DIGITAL WATERMARKS FOR AUDIO SIGNALS

Laurence Boney *Ahmed H. Tewfik* *Khaled N. Hamdy*

Department of Electrical Engineering
University of Minnesota
Minneapolis, MN 55455
{boney,tewfik,khamdy}@ee.umn.edu

ABSTRACT

In this paper, we present a novel technique for embedding digital "watermarks" into digital audio signals. Watermarking is a technique used to label digital media by hiding copyright or other information into the underlying data. The watermark must be imperceptible or undetectable by the user and should be robust to various types of distortion. In our method, the watermark is generated by filtering a PN-sequence with a filter that approximates the frequency masking characteristics of the human auditory system. It is then weighted in the time domain to account for temporal masking. We also discuss the detection of the watermark and assess the robustness of our watermarking approach to various signal manipulations.

1. INTRODUCTION

In today's digital world, there is a great wealth of information, which can be accessed in various forms: text, images, audio, and video. It is easy to ensure the security of "analog documents" and protect the author (author will be used to also denote composer, artist, designer, etc.) from having his work stolen or copied. For example, a painting is signed by the artist, books and albums have copyright labels imprinted inside the cover. The question is *how do you copyright or label digital information and preserve its security without destroying or modifying the content of the information.*

One approach to data security is to use cryptographic techniques. In cryptology, the information is scrambled using an encryption transformation before it is sent and the information can be viewed after de-scrambling with the inverse transformation. A public-key cryptosystem can be used to implement an electronic mail system in which messages are kept private and can be signed [1]. The security of the encryption algorithm is based on the fact that no one has discovered an algorithm which can factor composite numbers with two very large prime factors (on the order of 200 digits) in a *reasonable amount of time.* Note that cryptosystems restrict access to the document and do not label or stamp them. Once the documents are decrypted, the "signature" is removed and there is *no proof of ownership* such as a label, stamp, or watermark [2]. Cryptology, as discussed in [3], may be used for digital TV broadcasting to provide conditional access for pay TV, watermarking of images for copyright protection, and image signature for authentication. Note, that it is useful to consider the binary representation of these large numbers and their prime factors as codewords for the signatures.

Data hiding, or steganography, refers to techniques for embedding watermarks, signatures, tamper protection, and captions in digital data. Captioning is an application which requires a large amount of data; however, it need not be invariant to removal because it contains extra non-critical information which may be of benefit to the author and the user. On the other hand, watermarking is an application which embeds the least amount of data, but requires the greatest robustness because the watermark is required for copyright protection [4]. Note that data hiding does not restrict access to the original information as does encryption.

A watermark, or an invisible stamp, could be used to provide proof of "authorship" of a signal. Similarly, a signature is used to provide proof of ownership and track illegal copies of the signal. The watermark must be embedded in the data such that it is *imperceptible* by the user [3, 4, 5]. Moreover, the watermark should also have the following characteristics:

- **Inaudible**;
- **Statistical invisibility** to prevent unauthorized detection and/or removal;
- **Similar** compression characteristics as original signal;
- **robustness** to manipulation and signal processing operations on the host data, e.g., filtering, resampling, compression, noise, cropping, A/D-D/A conversions, etc;
- **Embedded** directly in the data, not in a header;
- **Support multiple watermarks, i. e. multiple users,**;
- **Self-clocking**.

The watermark should be characteristic of an author, but a "pirate" should not be able to detect the watermark by comparing several signals belonging to the same author. The signal should be degraded when the watermark is removed through unauthorized means.

In black and white document images, identification marks are hidden by shifting characters, words, lines, etc. randomly, in such a way that it is not observable upon inspection of the document [6, 7, 8, 9, 10]. These methods restricted to text documents and are easily defeated as shown by the authors.

Other techniques for data hiding in images have been developed. Two methods for watermarking images are proposed in [11]. The first approach embeds a PN-sequence on the least significant bit (LSB) of the data. This provides easy and rapid decoding of the watermark or signature. In the second approach, a PN-sequence (watermark) is added to the LSB of the data. This is more difficult to decode, providing more security. As with any approach which modifies the LSB of the data, however, these watermarks are highly sensitive to noise and are easily corrupted.

