four features for extracting the audio from the database, use of this multiple features increase the accuracy of the audio file which we are retrieving from the audio database.***

Keywords - Content Based Audio Retrieval, Feature Extraction, Classification based on Features, Content Based Retrieval.

I. INTRODUCTION

Audio, which includes voice, music, and various kinds of environmental sounds, is an important type of media, and also a significant part of audiovisual data. As there are more and more digital audio databases in place at present, people start to realize the importance of audio database management relying on audio content analysis. There are also distributed audio libraries in the World Wide Web, and content-based audio retrieval could be an ideal approach for sound indexing and search.

Existing research on content-based audio data management is very limited. There are in general two directions. One direction is audio segmentation and classification, where audio was classified into "music", speech", and "others". The second direction is audio **re**trieval, where audio is retrieval from the huge database using different audio features.

With the development of multimedia technologies, huge amount of multimedia information is transmitted and stored every day. Audio and video data from radio, television, databases, or on the Internet has been a source of recent research interest.

II. OVERVIEW OF AUDIO RETREIVAL

An easy way to comply with the conference paper formatting requirements is to use this document as a template and simply type your text into it.

The basic operation of the retrieval system is as follows. First, the feature vectors are estimated for both, the example signal from the user, and for the database signal[2]. One by one each database signal is compared to the example signal. If similarity criterion is fulfilled, the database sample is retrieved[1].

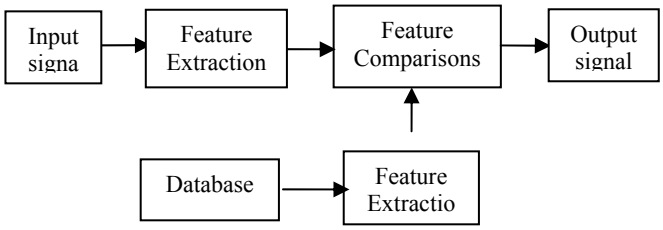

Figure 1.1 Block diagram for the Content Based Audio Retrieval

As shown in the figure 1.1, next step is comparing two samples, by suing different methods [3]. For example, signal is divided into frames and feature vector is calculated for each frame. Both, features which describe the frequency content of the signals, and features which describe the temporal characteristics were used in the system. Features are normalized over the whole database with zero mean and unity variance [5]. After these calculations, same method is applied on the input audio signal and then one by one feature values are access from the database and compare with the input signal value. When the similarity is found between the input and database signals then that signal we can say that require signal is found from the database, these signals need to be display as an out of the Content Based Audio Retrieval[4].

III. AUDIO RETRIVAL USING MULTIPAL FEATURES

All Feature extraction is the process of converting an audio signal into a sequence of feature vectors carrying 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 [6]. 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 [6]. These features can be divided into two categories: time domain and frequency domain features.

The temporal domain is the native domain for audio signals [2]. All temporal features have in common that they are extracted from the raw audio signal, without any preceding transformation.

A .Types of Audio Features

1) The Temporal Features –

2) Cepstral features

Cepstral features are frequency smoothed representation of the log magnitude spectrum and capture timber characteristics and pitch [2]. Cepstral features allow for application of the Euclidean metric as distance measure due to their orthogonal basis which facilitates similarity comparison.

A. Muliple Features

1) Zero-crossing rate: The zero crossing rate counts the number of times that the signal amplitude changes signs in the time domain during one frame of length N,

$$ZCR = 1/2 \sum_{n=1}^{N} |\, sgn(X[n]) - sgn(X[n-1])| \qquad (1)$$

Where the sign function is defined by

$$sgn(x) = \begin{cases} 1, & x >= 0 \\ -1, & x < 0 \end{cases}$$

2) *Spectral Centroid*: Centroid is the gravity of the spectrum [2] Where the sign function is defined by

$$C_r = \frac{\sum_{K=1}^{N/2} f[k]|X_r[k]|}{\sum_{K=1}^{N/2} |X_r[k]|} \qquad (2)$$

Where N is a number of FFT points, X_r [k] is the STFT of frame x_r, and f [k] is a frequency at bin k. Centroid models the sound sharpness. Sharpness is related to the high frequency content of the spectrum.

