A Novel Audio Fingerprinting Scheme based on Subband Envelop Hashing

Yu Liu, Hwan Sik Yun, June Sig Sung, and Nam Soo Kim
School of Electrical Engineering and INMC
Seoul National University, Seoul 151-742, Korea
E-mail: nkim@snu.ac.kr Tel: +82-2-880-1824

Abstract—**In this paper, we propose a novel audio fingerprinting technique based on subband envelop hashing (SEH). In this approach, the discrete cosine transform (DCT) is applied to each subband of the perceptual spectrogram, and the lower-ordered DCT coefficients representing the subband envelopes are retained to generate multiple hash values. The corresponding database matching algorithm is also extended which allows for an efficient search in the database given a short query audio clip. The algorithm is implemented and compared with the Philips Robust Hash (PRH) algorithm proposed in [1], and experimental results show that the SEH algorithm achieves a significant improvement over the baseline system under various distortions.**

I. Introduction

An audio fingerprint is a short summary of the content of an audio signal. The objective of an audio fingerprinting system is to identify short, unlabeled audio clips in an efficient and reliable way regardless of the audio format. The system consists of two fundamental processes: the fingerprint extraction stage and the database matching stage, as illustrated in Fig. 1. Given an unknown audio excerpt, the fingerprints are extracted and compared with the items in the database, and the metadata will be returned if the corresponding fingerprints are found in the database. The main difficulty in designing such a system comes from the high dimensionality, the significant variance of the audio data for perceptually similar content and the necessity to efficiently find the fingerprint in a huge collection of registered fingerprints [2]. The application scenarios include broadcast monitoring, connected audio, filtering technology for file sharing and automatic music library organization [1]. The application potential has boosted the interest from many researchers, and has given rise to a number of practical algorithms.

A review of audio fingerprinting is given in [2], in which the properties of a desired audio fingerprinting system were discussed and some practical algorithms were introduced. Among the various fingerprinting schemes, the Philips Robust Hash (PRH) algorithm [1] was claimed to be highly robust to degradations and theoretical framework that analyzed its robustness has been given in [3] and [4]. Due to the robustness and efficiency, the PRH algorithm has been used as the baseline system for evaluating several recently developed algorithms: the authors of [5] viewed the temporal-frequency differentiation of spectrogram as a filtering output, and they tried to improve the performance by using alternative frequency filters. However, the filters used are empirical and not founded on a theoretical basis. On the other hand, the authors of [6] treated the spectrogram as a 2-D image and transformed the music identification into a corrupted sub-image retrieval problem. The approach was further improved by [7] and [8] who utilized wavelets which is commonly used in the area of image processing. Although the spectrogram and a 2-D image do share something in common, they differ significantly in such a way that the spectrogram has a much larger correlation along the temporal axis than along the frequency axis, while a 2-D image typically have similar correlations along both x-axis and y-axis.

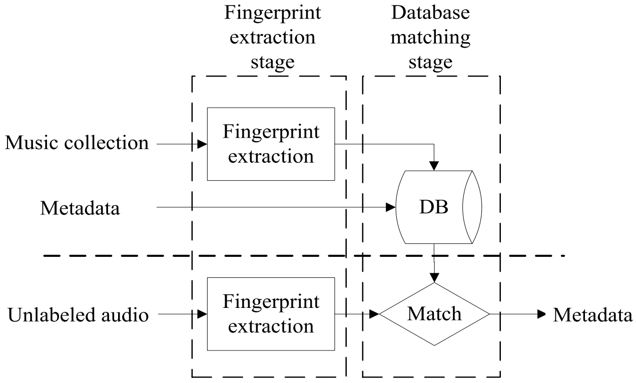

Fig. 1. Framework of an audio fingerprinting system

In this paper, we propose a novel audio fingerprinting scheme that generates multiple hash codes for each frame, and the corresponding database searching algorithm is also extended. Specifically, the discrete cosine transform (DCT) is applied to the temporal sequence of energies to extract the envelops of each subband, and the hash codes are computed from the low-ordered DCT coefficients. Experimental results have shown that the proposed approach outperforms the PRH algorithm under various environments.

