AUDIO SIGNAL IDENTIFICATION AND SEARCH APPROACH FOR MINIMIZING THE SEARCH TIME IN AUDIO FINGERPRINTING USING TEMPLATE MATCHING

Er. Mohammed Ahmed[1]
[1]Computer Engineering Department
M.H. Saboo Siddik College of Engineering
Mumbai, India
Swaleha Shaikh[2], Hera Ansari[3] and Arshi Siddiqui[4]
[2,3,4] Department of Computer Engineering, M.H.Saboo Siddik College Of Engineering

ABSTRACT

Audio or music fingerprints can be utilize to implement an economical music identification system on a million-song library, however the system needs great deal of memory to carry the fingerprints and indexes. Therefore, for a large-scale audio library, memory imposes a restriction on the speed of music identifications. So, we propose an efficient music identification system which used a kind of space-saving audio fingerprints. For saving space, original finger representations are sub-sample and only one quarters of the original data is reserved. In this approach, memory demand is far reduced and therefore the search speed is criticalincreasing whereas the lustiness and dependability ar well preserved. Mapping audio information to time and frequency domain for the classification, retrieval or identification tasks presents four principal challenges. The dimension of the input should be considerably reduced; the ensuing options should be strong to possible distortions of the input; the feature should be informative for the task at hand simple. We propose distortion free system which fulfils all four of these requirements. Extensive study has been done to compare our system with the already existing ones, and the results show that our system requires less memory, provides fast results and achieves comparable accuracy for a large-scale database.

KEYWORDS
Audio Fingerprint, Fingerprint Extraction, Audio Template, Fast Fourier transform, Database Search.

1. INTRODUCTION

Nowadays, due to rapid digitization, digital music libraries are becoming popular and expanding[1][2][3][4]. With their increasing demand it has become essential to study them. The potency to seek out the precise song from outsized audio information is especially a vital task for variety of application. Fingerprints are short summaries of multimedia content. It is similar to Human fingerprint that is used for identifying an individual; an audio fingerprint is used for matching an audio clip. The modified version of the audio should have the similar fingerprints with the original or real audio. Various systems have been developed consisting of different techniques and methods. Each of it had some or the other drawback's[5][6][7][8][9][10][11].

The audio fingerprinting technique has many applications in broadcasts monitoring, automatic generations of playlist of radio, TV, or web broadcast etc. Another application can be described as search (retrieving) the name of the artist and title of the songs given a short clip of an audio,

DOI:10.5121/acii.2016.3203

filtering technology for files sharing and so on. Since an audio contains many properties, extracting a sample of the audio fingerprint is a potential task that identifies each song uniquely.

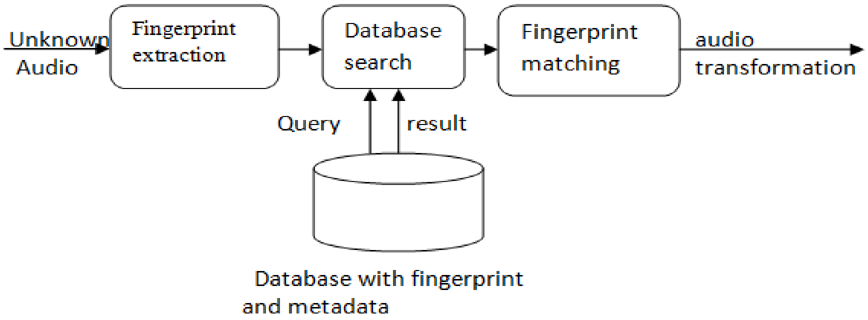


Figure 1: Overview of Audio signal identification

A fingerprinting system which is used for identification is generally made up of the following main modules:

1. Fingerprint extraction-This involves removing the unique characteristic of each audio input.
2. DB search-In this step, the audio is searched with the already stored audio in the song Meta data.
3. Fingerprint matching-Finally, the fingerprints are compared. If there exists a match of fingerprints then respective result is returned.

Our main focus is on how to increase the speed of searching as well as decrease the searching time by a considerable amount.