3) Spectral Roll-off: The roll-off is a measure of spectral shape useful for distinguishing voiced from unvoiced speech. The roll-off is defined as the frequency below which 85% of the magnitude distribution of the spectrum is concentrated [2]. That is, if K is the largest bin that fulfils,

$$\sum_{K=1}^{N/2} |X_r[k]| <= 0.85 \sum_{K=1}^{N/2} |X_r[k]| \qquad (3)$$

Then the roll-off is R_r = f[K][8].

4) Spectral Flux: The spectral flux is defined as the squared difference between the normalized magnitudes of successive spectral distributions that correspond to successive signal frames [5]. That means, a measure of the spectral rate of change, which is given by the sum across one analysis window of the squared difference between the magnitude spectra corresponding to successive signal frames,

$$F_r = \sum_{K=1}^{N/2} (|X_r[k]| - |X_{r-1}[k]|)^2 \qquad (4)$$

Flux has been found to be a suitable feature for the separation of music from speech, yielding higher values for music examples [8].

IV. HISTOGRAM MODELLING

1) Basic Algorithm

Fig. 1 outlines the algorithm. In this paper, a histogram is a frequency distribution of the feature vector occurrences over the window. The frequency distribution is obtained by classifying the feature vectors according to a certain vector. Since the feature vectors are not uniformly distributed in the feature space, feature vector density should be considered in the classification process in order to efficiently represent signals with a histogram.
Histogram is then defined as,

$$h=(h_1,h_2,\ldots h_l,\ldots h_L) \qquad (5)$$

where is the number of histogram bins, the typical window length is the query signal duration.

The reference signal is obtained by dividing the reference sound window into a number of fixed length frames, extracting a feature vector form each frame, and then finding the probability distribution of feature vectors in feature space over this window. A histogram is used as the non-parametric model for this distribution. The same process is applied to a window of test data to obtain a test template. Similarity between the reference and test templates h_R and h_T respectively, is calculated using histogram interaction, where B is the number of histogram bins [8][9]:

$$S(h_R, h_T) = \sum_{i=1}^{B} \min(h^i_R, h^i_T) \qquad (6)$$

The histogram intersection measure is used because it is computationally simple, and it has been used successfully in visual object detection [10]. As the window for the stored signal shifts forward in time [7], similarity based on the query- and stored-signal histograms shows certain continuity from one time step to the next. The time-series active search takes advantage of this by computing an

upper bound of the similarity measure as a function of the time step and skipping all intermediate time-step similarity evaluations until this upper bound exceeds the detection threshold [7].

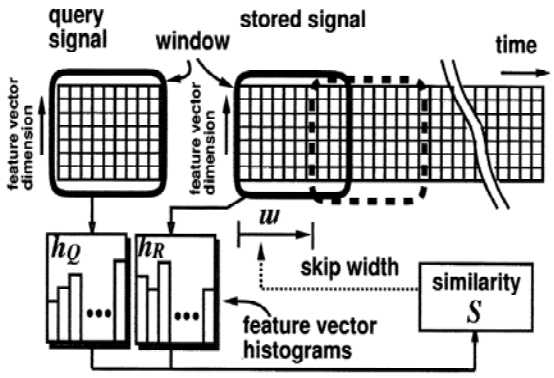

Figure 1.2: Block diagram of the Basic Histogram Modeling

VI. EXPERMENTAL RESULT

In this section, the methods used to implement the system for selective audio signals will be described in details. Also how the experimental results are obtained is described with some comments. The chapter is split into the following sub sections: data description, feature extraction, segmentation and classification.

