IMPROVING AUDIO WATERMARK ROBUSTNESS USING STRETCHED PATTERNS AGAINST GEOMETRIC DISTORTION

Ryuki Tachibana

Tokyo Research Laboratory, IBM Japan
1623-14, Shimotsuruma, Yamato-shi, Kanagawa-ken 242-8502, Japan
ryuki@jp.ibm.com

ABSTRACT

One of the problems for audio watermarks is robustness against signal processing causing de-synchronization of the pseudo-random sequences. To tackle the problem, we previously introduced an audio watermarking method using a two-dimensional pseudo-random array, which is robust against pitch shifting and random stretching to some extent. In this paper, we explain a modification to the detection algorithm to improve the robustness against excessive distortion. The method uses multiple pseudo-random arrays each of which is stretched assuming a certain amount of distortion. Since most of the detection process for the multiple arrays is shared, the additional computational cost is limited.

1. INTRODUCTION

Robustness of image watermarks against geometric distortion has been gathering increasing attention recently [1, 2]. When the image is rotated, translated, or scaled, the mis-synchronization of the embedded pseudo-random sequence (PRS) and the PRS which the detection algorithm uses prevents the detector from properly correlating the PRSs, and leads to serious damage on watermark detection.

Similarly, there is a class of audio signal processing which effects on audio watermark as geometric distortion effects on image watermark. Audio processing such as pitch shifting, random stretching, and wow-and-flutter changes the time and frequency property of the embedded PRS, and is hence difficult for watermark to survive. To tackle this problem, several audio watermarking techniques modifying magnitudes in the frequency domain were proposed in the past two years [3, 4, 5]. We introduced an audio watermarking method that is robust against pitch shifting and random stretching up to $\pm 4\%$ without any exhaustive search for the scale change in [6]. However, it was still difficult for the method to survive excessive geometric distortions. [7, 8] solved the problem by performing multiple correlation tests. One possible problem with the multiple correlation tests is that that may increase the false alarm rate.

In this paper, we modify the detection algorithm of [6] and improve the robustness against excessive geometric distortions. While multiple correlations are calculated also in the paper, the proposed algorithm chooses one correlation by the strength of a synchronization signal. Because the strength of the synchronization signal and that of the message signal are independent, the false alarm rate is preserved. Furthermore, because the same synchronization signal, which was necessary for the original detection algorithm to search the head of the message, is used for the scale selection, this method does not decrease the data payload.

2. THE PREVIOUS METHOD

In this section, we summarize the method introduced by [6].

The method can embed a multiple-bit message in the content by dividing it to short messages and embedding each of them in *a pattern block*. The patter block is defined as a two-dimensional segmented area in the time-frequency plane of the content (Fig. 1), which is constructed from the sequence of power spectrums calculated using short-term DFTs. A pattern block is further divided into *tiles*. We call tiles in row a *subband*. A tile consists of four consecutive overlapping DFT frames. A pseudo-random number is selected corresponding to each tile. If the pseudo-random value assigned to a tile is positive, the embedding algorithm increases the magnitudes of frequency bins of the first two frames in the tile and decreases those of the last two frames.

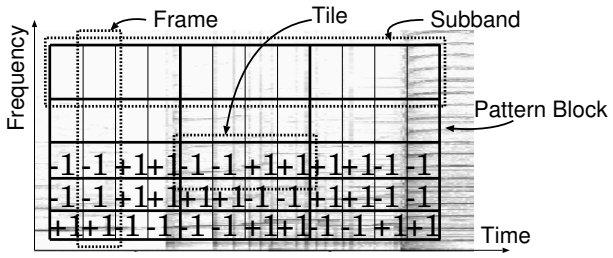

Fig. 1. A pattern block

The detection algorithm calculates the magnitudes for all tiles of the content and correlates them with the pseudo-random array by applying the following steps.

1. Windowing DFT
   The magnitude $a_{t,f}$ of the f-th frequency in the t-th frame of a pattern block of the content is calculated by the DFT analysis of a frame of the content. A frame overlaps the adjacent frames by a half window.