Broadly speaking, the formula may be logically rotten into 2 main parts[17]:

- Fingerprint creation (extracting distinctive sensory activity options from the song) and
- Fingerprint search (querying the database)

Because many elements are concerned in each activity, they'll be summarized along.

2. PROPOSED SYSTEM

The proposed search algorithm contains two phases as described below:

i. In the initial phase, all those songs are searched and retrieved which could probably contain the query songs fingerprint.
ii. In the second major phase, the entire fingerprint of each matched songs is retrieved from the database and is compared with the query fingerprint that is in question, if the song matches with the query fingerprint then it is retrieved.

In simple terms, if you want to compare audio or music files by their perceptual equality, you should create the so called "fingerprints" (same or similar to human fingerprints, which uniquely describe a person's identity), and sees if sets of these objects, gathered from different audio or music sources match or not. Logically, similar audio objects should generate similar fingerprints, whereas different file should emanate unlike signatures. A pre requisite for these fingerprints is that they should act as "hash", in order with format differences, difference in frequencies, distortions, noise, "intensity", etc. The simplified concepts of audio fingerprinting can be visualized below.

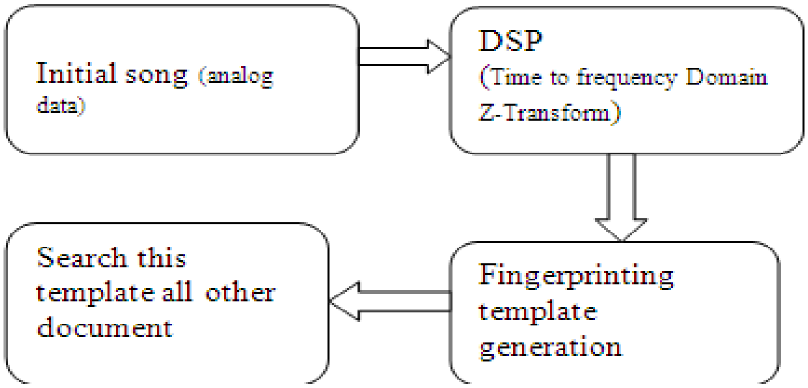

Initial song (analog data)
DSP
(Time to frequency Domain Z-Transform)
Fingerprinting template generation
Search this template all other document

Figure 2: Modules in the proposed system

- **Fingerprint as a hash value**

Briefly, audio fingerprint may be seen as a compressed outline of the corresponding audio object. Mathematically speaking, a finger mark operate F maps the audio object X consisting from an oversized variety of bits to a fingerprint of solely a restricted variety of bits. So, the complete mechanism of extraction of a fingerprint may be seen as a hash-function H, that maps Audio object X to a hash (a.k.a. message digest). The most advantage why hash functions are therefore widely utilized in the sphere of engineering science is that they permit scrutiny 2 massive objects X, Y; by simply scrutiny their individual hash values H(X), H(Y). The equality between latter pairs implies the equality between X, Y, with a really low error chance (even although collision cannot be eliminated entirely). Next, you'll be able to see an abstract diagram of the hash function's practicality.

