Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 5.258

IJCSMC, Vol. 5, Issue. 7, July 2016, pg.337 – 343

Accelerating Content Based Music Retrieval Using Audio Fingerprinting

Mr. Mahesh B. Sonje[1], Prof. S.M. Rokade[2]

[1]Department of Computer Engineering, Savitribai Phule University, India
[2]Department of Computer Engineering, Savitribai Phule University, India
[1] mahesh.sonje75@gmail.com; [2] smrokade@yahoo.com

Abstract— Audio Fingerprinting is a content-based signature to refer from original audio file. Audio Fingerprinting technique is commonly used to monitor the audio without the need of meta-data. Fingerprinting approaches are Pattern matching, Multimedia Information Retrieval and Cryptography. For browsing and identifying the audio content from large audio data collection set is a major strand for research. The best solution for identifying the audio content is Audio ID System in which the specified audio is identified by passing a short query audio clip from a large set of audio data set. As the main contribution, the system establishes the connection between the hash values which are mainly used for indexing and their ability to survive the presence of noise and distortions. This improves the performance of the audio ID system by using following four ways: First, the fingerprints are generated to increase its reliability. Second, by providing the priority to reliable hash values which are basically use during searching for reference entries, it improves the retrieval speed. Third increase the identification of reliable hash values by enlarging the query fingerprint. Fourth indexing only the reliable hash values, this lowers the server side storage requirements.

Keywords— Audio Identification, Content-based retrieval, Music, Indexing.

I. Introduction

Audio fingerprinting is the property to connect unlabelled audio to related metadata (e.g. song name and artist name), with respect to the audio type or format. Following are the applications of audio fingerprinting: Content-based verification or watermarking support, which focuses primarily on identification. Audio fingerprinting extract a part of audio content, i.e. fingerprint and store it in a database file. In case of unlabelled audio, its fingerprint is calculated and matched with the fingerprints which are stored in the database. Using fingerprints and matching algorithms, can be identified as the same audio content. A source of difficulty when automatically identifying audio content derives from its high dimensionality and the significant variance of the audio data for perceptually similar content. The simplest approach that one may think of – the direct comparison of the digitalized waveform - is neither efficient not effective. An efficient implementation of this approach could use a hash method, such as Message Digest-5 and Cyclic Redundancy Checking (CRC), to obtain a compact binary file. In this case one compares the hash values instead of the whole files. However, hash values is a single bit flip which is sufficient for the hash to completely change. Of course this setup is not robust to compression or minimal distortions of any kind and, in fact, it cannot be considered as content-based identification since it does not consider the content, understood as information, just the bits.

An ideal fingerprinting system should be able to accurately identify an item, with respect to the level of compression and distortion. Depending on the application, it should be able to identify whole titles from excerpts a few seconds long, which requires methods for dealing with shifting that is lack of synchronization between the extracted fingerprint and those stored in the database. It should also be able to deal with other sources of degradation such as pitching (playing audio faster or slower), equalization, background noise, D/A-A/D conversion, speech and audio coders (such as GSM or MP3), etc. The fingerprinting system should also be computationally efficient. This is related to the size of the fingerprints, the complexity of the search algorithm and the complexity of the fingerprint extraction. The design principles behind audio fingerprinting are recurrent in several research areas. Compact signatures that represent complex multimedia objects are employed in Information Retrieval for fast indexing and retrieval. In order to index complex multimedia objects it is necessary to reduce their dimensionality and perform the indexing and searching in the reduced space. In analogy to the cryptographic hash value, content-based digital signatures can be seen as evolved versions of hash values that are robust to content-preserving transformations. Also from a pattern matching point of view, the idea of extracting the essence of a class of objects retaining the main its characteristics is at the heart of any classification system.

II. LITERATURE SURVEY

Hendrik Schreiber and Meinard Muller presented various improvements for the audio ID system originally proposed by Haitsma and Kalker. Our main observation was that the probability of finding a matching reference sub-print is elevated in the case that multiple consecutive query sub-prints are identical. Which supports for mildly distorted audio files [1].

J. Haitsma and T. Kalker presented a new approach to audio fingerprinting. The fingerprint extraction is based on extracting a 32 bit sub fingerprint every 11.8 milliseconds. The sub-fingerprints are generated by looking at energy differences along the frequency and the time axes. A fingerprint block, comprising 256 subsequent sub-fingerprints, is the basic unit that is used to identify a song. The fingerprint database contains a two-phase search algorithm that is based on only performing full fingerprint comparisons at candidate positions pre-selected by a sub-fingerprint search with reference to the parameters like Robustness, Reliability, Fingerprint size, Granularity, Search speed and scalability [2].