2. Normalization
   The magnitudes are then normalized by the average of the magnitudes in the frame. A normalized magnitude is

$$\widehat{a}_{t,f} = \frac{a_{t,f}}{\frac{1}{N_{PCM}/2}\sum_{f=1}^{N_{PCM}/2} a_{t,f}}. \quad (1)$$

   The difference between the logarithmic magnitudes of a frame and the next non-overlapping frame is taken as $P_{t,f} = \log \widehat{a}_{t,f} - \log \widehat{a}_{t,f+2}$.

3. Magnitudes of tiles
   The magnitude of a tile located at the b-th subband of the t-th frame in the block is calculated by

$$Q_{t,b} = \frac{\sum_{f=f_b^L}^{f_b^H} P_{t,f}}{f_b^H - f_b^L + 1}, \quad (2)$$

   where f_b^H and f_b^L are the highest and lowest frequencies in the b-th subband, respectively.

4. Watermark strength
   The detected watermark strength for the j-th bit in the tile is calculated as the cross-correlation of the pseudo-random numbers and the normalized magnitudes of the tiles by

$$X_j = \frac{\sum_{k=1}^{D_B} \omega_{j,k}^B (Q_{t,b} - \overline{Q})}{\sqrt{\sum_{k=1}^{D_B} \left\{\omega_{j,k}^B (Q_{t,b} - \overline{Q})\right\}^2}}, \quad (3)$$

   where $\overline{Q} = \frac{1}{D_B}\sum_{k=1}^{D_B} Q_{t,b}$, D_B is the number of tiles assigned for a bit, and $\omega_{j,k}^B$ is the k-th pseudo-random number of the j-th bit corresponding to the tile at the b-th subband in the t-th frame. Similarly, the synchronization strength is calculated by

$$S = \frac{\sum_{k=1}^{D_S} \omega_k^S (Q_{t,b} - \overline{Q})}{\sqrt{\sum_{k=1}^{D_S} \left\{\omega_k^S (Q_{t,b} - \overline{Q})\right\}^2}}. \quad (4)$$

   Because this detection algorithm calculate the difference of magnitudes of a frame and the next non-overlapping frame per tile, the pattern block regarding detection can be illustrated as Fig. 2.

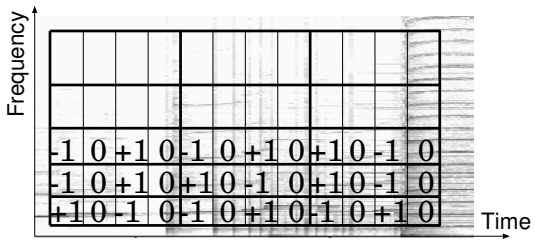

Fig. 2. A detection pattern block

3. IMPROVING ROBUSTNESS

In this section, we explain a method to improve the robustness of the audio watermarking method against pitch shifting and random stretching. In the experiment, we used a software system that can embed and detect a 64-bit message in 30-second pieces of music. Its details and parameters that are not explained below are same as explained in [6]. All the following graphs are experimental results using ten 100-second music samples. The watermark strength data plotted in the figures or shown in the table are measured after the accumulation of doubly-encoded watermarks. *Pitch shifting* is performed using linear interpolation without anti-alias filtering. *Random stretching*[1] is a transformation that changes the length of the total content to a different length by omitting or inserting a random number of sample blocks from 50 up to 500 samples per block.

When the content is distorted by pitch shifting or random stretching, the time and frequency location of the embedded tiles are displaced. Accordingly, our idea for improving robustness is to detect watermark using multiple patterns each of which is stretched in advance assuming a certain amount of distortion. As for random stretching, because it changes the length of the content, watermark is expected to be detectable using a pattern that is also stretched

[1] *Random sample cropping* can be considered as random stretching with the target length smaller than 100%.

with respect to time(Fig. 3). The watermark strength using the pattern that is stretched at the rate of r_t is calculated by

$$X_j = \frac{\sum_{k=1}^{D_B} \omega_{j,k}^B (Q_{\lfloor r_t t + 0.5 \rfloor, b} - \overline{Q})}{\sqrt{\sum_{k=1}^{D_B} \{\omega_{j,k}^B (Q_{\lfloor r_t t + 0.5 \rfloor, b} - \overline{Q})\}^2}}. \quad (5)$$