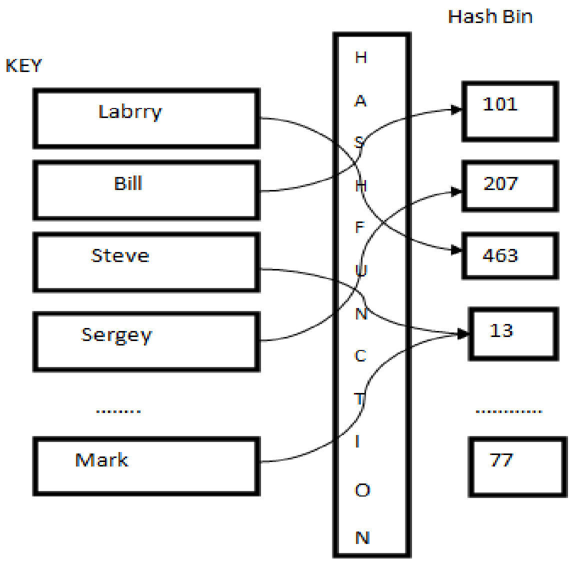


Figure 3: Conceptual diagram of the hash function's functionality

Here within the figure, string values act as keys, mapping into hash bins (integers) that are smaller purposely and far easier to check. Nonetheless, there's forever little chance that fully completely different keys can occupy a similar bin (Steve and Mark collision). So, why will we get to develop a completely new system supported fingerprints, and not simply use crypto logic hash functions? The solution is pretty straightforward, though the sensory activity distinction of audio object compressed to MP3 format and therefore the same one compressed to WAVE is incredibly little, the binary illustration of these are altogether completely different, that means that H(MP3(X)) and H(WAVE(X)) can end in fully dissimilar message digests (hash bins). Even worst, crypto logic hash functions are typically terribly sensitive: one little bit of distinction within the original object ends up in {a completely an altogether a very} different totally completely different completely different} hash price (thus a little degradation within the key can result in totally different hash digest). So we propose a system which will not take under consideration any low level details (binary signatures) and can analyse solely the audio signal as any human will (will act as a "forgiving hash" that will not pay abundant attention to irrelevant differences).

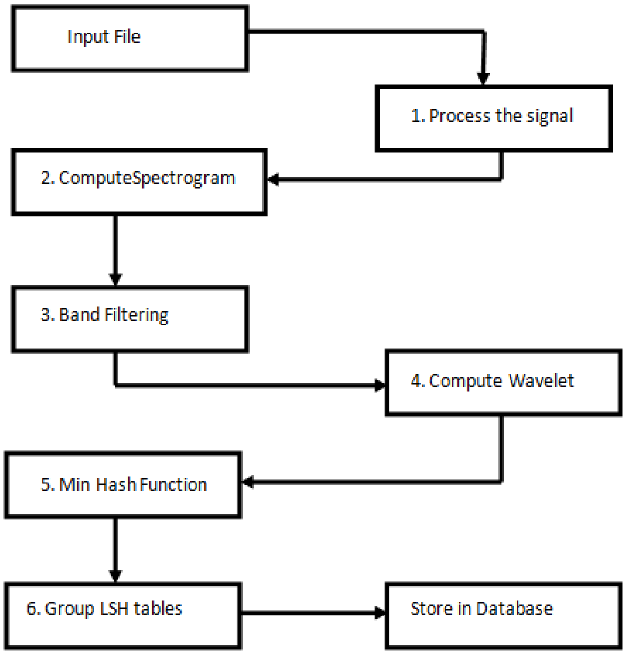
Thus, audio fingerprinting algorithms have the potential to link small, unknown chunks of audio templates i.e., finding similar properties of a song that can be afterwards used to recall a new unknown sample, or find a duplicate in the database of already processed songs. These types of algorithms can be used in a variety of different application:

i. Identifying the currently playing song
ii. Managing sound effects libraries
iii. Observing and controlling the broadcast monitoting[2][3][4[5][12].

With all the advantages that the sounds fingerprinting system has, there are several challenge that it has to address. One of them is a huge database to searches (imagine YouTube's database of video content that is monitors for audio copyright issues). Normally, each song will generates a big amount of fingerprints (in the described algorithms , the granularity of each of them is going to be 1.48 sec, so we'll have about 200 object for a song). Lastly, there will be a requirement for developing an economical yet efficient search rule that works well within scalable solution bounds.

1. IMPLEMENTATION

The framework that goes to be designed (upon that the complete system can reside) can comprises many completely different abstract elements. Within the next figure, you'll be able to visualize the activity diagram that abstractly describes the logical flow of the audio fingerprint generation. Following, can describe in deeper details every activity concerned and what element is accountable for it.



.

