Comparison and Implementation of Audio based Searching for Indian Classical Music

Akshat Arora, Nikhil Panwar, Pankaj Sikka

Jaypee Institute of Information Technology, Noida

akshatarora13@gmail.com, nikhilrajput4u@gmail.com, sikka.pankaj@yahoo.co.in

Abstract: *Increase in the amount of digital audio media in recent years, demands for content based retrieval of multimedia. However India classical Music is untouched from this revolution. Hence, this paper presents a comparative study on content based identification and retrieval for Indian Classical Music. The paper focuses on comparative implementation of various features like mfcc, bark scale based features etc for reliable, fast and robust identification of Indian classical music which can be run on the resources provided by today's standard computing platforms. Besides, a new modified feature has also been proposed to minimize the drawbacks (i.e. space and time complexity) of already existing features. The algorithms are tested on Indian Classical Music, which exhibits certain characteristics that make the identification task harder when compared to Western music.*

Keywords: *Content based audio retrieval, Indian classical Music, Audio Fingerprinting, MFCC, FFT*

1. Introduction

The exponential growth of the Internet and compression technologies has made huge amounts of audio data easily available. This, in turn has spurred the growth of multimedia databases spanning in diverse fields such as arts, medicine, music, science, engineering. However currently, retrieval and management of audio data rely mostly on textual information which is added by human/user, which is an

extremely inaccurate task and furthermore, this information is often incomplete or not available at all. Hence to retrieve the information from database is an extremely arduous task thereby hampering the growth of fields such as Indian Classical Music. However search based on actual content rather than on the keywords associated with it can provide the much needed impetus to it. Efficient and accurate automatic music information processing (accessing and retrieval, in particular) will be an extremely important issue in the coming future. This is why automatic systems are required to lighten the job.

The retrieval function used is query by example which aims at automatic retrieval of audio excerpts similar to a user provided audio sample from his/her personal audio database. In the query by example approach the user provides an example of an audio signal and based on that example, similar samples are retrieved from the database. The retrieval is based on some similarity measure between the example and database samples. There are other ways of querying audio data e.g. by audio features like pitch or words/text. Query by example [1] [2] is most convenient for a user and is often also a relatively accurate way of specifying the information needed. Although audio information retrieval has been enjoying a great amount of attention, Indian classical music is yet to be impacted by the influence of computer based automation tools. Indian Classical Music deserves special attention as it is different from western music in many aspects. Western music and instruments are based on equal tempered scale which is not the case with Indian music and instruments. Indian Classical music is based on a scale which is not tempered [3]. Western music divides an octave into the twelve notes of the chromatic scale, Indian Classical music does the same but the notes are not tuned like the notes of the chromatic scale. Another significant difference is that Western Music is polyphonic i.e. it depends on the resonance of multiple musical notes occurring together but Indian Classical music is essentially monophonic.

In this paper, Fingerprinting algorithms have been implemented and tested on a database containing various forms of Indian Classical Music (Dhrupad, Thumri, Dhamar, Ragas etc). An audio fingerprint basically summaries and differentiates each and every audio clip from the others as in the case of human fingerprinting. All the algorithms under consideration are based on frame based features [4][5] i.e. a long audio clip is divided into many frames to catch the short time property, but the features extracted are absolutely different.

2. Algorithm Overview

In spite of the different justifications behind various identification tasks, retrieval methods share certain aspects. As depicted in Fig.1, there are three fundamental processes**:** pre-processing, feature extraction, and matching algorithm. Fig 2 shows the process of fingerprint extraction which is further explained in the next section. Efficient features must be derived for robust identification and retrievals. The feature requirements include discrimination power over huge numbers of other features, invariance to distortions, compactness, and computational simplicity. Robust features must fulfill the above requirements and therefore imply a trade-off between dimensionality reduction and information loss. After the pre-processing is achieved, the feature vectors are calculated for both, i.e. the example signal from the user, and for the database signal. One by one each database signal is compared to the example signal. If similarity criterion is fulfilled, the database sample is retrieved [6]. According to [7], efficient searching methods should be fast, correct and memory efficient. The matching algorithm actually implemented fulfilled most of these conditions.

