Digital Audio Watermarking Technique Using Pseudo-Zernike Moments

Xiangyang Wang[1], Tianxiao Ma[1], and Panpan Niu[2]

[1] School of Computer & Information Technology, Liaoning Normal University,
Dalian 116029, China
[2] School of Information Science &Technology, Dalian Maritime University,
Dalian, 116026, China
{wxy37,tianxiao87,niupanpan333}@126.com

Abstract. Based on Pseudo-Zernike moments and synchronization code, we propose a new digital audio watermarking algorithm with good auditory quality and reasonable resistance toward de-synchronization attacks in this paper. Simulation results show that the proposed watermarking scheme is not only inaudible and robust against common signals processing such as MP3 compression, noise addition, re-sampling, and re-quantization etc, but also robust against the de-synchronization attacks such as random cropping, amplitude variation, pitch shifting, etc.

1 Introduction

Due to the advent of network and computer technology, there has been an explosion in growth of the use of digital media through electronic commerce and on-line services. Since digital media is easily reproduced and manipulated, anyone is potentially capable of incurring considerable financial loss. Digital watermarking is introduced to safeguard against such loss. [1].

Nowadays, there is an unprecedented development in the audio watermarking field. On the other hand, attacks against audio watermarking systems have become more sophisticated [2]. In general, these attacks can be categorized into common signal processing and de-synchronization attacks. Most of the previous audio watermarking schemes are robust to common signal processing, but show severe problems to de-synchronization attacks. Fortunately, several approaches against the de-synchronization attacks have been developed in recent years. These schemes [2] can be roughly divided into exhaustive search, invariant watermark, self-synchronization, and synchronization pattern.

Exhaustive search: In [3] and [4], by performing multiple correlation tests, the authors applied the detection engine to search for resynchronization. However, exhaustive search schemes need large amount of calculation, and often cause false alarm. *Invariant watermark*: Mansour et al. [5-6] proposed a time-scale invariant watermarking embedding strategy by changing the relative length of the middle segments between two successive maximum peaks of the smoothed waveform. In [7],

S. Qing, C.J. Mitchell, and G. Wang (Eds.): ICICS 2009, LNCS 5927, pp. 459 – 473, 2009.

by using music content analysis, the authors presented a new audio watermarking method. The watermark is robust to pitch-invariant TSM but vulnerable to the common signal processing. *Self-synchronization*: In [8], self-synchronization was implemented by applying special peak point extraction scheme. In general, the current self-synchronization algorithm cannot extract invariant audio feature steadily. *Synchronization pattern*: In [9], the authors chose Bark code which has better self-relativity as synchronization mark and embedded it into temporal domain. However, this method is vulnerable to some de-synchronization attacks such as amplitude variation, pitch shifting, jittering, time-scale modification (TSM) etc.

Pseudo-Zernike moment is an ideal region-based shape descriptor, with the characteristics of rotation invariant, low noise sensitive, expression effectiveness and fast computation. It has been widely used for pattern recognition, signal analysis and other fields [10]. Based on Pseudo-Zernike moment and synchronization code, we propose a new digital audio watermarking algorithm. The watermark bit is embedded into the average value of modulus of the low-order Pseudo-Zernike moment. Meanwhile combining the two adjacent synchronization code searching technology, the algorithm can extract the watermark without the help from the original digital audio signal.

2 Fundamental Theory and Synchronization

2.1 Fundamental Theory

In our audio watermarking scheme, the watermark can be embedded into the host audio by 3 steps. Firstly, the original digital audio is segmented and then each segment is cut into two parts. Secondly, with the spatial watermarking technique, synchronization code is embedded into the statistics average value of audio samples in the first part. And then, map 1-D digital audio signal in the second part into 2-D form, and calculate its Pseudo-Zernike moments. Finally, the watermark bit is embedded into the average value of modulus of the low-order Pseudo-Zernike moments.

2.2 Synchronization Code

Synchronization is one of the key issues of audio watermarking. Watermark detection starts by alignment of watermarked block with detector. Losing synchronization causes false detection. Time-scale or frequency-scale modification makes the detector lose synchronization. So we need exact synchronization algorithms based on robust synchronization code.

Generally, we should avoid false synchronization during selecting synchronization code. Several reasons contribute to false synchronization: (1) the style of the synchronization code, (2) the length of synchronization code, (3) the probability of “0”and“1”in synchronization code. Among of them, the length of synchronization code is especially important. The longer it is, the more robust it is.

The proposed scheme embeds Barker code in front of the watermark to locate the position where watermark is embedded. Barker codes, which are subsets of PN sequences, are commonly used for frame synchronization in digital communication systems. Barker codes have low correlation side lobes. A correlation side lobe is the correlation of a codeword with a time-shifted version of itself.