II. Scheme of the baseline system

As the baseline system, the block diagram of the fingerprint extraction stage of the PRH algorithm is illustrated in Fig. 2. Specifically, the audio signal is first divided into overlapping frames with the length of about 370 ms, and the frame shift is 1/32 of the frame length. The large overlap of the frames assures that the hash values possess a large correlation along the time-axis so as to be robust against shifting. Second,

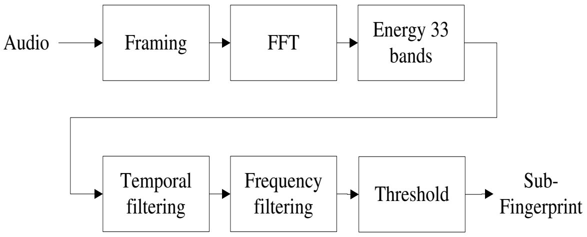

Fig. 2. Fingerprint extraction stage of the PRH algorithm

the power spectrum is obtained by performing FFT. Third, the energies for 33 non-overlapping logarithmically spaced subbands (e.g. Bark Scale) covering the frequency range of 300 Hz to 2000 Hz are calculated. The band division process reflects the perceptual characteristics of an audio signal. Finally, the obtained spectrogram is passed through a temporal filter $H_T(z)$ and a frequency filter $H_F(z)$ which can be shown as:

$$H_F(z) = H_T(z) = 1 - z^{-1}, \tag{1}$$

and the output is represented as:

$$\begin{aligned} ED(n,m) &= E(n,m) - E(n,m+1) \\ &- (E(n-1,m) - E(n-1,m+1)), \end{aligned} \tag{2}$$

where $E(n,m)$ denotes the m-th subband energy of n-th frame, and $ED(n,m)$ is the output of the filters that represents the difference between energies from successive frames and neighboring frequency bands. Finally a 32-bit hash value for each frame (which is referred to as a *subfingerprint*) is obtained by a threshold process:

$$F(n) = [F(n,0), \cdots, F(n,31)], \tag{3}$$

$$F(n,m) = \begin{cases} 1, & ED(n,m) > 0 \\ 0, & ED(n,m) \leq 0, \end{cases} \tag{4}$$

where $F(n)$ is the subfingerprint of frame n and $F(n,m)$ is the m-th bit of it.

The extracted hash values can be highly unique and thus enable an efficient database matching algorithm illustrated in Fig. 3. In the offline processing of the audio files stored in the database, all the subfingerprints computed are registered in a hash table with the subfingerprints being treated as the keys. Each entry of the hash table stores a list of pointers to the positions in the audio files where the subfingerprint occurs. In the stage of database matching, 256 subfingerprints which amount to approximately 3 seconds are extracted from the query audio, and each subfingerprint is matched with the hash table contents to find the candidate positions where it may come from. A fingerprint block with the same size as the query block ($256 \times 32 = 8192$ bits) from the candidate position is obtained, the bit error rate (BER) between the two blocks is computed and compared with a threshold which is set to 0.35 in [1]. If the BER is less than the threshold, the two signals are considered similar and the candidate audio is declared as the result.

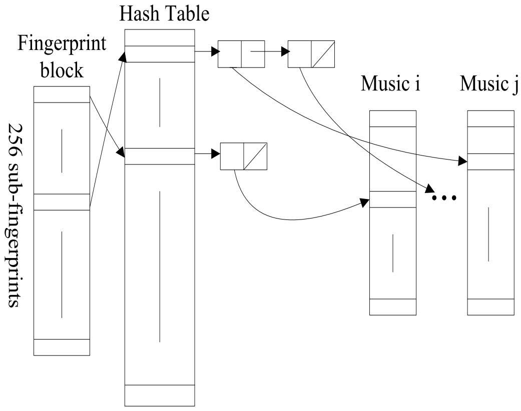

Fig. 3. Database matching stage of the PRH algorithm

III. Subband envelop hashing algorithm

The PRH algorithm is based on the sign of the temporal-frequency differentiation of the subband energies in the perceptual spectrogram. It is reasonable to summarize the discriminative information in the spectrogram in that way. However, for the temporal filter, it only makes use of the neighboring two frames, and is vulnerable to the possible local disturbances. The bits in subfingerprints can be flipped due to the existence of local noise and mislead the database matching algorithm. Moreover, the flips of bits in the fingerprints will reduce the robustness of the algorithm. As a result, to enhance the robustness of the algorithm both in the extracting phase and the database matching phase, including more frames and performing low-pass filtering to extract the subband envelops will provide a good way to extract the stable information for a robust discrimination by the frequency filter. The set of low-pass filters are designed to be orthogonal to assure that the features obtained can be more distinguished from each other in the same frame.