In other coding schemes, the watermarks are made to appear as quantization noise as the are embedded into the images [12, 13]. The first method uses a predictive coding scheme to embed the watermark into the image. In the second method, the watermark is embedded image by dithering the image based on the statistical properties image. These scheme is not robust to attacks such as requantization and cropping.

In [14], a watermark for an image is generated by modifying the luminance values inside 8x8 blocks of pixels, adding one extra bit of information to each block. The choice of the modified block is secretly made by the encoder. In [15], a 2-D signature is generated and is embedded into the image by modifying the intensity levels of the image, whose corresponding signature pixels is one. A method using a JPEG model based, frequency hopped, randomly sequenced pulse position modulated code in [16] is robust to operations such as lossy data compression, lowpass filtering, and color space conversion. The watermarking problem is viewed as a problem in digital communications in [17]: a codeword is generated and used to modulate selected coefficients of the DCT or wavelet transform of a block in an image.

Ref. [4] discusses data hiding in images by exploiting the properties of the human visual system (HVS), such as sensitivity to contrast as a function of spatial frequency, the masking effect of edges, and sensitivity to changes in grayscale. In [4, 18], techniques for data hiding in images are discussed. The first, an LSB method called "Patchwork," is a statistical technique which randomly chooses n pairs (a_i, b_i) of points in an image and increases the brightness of a_i by one unit while simultaneously decreasing the brightness of b_i. The second, texture block coding, hides data by mapping a random texture pattern in an image to another region in the image with a similar texture pattern. This method is limited to images that possess large areas of random texture. In [18], an encoding scheme is made resistant to affine transformations (scaling, translations, rotations) by embedding crosses in an image. Xerox DataGlyph technology [4, 19] adds a barcode to its images according to a predetermined set of geometric modifications. In [20] data is hidden in the chrominance signal of NTSC by exploiting the HVS temporal over-sampling of color. Adelson [21] proposes a scheme that embeds digital data into analog TV signals. The method substitutes high-spatial frequency image data for "hidden" data in a pyramid-encoded image. However, the scheme is particularly susceptible to filtering and rescaling.

A method similar to ours is proposed in [5], where the N largest frequency components of an image are modified by Gaussian noise. However, the scheme only modifies a subset of the frequency components and does not take into account the HVS. The image watermark we propose here embeds the *maximum* amount of information throughout the spectrum while still remaining perceptually invisible. By placing the maximum amount of watermark information in the signal, we maximize the probability of detection of the watermark if the host signal has been distorted. Moreover, it simultaneously minimizes probability of false alarm, i. e. detection of a signature when one is not present or falsely detecting another author's signature.

Data hiding techniques have also been applied to audio signals [4, 18]. In Direct Sequence Spread Spectrum Coding (DSSS), the signature, a binary codeword, is modulated by both a PN-sequence and the audio signal using bi-phase shift keying. It is then added to the original signal as additive random noise. The perceivable noise added to the signal can be reduced by adaptive coding and redundant coding. In Phase Coding, binary information is embedded in the audio signal by modifying the phases of each frequency component of the Discrete Short Time Fourier Transform of the signal. Because the human auditory system (HAS) is not highly sensitive to phase distortion, the data produce no audible distortion. With Echo Coding, moderate amounts of data are hidden as peaks in the spectrum and are robust to analog transmission.

In this paper, we present a novel technique for embedding digital watermarks into audio signals. The watermark is generated by filtering a PN-sequence with a filter that approximates the frequency masking characteristics of the HAS. It is then weighted in the time domain to account for temporal masking. Note that our approach is similar to that of [4, 18] in that we shape the frequency characteristics of a PN-sequence. However, unlike [4, 18] we use perceptual masking models of the HAS to generate the watermark. In particular, our scheme for images, audio, and video is the only one that uses the frequency masking models of the HAS/HVS along with the temporal masking models and spatial masking models to hide the copyright information in the signal [?].