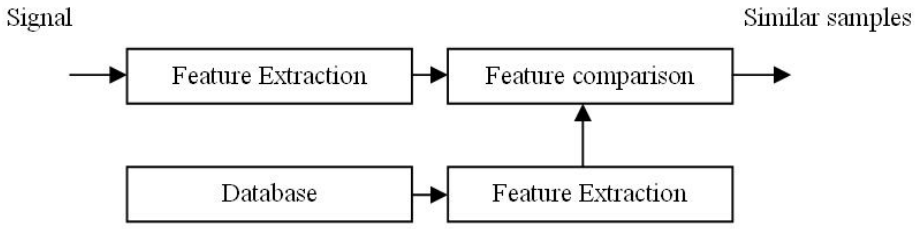


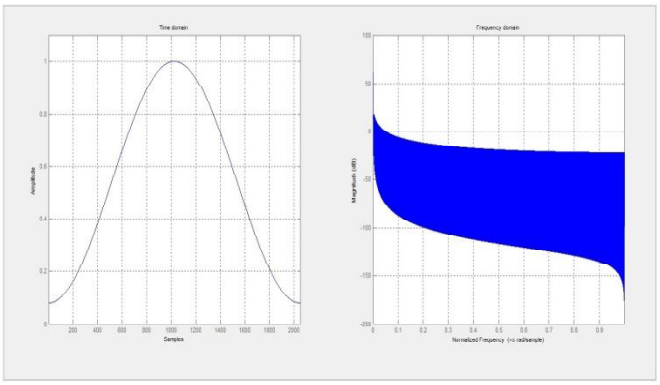
Fig. 1 a block diagram of the overall system

3. Pre- Processing

As shown in Fig 2, the audio needs to be digitalized in this first step. This has been achieved by using DirectX 9.0 library. In some cases, audio was down sampled. But no such changes were done when MFCC and new 4-bit fingerprint were used for matching audio documents and the sampling rate was maintained at default 44.1 KHz. Although, bark scale based features were found to be working better when the audio samples were down sampled to 5 KHz.

4. **Framing and Overlap**

In audio retrieval, frame-based representation is instinctive because audio frame is the basic composition. The audio signal can be regarded as stationary over an interval of a few milliseconds (i.e. quasi-stationary). Hence, short-time spectral analysis of audio signals is the most preferred way to classify them. The signal has been divided into frames of a size equivalent to the variable velocity of the underlying auditory events. This step also includes application of window function to frames. The main aim of applying window function to each individual frame is to minimize the signal discontinuities at the beginning and end of each frame. The concept here is to minimize the spectral distortion by using the window to taper the signal to zero at the beginning and end of each frame. Overlap must be applied to mitigate the loss at the edges of the window. Typically, Hamming window has been used in all the feature extraction algorithms which have the form:



Hamming window time and frequency domain mapping

$$W(n) = 0.53836 - 0.46164 \cos(2\pi n/N-1) \quad (1)$$

N = No. of samples in a frame

n = Sample no. in the frame

Choosing the right frame size and overlap factor is always an important issue as it decides the robustness and time complexity to a large extent. Bark scale based 32 bit fingerprint features were found to give best results when the frame size was 2048 samples and the hop size was as low as 1/32 to ensure accurate results even in worst case scenarios.

The new, modified 4 bit fingerprint feature, used the same frame size i.e. 2048 samples but the overlap factor in this case was reduced to 15/16, thus decreasing the number of frames, and hence improving the time taken in the feature calculation process. MFCC worked efficiently when frame size was reduced to 512 samples with a hop size of 256 frames.

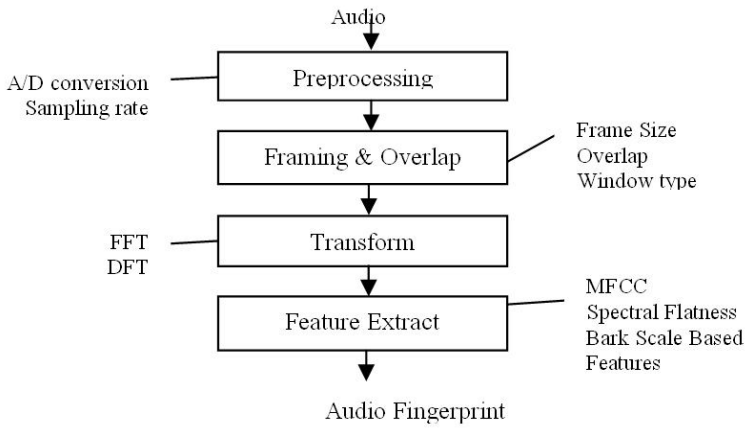


Fig 2: Fingerprint Extraction Framework

5. Spectral Estimates and Feature Extraction

As shown in Fig 2**,** transformations are applied on the time representation of audio document to extract the auditory vectors. The aim is again to reduce the complexity and, at the same time, to increase the resistance to noise.