3 The Pseudo-Zernike Moments

Pseudo-Zernike moments consist of a set of complex polynomials [11] that form a complete orthogonal set over the interior of the unit circle, $x^2 + y^2 \leq 1$. If the set of these polynomials is denoted by $\{V_{nm}(x, y)\}$, then the form of these polynomials is as follows

$$V_{nm}(x, y) = V_{nm}(\rho, \theta) = R_{nm}(\rho)\exp(jm\theta) \tag{1}$$

where $\rho = \sqrt{x^2 + y^2}$, $\theta = \tan^{-1}(y/x)$. Here n is a non-negative integer, m is restricted to be $|m| \leq n$ and the radial Pseudo-Zernike polynomial $R_{nm}(\rho)$ is defined as the following

$$R_{nm}(\rho) = \sum_{s=0}^{n-|m|} \frac{(-1)^s (2n+1-s)!\rho^{n-s}}{s!(n+|m|+1-s)!(n-|m|-s)!} \tag{2}$$

Like any other orthogonal and complete basis, the Pseudo-Zernike polynomial can be used to decompose an analog 2-D signal $f(x, y)$

$$f(x, y) = \sum_{n=0}^{\infty} \sum_{\{m:|m|\leq n\}} A_{nm} V_{nm}(x, y) \tag{3}$$

where A_{nm} is the Pseudo-Zernike moment of order n with repetition m . Given a 2-D signal of size $M \times N$, its Pseudo-Zernike moments (approximate version) are computed as

$$\hat{A}_{nm} = \frac{n+1}{\pi} \sum_{i=1}^{M} \sum_{j=1}^{N} h_{nm}(x_i, y_j) f(x_i, y_j) \tag{4}$$

where the value of i and j are taken such that $x_i^2 + y_j^2 \leq 1$, and

$$h_{nm}(x_i, y_j) = \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} V_{nm}^{*}(x, y)\, dx dy \tag{5}$$

where $\Delta x = \frac{2}{M}$, $\Delta y = \frac{2}{N}$, $h_{nm}(x_i, y_j)$ can be computed to address the nontrivial issue of accuracy. In this research, we adopt the following formulas (6) which are most commonly used in literature to compute Pseudo-Zernike moments of discrete 2-D signals, and the orthogonality and completeness of the Pseudo-Zernike yield the following formula (7) for reconstructing the 2-D signal.

$$\hat{A}_{nm} = \frac{n+1}{\pi} \sum_{i=1}^{M} \sum_{j=1}^{N} V_{nm}^{*}(x_i, y_j) f(x_i, y_j) \Delta x \Delta y \tag{6}$$

$$\hat{f}(x, y) = \sum_{n=0}^{n_{\max}} \sum_{m=-n}^{n} A_{nm} V_{nm}(x, y) \tag{7}$$

4 Analysis of Audio Pseudo-Zernike Moments

In this paper, the digital watermark will be embedded into original audio signal by utilizing the excellent characteristics of Pseudo-Zernike moments. However, in transmission process, audio signal (or watermarked audio) may suffer from attacks such as low-pass filtering, MP3 compression, and amplitude variation etc, which will impact the audio Pseudo-Zernike moments unavoidably. So, it is very necessary to analyze the audio Pseudo-Zernike moments and select the stable audio Pseudo-Zernike moments for embedding.

4.1 Decomposition and Reconstruction of Audio Pseudo-Zernike Moments

Digital audio signal is one-dimensional (1-D) discrete signal, and the two-dimensional (2-D) Pseudo-Zernike transform cannot be performed on digital audio directly, so the 1-D digital audio $\{g(i)\}$ must be mapped into a 2-D form $\{f(x,y)\}$ by using the following

$$\begin{cases} L = R \times R + M, & 0 \le M < 2R+1 \\ f(x,y) = g(x \cdot R + y), & 0 \le x, y \le R \end{cases} \tag{8}$$

where $f(x,y)$ is corresponding 2-D audio version after projection, L is the length of 1-D digital audio, M is the rest of audio samples, and R is the width or height in $f(x,y)$ which should be as large as possible under the constraint of Equation (8).

