Digital audio watermarking robust against Locality Sensitive Hashing

Kotaro Sonoda and Kentaro Morisaki

Graduate school of engineering, Nagasaki University, Bunkyo-machi 1-14, Nagasaki,
852-8521 Nagasaki, Japan
sonoda-iihmsp16@cis.nagasaki-u.ac.jp

Abstract. Audio fingerprinting is an audio feature vector and a technique to identify the music source by comparing with original fingerprints in a fingerprint database. The fingerprints are often hashed to shorten codes. The content similarities between fingerprints are maintained in the hashed code. Such a hash function family is called Locality Sensitive Hashing (LSH). Because audio fingerprint using LSH is known to be resistant against CPO (Content Preserving Operations, perceptively acceptable manipulations) such as compression, noise adding, mean filtering, it is possible to identify the original source even if the source was slightly modified.

On the other hand, mixed arrangement (mashup) of several music sources is allowed as legitimate artistic expression. In the conventional fingerprint based retrieval system, the mixed arrangements could identify the origins segmentally but the arranger's authorization is ignored.

In this report, we propose an audio watermarking method robust against LSH coding. That is, the arranger information is watermarked in the audio signal and it is detectable from not only stego audio signal but also stego audio fingerprint of LSH.

Keywords: Audio Fingerprinting, Locality Sensitive Hashing, Audio watermarking

1 Introduction

1.1 Identification of the music source

In regard to using a similar music search system, to identify the music source, an audio fingerprinting technique is often used. The audio fingerprint is an inherent feature of the audio signal like a human fingerprint or biometric feature. As the audio feature vector, for example, one can utilize the short-time Fourier transform (STFT) coefficients, the chroma codes, and zero-crossing intervals and so on.

The similarity is evaluated in the distance norm between the vectors in the feature domain space. However, in the case that the vector dimension is high, the calculation of the norm also becomes high. Therefore, the feature vector is shortened by hashing technique which fulfills both keeping the dimension small and yet providing a high level of accuracy in similarity measurement.

Fridrich et al. proposed a robust hash function for still image fingerprinting [2]. In their study, the still image hash is constructed by projecting its each descrete cosine transform (DCT) block on the zero-mean random patterns. Their hash function is robust against CPO (Content Preserving Operations, visually acceptable manipulations) such as compression, noise adding, and mean filtering.

Radhakrishnan et al. proposed audio signature hashing by applying Fridrich's hashing method to the audio spectrogram [4].

1.2 Detection of arrangement

By using the above mentioned audio fingerprinting techniques, the music's original sources are identified in the hashing domain. However, we should consider arrangements of music source. How should the arranged music be differenciated from the original music sources? Particularly, how should we treat mashup music (e.g. one made by concatenating several music sources)?

There are three options that may be considered.

- One possible option is to ignore the original sources and fingerprint the mashup as an independent music sources.
- The second option is to ignore the mashup and preserve the original fingerprints of the original sources separately.

In these two options, both the arranger and composer cannot coexist. We would propose another third option.

- Detecting every original fingerprint of the original sources separately as well as detect the watermark of the arrangement from the fingerprints.

In the third options, one can detect both the originals and the arrangement as shown in Fig.1.

In this concept, the arranger S3 embeds his watermark into the arranged source before broadcasting it. In the authorization checking system, the hashed audio fingerprints are extracted from the uploaded arranged source and the segmental fingerprints identify the original sources S1 and S2 as in conventional fingerprint retrieval. Moreover as in conventional watermarking method, the arranger S3's watermark is detected by analyzing the uploaded music signal before the hashing. Providing that the watermarking scheme is robust against the hashing, the hashed audio fingerprints preserve the arranger's watermark and the arranger S3's watermark is detected from the hashed fingerprints.

To utilize this third option, we propose a digital audio watermarking scheme that is robust against Locality Sensitive Hashing in this paper. In section 2, detail of the Locality Sensitive Hashing of the audio fingerprinting is introduced. And we propose a watermarking scheme that is robust against LSH in section 3. The inaudibility and the robustness of the proposed method is evaluated in section 4. The paper is brought to a conclusion in section 5.