A.Wang processes the fingerprints from the unknown sample and matched with a large set of fingerprints derived from the music database. The candidate matches are subsequently evaluated for correctness of match. Some guiding principles for the attributes to use as fingerprints are that they should be temporally localized, translation-invariant, robust, and sufficiently entropic. The temporal locality guideline suggests that each fingerprint hash is calculated using audio samples near a corresponding point in time, so that distant events do not affect the hash [3].

H. Schreiber, P. Grosche, and M. Muller presented an optimization scheme of the Haitsma/Kalker audio fingerprinting search algorithm. The suggested approach exploits strong temporal correlations between sub-prints as an indicator for sub-print robustness. This can lead to significant savings in the number of required lookups leading to a significant overall speed-up for the identification task[4].

P. Grosche, M. Müller, and J. Serra presented three representative retrieval strategies based on the query. (a)Traditional retrieval using textual metadata (e. g., artist, title) and a web search engine. (b)Retrieval based on rich and expressive metadata given by tags. **(c)** Content-based retrieval using audio, MIDI, or score information. Such content-based approaches provide mechanisms for discovering and accessing music even in cases where the user does not explicitly know what is actually looking for. Such approaches complement traditional approaches that are based on metadata and tags[5].

F. Kurth and M. Muller proposed a novel index-based audio matching algorithm, which allows for identifying and retrieving musically similar audio clips irrespective of a specific interpretation or arrangement—a task which previous audio identification algorithms cannot cope with. First absorb a high degree of the undesired variations at the feature level by using Chroma-based audio features. To cope with global variations in tempo and pitch, multiple queries strategy is employed. Finally, introduced a further degree of robustness into the matching process by employing fuzzy and mismatch search. The combination of these various deformation- and fault-tolerance mechanisms allowed to employ standard indexing techniques to obtain an efficient as well as robust overall matching procedure, which can cope with significant, musically motivated variations that concern tempo, articulation, dynamics, and instrumentation [6].

D. P. Ellis and G. E. Poliner present a method for identification of cover tracks from music audio databases with two features i.e. Beat Tracker and Chroma feature. Beat tracker generate a beat-synchronous representation with one feature vector per beat. The representation of each beat is a normalized Chroma vector. It provides the most efficient coverage of large music databases. These can then be used as 'index terms' to permit the use of more rapid indexing schemes, as well as potentially revealing interesting repeated and shared structure within music collections[7].

III. IMPLEMENTATION STRATEGY

Fingerprinting is one of the reliable techniques which allow the identification of audio independently of its format and without the need of metadata. Approaches for fingerprinting are: Information Retrieval or Cryptography and Pattern matching. An ideal fingerprinting system has ability to solve several requirements. It should be able to indentify an item accurately with respect to the level of compression and distortion. When presented with unlabelled audio, its fingerprint is calculated and matched against those stored in the database. Using fingerprints along with matching algorithms, distorted recording can be identified as the same audio content. To identify an unknown audio fragment (the query) which consists of series of hash values, fingerprints i.e. compact representations of an audio file or fragment are extracted from the query and compared with fingerprints stored in a database containing hash values for reference files that were computed the same way as the query.