After mapping, Pseudo-Zernike decomposition and reconstruction procedures on audio signal are performed by using Equation (6) and (7). For the convenience of explaining the audio Pseudo-Zernike decomposition and reconstruction, we choose a clip from our test

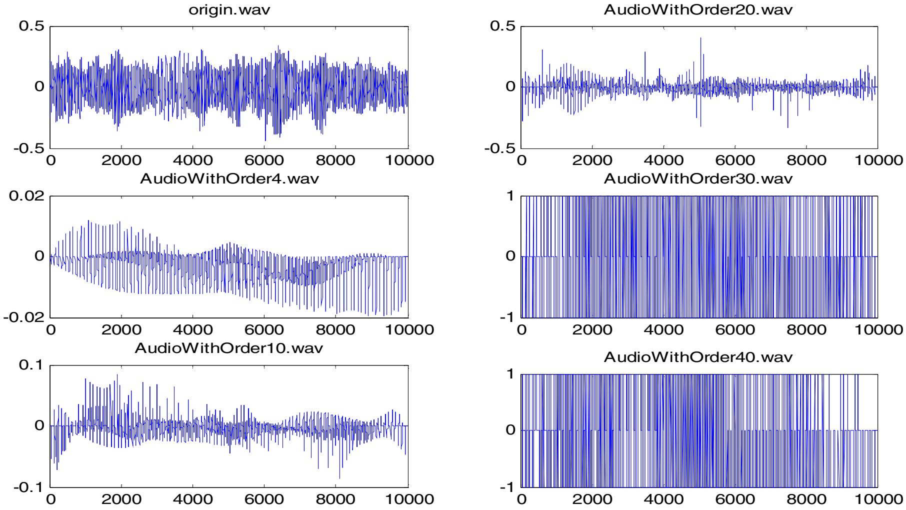

Fig. 1. The original audio and the reconstructed audios under the different order

data set, flute music denoted as *origin.wav* (16-bit signed mono audio file with the length of 1.25s), for testing under different sampling rate from 8 kHz to 44.1 kHz. The number of the given max order $N_{\max}$ is assigned to 4, 10, 20, 30, and 40, respectively. The waveform of original one and the reconstructed audios are aligned in Fig.1 (Here, the sampling rate is 8 kHz, and the audio signal is mapped into 100×100 form).

In Fig. 1, *origin.wav* is the original audio while AudioWithOrder*.wav denote the reconstructed ones, in which $N_{\max}$ is assigned as '*'. It is noted that the low-order moments captured the basic shape of audio signal while the higher order ones fill the high frequency details. This observation is similar to that in images [11]. The degradation caused in reconstruction procedure is due to that when $N_{\max}$ is lower the high frequency information is discarded, while $N_{\max}$ is higher the cumulative computation error occurs in the reconstruction [11]. Referred to Fig.1, it is evident that the reconstruction degradation from limited moments is unavoidable. As to other kinds of audio, such as pop music, piano music and speech, etc., the simulation results are similar.

4.2 Selection of Audio Pseudo-Zernike Moments

In the following experiment, we investigate the robustness performance of audio Pseudo-Zernike moments to common signal processing. We choose a clip flute music denoted as *music.wav* (16-bit signed mono audio file sampled at 44.1 kHz with 250000 audio samples), for testing. First, the digital audio is mapped into 500×500 form. And then, the following mathematical expression is designed to compute the modification of moments before and after audio processing.

$$E_{an} = \sum \left\| Z_{nm} \right\|, E_{bn} = \sum \left\| Z''_{nm} \right\| \tag{9}$$

where $\left\| Z_{nm} \right\|$, $\left\| Z''_{nm} \right\|$ are the modulus of Pseudo-Zernike moments of order n ($0 \le n \le N_{\max}$) with repetition m . E_{an} and E_{bn} denote the total amplitude of all moments with the given order n before and after audio processing, respectively.

1) The influence of low-pass filtering on audio Pseudo-Zernike moments

Fig. 2 shows the influence of low-pass filtering on audio Pseudo-Zernike moments under different cutoff frequency. We can see that the Pseudo-Zernike moments under order 20, with cutoff frequency of 0.8 kHz, are very robust to low-pass filtering.

2) The influence of MP3 compression on audio Pseudo-Zernike moments

Fig. 3 shows the influence of MP3 compression on audio Pseudo-Zernike moments under different bit rates. We can see that the Pseudo-Zernike moments under order 10, with the lowest bit rate of 32 kbps, are very robust to MP3 compression.

3) The influence of amplitude variation on audio Pseudo-Zernike moments

Fig. 4 gives the influence of amplitude variation on audio Pseudo-Zernike moments (Scaling factor α is 0.8 and 1.2, respectively). We can see that the relation between audio amplitude and its Pseudo-Zernike moments is linear.