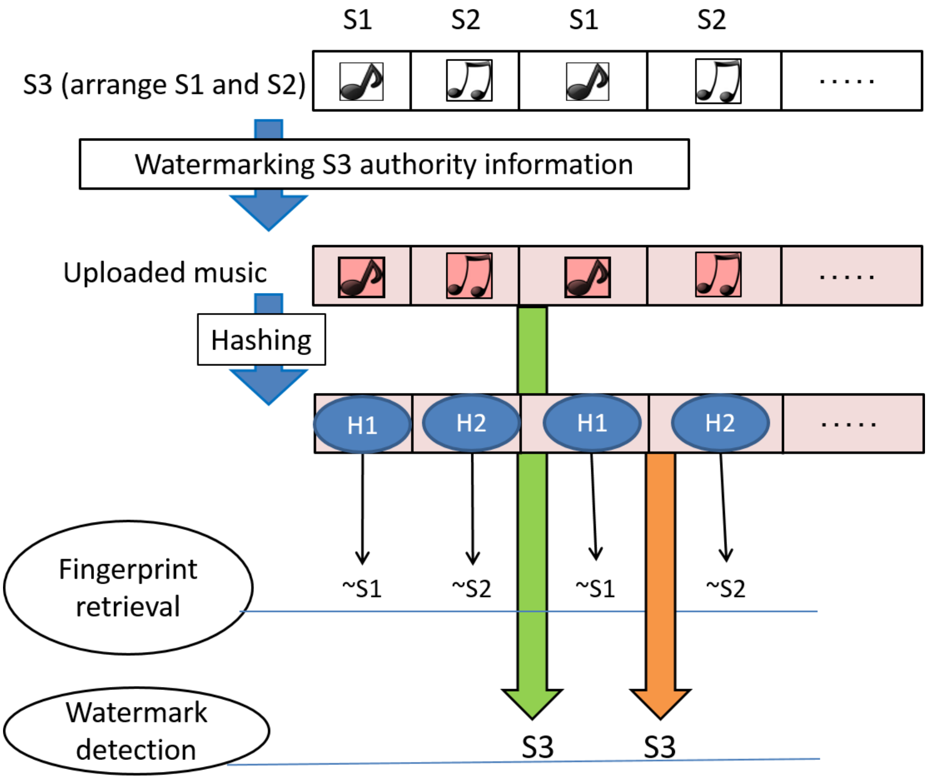

Fig. 1. Proposed system: Identification of each original and watermark detection of the arrangement

2 Audio fingerprinting based on STFT and Local sensitive hashing (LSH)

In this section, we introduce the audio fingerprinting scheme based on STFT and LSH which is proposed by Regunathan et al. [4]. Fig.2 shows the block diagram of Regunathan's LSH audio fingerprinting.

2.1 Audio feature vector based on STFT spectrogram

The audio signal is segmented into chunks (time length of a chunk is T_{ch}) with overlapping T_o. Similarity measurement is carried out with chunk as an unit. In every chunk, segmented signal $(X_i, i = 1, 2, \ldots, T_{ch})$ is transformed to short-time Fourier coefficients and concatenated. The concatenated coefficient matrix (fine spectrogram $\mathbf{S}$) is segmented to $F \times T$ blocks and their blocks are averaged in both the time and frequency domains, and as a result we can get a coarse