For the transformation FFT [8] has been computed on the frames for all the algorithms to represent the signal on the time frequency scale. This paper focuses on Audio fingerprinting techniques which function by extracting features of audio files and storing them in database. To calculate robust features, knowledge of the transduction phases of the human auditory system has been incorporated. The acoustic features evaluated in this paper are as follows:

Mell Frequency Cesptral Coefficients

For calculating the MFCC[9] mel-frequency scale is defined, which is a linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz . As a

reference, 1000 Mels is defined as the pitch of a 1 kHz tone, 40 dB above the noncognitive hearing threshold. In order to capture the phonetically important characteristics of audio clip filters spaced linearly at low frequencies and logarithmically at high frequencies have been used Therefore we can use the following approximate formula to compute the mels for a given frequency f in Hz:

$$Mel(f) = 2595 * \log(1 + f/700) \quad (2)$$

Thus for each tone with an actual frequency, f, measured in Hz, a subjective pitch is measured on the mel scale.

Now the log mel spectrum in frequency domain are converted back to time domain which is called the mel frequency cepstrum coefficients (MFCC) as cepstral representation provides a descriptive representation of the local spectral properties of the audio signal for the given frame analysis. Because the Mel spectrum coefficients (and so their logarithm) are real numbers, we can convert them to the time domain using the Inverse Fourier Transformation.

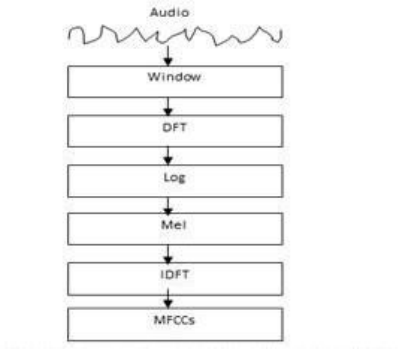


Fig. 4: Audio parameterization into mel frequency cepstral coefficients

Bark Scale based Feature

This feature is based on the energies of 33 bark scaled features as proposed by Haitsma et al [10]. The feature proposed by Haitsma has been modified to reduce its time complexity and make it more accurate (Fig 5). Relevant 32 bit features are computed for every set of frames, which are then referred to, as sub-fingerprint. Finally a fingerprint block is formed which consists of 256 subsequent sub-fingerprints. Based on the Human Auditory System(HAS) , power spectral density of each frame is calculated which is then followed by the selection of 33 non-overlapping frequency

bands lying in the range from 300 to 2000 Hz. The calculation of fingerprints is shown below:

F(n,m)=1

if E(n,m)–E(n,m+1) – (E(n-1,m)–E(n-1,m+1)) > 0

F (n, m) =0

if E(n,m)–E(n,m+1) – (E(n-1,m)– E(n-1,m+1)) < 0 (3)

F (n, m) – m^{th} bit of sub-fingerprint of frame n

E (n, m) – Energy of band m of frame n

The feature, although very promising, used too many resources as it had a high space complexity and time complexity. The process of down sampling the audio excerpts, introduced unnecessary noise and increased the time complexity of the algorithm. At the same time, the files containing features of Indian classical music took as much as 600 KB of space when each audio file was of approximately 10 seconds in duration. Hence, we have proposed a new feature, based on the above feature, which reduces the time and space complexity of algorithm by great extent. A new modified logarithmic scale is used which is based on 5 non-overlapping frequency bands. As a result, just 4 bit sub-fingerprints were obtained for every frame, reducing the size of feature of each file by 95% .

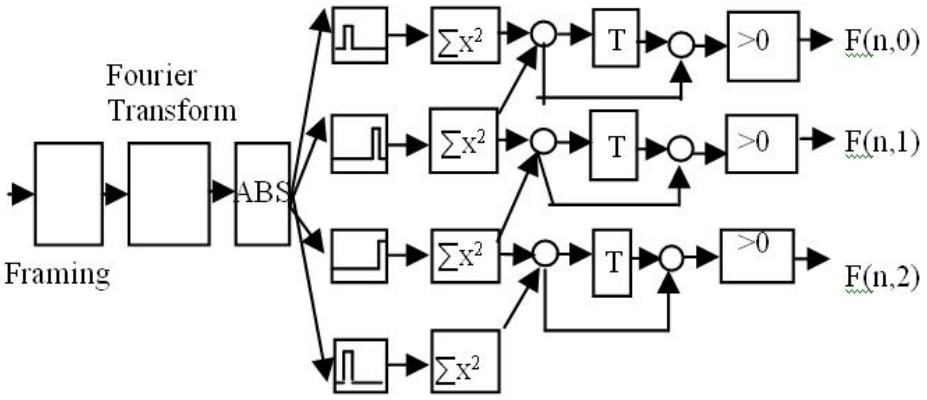


Fig 5: Overview of Bark Scale based Feature

6. Post Processing and Matching Algorithms

A fundamental issue for the usability of a system is how to efficiently do the comparisons of the unknown audio against the possibly million fingerprints. The method depends on the fingerprint representation. In case of 32 bit fingerprints and 4 bit fingerprints, an approach similar to linear search was used. To identify a fingerprint block originating from an unknown audio clip, the most similar fingerprint block in the database was found. In other words, fingerprint blocks were identified in the database where the bit error rate was minimal

(i.e. number of bits not matching) and less than certain threshold value. The clips in database with least bit error rate were displayed (Fig 6) and user had the option of playing those clips.