A. System Architecture

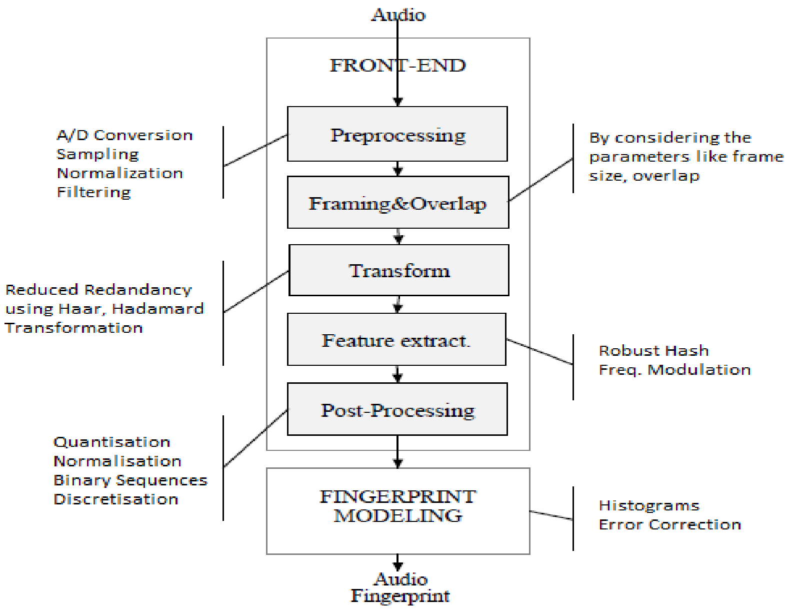

Fig. 1 System architecture of Audio Fingerprinting

Fig.1 shows the architecture of audio fingerprinting which consists of mainly two blocks one is a front-end and second is a fingerprint modelling. The first block that is front end block computes measurement set from the signal. The model block defines the fingerprint representation. Index sequences to HMM sound classes, error correcting words sequence. By using matching algorithm a fingerprint derived from recording, searches the best match from a database of fingerprints. Thus the fingerprints comparison is needed as the number of comparison is high and the distance which can be compute is expensive. The common methods use a simpler distance to quickly discard candidates and the more correct, but it is an expensive distance.

FRONT-END OF SYSTEM

The front-end block of a system converts an audio signal into a sequence of related features to pass as an input to the fingerprint model block.

PREPROCESSING

With the help of pre-processing the audio is digitalized and converted to a general format.

FRAMING AND OVERLAP

By using framing the signal is divided into number of frames of a size. Frame rate is the number of frames computed per second. Tapered window function is mainly used to each block to minimize the discontinuities at the beginning and end. Overlap is applied when the input data is not aligned to the recording that was used for generating the fingerprint.

LINEAR TRANSFORMS: SPECTRAL ESTIMATES

With the help of this steps the transformation of the set of measurements to a new set of features.

FEATURE EXTRACTION

Feature extraction finds a great diversity of algorithms. The main purpose is to reduce the dimensionality and increase the in variance to distortions.

POST-PROCESSING

Propose system uses low resolution quantization to the feature extraction.

B. Methodology

Module 1: Reading Song and converting song to time domain
Input : Song MP3,Wav File
OutPut : Time Domain Representation of Song

Module 2: Time to Frequency Domain
Input : Time Domain
OutPut : Frequency Domain Output

$$X(n) = \sum_{k=0}^{N-1} x[k]\, e^{-j(2\pi kn/N)}$$

N is the size of the **window**: the number of samples that composed the signal
X(n) represents the nth **bin of frequencies**
x(k) is kth sample of the audio signal

Module 3: Find Key Points
Input : Frequency Domain Signal
Output : Keypoints after predefined interval
KeyPoints=Log(Max(Upper_Limit))+1;
Where Upper Limit is Frequency at peak point

Module 4: Storing Keypoints
Input :Set of Keypoints
Output :HashTable index
Index=Hash<List<K>,V>;
K=keypoint Nam
V=value of Keypoint

Module 5: Noise Removal
Input : Song to Search
Output : noise free Song

C. Algorithm

Algorithm for Noise Removal

1. Calculate the fft of samples
   fft_values = fft(samples);
2. Get the mean value and calculate a threshold
   mean_value = mean(abs(fft));
   threshold = 1.1 * mean_value; % fine-tune this
3. Remove everything that's below the threshold (assume that it corresponds to noise) :
   fft_values [abs(fft_values) < threshold] = 0 ;
4. Get the filtered samples :
   filtered_samples = ifft(fft_values);
5. Return Audio

IV. EXPERIMENT AND RESULTS

A real time audio database of mp3, Hindi songs, with stereo format, 16 bit depth and 44.1 KHz sampling rate, sample size 8-bits are referred for experimentation. The segments are depends on the length of the audio, for every 50s length the audio is cut to function as query in our audio fingerprinting system. For audio length up to 50s is consider as one segment. The fingerprints are generated for these song segments. Each Fingerprint block contains 256 subsequent 32-bit sub-fingerprints. All these song segments are recorded using a microphone / hard drive.

TABLE I shows the result for 1000 songs, segments are analysed and Precision are calculated for top k-values.

TABLE I
SHOWS RESULTS OF SONG PRECISION

No of Input Song	Top K Songs	Precision
3	2	1
10	5	1
30	5	0.8
100	5	0.8
300	5	0.8
500	5	0.8
1000	5	0.8