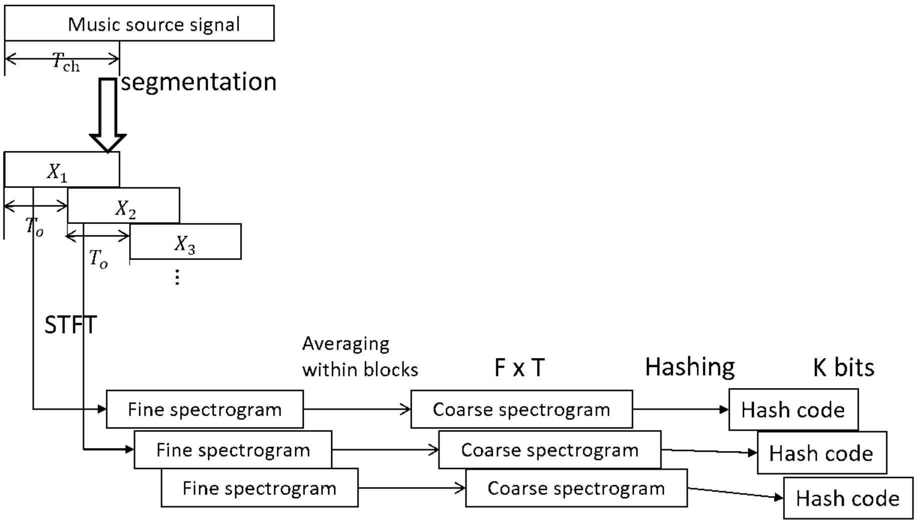

Fig. 2. LSH fingerprinting

spectrogram $\mathbf{Q}$ (size: $F \times T$). Through this averaging process, the spectrogram is expected to be robust against slight modifications.

Therefore, the resulting coarse spectrogram $\mathbf{Q}$ is calculated from the fine spectrogram $\mathbf{S}$ as Eq.1:

$$Q_{k,l} = \frac{1}{W_f * W_t} \sum_{i=(k-1)W_f}^{k \cdot W_f} \sum_{j=(l-1)W_t}^{l \cdot W_t} S_{i,j}, \\ k = 1, 2, ...F; \quad l = 1, 2, ...T, \tag{1}$$

where W_t and W_f are the matrix length and width of an average block size in the fine spectrogram, respectively.

This coarse spectrogram $\mathbf{Q}$ is produced for every chunk.

2.2 Locality Sensitive Hashing

The coarse spectrogram $\mathbf{Q}$ is an audio feature vector and it can be used to evaluate the audio similarity by calculating the vector distance norm from other audio content. However, the precise similarity requires a $\mathbf{Q}$ with a higher dimension of F and T. Therefore, the feature vector is transformed to be short in keeping with their geometric features using the Locality Sensitive Hashing technique. Radhakrishnan et al. proposed a hashing scheme for music signal [4].

In their method, the coarse spectrogram $\mathbf{Q} \in \mathbb{R}^{F \times T}$ is hashed to a K-bits code $\mathbf{H} \in \mathbb{B}^K$. Initially K random matrices $\mathbf{P}^{(k)} \in \mathbb{B}^{F \times T}, k = 1, 2, \ldots K$ are prepaired. They are equally distributed in $[-1, 1]$. Using these random matrices, K-bits hashed code $\mathbf{H}$ is calculated as following Eq.3,

$$h_k = \sum_{i=1}^{F} \sum_{j=1}^{T} Q_{i,j} * P_{i,j}^{(k)} \tag{2}$$

$$H_k = \begin{cases} 1 & \text{if} \quad h_k > \text{Median}(\mathbf{h}), \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

3 Audio watermarking method robust against LSH

As our goal, we wished to detect the original music sources and simultaneously detect their arrangement authority by analyzing hashed audio features. Moreover, we are required to detect them by analyzing the raw signal before hashing.

In conventional watermarking methods, it is possible to embed and detect robustly against some signal processing attacks, such as MP3 coding, noise adding and resampling. However the hashing process had never been considered.

The hashing process is a mapping process whereby the original signal is transformed by a random matrices set of variables, and their mapped components are fully quantized. Because of that, segment-based watermarks are randomized and can't be identified autonomously.

The hashing processes every segment by using the same hashing matrices set. While the hashed codes are randomized in a segment, the similarity relations between each couple of segments are expected to be preserved in the similarity measurement of hash codes after LSH. Therefore, the watermarking embedded in the relationship between time series segments is expected to be preserved. As a such type of watermarking method, the simplest one is the echo hiding method [3].

In this study, we apply conventional echo hiding to the system.