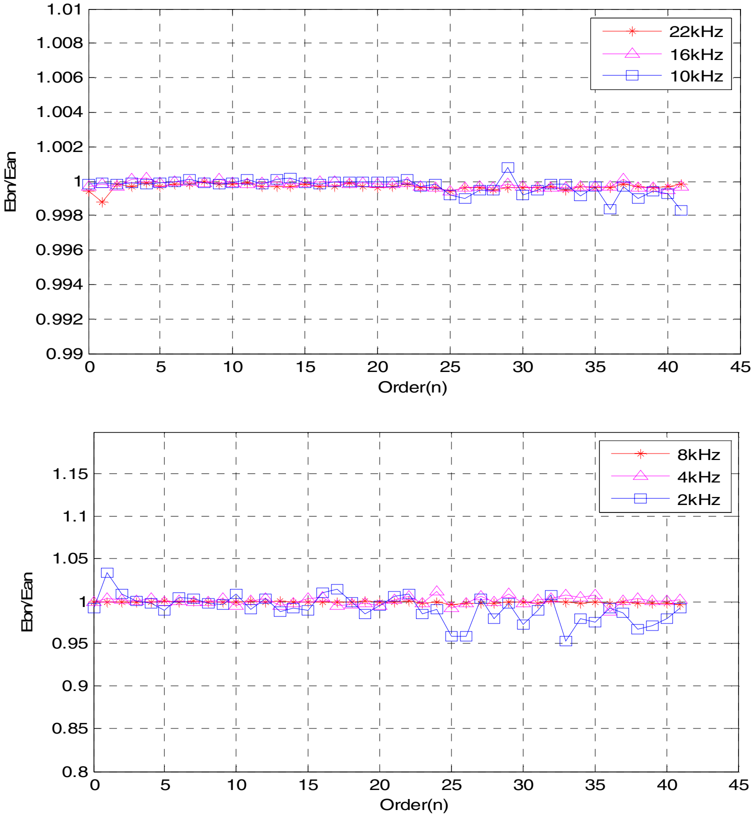

Fig. 2. The effects of low-pass filtering on audio Pseudo-Zernike moments under different cutoff frequency

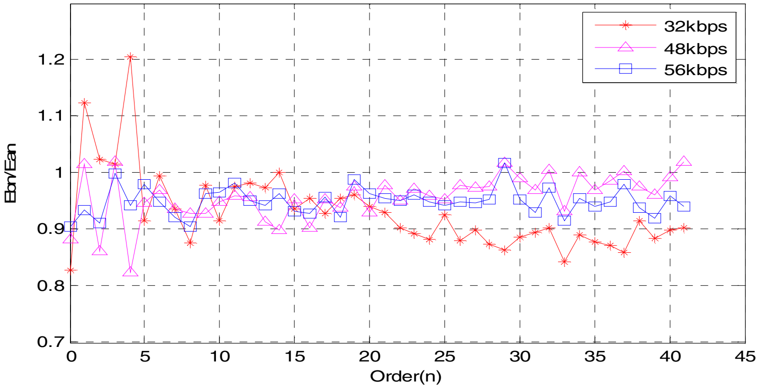

Fig. 3. The effects of MP3 compression on audio Pseudo-Zernike moments under different bit rates

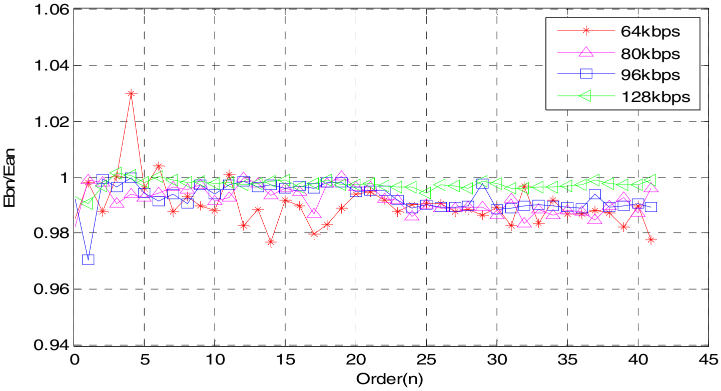

Fig. 3. (*Continued*)

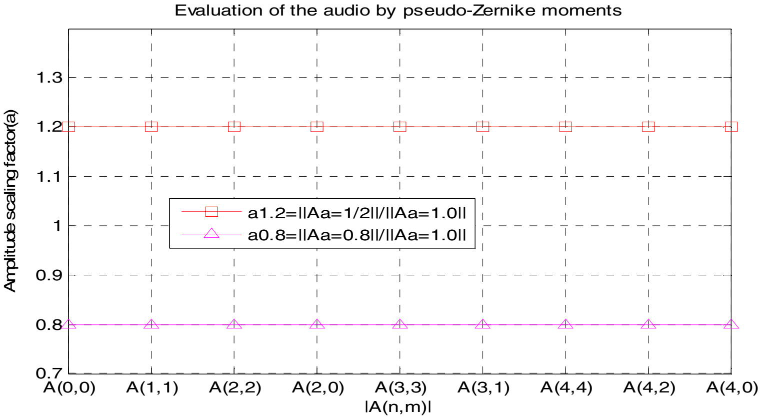

Fig. 4. The relationship between amplitude variation and Pseudo-Zernike moments

We select flute music as the example clip to test the effect of MP3 compression, low-pass filtering and amplitude variation. As to other kinds of audio, such as pop music, piano music and speech, etc., the simulation results are similar. Based on the extensive testing with different audio signals, we have the following observations:

i) Pseudo-Zernike transform of 1-D signal may be achieved by mapping the signal into 2-D form. It is noted that the low-order moments capture the basic shape of the signal but the reconstruction degradation from Pseudo-Zernike moments is large and unavoidable.