While adding minimal information, the watermark should create no audible distortion by exploiting the masking effects of the HAS. For audio applications, the watermark should be robust in the presence of various types of colored noise, lossy coding/decoding, D/A - A/D conversion, timescale modifications, and filtering; and, the signal should be degraded when the watermark information is removed through unauthorized means. Each author assigns to each of his signals a unique secret codeword that only he can detect in any time segment of the audio signal with a high degree of certainty, even if the signal has been modified. We provide a study of the detection performance of our watermarking scheme. Our results indicate that the scheme is robust to the types of signal manipulations listed above.

2. BACKGROUND

2.1. Masking

Masking is the effect by which a faint but audible sound becomes inaudible in the presense of another louder audible sound, masker [22]. The masking effect depends on the both spectral and temporal characteristics of both the masked signal and the masker [22]. Frequency masking refers to masking which occurs in the frequency domain. If two signals which occur simultaneously are close together in frequency, the stronger masking signal will make the weaker masked signal inaudible. The masking threshold of a masker depends on the frequency, sound pressure level (SPL), and tone-like or noise-like characteristics of both the masker and the masked signal [23]. It is easier for a broadband noise to mask a tonal, than for a tonal signal to mask out a broadband noise. Moreover, higher frequency signals are more easily masked. Temporal masking refers to both pre- and post-masking. Pre-masking effects render weaker signals inaudible before the stronger masker is turned on, and post-masking effects render weaker signals inaudible after the stronger masker is turned off. Pre-masking occurs from 5-20 msec. before the masker is turned on while post-masking occurs from 50-200 msec. after the masker is turned off [23].

Using the frequency masking information of the HAS, we can shape the spectral characteristics of the watermark. Processing of impulsive signals such as castanets can cause audible pre-echos. Similarly, we can use temporal masking information to eliminate these effects.

2.2. Frequency Masking: MPEG-1 Psychoacoustic Model

Audio signals consist of telephone quality speech, wideband speech, and wideband audio. The frequency ranges for these types of audio signals are 300-3400 Hz for telephone speech signals, 50-7000 Hz for wideband speech range from 50-7000 Hz, and 20-20000 Hz for high quality wideband audio. The human ear acts as a frequency analyzer and can detect sounds with frequencies which vary from 10 Hz to 20000 Hz. The HAS can be modeled by a set of 26 bandpass filters with bandwidths that increase with increasing frequency. The 26 bands are known as the critical bands. The critical bands are defined around a center frequency in which the noise bandwidth is increased until there is just noticeable difference in the tone at the center frequency. Thus if a faint tone lies in the critical band of a louder tone, the faint tone will not be perceptible.

Frequency masking models have already been defined for the perceptual coding of audio signals because it is not necessary to code perceptually irrelevant information. In this work, we use the masking model defined in MPEG Audio Psychoacoustic Model 1, for layer I [24]. The masking method is summarized as follows for a 32 kHz sampling rate [24, 25]. The MPEG model also supports sampling rates of 44.1 kHz and 48 kHz.

- First Step:
  Each 16 ms segment of the signal $s(n)$, N=512 samples, is weighted with a Hann window, $h(n)$:

$$h(n) = \frac{\sqrt{8/3}}{2}[1 - \cos(2\pi \frac{n}{N})] \tag{1}$$

  The power spectrum of the signal $s(n)$ is calculated as follows:

$$S(k) = 10 * \log_{10} [\frac{1}{N} \| \sum_{n=0}^{N-1} s(n)h(n) \exp{(-j2\pi \frac{nk}{N})} \|^2] \tag{2}$$

  The maximum is normalized to a reference sound pressure level of $96dB$.

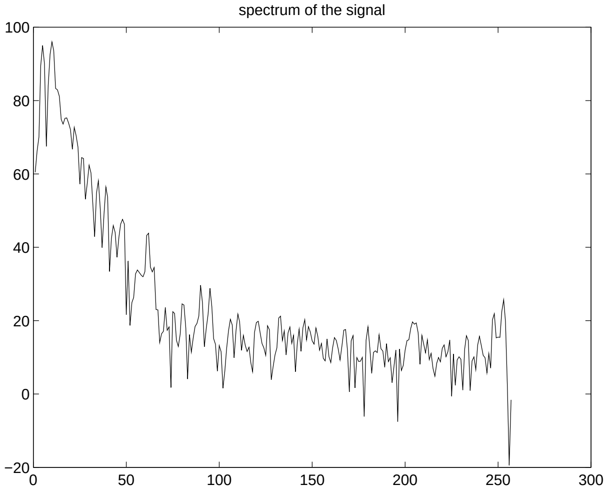

Figure 1. First step

- Second Step: *Identify Tonal Components* Tonal (sinusoidal) and non tonal (noisy) components are identified because their masking models are different.
  A tonal component is a local maximum of the spectrum ($S(k) > S(k+1)$ et $S(k) \geq S(k-1)$) satisfying:

$$S(k) - S(k+j) \geq 7dB$$
$$j \in [-2, +2] \text{ if } 2 < k < 63$$
$$j \in [-3, -2, +2, +3] \text{ if } 63 \leq k < 127$$
$$j \in [-6, ..., -2, +2, ..., +6] \text{ if } 127 \leq k \leq 250$$

  We add to its intensity those of the previous and following component. Other tonal components in the same frequency band are no longer considered.
  Non-tonal components are made of the sum of the intensities of the signal components remaining in each of the 24 critical bands between 0 and 15500 Hz. (The auditory system behaves as a bank of bandpass filters, with continuously overlapping center frequencies. These “auditory filters” can be approximated by rectangular filters with critical bandwidth increasing with frequency. In this model, the audible band is therefore divided into 24 non-regular critical bands.)
- Third Step: *Remove Masked Components*
  Those components below the absolute hearing threshold and tonal components separated by less than 0.5 Barks.
- Fourth Step: *Individual and Global Masking Thresholds*
  In this step, we account for the frequency masking effects of the HAS. We need to discretize the frequency

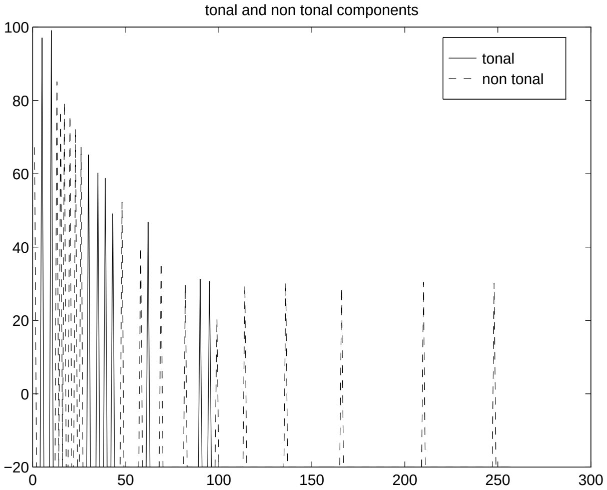

Figure 2. Second step

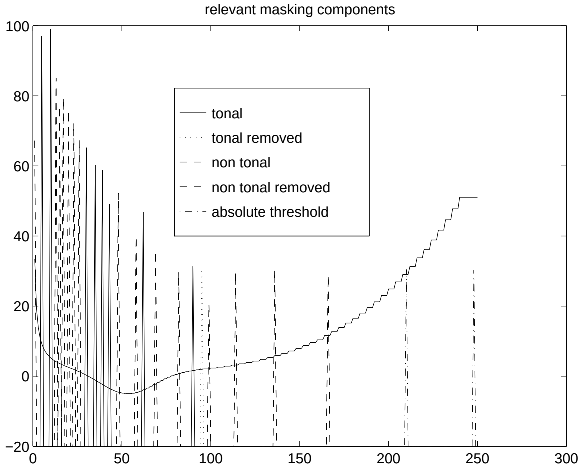

Figure 3. Third step

axis according to the hearing sensitivity which is better for low frequencies and express frequencies in Barks. Masking curves are then almost linear (with different lower and upper slopes depending on the distance between the masked and the masking component) and depend on a masking index different for tonal and non-tonal components. We use f_1 to denote the set of frequencies present in the test signal. The global masking threshold for each frequency f_2 takes into account the absolute hearing threshold S_a and the masking curves P_2 of the N_t tonal components and N_n non-tonal components:

$$\begin{aligned} S_m(f_2) &= 10 * \log_{10}[10^{S_a(f_2)/10} \\ &\quad + \sum_{j=1}^{N_t} 10^{P_2(f_2,f_1,P_1)/10} \\ &\quad + \sum_{j=1}^{N_n} 10^{P_2(f_2,f_1,P_1)/10}] \end{aligned} \tag{3}$$

The masking threshold is then the minimum of the local masking threshold and the absolute hearing threshold in each of the 32 equal width subbands of the spectrum.