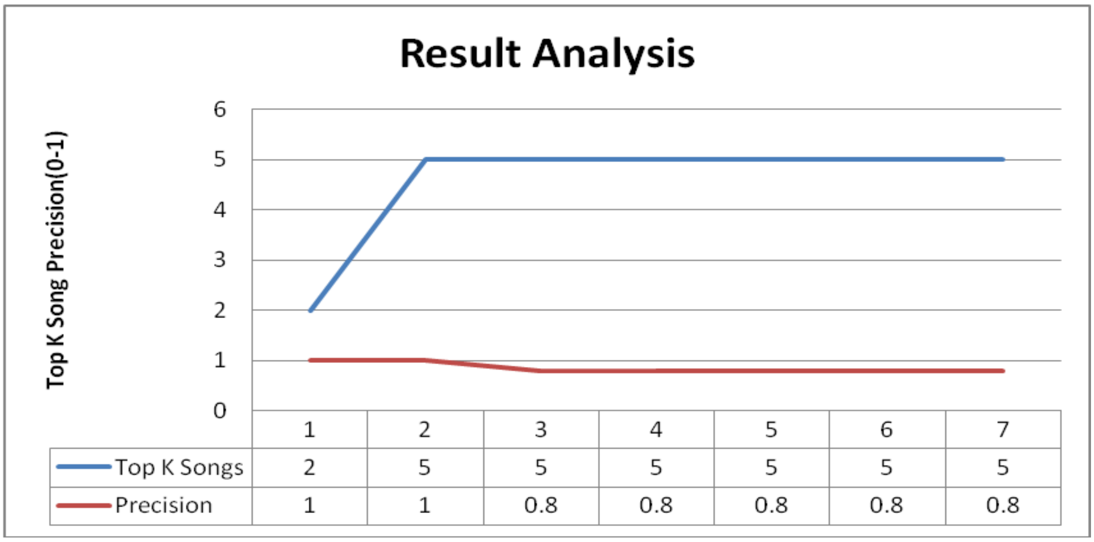

Fig 2: Results of Song Precision

TABLE II shows the result for fingerprint creation. Fingerprints are calculated for input query and it is used to compare with the stored fingerprints in database.

TABLE II
SHOWS RESULTS OF FINGERPRINT CREATION

Total Number of Songs	Time in MS
3	45
10	458
30	758
100	987
300	1248
500	1485
1000	2785

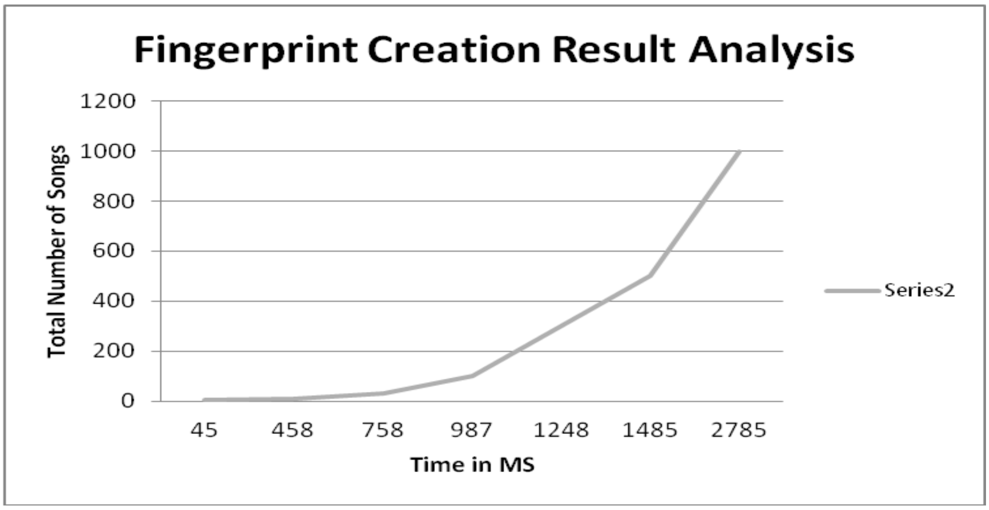

Fig 3: Result of Fingerprint Creation

TABLE III shows the result for the matching fingerprint of input song by calculating the Euclidian distance and compared with the threshold value and the metadata is displayed for that song. It includes Title, Album, Singer, Lyrics and Music information.