Similar approach was used in case of MFCC algorithm as well. Here, Euclidean Distance was used instead of bit error rate as the feature values were non-binary.

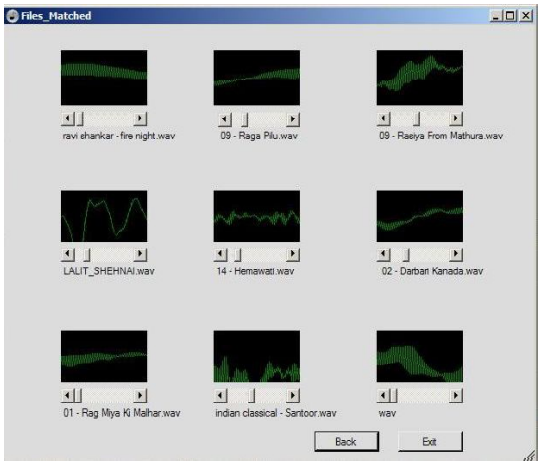


Fig. 6 GUI showing the results

7. Results

The performance of the proposed methods was measured through the following simulations.

The database contained audio signals, each of around 10 seconds (average length) taken from CD recordings and converted to '.wav' files. Audio signals consist of songs and albums of various Indian artistes and have various genres of Indian

classical music. Excerpts of around 3 seconds (average length) were used as the query clips.

Category	MFCC	32 bit Fingerprint	4 bit Fingerprint
Sitar	85	86	86
Flute & Mohan Veena	54	78	83
Santoor & Shenai	99	95	95
Bhairavi & Multani	87	98	79
Bahar & Bhageshwari Bahar	75	95	95
Ahir & Anand Bhairva	75	98	98
Tabla & Asvari	75	96	96
Hemavati	99	99	99
Brindavani Sarang	99	94	94
Yaman	50	98	98
Darbari Kannada	99	92	92
Madhuvanti & Malhar	83	99	67
Bihag & Puriya	99	99	99

The results for MFCC, 32 bit and 4 bit fingerprints using above mentioned database are presented in Table 1.The experimental results of the features on the various genres of Indian classical music have been shown. The number in every cell shows the percentage of successful matching i.e. when a query was given then the signal of which the query was a part was correctly identified and displayed in the top 3 results (Fig 6).

8. Conclusion and future work

In this paper, a novel approach to the query by example of Indian classical music was presented. The results show that the features that have been considered in this paper responded in a different manner to Indian classical music, as expected. MFCC and Bark Scale based features were used in the feature extraction process. The

features have proved to be quiet efficient except in certain cases, showing that useful retrieval can be performed even for complex audio.

The basic problem in query by example using only a single example is the definition of similarity itself. Based on only one example it is difficult even for a human to say what the user means with similarity. Therefore, the future work will consider taking the feedback from a user. When the first query is done, the user could guide the algorithm by telling which retrieved samples were correct or which were not and the system could learn from this feedback. This way the system gains information regarding the user's idea of similarity. A new query could then be done based on this improved knowledge. If the entered query clip is not found in the database then it should be automatically classified among the different genre of Indian classical music based on its features.

9. References

1. Z. Liu, Q. Huang, "Content-based Indexing and Retrieval by-Example in Audio," ICME 2000, Vol. 2, pp:877-880, New York, July 30 - Aug. 2, 2000

2. Velivelli, A.; ChengXiang Zhai; Huang, T.S, " Audio segment retrieval using a short duration example query", ICME 2004,pp: 1603-1606, June 27 - June 30, 2004

3. Catherine Schmidt-Jones, Indian Classical Music: Tuning and Ragas; http://cnx.org/content/m12459/latest/

4. L. Lu, H. J. Zhang and H. Jiang, "Content Analysis for Audio Classification and Segmentation", IEEE Transactions on Speech and Audio Processing, Vol.10, No.7, pp.504-516, Oct. 2002.

5. E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content based Classification, Search, and Retrieval of Audio", IEEE Multimedia, 3(3): 27-36, 1996.

6. P. Cano, E. Batlle, H. Mayer, and H. Neuschmied, "Robust sound modeling for song detection in broadcast audio," in Proc. AES 112th Int. Conv., Munich, Germany, May 2002.

7. R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval. Addison Wesley, 1999

8. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein; Introduction to Algorithms, MIT Press

9. J. Foote. Content-based retrieval of music and audio.In In Multimedia Storage and Archiving Systems II, Proceedings of SPIE, pages 138–147, 1997.

10. Haitsma, Kalker. A Highly Robust Audio Fingerprinting System. Proc. ISMIR, 2002.