To be inaudible, the watermark must fall below the masking threshold.

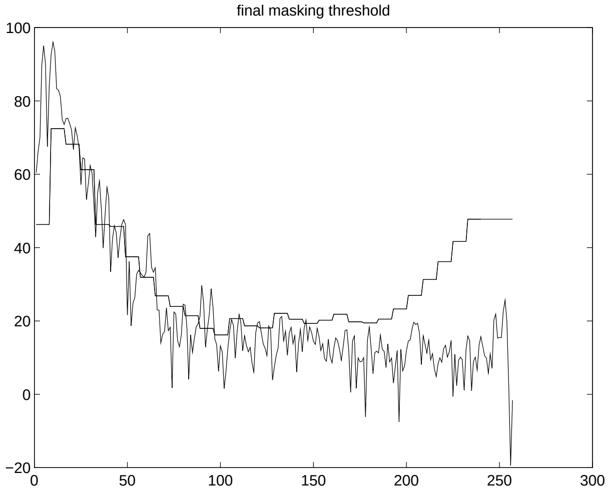

Figure 4. Fourth step

2.3. PN-sequences

PN-sequences form the basis of our watermarking scheme because of their noise-like characteristics, resistance to interference, and good auto-correlation properties. Spread spectrum communication systems use pseudo-noise (PN) sequences to modulate transmitted data into noise-like wideband signals so they blend into the background [26]. Spread spectrum signals are resistant to interference such as unintentional interference, channel noise, multiple users, multipath interference, or intentional jammers [26].

PN-sequences are periodic noise-like binary sequences generated by feedback shift register of fixed length m [26]. The feedback is linear, that is, it consists of only modulo-2 adders. This prevents the zero state from occuring, which provides an output of only zeros. The maximum period of a PN-sequence is $N = 2^m - 1$ [26]. The feedback connections for maximal length PN-sequences with m varying from 1 to 89 are provided in [27].

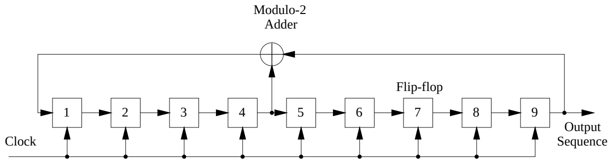

Figure 5. Shift register with m=9, N=511

Maximum length PN-sequences, also called m-sequences, are used in our watermarking scheme because they provide an easy way to generate a unique code for an author's identification. Moreover, like random binary sequences, m-sequences have 0's and 1's occur with equal probabilities. Also, the number of 1's is always one greater than the number of 0's. M-sequences also have good autocorrelation properties [26]: the autocorrelation function (ACF) has period N and it is binary valued. The ACF has peaks equal to 1 at 0, N, 2N, etc. and is approximately $1/N$ elsewhere. Because of these periodic peaks, the m-sequence is self-clocking. This allows the author to synchronize with the embedded watermark during the detection process. This is important if the signal is cropped and resampled.

3. WATERMARK DESIGN

Each audio signal will be watermarked by a unique codeword. Our method uses PN-sequences to watermark the audio signal as suggested in [11]. In order to take advantage of the frequency masking characteristics of the HAS, we shape the PN-sequence with the masking threshold of the signal in frequency domain. But we must also account for temporal masking effects. The watermark should be hidden in the signal according to its energy. We do not want to add too much information where the signal has low energy; Otherwise, the information that is introduced will become audible. This is because a fixed length Fourier transform cannot have both good time and good frequency localization. If high energy signal has a time duration much less than the window length, its energy is spread across all frequencies for the duration of the window. Therefore, we weight the watermark in time with the energy of the signal's envelope.