Figure 4: Phases involved in Fingerprint extraction algorithm

4. PREPROCESSING THE INPUT SIGNAL

Merely, all audio files that anyone has on his laptop are encoded in some format. Therefore, the primary step (1) within the formula is decipherment the input data to PCM (Pulse Code Modulation format). The PCM format may be thought of the raw digital format of analog signal, within which the magnitude of the signal is sampled often at uniform intervals. For time-domain signals, the unit for rate is 1/s. The inverse of the frequency is that the sampling amount or sampling interval, that is that the time between samples. because the analysed band within the formula lies inside the vary of 318 Hz and 2000 Hz, down sampling the input to 5512 Hz is taken into account a secure and, at a similar time, a needed operation. Whereas reducing the sample rate (normally from 44100 Hz), we'll get eliminate data that isn't relevant from the human sensory activity purpose of read, and can specialize in vital options of the signal. The pre-processing is finished victimization the Bass.Net library (in my opinion, the simplest one for operating with audio files). With the refashion of the Sound Fingerprinting framework, you'll be able to use library for the needs printed higher than. The sole bother with Audio is that you simply do not have that abundant flexibility in terms of supported audio formats.

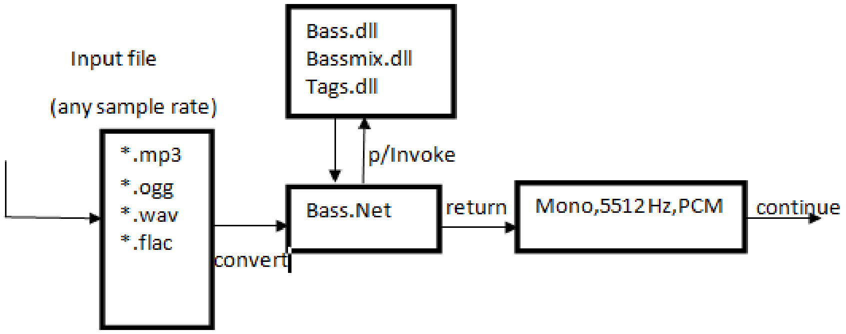


Figure 5: Input Signal Processing

5. SPECTROGRAM CREATION

After pre-processing the signal to 5512 cycles/second PCM, successive (2) step within the procedure algorithmic rule is building the spectrograph of the audio input. In digital signal process, a spectrograph is a picture that shows however the spectral density of a symbol varies in time. Changing the signal to spectrograph[6] involves many steps. 1st of all, the signal ought to be sliced into overlapping frames. Every frame ought to then be skilled a quick Fourier remodel so as to urge the spectral density varied in time domain. The parameters utilized in these transformation steps are going to be adequate those who are found to figure well in different audio procedure studies (specifically in an exceedingly extremely sturdy Audio procedure System): audio frames that area unit 371 Ms long (2048 samples), taken each 11.6 Ms (64 samples), so having associate overlap of 31/32, as have been used in the previous study. H an k 2002 ref

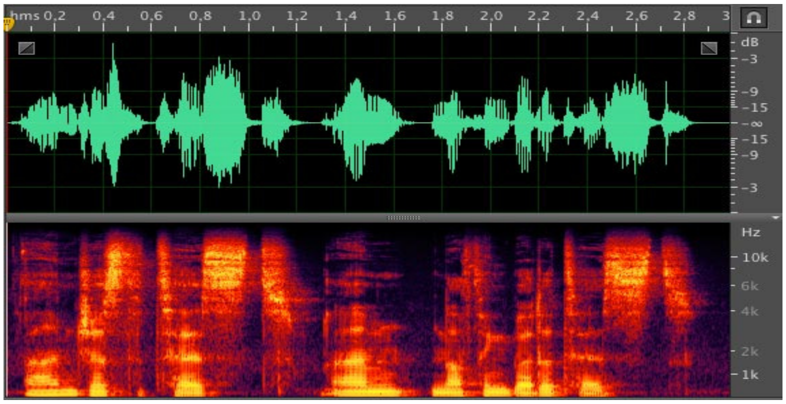


Figure 6: Spectrogram of the audio input

Building the spectrograph of the image involves the foremost high-ticket (from a central processing unit purpose of view) operation within the entire algorithmic rule - quick Fourier remodel (O (n*log (n))).Fourier analysis could be a family of mathematical techniques, all supported rotten signals into sinusoids.