ii) Based on the extensive experiments, it is also found that the low-order Pseudo-Zernike moments are robust to common signal processing. The moments under order 10 are very robust to MP3 compression even with the lowest bit rate of 32 kbps. The moments under order 20 are robust to low-pass filtering up to with cutoff frequency of 0.8 kHz.

iii) Scaling linearly audio amplitude not only can change the audio Pseudo-Zernike moments, but also is insensitive to human perceptual system. The relation between audio amplitude and its Pseudo-Zernike moments is linear.

As a conclusion, if we embed the digital watermark into that Pseudo-Zernike moments under order 10 and try to avoid the degradation in reconstruction procedure, it is expected that the watermark will be very robust to common signal processing and some hostile attacks.

5 Watermark Embedding Scheme

We propose a new audio watermarking scheme in which the Pseudo-Zernike moments and synchronization code are utilized. Firstly, the original audio is segmented and then each segment is cut into two parts. Secondly, with the spatial watermarking technique, synchronization code is embedded into the statistics average value of audio samples in the first part. And then, map 1-D digital audio signal in the second part into 2-D form, and calculate its Pseudo-Zernike moments. Finally, the watermark bit is embedded into the average value of modulus of the low-order Pseudo-Zernike moments.

Let $A = \{a(i), 0 \le i < Length\}$ represent a host digital audio signal with $Length$ samples.

$W = \{w(i, j), 0 \le i < M, 0 \le j < N\}$ represent a binary image to be embedded within the host audio signal, and $w(i, j) \in \{0,1\}$ is the pixel value at (i, j).

$F = \{f(i), 0 \le i < Lsyn\}$ represent a synchronization code with $Lsyn$ bits, where $f(i) \in \{0,1\}$.

The main steps of the embedding procedure based on Pseudo-Zernike moments and synchronization code can be described in detail as follows.

5.1 Watermark Preprocessing

In order to dispel the pixel space relationship of the binary watermark image, and improve the robustness of the whole digital watermark system, watermark scrambling algorithm is used at first. In our watermark embedding scheme, the binary watermark image is scrambled from W to W_1 by using Arnold transform, where Then, it is transformed into a 1-D sequence of ones or zeros, and each bit of the watermark data is mapped into an antipodal sequence using BPSK modulation:

$$W_1 = \{w_1(i, j), 0 \le i < M, 0 \le j < N\}$$

$$W_2 = \{w_2(k) = w_1(i, j), 0 \le i < M, 0 \le j < N, k = i \times N + j, w_2(k) \in \{0,1\}\} \tag{10}$$

$$W_3 = \{w_3(k) = 1 - 2 \times w_2(k) \quad , k = 0,1,\cdots, M \times N - 1, \quad w_3(k) \in \{-1,1\}\}$$

In order to improve the robustness of proposed scheme, audio segmenting is used at first. Then, each segment is cut into two parts with L_1 and L_2 samples, respectively, where

$$A(i) = \{a(iL + k), 0 \le k < L\} \quad (0 \le i < \lfloor {}^{Length}/_{L} \rfloor) \tag{11}$$

Where $L = L_1 + L_2, L_1 = Lsyn \times n$, n is a constant and is chosen to be 5 samples in our experiment, $L_2 = R \times R \times M \times N \times 9$ (We use 9 sub-segments to embed 1 watermark bit).

5.2 Synchronization Code Embedding

In order to guarantee robustness and transparency of watermarking, the proposed scheme embeds synchronization code into the statistics average value of audio samples as follows.

1) The first part A_1^0 of audio segment A^0 is cut into $Lsyn$ audio sub-segments, and each audio sub-segment $PA_1^0(m)$ having n samples, where

$$PA_1^0(m) = \left\{pa_1^0(m)(i) = a_1^0(i + m \times n), 0 \le i < n, 0 \le m < Lsyn\right\}$$

2) Calculating the average value of $PA_1^0(m)$:

$$\overline{PA_1^0(m)} = \frac{1}{n}\sum_{i=0}^{n-1} pa_1^0(m)(i), (0 \le m < Lsyn)$$

3) The synchronization code can be embedded into each $PA_1^0(m)$ by quantizing the average value $\overline{PA_1^0(m)}$, the rule is given by

$$pa_1^{0'}(m)(i) = pa_1^0(m)(i) + (\overline{PA_1^{0'}(m)} - \overline{PA_1^0(m)})$$

where $PA_1^0(m) = \{pa_1^0(m)(i), 0 \le i < n\}$ is original sample, and $PA_1^{0'} = \{pa_1^{0'}(m)(i), 0 \le i < n\}$ is modified sample.