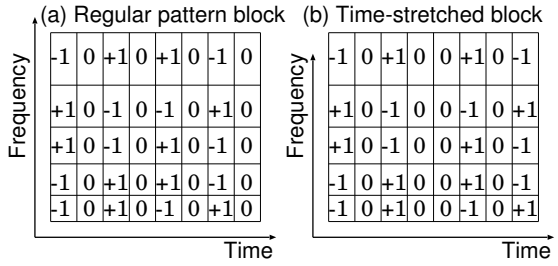

Fig. 3. Pattern block stretched with respect to time

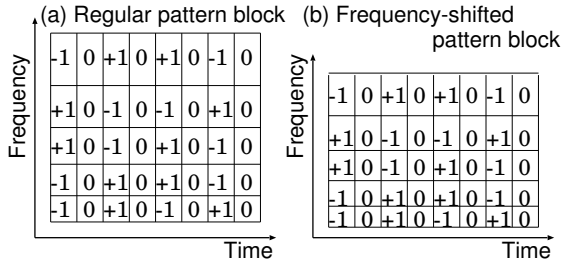

Fig. 4. Pattern block stretched with respect to frequency

To detect watermark from a sample whose pitch is shifted at the rate of r_f, we correspondingly shift the subbands (Fig. 4) as $f_b^{L'} = r_f f_b^L$ and $f_b^{H'} = r_f f_b^H$. Moreover, since linear pitch shifting changes the duration of the block as well as its frequency, we also stretch the pattern using Eq. 5.

In this way, we define a *stretched detector*, $D(r_t, r_f)$, which matches best to a time expansion rate, r_t, and a frequency shifting rate, r_f. Figure 5 and Figure 6 show mean watermark strengths detected by $D(0.90, 1.00)$, $D(0.935, 1.07)$, $D(1.00, 1.00)$, $D(1.075, 0.93)$, and $D(1.10, 1.00)$ from distorted content. While the strength detected by the regular detector, $D(1.00, 1.00)$, decreases as the content is severely distorted, the stretched detectors have their maximum strengths approximately at their assumed distortion rates.

These experiments indicate that if several stretched detectors detect watermark in the music sample using differently stretched patterns in parallel, and an appropriate stretched detector is selected, we can detect watermark even from an excessively distorted music sample by some of the stretched

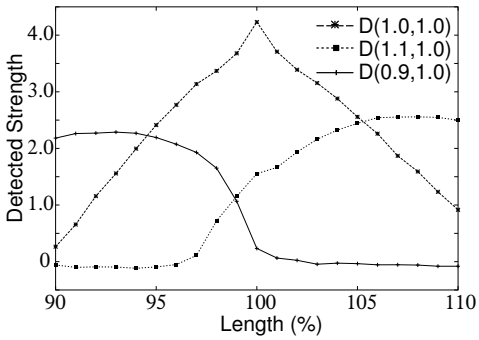

Fig. 5. Mean strengths from randomly stretched samples

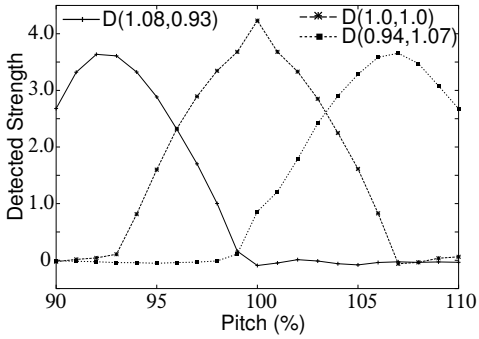

Fig. 6. Mean strengths detected from pitch-shifted samples

detectors. In this way, the detection flow becomes as shown in Fig. 7. The selection of a stretched detector is done approximately every 30 seconds based on the *accumulated synchronization strength*,

$$A^{(i)} = \frac{1}{\sqrt{N_S}} \sum_{n=1}^{N_S} S_n^{(i)}, \quad (6)$$

where i is the index of stretched detectors, n is the index of synchronization signals detected in the 30-second period, and N_S is the number of synchronization signals detected in the period. After the stretched detector that gives the maximum $A^{(i)}$ is selected, the watermark strengths detected from the stretched detector are used for the message reconstruction.