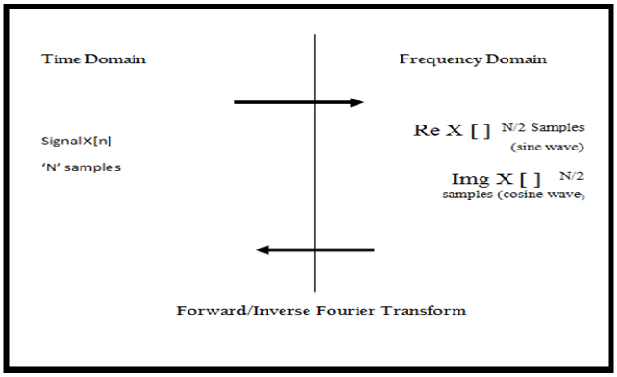

Time Domain
Frequency Domain
SignalX[n]
'N' samples
Re X [ ] N/2 Samples (sine wave)
Img X [ ] N/2 samples (cosine wave)
Forward/Inverse Fourier Transform

Figure 7: Forward / Inverse Fourier Transform

6. BAND FILTERING

After the FFT remodel is accomplished on every slice, the output spectrograph is cut such 318Hz-2000Hz domain is obtained (researchers set that this domain is enough for a relevant illustration of the sensory activity content of a file). Generally, this domain is thought of to be one among the foremost relevant frequency areas for Human sensory system. whereas the vary of frequencies that any person will hear is basically associated with environmental factors, the commonly accepted customary vary of perceptible frequencies is twenty to 20000Hz. Frequencies below 20Hz will typically be felt instead of detected, assumptive the amplitude of the vibration is high enough. Frequencies on top of 20000Hz will generally be detected by youth, however high frequencies area unit the primary to be plagued by hearing disorder as a result of age and/or prolonged exposure to terribly loud noises. Within the next table, you'll be able to visualize the frequency domains and their main descriptions. Because it is seen, the foremost relevant audio content lies inside the 512-8192Hz frequency vary, because it defines traditional speech.

7. WAVELET DECOMPOSITION:

Once the power spectrograph is "drawn" (having thirty two bins contact the 318-2000Hz vary, for every eleven.6 MS), successive step within the algorithmic rule (4) is dividing the whole image into slices (spectral sub-images one28x32) that correspond to a 1.48 sec roughness. In terms of programming, once you build the log-spectrogram, you bought its entire image encoded into associate [N] [32] double-dimensional float array, wherever N equals the "length of the signal in milliseconds divided by eleven.6 MS", and thirty two represents the amount of frequency bands. Once you get the divided spectral sub-images, they're going to be more reduced by applying customary Haar[7] ripple decomposition on every of them. The utilization of wavelets within the audio-retrieval task is finished as a result of their flourishing use in image retrieval (Fast Multiresolution Image Querying). In different words, a ripple could be a wave-like oscillation with amplitude that starts at zero, increases, and so decreases back to zero. It will generally be unreal as a "brief oscillation" just like the one seen on a measuring device or monitor.