3.1 Embedding process

In the conventional echo hiding method, the host signal is added slight gained echo on about 10 ms delay in order to maintain inaudibility, and the echo kernels are exchanged as according to the embedding rate.

Robust hash echo hiding we propose is the same process as the conventional method but it adds slight echoes on about over segment unit (e.g., two segments length delayed echo is added for watermark bit '0' and three segments length for bit '1') and the embedding bit rate is about 1 bps. That is the delay times of echo are large compared to the conventional echo hiding. Such a large delay time of echoes distort the audio signal. However we study the robust detection method rather than inaudibility in this report.

3.2 Detection process

Before LSH coding, the detection process is also the same as that of conventional echo hiding. And its robustness is inherently the same as the conventional method.

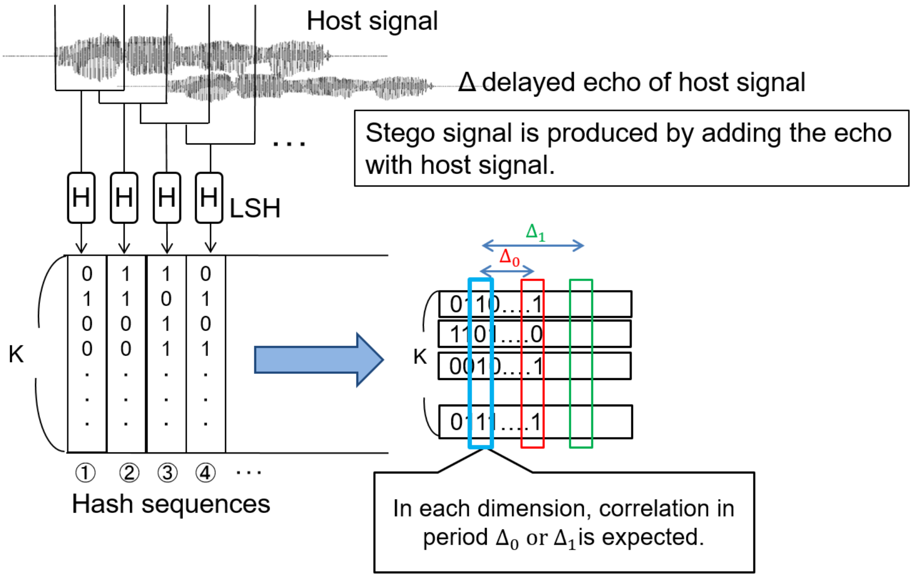
Host signal
Δ delayed echo of host signal
Stego signal is produced by adding the echo with host signal.
H
H
H
H
LSH
K
0 1 0 0
1 1 0 0
1 0 1 1
0 1 0 1
①
②
③
④
Hash sequences
Δ_1
Δ_0
K
0110....1
1101....0
0010....1
0111....1
In each dimension, correlation in period Δ_0 or Δ_1 is expected.

Fig. 3. Detection process

After LSH coding, each segment is mapped to orthogonal K-dimensional codes. When $\varDelta$ segments delayed echo was added, a similar bit is expected to appear every $\varDelta$ segments in the hashed codes. In this study, we tried to detect the watermark by finding peaks in the averaged auto-correlation of each K's dimension as shown in Fig. 3

4 Experiment

To evaluate the robustness against LSH coding, we tried to detect watermarks from LSH coded sequences of the music sources. The testing music source is embedded random payload by echo hiding and converted to the LSH code sequences.

The LSH configurations are the same as Radhakrishan's settings [4]:

- Segment length (Chunk Size T_{ch}): 133 ms (6400 samples at 48 kHz sampling)
- Segmentation stepping length (T_0): 10 ms (512 samples at 48 kHz sampling)
- Spectrogram resolution : Fine (128 × 25) ⇒ Coarse (F:20 × T:10)
- Hash code size (K) : 18

Testing music source is selected from SQAM (Sound Quality Assessment Material) set and 8 tracks are used [1]. Table 1 shows the evaluated-source list. In this report, any source mixing arrangement is not carried out.