The mean of $A^{(i)}$ detected from five stretched detectors are shown in Fig. 8 and Fig. 9. It can be seen in Fig. 9 that, for example, $D(1.08, 0.93)$ is selected for the pitch-shifting rate ranging from 90% up to 96%.

Consequently, the mean strengths by the selected stretched detectors become as shown in Fig. 10 and Fig. 11, and are enough high for every degree of tested distortion. Corresponding bit error rates are also shown in the figures.

Table 1 shows (1) the means of the detected strengths, (2) the bit error rates (BER) which are plotted in Fig.10 and Fig.11, and (3) the correct detection rates (CDR) at which

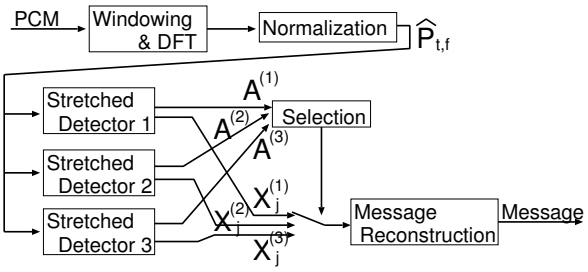

Fig. 7. Detection flow with three stretched detectors

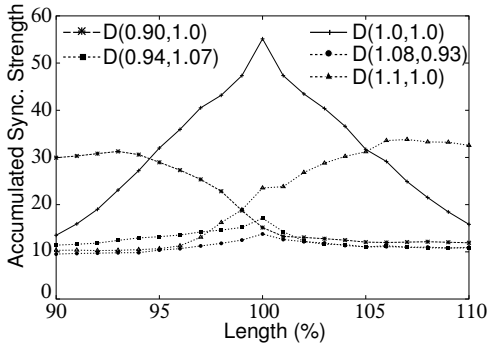

Fig. 8. Mean accumulated sync. strength detected from randomly stretched music samples

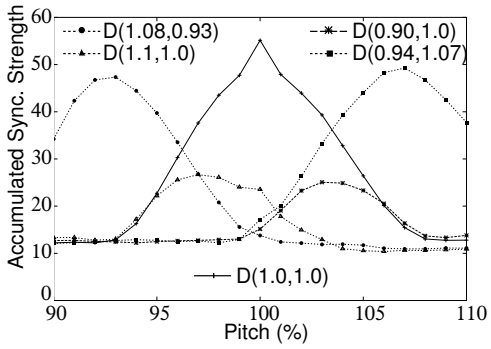

Fig. 9. Mean accumulated sync. strength detected from pitch-shifted music samples

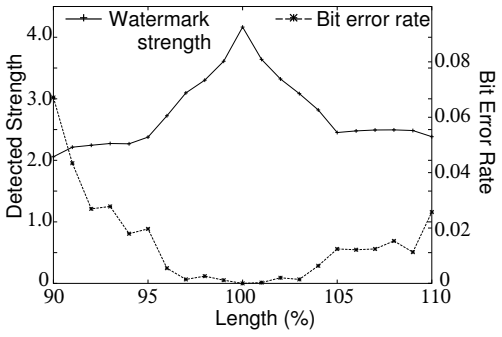

Fig. 10. Mean strength and bit error rate(BER) for randomly stretched samples

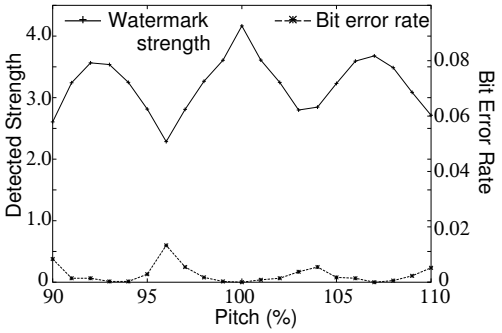

Fig. 11. Mean strength and BER for pitch-shifted samples

correct 64-bit message was detected. CDR over 80% were seen for every one of the tested degradation. The error correction and detection algorithm and the counting of weak bits successfully avoided detection of an incorrect message.