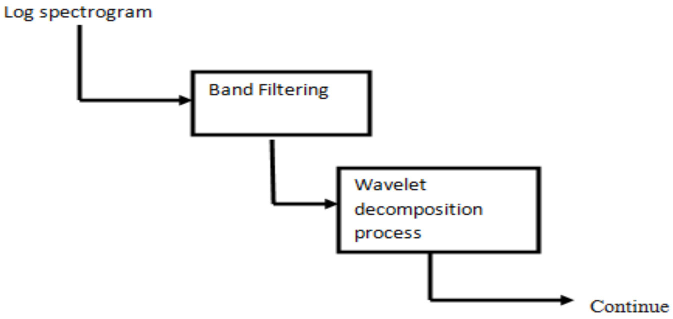


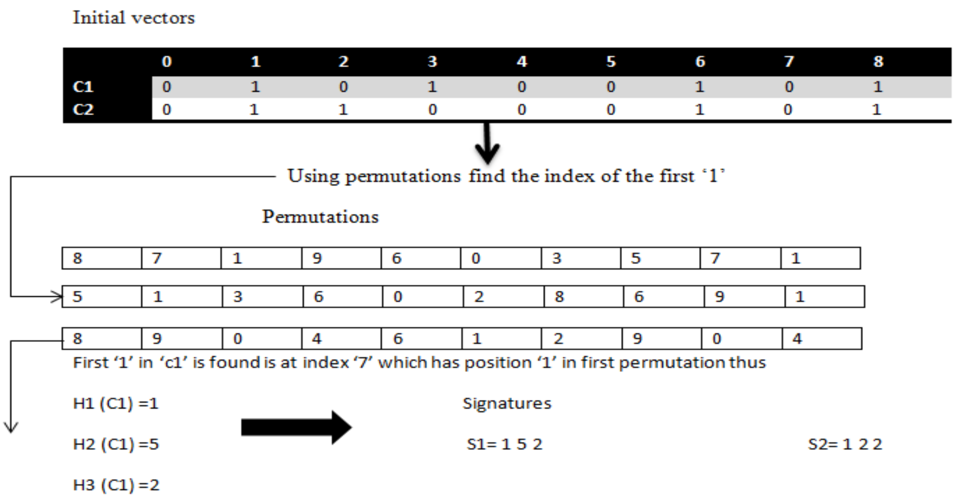
Figure 8: Framework for Audio-retrieval task

Wavelets area unit crafted purposefully to possess specific properties that build them helpful for signal/image process. They'll be combined by employing a "shift, multiply and sum" technique referred to as convolution, with parts of associate unknown signal to extract info from this signal. The elemental plan behind wavelets is to investigate in step with scale. So, turning back to our algorithmic rule, for every spectral picture, a ripple signature is computed. Rather than exploitation the whole set of wavelets, we have a tendency to solely keep those that almost all characterize the song (top two hundred is taken into account an honest choice), so creating the signature proof against noise and different degradation. One among the fascinating things found within the previous studies of image process was that there's no have to be compelled to keep the magnitude of high wavelets; instead, we are able to merely keep the sign of it (+/-). This info is enough to stay the extract sensory activity characteristics of a song.

8. MIN HASH THE FINGERPRINT

At this stage of process, we've got fingerprints that area unit 8192 bits log, that we'd wish to more cut back in their length, by making a compact illustration of every item. Therefore, within the next step (5) of the algorithmic rule, we have a tendency to explore the utilization of Min-Hash[16] to reason sub-fingerprints for these distributed bit vectors. Its usage has proved to be effective within the economical search of comparable sets. It works as follows: think about a column as a group of rows during which the column contains a one (consider C1 and C2 as 2 fingerprints). Then the similarity of 2 columns C1 andC2 is Sim (C1, C2) = (C1∩C2)/(C1∪C2).

This similarity index is named the Jaccard similarity: it's variety between zero and 1; it's zero once the 2 sets area unit disjoint, one once they area unit equal, and strictly between zero and one otherwise. Before exploring however Min-Hash works, I am going to purpose to an awfully helpful characteristic that it has: the likelihood that MinHash (C1) = MinHash (C2) equals specifically to Jaccard similarity. this can permit U.S. to use the algorithmic rule by with efficiency hashing similar things (with a similarity index over 60-70%) into identical bins (exactly what we have a tendency to have an interest in - "forgiving hashing" technique that will not pay a lot of attention to little variations between 2 keys). The Min-Hash technique works with binary vectors as follows: choose a random, but known, rearrangement of all the vector positions. Then, for every vector permutation live during which position the primary '1' happens. As a result of the system can work with fingerprints of 8192 bit size, the permutation vector can contain random indexes ranging from zero to 8192. For a given permutation, the hash values agree if the primary position with a one is that the same in each bit vectors, and that they disagree if the primary such position could be a row wherever one, however not each, vectors contained a one. Note that this can be specifically what's required; it measures the similarity of the distributed bit vectors supported matching "on" positions. We are able to repeat the on top of procedure multiple times, on every occasion with a brand new position-permutation. If we have a tendency to repeat the method p times, with p totally different permutations, we have a tendency to get p for the most part freelance projections of the bit vect (e.g. p=100permutations).