5.3 Watermark Embedding

1) The second part A_2^0 of audio segment A^0 is cut into audio sub-segments and each audio sub-segment $A_2^0(k)$ $(k = 0,1,\cdots,M \times N \times 9 - 1)$ is chosen to have $R \times R$ samples.

2) The audio sub-segment $A_2^0(k)$ is mapped into 2-D form, and low-order Pseudo-Zernike moments are calculated. Then, the total modulus of the low-order Pseudo-Zernike moments is given

$$S_k = \sum_{n=0}^{N_{\max}} \sum_{m} \|Z_{nm}\|$$

where S_k is the total modulus of the Pseudo-Zernike moments for the k^{th} audio sub-segment $A_2^0(k)$, Z_{nm} is the low-order Pseudo-Zernike moments of order n with repetition m . And in this paper, $N_{\max}$ =10.

3) Three successive sub-segments are selected as a group (sub-segment group), and the average value of modulus of the low-order Pseudo-Zernike moments for the r^{th} Sub-segment group is

$$AVE_r = {S_{k-1} + S_k + S_{k+1}}\Big/{3} \tag{12}$$

4) Let the average values of modulus in the three consecutive sub-segment groups be represented as AVE_{r-1}, AVE_r and AVE_{r+1}. Their relations may be obtained from the following Equation

$$\begin{cases} A = E_{\max} - E_{med} \\ B = E_{med} - E_{\min} \end{cases} \tag{13}$$

where A and B stand for the differences, respectively. And

$$E_{\max} = Max(AVE_{r-1}, AVE_r, AVE_{r+1})$$

$$E_{\min} = Min(AVE_{r-1}, AVE_r, AVE_{r+1})$$

So we can exploit the following Equation to embed one digital watermark bit $w(i)$

$$\begin{cases} A - B \geq T \quad if \quad w_3(i) = 1 \\ B - A \geq T \quad if \quad w_3(i) = -1 \end{cases} \tag{14}$$

where $T = d \cdot {(AVE_{r-1} + AVE_r + AVE_{r+1})}\Big/{3}$ is the embedding strength, and d is the intensity factor.

If the Equation (14) holds, we will embed the watermark bit $w(i)$ directly; else we use the strategy as followed in Table I to adjust $E_{\max}$, E_{med} and $E_{\min}$ until they satisfy Equation (14).

5) The sub-segments are reconstructed by using the modified Pseudo-Zernike moments. Assumed that after embedding one digital watermark bit, AVE_{r-1}, AVE_r and AVE_{r+1} go to AVE'_{r-1}, AVE'_r and AVE'_{r+1}, respectively. It is equivalent to scale AVE_{r-1}, AVE_r and AVE_{r+1} by using the corresponding factor α_{r-1}, α_{r} and α_{r+1}, which may be computed by the following expressions

$$\alpha_{r-1} = \frac{AVE'_{r-1}}{AVE_{r-1}} \quad \alpha_r = \frac{AVE'_r}{AVE_r} \quad \alpha_{r+1} = \frac{AVE'_{r+1}}{AVE_{r+1}} \tag{15}$$

According to the analysis of audio Pseudo-Zernike moments, we can know that the relationship between audio amplitude and its Pseudo-Zernike moments is linear (see Section 4.2, Fig.4). It means that the modification of Pseudo-Zernike moments may be mapped as the operation of scaling audio amplitude. Using this conclusion, we

Table 1. The adjust Strategy of $E_{\max}, E_{med}, E_{\min}$

If embedded watermark bit w3(i) is '1' and A-B<T:	If embedded watermark bit w3 (i) is '-1' and B-A<T:
Step 1: Emax increase.	Step 1: If (Emin >0)
Step 2:	Emin reduce.
If (Emed>Emin)	Step 2: If (Emax>Emed)
Emed reduce, Emin increase.	Emax reduce□ Emed increase.
Else if (Emed<=Emin & Emin >0)	Else if (Emax<=Emed)
Emed reduce, Emin reduce.	Emax increase, Emed increase.

introduce the following strategy to generate the watermarked audio by scaling the sample values in each sub-segment, referred to Equation (16).

$$\begin{cases} f_{k-1}^{'}(x,y) = \alpha_r \cdot f_{k-1}(x,y) \\ f_k^{'}(x,y) = \alpha_r \cdot f_k(x,y) \\ f_{k+1}^{'}(x,y) = \alpha_r \cdot f_{k+1}(x,y) \end{cases} \tag{16}$$

where α_{r} is the amplitude scaling factor of the r^{th} audio sub-segment group (It has three successive sub-segments, $f_{k-1}(x,y)$, $f_k(x,y)$, and $f_{k+1}(x,y)$), computed by using Equation (14). $f_k(x,y)$ and $f_{\mathrm{k}}^{'}(x,y)$ denote the k^{th} sub-segment of the original 2-D signal and the watermarked 2-D signal, respectively.