To generate the watermark, we first calculate the masking threshold of the signal using the MPEG Audio Psychoacoustic Model 1, as described above. The masking threshold is determined for audio segments of 512 samples, weighted by a Hanning window, with 50% overlap in successive blocks. It is then approximated with a 10^{th} order all-pole filter, $M(\omega)$, using a least squares criterion. The PN-sequence $seq(\omega)$, is filtered with the approximate masking filter, $M(\omega)$, in order to ensure that the spectrum of the watermark is below the masking threshold, as shown in Fig. 6. In this example, we used a m-sequence with $m = 9$.

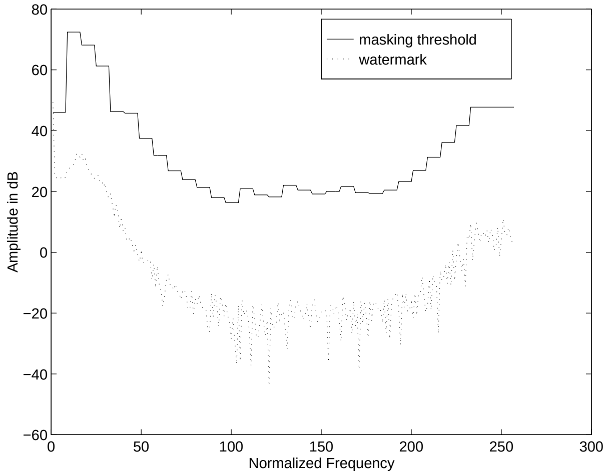

Figure 6. Filtered PN-sequence

Manipulating signals with sudden changes in energy, such as castanets, may lead to pre-echos. The watermark computed below the masking threshold in frequency domain, is spread in time on the block of 512 samples. If there is a sudden change of amplitude inside the block, high energy regions of the signal will spread into the low energy regions of the signal, creating audible distortion. The audible result is noise preceding the sudden changes in the signal. Therefore, the watermark, $w(n)$ is weighted in the time domain with the squared and normalized envelope of the signal.

$$w(n) = w(n) * \frac{envelope(n)^2}{\sum_{k=1}^{N} envelope(k)^2}$$

The filtered watermark is then multiplied by a scale factor for the following reason. To make the detection of the watermark easier, we would like to increase the power of the watermark, while keeping its spectrum below the masking threshold. The "computed watermark" is smaller than the quantization step size and, therefore must be scaled such that it is not lost in the quantization process when it is written as a 16 bit integer. We have found that amplifying the watermark by 40dB before weighting the signal in time gives good results. Weighting in time lowers the watermark below the masking threshold often enough for it to be unhearable.

Figures 7 and 8 show our technique for watermarking audio signals. In our basic watermarking scheme, we filter a codeword, with a filter which approximates the filter characteristics of the HAS. This result is weighted in time with the envelope of the original audio signal to prevent the introduction of temporal effects such as pre-echoes. This is then added to the original signal giving

$$watermark_{firststage} = (originalsignal) + w,$$

where w denotes the filtered PN-sequence.

In order for the watermark to be robust to coding/decoding at low bit rates, we must generate w_{64} as shown in Fig 8. This includes the coding/decoding effects in the watermark.

$$w_{64} = (watermark_{firststage})_{64} - (originalsignal)_{64}$$

In our notation, the subscript 64 refers to a signal which has been coded/decoded at a bit rate of 64 kbits/second. A recent technical report [5] showed us that it is necessary to place the watermark in the perceptually significant components of the signal, i. e. high frequencies. This is contained in coding error of the original audio signal. Therefore, we add the filtered PN-sequence to the coding error,

$$w_{err} = (originalsignal) - (originalsignal)_{64}.$$

Thus the final watermark is given by

$$wat = w_{64} + w_{err}.$$

Figures 9 and 10 show the various stages of watermarking and the final watermarked signal for two different musical signals. In Fig. 9, the original signal is the beginning of the third movement of the sonata in B flat major D 960 of Schubert, interpreted by Vladimir Ashkenazy. In Fig. 10, the original signal is a castanet signal.