Initial vectors

	0	1	2	3	4	5	6	7	8
C1	0	1	0	1	0	0	1	0	1
C2	0	1	1	0	0	0	1	0	1

Using permutations find the index of the first ‘1’

Permutations

8	7	1	9	6	0	3	5	7	1
5	1	3	6	0	2	8	6	9	1
8	9	0	4	6	1	2	9	0	4

First ‘1’ in ‘c1’ is found is at index ‘7’ which has position ‘1’ in first permutation thus

H1 (C1) =1

H2 (C1) =5

H3 (C1) =2

Signatures

S1= 1 5 2

S2= 1 2 2

Figure 9: Min-Hash to compute sub-fingerprints

9. Retrieval

After the min-hashed fingerprint is partitioned off into LSH tables, the LSH technique is efficient in the number of comparisons that are required and provides noise-robustness properties[13]. The method of extracting signatures from audio files is thought of completed. Now, however will we discover similar songs or retrieve info regarding associate unknown audio by having solely a tiny low piece of it? The retrieval method (detection of the question song) is analogous to the creational one, with associate exception that not the whole song is fingerprinted at now. You will choose as several seconds to fingerprint as you wish, creating an exchange between accuracy and interval. The steps area unit recurrent until the fingerprint is partitioned off into hash tables. Once done, the question is shipped to the storage to envision what percentage hash tables contain corresponding hash values. If over v votes are unit non heritable throughout the question (more than v tables have come back identical fingerprint), the fingerprint is taken into account a possible match (in Content procedure exploitation wavelets, a threshold of 2-5 votes was used). All matches area unit elite and analysed so as to search out most effective} candidate (a simple overacting distance calculation between the question and potential candidates is exhausted order to envision what item is that the most relevant one). Just in case of developing a duplicates detector algorithmic rule, the principle of extraction could be a bit totally different. Range of threshold votes is inflated to eight (8/25 tables ought to come identical result), and a minimum of five-hitter of total question fingerprints ought to vote for identical song. Once this criterion is met, the song is taken into account a reproduction. Overall, of these parameters area unit outlined through empirical observation, thus you may notice even higher configuration which will work additional correct because the one delineated here. As associate example, you will lower the edge votes so as to get results which will additionally contain the remixes of the songs matched as duplicates (it's perpetually a trade-off between false-positives and therefore the objective function).

10. FUTURE SCOPE

Here in this paper, an alternative searching method has been elaborated. On comparison with the older schemes a significant amount of decrease in the audio retrieval was noticed. On one hand making use of hash function provided uniqueness to each song so on the other hand, fingerprinting on only required length of song saved a large amount of disk space. The above scheme has a scope and can be applied to areas of military and telecommunications. Also big music industries such as radio stations, airtel music stores, online song sites etc. can make use of this to avoid repetitive data and thereby save space. Further it can be used by making it for sppech datasets and for variety of other audio formats also.

CONCLUSION

In this paper, we propose a new and efficient process for environments where large audio databases need to be systemized and non-redundant. The fingerprinting scheme theoretically gives better results for large databases and saves time. Also only small fragments of each song fingerprint is created which saves time and reduces the memory requirement. This scheme gives more emphasis on extracting unique fingerprint value and avoids the other distortions or dissimilarity such as difference in genres, pitch[14], level of noise content, zero cross rating[15] etc. present in the audio. Also this scheme will avoid redundancy of songs existing in different files, folders or even root directories. Fingerprint along with hashing provides optimal and efficient search results.

ACKNOWLEDGEMENT

No project can be completed without the support of a lot of people. When we are concluding our project synopsis work by submitting this report, we reflect upon all the times when we needed support in various forms and were lucky enough to receive it.

We wish to express our sincere gratitude to our principal **Dr. Mohiuddin Ahmed**, M. H. Saboo Siddik College of Engineering, Mumbai, for providing the facilities to carry out the project work.

A special mention here to **Prof. Z. A. Usmani** (H.O.D., Computer Engineering Department, MHSSCOE) for his valuable support. We are also thankful to all staff members of Computer Department, without whom the completion of this report would have been impossible.

This entire journey would not have been possible without the efforts put in by our guide, **Er. Mohd Ahmed**. They have been a constant source of encouragement and guidance through the entire semester.