In the proposed SEH algorithm, the temporal filters are substituted into low-pass filters to extract the stable information in each subband. Specifically, DCT is applied to the temporal sequence of energies in each subband, and only the low-frequency components of DCT are used as inputs for the frequency filters. The low-frequency components of DCT are the envelops in each subband, which represent the stable information. The reasons for employing DCT as the temporal filters lie as follows: First, among all the orthogonal transforms, the decorrelation performance of DCT is closest to the Karhunen-Loéve transform [9]. Second, DCT has a strong energy compaction property [10] implying that most of the signal energy tends to be concentrated in a few low-frequency components. The decorrelation property ensures that each subfingerprint can be treated separately and performance improvement is possible via generating more subfingerprints. The energy compaction property enables the reduction in the number of subfingerprints, thus reduces the dimension of the fingerprinting block. Since the subband energies are evolving slowly, only a few DCT coefficients are sufficient to describe

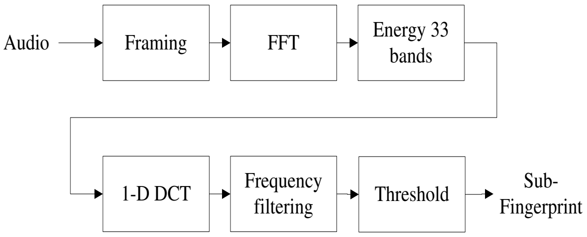

Fig. 4. Fingerprint extraction stage of the SEH algorithm

the subfingerprints.

The framework of fingerprint extraction stage in the SEH system is depicted in Fig. 4. The first three parts, i.e. framing, FFT and band energy calculation are the same as those in the PRH algorithm. However, the temporal filter in the PRH algorithm is substituted by DCT, i.e. L-point DCT is performed on the L consecutive subband energies $E(n,m), E(n+1,m), \cdots, E(n+L-1,m)$. Among the L DCT coefficients, the lower-ordered K values are retained as the input for the frequency filter. The obtained K coefficients for each frame n and subband m are denoted by $C_k(n,m)$, $k = 1, 2, \cdots, K$. The DCT coefficients are then passed through the same frequency filter as in the PRH algorithm, and the output can be represented as:

$$ED_k(n,m) = C_k(n,m) - C_k(n,m+1), \tag{5}$$

where $ED_k(n,m)$ represents the k-th output of subband m in frame n. Let $F_k(n)$ denote the k-th subfingerprint in frame n. Then,

$$F_k(n) = [F_k(n,0), \cdots, F_k(n,31)], \tag{6}$$

in which

$$F_k(n,m) = \begin{cases} 1, & ED_k(n,m) > 0 \\ 0, & ED_k(n,m) \leq 0. \end{cases} \tag{7}$$

The database matching algorithm of the PRH algorithm is also expanded for the SEH algorithm as depicted in Fig. 5. In the offline processing of the audio files stored in the database, all the subfingerprints computed are registered in hash tables with the subfingerprints being treated as the keys. Since K subfingerprints are computed for each frame in the SEH algorithm, K hash tables are constructed, for example, the subfingerprints obtained from the second DCT coefficients in each frame are registered in the second hash table. The database matching scheme consists of three steps: First, the query audio is divided into 256 frames, and K subfingerprints are obtained in each frame as in the fingerprint extraction phase. Consequently, the query fingerprint block consists of $K \times 32 \times 256 = K \times 8192$ bits, and $\{F_k(n), n = 1, 2, \cdots, 256\}$ forms the fingerprint block k. Second, the candidate positions are generated in each hash table separately, i.e. the subfingerprints in fingerprint block k are matched with the contents in the k-th hash table as in the PRH algorithm, and a candidate list is created by accumulating all the search results in all included hash tables. Finally, BERs are computed by comparing the query fingerprint block with those stored at the candidate positions in the database, and the candidate with most number of hits and BER less than the specified threshold is returned as the result.