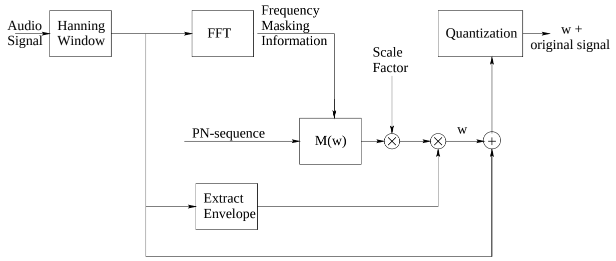

Figure 7. Watermark Generator: First stage for audio

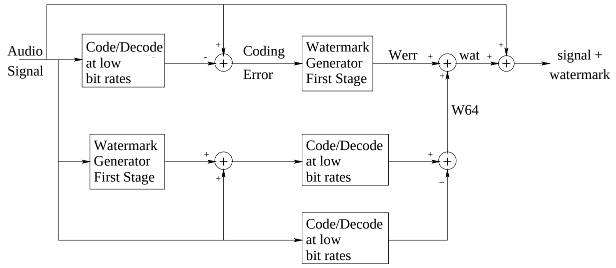
Audio Signal
Code/Decode at low bit rates
Coding Error
Watermark Generator First Stage
Werr
wat
signal + watermark
W64
Watermark Generator First Stage
Code/Decode at low bit rates
Code/Decode at low bit rates

Figure 8. Full Watermark Generator for audio

4. DETECTION OF THE WATERMARK

An unauthorised user of the signal would try to make detection of the watermark impossible by adding colored noise, filtering, coding, D/A - A/D converting, time compressing, etc. For the detection problem, it is assumed that the original signal is available to the detector along with the author's PN-sequence.

We want to be able to decide whether the difference between the "pirated" audio signal $r(t)$ and the original audio signal $s(t)$ is only noise $n(t)$ or the jammed (corrupted) watermark $w'(t)$ and noise. The hypotheses to test are the following :

- $H_0 : x(t) = r(t) - s(t) = n(t)$
- $H_1 : x(t) = r(t) - s(t) = w'(t) + n(t)$

Note that the watermark is inaudible and that we are interested in cases where the unauthorized distortion added to the watermarked signal is also inaudible. As suggested in [3], we can use the correlation of x with w to detect the presence of a jammed watermark, by comparing it to a threshold. Our study indicates that it is possible to reliably detect the watermark when 50 or more blocks of 512 samples of the difference signal are used, and for a detection threshold around 0.5. Note that this corresponds to 0.8 seconds of the audio signal (at 32 kHz sampling rate).

We can then calculate the probabilities of detection and false alarm for each segment of 50*512 samples, after determining the optimal threshold. Note also that, even if the watermark is generated with the same PN-sequence for the whole original signal, the watermark changes across the signal, depending on the masking threshold and the energy of the signal for each block of 512 samples.

The author should choose different PN-sequences for each of his audio signals, so that his signature cannot be found by comparing or correlating several of his audio signals. Note also that it is possible to make the watermark difficult to detect by an unauthorized user, by using long PN-sequences or embedding long cryptographic digital signatures.