5.4 Repeat Embedding

In order to improve the robustness against cropping, the proposed scheme repeats 5.2 and 5.3 sections to embed synchronization code and watermark into every audio segment. Finally, by using Equation (8) we obtain the reconstructed watermarked audio $A^{'}$.

6 Watermark Detecting Scheme

The watermark detecting procedure in the proposed scheme neither needs the original audio signal nor any other side information. The synchronization code detection and digital watermark extraction are two key steps in the whole watermark detection procedure.

6.1 Synchronization Code Detection

Synchronization code detection refers to check the synchronization code in the audio data segment covered by the window (the size is L_1), and synchronization code detection can be described as follows.

1) According to Section 5.2, the average value $\overline{PA^*(m)}$ of former $n \times m$ audio samples $PA^*(m)$ in the audio data segment (covered by the window) is calculated.

$$PA^*(m) = \{pa^*(m)(i) = a^*(i + m \times n), 0 \le i < n, 0 \le m < Lsyn\}$$

2) The synchronization code is extracted by using below rule

$$F' = \{ f'(m) = \left\lfloor \overline{PA^*(m)} \Big/ S_1^* \right\rfloor \bmod 2, \quad 0 \le m < Lsyn \}$$

where S_1^* is the quantization step size.

3) In order to avoid effectively false synchronization, the frame synchronization technology of digital communications (The bit comparison) is utilized for identifying the synchronization code. That is to say, if the extracted synchronization code is same completely as the original one, the synchronization code is thought to be found, and the corresponding position is recorded.

6.2 Digital Watermark Extraction

In this paper, digital watermark is extracted from the audio data segment between two adjacent synchronization codes.

1) The audio data segment $A_2^{0^*}$ (the size is L_2^*), which is candidate audio segment for watermark extraction, is defined by two adjacent synchronization codes.

2) The audio data segment $A_2^{0^*}$ is cut into audio sub-segments and each audio sub-segment $A_2^{0^*}(k)$ is mapped into 2-D form. Then, the low-order Pseudo-Zernike moments are calculated. See Section 5.3.

3) As in Equation (12), we compute AVE_{r-1}^*, AVE_r^* and AVE_{r+1}^*, which are ordered to obtain $E_{\max}^*$, E_{med}^* and $E_{\min}^*$. Similar to Equation (13), we have

$$\begin{cases} A^* = E_{\max}^* - E_{med}^* \\ B^* = E_{med}^* - E_{\min}^* \end{cases}$$

Comparing A^* and B^*, we can get the hidden watermark bit by using the following rule

$$w_3^*(i) = \begin{cases} 1 & if \quad A^* - B^* \ge 0 \\ -1 & if \quad A^* - B^* < 0 \end{cases}$$

The process is repeated until all hidden watermark bits are extracted.

4) Step *1)*-step *3)* is repeated until several copies of watermark are extracted, and the optimal digital watermark $W_4^* = \{ w_4^*(i) \} \ (i = 0,1,\cdots, M \times N - 1)$ is obtained according to the majority rule.

5) The 1-D binary sequences W_2^* is obtained by BPSK demodulating the optimal watermark W_4^*

$$W_2^* = \{ w_2^*(k) = (1 - w_4^*(k))/2 \quad , k = 0,1,\cdots, M \times N - 1, w_2^*(k) \in \{0,1\} \}$$

6) The 1-D binary sequences W_2^* are rearranged to form the binary watermark image W_1^*.

7) Finally, the watermark image

$$W^* = \{w^*(i, j), 0 \le i < M, 0 \le j < N\}$$

can be obtained by descrambling.

7 Experimental Results

In order to evaluate the performance of our scheme, performance test and robustness test are illustrated for the proposed watermarking algorithm. All of the audio signals in the test are music with 16 bits/sample, 44.1kHz sample rates, and 20 seconds. We use a 16×16 binary image as our watermark for all audio signals and a 16-bit Barker code 1111100110101110 as synchronization code. The quantization step S_1 =0.2, the intensity factor d =0.2, the audio sub-segment is mapped into $R \times R$ =8×8 2-D form, and $N_{\max}$ =8.