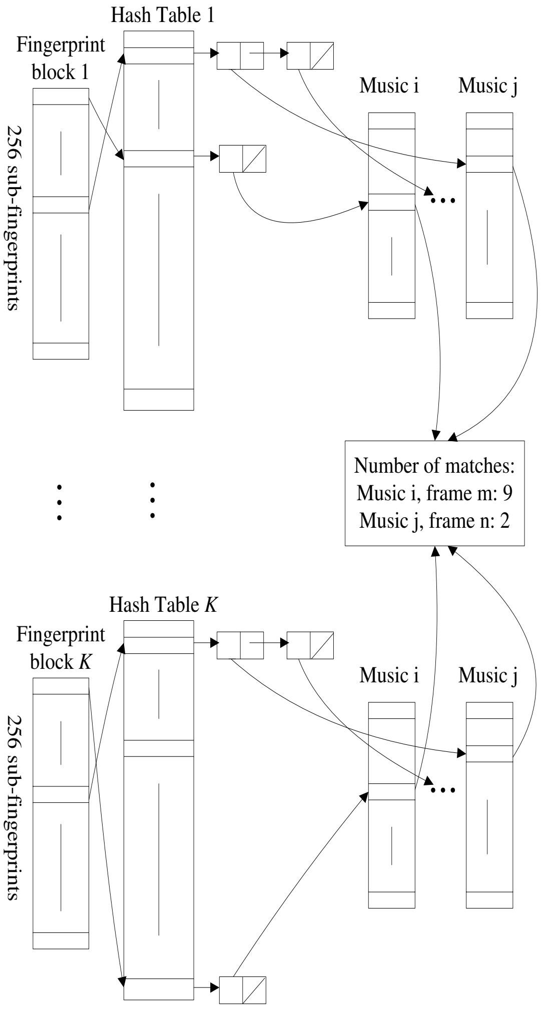

Fig. 5. Database matching stage of the SEH algorithm

IV. Experimental Results

In implementing the system, the DCT length L was set to be 16 to provide a good compromise between the frequency and temporal resolutions. As for K, we used $K = 4$ since more than 90% of the total energy was found to concentrate on the first 4 coefficients from the experiment. The threshold for BER was set to be 0.35 as in [1]. Finally, to speed up the computation and further reduce the computation load, we applied a running DCT algorithm [11] since computation of DCT shifts one sample each time.

Several experiments were conducted to evaluate the performance of the SEH algorithm. In the experiments, a database consisting of 1500 music files extracted from commercial compact discs is constructed. The genres include classical, pop and rock/roll, and the average audio length is 4 minutes. As for the test sets, 1200 query audio clips with the length of about 3 seconds were randomly chosen from the database. The following distortions were applied to the query audio clips to compare the robustness of the two algorithms:

Set 1: Additive white noise with the SNR at 5db.
Set 2: Additive white noise with the SNR at 0db.
Set 3: Playing and recording in a very quiet environment.
Set 4: Playing and recording in office noise environment.

The hash tables used in SEH are denoted as HT1, HT2, HT3 and HT4, and the only table used in PRH is denoted as HT0. Here HTk was built from the k-th DCT coefficient; for example, HT1 was constructed from the DC components. For each query set, the SEH algorithm using various combinations of hash tables were tested along with PRH. The recognition rate (in percentage) for each algorithm is given in Table I. Note that 'HT' of HTk is omitted in the table.

Since the temporal filter used in the PRH algorithm is a high pass filter while in the SEH algorithm a set of low-pass filters are used, it can be inferred that the hash values will reflect more stable information in the temporal axis which would be less influenced by the disturbances at the frame. As a result, the results of the SEH for clips recorded in real noise which was not stationary should display higher robustness and thus better recognition rate compared to the PRH algorithm. It is supported by the results in Table I. Although the results using HT1 in SEH are compatible with the results from PRH for the first three query sets, the result for set 4 is significantly higher than that of PRH. Also note that the recognition rates increased significantly as more hash tables are used in the experiment. However, it is worth pointing out that,if more hash tables are used, the memory usage and the computational burden also increases. Thus it requires careful consideration how many hash tables are needed according to the environment in which the audio fingerprinting would be used.