Table 1. The means of the detected strengths (μ), the bit error rates (BER), and the correct detection rates (CDR)

Processing	μ	BER	CDR
Original watermark	4.17	0.000	100%
Pitch shifting -10%	2.61	0.008	96%
Pitch shifting -8%	3.56	0.001	100%
Pitch shifting -6%	3.25	0.000	100%
Pitch shifting -4%	2.29	0.013	90%
Pitch shifting -2%	3.27	0.002	100%
Pitch shifting +2%	3.25	0.001	100%
Pitch shifting +4%	2.85	0.005	100%
Pitch shifting +6%	3.59	0.001	100%
Pitch shifting +8%	3.49	0.001	100%
Pitch shifting +10%	2.71	0.005	100%
Random stretching -10%	2.06	0.067	83%
Random stretching -8%	2.25	0.027	87%
Random stretching -6%	2.27	0.018	87%
Random stretching -4%	2.72	0.005	100%
Random stretching -2%	3.30	0.003	100%
Random stretching +2%	3.32	0.002	100%
Random stretching +4%	2.82	0.006	100%
Random stretching +6%	2.48	0.012	93%
Random stretching +8%	2.49	0.015	93%
Random stretching +10%	2.38	0.026	87%

Performance We also measured the detection speed using a PC with 600 MHz Pentium III running Windows NT. While, when the detector uses only a regular detector, detection takes 7.72% of the length of the content, when the detector uses five stretched detectors, it is performed within 8.96% of the length of the content. That is only 16% in-

crease for four more stretched detectors. This is because the Fourier transform and calculation of the normalized magnitudes take most of the processing time, and the stretched detectors can share this part of calculation. Therefore, using stretched detectors in a detector is much faster than simply using multiple detectors.

4. SUMMARY

We improved robustness of our previous audio watermarking method by using multiple stretched pattern blocks. With the improvement, robustness against pitch shifting and random stretching up to $\pm 10\%$ was achieved with only 16% additional computational time. Further improvement is required to shorten the duration of content required to carry a message.

5. REFERENCES

[1] M. Kutter, "Watermarking resisting to translation, rotation, and scaling," in *Proc. of SPIE Int. Conf. on Multimedia Systems and Applications*, 1998, vol. 3528, pp. 423–431.

[2] C-Y. Lin, M. Wu, J.A. Bloom, I.J. Cox, M.L. Miller, and Y-M. Lui, "Rotation, Scale, and Translation Resilient Public Watermarking for Images," in *Proc. of SPIE Int. Conf. on Security and Watermarking of Multimedia Contents II*, San Jose, USA, January 2000, vol. 3971, pp. 90–98.

[3] J. Haitsma, M. van der Veen, F. Bruekers, and T. Kalker, "Audio Watermarking for Monitoring and Copy Protection," in *Proc. of ACM Multimedia*, Marina del Rey, USA, November 2000, pp. 119–122.

[4] C-P. Wu, P-C. Su, and C-C. J. Kuo, "Robust and efficient digital audio watermarking using audio content analysis," in *SPIE Int. Conf. on Security and Watermarking of Multimedia Contents II*, San Jose, USA, January 2000, vol. 3971, pp. 382–392.

[5] D. Kirovski and H. Malvar, "Robust spread-spectrum audio watermarking," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Salt Lake City, Utah, USA, May 2001, pp. 1345–1348.

[6] R. Tachibana, S. Shimizu, T. Nakamura, and S. Kobayashi, "An audio watermarking method robust against time- and frequency-fluctuation," in *SPIE Conf. on Security and Watermarking of Multimedia Contents III*, San Jose, USA, January 2001, vol. 4314, pp. 104–115.

[7] D. Kirovski and H. Malvar, "Spread-spectrum audio watermarking:requirements, applications, and limitations," in *IEEE Forth Workshop on Multimedia Signal Processing*, Cannes, France, October 2001, pp. 219–224.

[8] D. Kirovski and H. Malvar, "Robust covert communication over a public audio channel using spread spectrum," in *4th Int. Workshop on Information Hiding*, Pittsburgh, USA, April 2001, vol. LNCS 2137, pp. 354–368.