Table 1. Evaluated source lists from SQAM

Track	SQAM
27	Castanets
32	Triangles
35	Glockenspiel
40	Harpsichord
65	Orchestra
66	Wind ensemble
69	ABBA
70	Eddie Rabbit

Table 2. Configurations for LSH robust echo hiding

Gain α	0.3
Delay time for watermark ”0” Δ_0	25 segments
Delay time for watermark ”1” Δ_1	40 segments
Frame length for 1 bit embedding L	62 segments
Embedding bit rate	1.15 bps

The configurations for echo hiding are listed in Table 2. Note the delay times for watermarks are set as very long as the length of 40 segments (523 ms). Such a long delay echo may deteriorate inaudibility, but it is overlooked in this experiment.

As a result of experiment, the averaged detection bit error rate (BER) is 46.2%. All the eight testing stego signals show almost similar detection bit error rates. This is clearly unacceptably high and only slightly below the chance level.

However some frames were successfully embedded and detected. Such an example of correctly detected correlation peak is shown in Fig.4. Therefore, further improvement of the embedding and detecting method is clearly essential.

5 Conclusion

Audio fingerprinting generates a hash code for every segment based on an audio feature vector. However, the mixed arrangement is ignored in the audio hash. In this report, we proposed to detect an additional arrangement authority by embedding a watermark. To utilize this protocol, we tried to apply the echo hiding watermarking method that is robust against Locality Sensitive Hashing.

However the proposed watermarking method outlined in this report is not yet full functional. Further improvement of the embedding and detecting method is clearly essential.

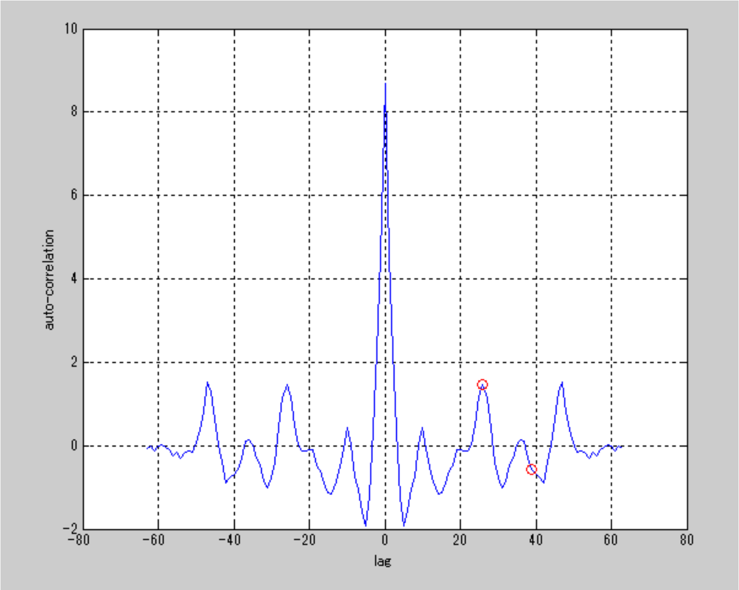

Fig. 4. Averaged auto correlation graph of successfully detected frame. Higher peak value at $\Delta_0 = 25$ or $\Delta_1 = 40$ (two red circles) indicates bit "0" or "1". In this example's case, Peak is found at Δ_0 (watermark bit "0").

References

1. European Broadcasting Union (EBU): SQAM (2008), `http://tech.ebu.ch/publications/sqamcd`
2. Fridrich, J., Goljan, M.: Robust hash functions for digital watermarking. In: In Proceeding of ITCC (May 2000)
3. Gruhl, D., Lu, A., Bender, W.: Echo hiding. In: In Proceeding of International Workshop on Information Hiding. pp. 295–315 (1996)
4. Radhakrishnan, R., Bauer, C., Cheng, C., Terry, K.: Audio signature extraction based on projections of spectrograms. In: In proceeding of IEEE International Conference on Multimedia and Expo (ICME). pp. 2110–2113 (2007)