TABLE III
SHOWS RESULTS OF SONG SEARCHING

Database Songs	Search Time(Ms)
3	3
10	12
30	15
100	33
300	40
500	52
1000	60

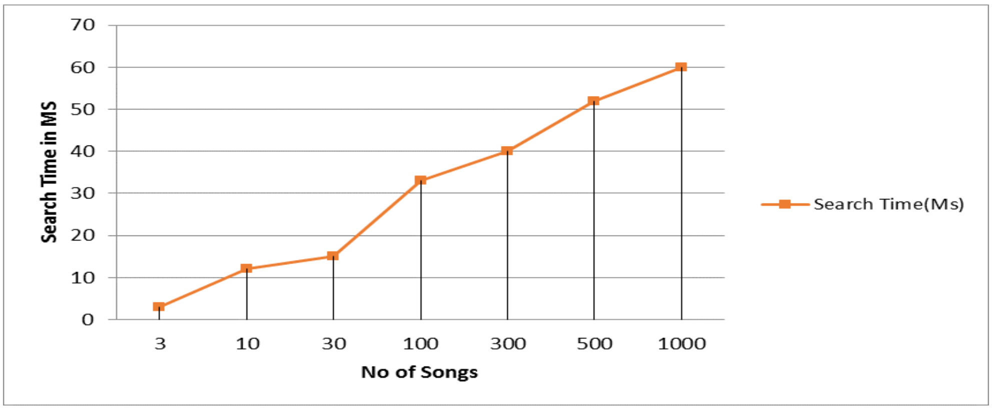

Fig.3: Shows Results Of Song Searching

V. CONCLUSION AND FUTURE SCOPE

The main purpose of audio fingerprinting is used for searching, retrieval and its applications in a common framework. The main observation found during the probability of finding a matching reference sub-print is elevated in the case that multiple query sub-prints are identical. At that time the fingerprint properties are explored with regard to robustness, hash quality, and temporal correlation and developed a set of indicators to measure how well a fingerprint can perform in the audio ID context. By using longer frames and less overlap in combination with enlarged fingerprints and a search strategy to find a correct hash bucket much quicker. The system focus on other feature extraction techniques and optimization of the search algorithm supports for strong distortions such as audio recorded in a noisy club, cropped or framed audio. In future it can be used to identify the problem for two mix audio recorded song.

ACKNOWLEDGEMENT

I am glad to express our sentiments of gratitude to all who rendered their valuable guidance to me. I would like to express my appreciation and thanks to Prof. Dr. S.A. Patil, Principal, SVIT College of Engg., Chincholi, Nashik. I am also thankful to Prof. S. M. Rokade, Head of Computer Engineering Department, SVIT College of Engg., Chincholi, Nashik. I thank the anonymous reviewers for their comments.

REFERENCES

[1] Hendrik Schreiber and Meinard Muller, Member, IEEE, "Accelerating Index-Based Audio Identification" *IEEE TRANSACTIONS ON MULTIMEDIA, VOL. 16, NO. 6, OCTOBER 2014*

[2] J. Haitsma and T. Kalker, "A highly robust audio fingerprinting system," in Proc. Int. Conf. Music Inf. Retrieval (ISMIR), Paris, France, 2002, pp. 107-115.

[3] A.Wang, "An industrial strength audio search algorithm," in Proc. Int. Conf. Music Inf. Retrieval (ISMIR), Baltimore, USA, 2003, pp. 7-13.

[4] H. Schreiber, P. Grosche, and M. Muller, "A re-ordering strategy for accelerating index-based audio fingerprinting," in Proc. 12th Int. Conf. Music Inf. Retrieval (ISMIR), Miami, FL, USA, 2011, pp. 127-132.

[5] P. Grosche, M. Mller, and J. Serr, "Audio content-based music retrieval "in Multimodal Music Processing, ser. Dagstuhl Follow-Ups,M. Miller, M. Goto, and M. Schedl, Eds. Dagstuhl, Germany:Schloss Dagstuhl-Leibniz- Zentrumfr Informatik, 2012, vol. 3, pp.157-174.

[6] F. Kurth and M. Mller, "Efficient index-based audio matching," IEEE-Trans. Audio, Speech, Language Process., vol. 16, no. 2, pp. 382-395,Feb. 2008.

[7] D. P. Ellis and G. E. Poliner, "Identifying 'cover songs' with chroma features and dynamic programming beat tracking," in Proc. *IEEE Int.Conf. Acoustics, Speech Signal Process. (ICASSP), Honolulu, HI,USA, Apr. 2007, vol. 4, pp. IV-1429-IV-1432.*

[8] F. Kurth, A. Ribbrock, and M. Clausen, "Identification of highly distorted audio material for querying large scale data bases," in Proc.112th AES Conv., 2002.

[9] C.Bellettini and G. Mazzini, "A Framework for Robust Audio Fingerprinting," J. Comm.,vol.5,no.5, pp. 409-424, 2010.

[10] E. D. Scheirer, "Tempo and beat analysis of acoustical musical signals," J. Acoust. Soc. Amer., vol. 103, no. 1, pp. 588-601, 1998.

[11] Juan Pablo Bello. "Audio-based cover song retrieval using approximate chord sequences: testing shifts, gaps, swaps and beats. In Proceedings of the International Conference on Music Information Retrieval," (IS-MIR), pages 239-244, Vienna, Austria, 2007.