V. Conclusions

In this paper, we present a novel audio fingerprinting technique based on hashing of the subband envelops in the spectrogram. The temporal filter in the PRH algorithm is substituted by DCT, and multiple hash tables are built for the corresponding lower-ordered DCT coefficients. Experimental results have shown that the proposed SEH algorithm outperformed the conventional PRH algorithm under various conditions. Future work may include retaining the accuracy of the SEH algorithm while using a reduced number of subfingerprints and hash tables, and deriving a mathematical framework for the performance of the SEH algorithm.

TABLE I
Recognition rates (%) of the PRH and SEH algorithms with different combinations of hash tables

Algorithm	Hash tables used	Set 1	set 2	set 3	set 4
PRH	0	97.75	92.25	96.83	34.75
SEH	1	98.08	93.33	96.08	49.08
	2	98.00	90.67	94.92	30.17
	3	97.75	88.42	93.50	20.58
	4	98.17	90.08	94.42	23.83
	1,2	98.75	96.26	98.50	62.92
	1,3	98.92	95.92	98.75	59.25
	1,4	98.67	96.58	98.67	60.58
	2,3	98.67	93.42	96.50	40.00
	2,4	98.75	94.67	96.67	42.25
	3,4	99.00	93.42	96.17	35.00
	1,2,3	99.08	97.00	98.92	68.83
	1,2,4	99.00	97.75	98.83	70.25
	1,3,4	99.17	97.33	99.08	67.00
	2,3,4	99.08	95.42	97.17	49.50
	1,2,3,4	99.25	98.08	99.08	75.00

Acknowledgment

This work was supported in part by the Korea Research Foundation Grant funded by the Korean Government(MOEHRD) (KRF-2008-313-D00783) and in part by the Korea Science and Engineering Foundation (KOSEF) Grant funded by the Korean Government (MOST) (No. R0A-2007-000-10022-0).

References

[1] J. Haitsma and T. Kalker, "A highly robust audio fingerprinting system," *Proc. 3rd Int. Conf. Music Information Retrieval*, pp. 107-115, Oct. 2002.

[2] P. Cano, E. Batlle, T. Kalker, and J. Haitsma, "A review of audio fingerprinting," *Journal of VLSI Signal Processing*, vol. 41, no. 3, pp. 271-284, Nov. 2005.

[3] F. Balado, N. Hurley, E. McCarthy, and G. Silvestre, "Performance analysis of robust audio hashing," *IEEE Trans. Information forensics and security*, vol. 2, no. 2, pp. 254-266, Jun. 2007.

[4] P. Doets and R. Lagendijk, "Distortion estimation in compressed music using only audio fingerprints," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 302-317, Feb. 2008.

[5] M. Park, H. Kim, Y. Ro, and M. Kim, "Frequency filtering for a highly robust audio fingerprinting scheme in a real-noise environment," *IEICE Transactions on Information and Seystems*, vol. E89-D, no. 7, pp. 2324-2327, Jul. 2006.

[6] Y. Ke D. Hoiem, and R. Sukthankar, "Computer vision for music identification," *Proc. Computer Vision and Pattern Recognition*, vol. 1, pp. 597-604, 2005.

[7] S. Baluja and M. Covell, "Content fingerprinting using wavelets", *Proc. Conf. Visual Media Production*, pp. 198-207, Nov. 2006.

[8] S. Baluja and M. Covell, "Audio fingerprinting: Combining computer vision and data stream processing, " *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 2, pp. 213-216, Apr. 2007.

[9] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transform," *IEEE Trans. Computers*, pp. 90-93, Jan. 1974.

[10] K. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*. Academic Press, 1990.

[11] J. Xi and J. Chicharo, "Computing running DCTs and DSTs based on their second-order shift properties," *IEEE Trans. Circuits and Systems-I: Fundamental Theory and Applications*, vol. 47, no. 5, pp. 779-783, May 2000.