We conducted an experiment with real music data. Following types of experiments were conducted,

i) Feature extraction for the stored signal and
ii) Vector quantization for the stored signal.
iii) After a query signal is given.
iv) Feature extraction for a query signal.
v) Vector quantization for the query signal and
vi) Matching between the query signal and each section of the stored signal, are performed.
vii) Processing time for calculating the feature vectors.
viii) Histogram Processing

A. Description of the audio data: The audio files used in the experiment were randomly collected from the different Indian songs. The speech audio files were selected from both male and female speakers. The music audio samples were selected from a variety of categories and consist of almost all musical genres. These files were in wav formats to be able to use them in mat lab programs, it was necessary to use these files with a wav format with a common sampling frequency. Gold Wave (open Source) software is used, for creating this database. The recorded audio files were further partitioned into according to their category: rock music, dance, etc.

Table 1: Training Data

Audio type	Number of files	Average length
Speech	20	30 sec.
Music	100	30 sec.

Table 2: Test Data

Audio type	Number of files	Average length
Speech	20	30 sec.
Rock	16	30 sec.

A. Experimental Results Using Multiple Features: As feature extraction has already been mentioned in the previous session. Here, it is focused on how ZCR, Centroid, Roll off, Flux, features are extracted from the row audio data and how they are used in classification and segmentation modules.

1. Zero-Crossing Rate

In order to extract ZCR features from the audio signal, the signal was first partitioned into short overlapping frames each consisting of 512 samples. The overlap size was set to half the size of the frame. The actual features used for classification task were the means of the ZCR taken over a window containing 20 frames. The following examples show result of ZCR feature. The following examples shows result of the speech and music signals,

featureZCR() – Function implemented for ZCR, we need to pass the parameter as an audio file to calculate the ZCR of that audio file.

For example –

1. featureZCR ('D:\Documents\CBAR\project\sound files\aud1.wav')

   Output - 0.0725
2. featureZCR ('D:\Documents\CBAR\project\sound files\aud22.wav')
   Output - 0.0557

4.2.2 Spectral Centroid

In order to extract Centroid features from the audio signal, the actual features used for classification task were the means of the Centroid taken over a window containing 20 frames. The following examples show result of Centroid feature. The following examples shows result of the speech and music signals,

featureSC() – Function implemented for Centroid, we need to pass the parameter as an audio file to calculate the Centroid of that audio file.

For example –

1. featureSC ('D:\Documents\CBAR\project\sound files\aud1.wav')
   Output - 0.1364

2. featureSC ('D:\Documents\CBAR\project\sound files\aud22.wav')
   Output - 0.1591

4.2.3 Spectral Roll Off

The following examples show result of Roll Off feature. The following examples shows result of the speech and music signals,

featureSRO() – Function implemented for Roll Off, we need to pass the parameter as an audio file to calculate the Roll Off of that audio file.

For example –

1. featureSRO ('D:\Documents\CBAR\project\sound files\aud1.wav')

   Output - 0.0028
2. featureSRO ('D:\Documents\CBAR\project\sound files\aud22.wav')
   Output - 0.0028

4.2.4 Spectral Flux

The following examples show result of Flux feature. The following examples shows result of the speech and music signals.

featureSF() – Function implemented for Flux, we need to pass the parameter as an audio file to calculate the Flux of that audio file.

For example –

1. featureSF ('D:\Documents\CBAR\project\sound files\aud1.wav')

   Output - 41.0537
2. featureSF ('D:\Documents\CBAR\project\sound files\aud2.wav')
   Output - 11.1627

Following fig.1, shows the result of accuracy graph for the rock music.

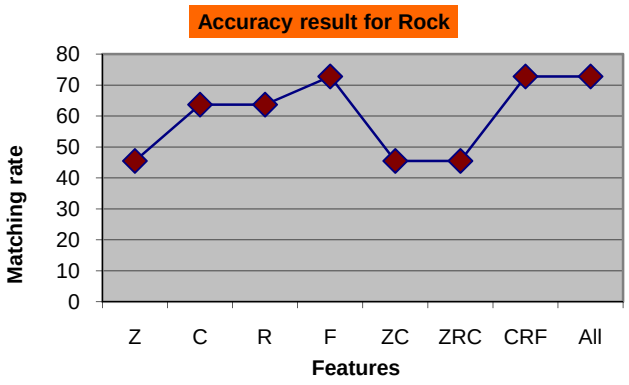

Fig. 1: Accuracy result for rock music

Following fig. 2 show the graphical representation of the processing time for calculating the different feature and their combinations.