This acknowledgment would indeed be incomplete without rendering our sincere gratitude to our family. They have always been a pillar of strength and support in our past and current endeavors.

REFERENCES

1. Cheng Y., “Music Database Retrieval Based on Spectral Similarity”, International Symposium on Music Information Retrieval (ISMIR) 2001, Bloomington, USA, October 2001.
2. Allamanche E., Herre J., Hellmuth O., Bernhard Fröbach B. and Cremer M., “AudioID: Towards Content-Based Identification of Audio Material”, 100th AES Convention, Amsterdam, The Netherlands, May 2001.

3. Neuschmied H., Mayer H. and Battle E., "Identification of Audio Titles on the Internet", Proceedings of International Conference on Web Delivering of Music 2001, Florence, Italy, November 2001.

4. Fragoulis D., Rousopoulos G., Panagopoulos T., Alexiou C. and Papaodysseus C., "On the Automated Recognition of Seriously Distorted Musical Recordings", IEEE Transactions on Signal Processing, vol.49, no.4, p.898-908, April 2001.

5. J.S. Seo, J. Haitsma, T. Kalker, and C.D. Yoo, "A robust image fingerprinting system using the Radon transform," Signal Processing: Image Communication, vol. 19, pp. 325–339, 2004.

6. J. Haitsma and T. Kalker, "A Highly Robust Audio Fingerprinting System With an Efficient Search Strategy," J. New Music Research, vol. 32, no. 2, pp. 211–221, 2003.

7. M.L. Miller, M.C. Rodriguez, and I.J. Cox, "Audio Fingerprinting: Nearest Neighbor Search in High Dimensional Binary Spaces," J. VLSI Signal Processing, vol. 41, pp. 285–291, 2005.

8. Avery Wang "An Industrial-Strength Audio Search Algorithm", Proc. 2003 ISMIR International Symposium on Music Information Retrieval, Baltimore, MD, Oct. 2003.

9. Yu Liu, Kiho Cho, Hwan Sik Yun, Jong Won Shin, and Nam Soo Kim "DCT Based Multiple Hashing Technique for Robust Audio Fingerprinting" –IEEE trans. ICASSP, pp.61-64, 2009.

10. Guang-Ho Cha, "An Effective and Efficient Indexing Scheme for Audio Fingerprinting", IEEE trans. Fifth FTRA International Conference on Multimedia and Ubiquitous Engineering, pp.48-52.

11. Haitsma J., Kalker T. and Oostveen J., "Robust Audio Hashing for Content Identification, Content Based Multimedia Indexing 2001, Brescia, Italy, September 2001.

12. Gionis, Indyk, Motwani. Similarity search in high dimensions via hashing. Proc. VLDB, pp. 518–529, 1999.

13. Anonymous ICME submission," Audio-To-Tag Mapping: A Novel Approach for Music Similarity Computation", IEEE trans.2011, pp.1-4.

14. Arijit Ghosal, Bibhas Chandra Dhara, Sanjoy Kumar Saha, "Speech/Music Classification Using Empirical Mode Decomposition"- IEEE, Second International Conference on Emerging Applications of Information Technology, pp.49-52, 2011.

15. https://www.researchgate.net/publication/3296928_Finding_interesting_associations_without_support_pruning

16. HTTP://WWW.IJSR.NET/ARCHIVE/V3I9/U0VQMTQ2MDU%3D.PDF

Author

Mohd. Ahmed is currently Assistant professor at the M.H. Saboo Siddik College of Engineering and pursuing Phd, Mumbai and formerly trainee hardware & network engineer in DSS Mobile communication, Mumbai. He is a M. E. in Computer Engineering. Having 11 years teaching and one year industry experience.

Shaikh Swaleha has completed Diploma in computer engineering from M.H. Saboo Siddik College of Engineering and currently pursuing B.E in computer engineering

Ansari Hera has completed Diploma in computer engineering from Babasaheb Gawde Institute of Technology and currently pursuing B.E in computer engineering.

Siddiqui Arshi has completed Diploma in computer engineering from Babasaheb Gawde Institute of Technology and currently pursuing B.E in computer engineering.