In order to illustrate the robust nature of our watermarking scheme, common signal processing and de-synchronization attacks are used to estimate the robustness of our scheme, as shown in Table 2. The PSNR of proposed algorithm is 40.39dB.

Table 2. The watermark detection results for various attacks (BER)

Attack free	Re-quantization	Re-sampling (22.05kHz~8kHz)	Low-pass filtering (9kHz-3kHz)	Low-pass filtering (6kHz)
0	0	0	0	1.75
MP3 (256kb~56kb)	Equalization	Noise addition	Echo addition	Low-pass filtering (2kHz)
0	0	0	6.25	4.47
Cropping (1s~6s)	Adding (1s~2s)	Pitch shift one degree higher	Pitch shift one degree lower	Amplitude-scaling (180%~10%)
0	0	0	0	0
TSM（+1%）	TSM（-1%）	TSM (-2%)	TSM（-3%）	TSM（-4%）
38.28	46.48	46.09	50.78	53.52
Low-pass filtering(4kHz) + Cropping 1s	Re-sampling (11.025kHz) + Low-pass filtering (8kHz)	Re-sampling (22.05kHz) + Cropping 1s	Low-pass filtering(8kHz) + Amplitude -scaling 150%	Noise addition + Amplitude -scaling 50%
0	0	0	0	0
Noise addition + MP3(112k)	Re-quantization + Noise addition	Equalization + Cropping 1s	Re-quantization + Adding 1s	Noise addition + Cropping 1s
0	0	0	0	0

8 Conclusion

De-synchronization attacks are the Achilles heel for many audio watermarking schemes. Based on Pseudo-Zernike moments and synchronization code, we propose a new digital audio watermarking algorithm with good auditory quality and reasonable resistance toward de-synchronization attacks in this paper. Simulation results show that the proposed watermarking scheme is not only inaudible and robust against common signals processing but also robust against the de-synchronization attacks. In addition, the watermark can be extracted without the help of the original digital audio signal and can be easily implemented.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant No. 60773031 & 60873222, the Open Foundation of State Key Laboratory of Networking and Switching Technology of China under Grant No. SKLNST-2008-1-01, the Open Foundation of State Key Laboratory of Information Security of China under Grant No. 03-06, the Open Foundation of State Key Laboratory for Novel Software Technology of China under Grant No. A200702, the Open Foundation of Key Laboratory of Modern Acoustics Nanjing University under Grant No. 08-02, and Liaoning Research Project for Institutions of Higher Education of China under Grant No. 2008351.

References

[1] I. J. Cox, L. M. Matthew, A. B. Jeffrey, et al. Digital watermarking and steganography, Second Edition, Burlington, MA, Morgan Kaufmann Publishers (Elsevier), 2007.

[2] N. Cvejic, T. Seppänen. Digital audio watermarking techniques and technologies: applications and benchmarks, Information Science Reference, Hershey, PA, USA, 2007.

[3] D. Kirovski, H. Malvar. Robust covert communication over a public audio channel using spread spectrum. In Proceedings of the 4th International Workshop on Information Hiding (IH 2001), 2001, Lecture Notes in Computer Science 2137, Springer 2001: 354-368.

[4] D. Kirovski, H. Malvar. Spread-spectrum watermarking of audio signals. IEEE Trans. on Signal Processing, 2003, 51(4): 1020-1033.

[5] M. Mansour, A. Tewfik. Audio watermarking by time-scale modification. In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 2001, 3: 1353-1356.

[6] M. Mansour, A. Tewfik. Data embedding in audio using time-scale modification. IEEE Trans. on Speech Audio Processing, 2005, 13(3): 432-440.

[7] W. Li, X. Xue. Localized audio watermarking technique robust against time-scale modification. IEEE Trans. on Multimedia, 2006, 8(1): 60-69.

[8] Xiaohong Ma, Bo Zhang, Xiaoyan Ding. Self-synchronization blind audio watermarking based on feature extraction and subsampling. International Symposium on Neural Networks 2007, Lecture Notes in Computer Science 4492, 2007:40-46.

[9] X. Y. Wang, W. Qi, P. P. Niu. A new adaptive digital audio watermarking based on support vector regression. IEEE Trans. On Audio, Speech and Language Processing, 2007, 15(8): 2270-2277.

[10] J. Haddadnia, M. Ahmadi, K. Faez. An efficient feature extraction method with Pseudo-Zernike moments in RBF neural network-based human face recognition system. EURASIP Journal on Applied Signal Processing, 2003, (9): 890-901.

[11] Javad Haddadnia, Majid Ahmadi, and Karim Faez. An efficient feature extraction method with Pseudo-Zernike moments in RBF neural network-based human face recognition system. EURASIP Journal on Applied Signal Processing, 2003, 3(9): 890-901.