Table 3: Processing time for calculating the feature vectors

Features	Processing Time(ms)
ZCR	0.614
Spectral Centroid	1.375
Spectral Flux	1.427
Spectral Roll Off	0.556
ZCR + SC	1.989
ZCR + SF	2.041
ZCR + SRO	1.17
SC + SRO	1.931
SC + SF	2.802
SRO + SF	1.983
SC + SRO + SF	3.358
ZCR + SC + SRO	2.545
ZCR + SRO + SF	2.597
SCR + SC + SF	3.416
All	3.972

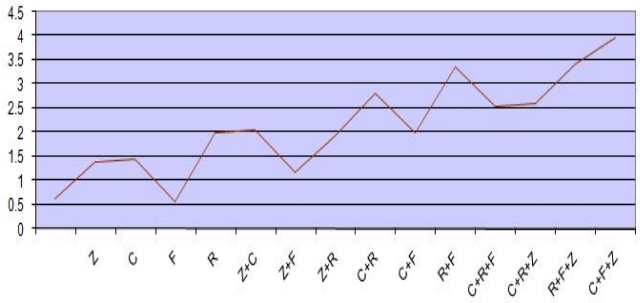

Fig. 4.1: Processing time for calculating the different features

V. CONCLUSIONS

In this paper a quick search method can quickly detect known sound in a long audio stream. Other papers mainly proposed a search method using the single feature. However in this paper multiple combinations of the features are used for more accurate search.

In addition, the preprocessing stage has been introduced for reduce the searching time. As a result of tests, the proposed method is approximately 10% more accurate.

REFERENCES

[1] Ki-Man Kim, Se-Young Kim, Jae-Kuk Jeon, and Kyu-Sik Park, **"**Quick Audio Retrieval Using Multiple Feature Vectors", IEEE Transactions on Consumer Electronics, Vol. 52, No. 1, FEBRUARY 2006.

[2] Dalibor Mitrovic, Matthias Zeppelzauer and Christuan Breitender, "Audio Feature survey – Features for Content-Based Audio Retrieval", Advances in computers Vol.78,pp. 71-10,2010.

[3] Qing Li, Byeong Man Kim, Dong Hai Guan, Duk whan Oh, "A Music Recommender Based on Audio Features", Kumoh National Institute of Technology, 188 Shinpyung-Dong, Kumi, Kyungpook, 730-701, South Korea.

[4] David Gerhard, "Audio Signal Classification: An Overview", School of Computing Science Simon Fraser University Burnaby.

[5] Prof. Preeti Rao and Dr. Sumantra, Hariharan Subramanian. D. Roy Credit, "AUDIO SIGNAL CLASSIFICATION", Seminar Report, Electronic Systems Group, EE. Dept, IIT Bombay, Submitted November2004.

[6] Tong Zhang and C. –C. Jay Kuo, "Content-Based Classification and Retrieval of Audio", Integrated Media Systems Center and Department of Electrical Engineering-Systems, university of Southern California, Los Angeles, CA 90089-2564.

[7] Kunio Kashino, Takayuki Kurozumi, and Hiroshi Murase, "A Quick Search Method for Audio and Video Signals Based on Histogram Pruning", IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 5, NO. 3, SEPTEMBER 2003.

[8] J. Ose Burred and A. Learch, "Hierarichical automatic audio signal classification", Proc of J. Audio Eng. Sec., vol.52, pp. 724-739, July/August 2005.

[9] M. J. Swain and D. H. Ballard, "Color indexing", Int. J. Comput., Vis., vol. 7, no. 1, pp. 11-32, 1991.

[10] L. Rabinar, R. Schafer,"Digital Processing of Speech Signals", Prentice-Hall, Inc., New Jersey, 1978.