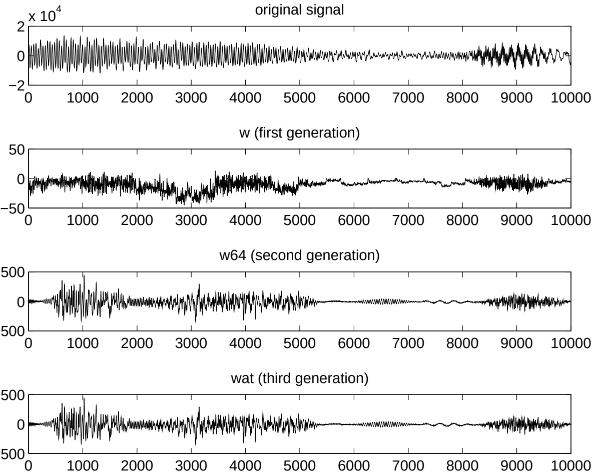

Figure 9. Various watermark stages for Schubert in time domain

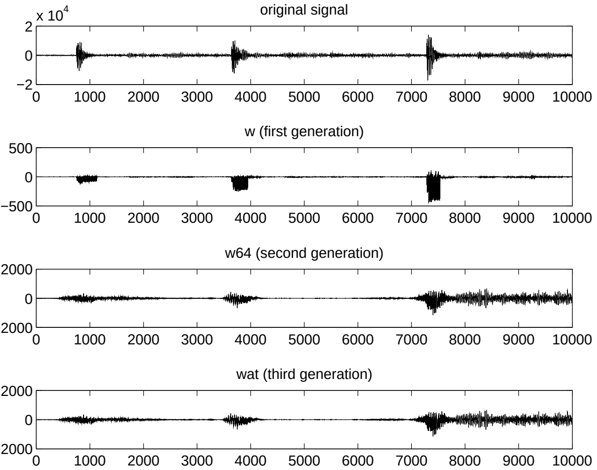

Figure 10. Various watermark stages for castanet signal in time domain

4.1. Robustness to additive noise

We have studied an approximation of the worst possible additive distortion of the watermark: a noise which follows the masking threshold of the watermarked signal. This type of distortion is a good worst case model for distortions due to coding and D/A-A/D conversions.

The noise we used follows the masking threshold of the watermarked signal, $s_w(t)$ and also is weighted in time domain by the energy of the signal. The colored noise is computed in the same way as the watermark: the masking threshold is first shifted $+40dB$ and multiplied by the FFT of a gaussian white noise. The resulting noise is then weighted in time by the squared and normalized envelope of the signal. After requantization, we filter this shaped noise by the masking threshold and requantize it again in time. The result is almost completely inaudible and is an approximation of the maximum noise we can add below the masking threshold.

We have performed detection tests on an audio segment of length 50*512 samples, with and without watermark, cor-

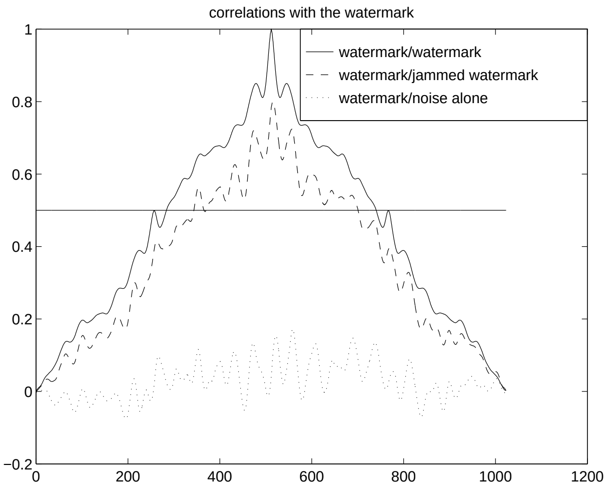

Figure 11. Detection of the watermark in a noisy Schubert file with 50 blocks

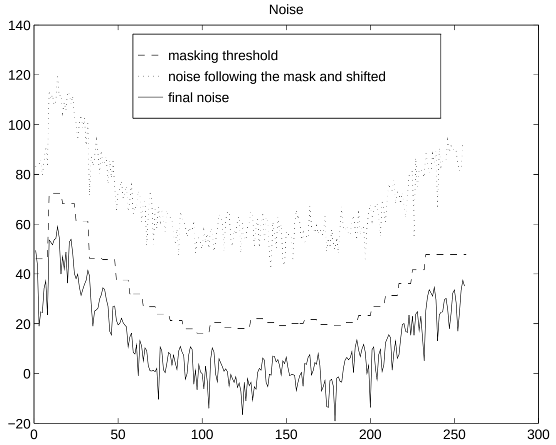

Figure 12. final noise after weighting in time, requantization, filtering by the mask and last requantisation

rupted by the worst case noise, using 12300 different noises. The probabilities of detection and false alarm were found to be 1 and 3.1285e-04, respectively, for a threshold of 0.5. The same test was performed on two other signals: a castanet and a clarinet piece. Castanets are one of those signals prone to pre-echoes.

4.1.1. Example: Robustness to coding/decoding

We also studied the robustness of the watermark to the coding/decoding of the audio signal. The coding/decoding was performed using a software implementation of the ISO/MPEG-1 Audio Layer III coder [28] with several different bitrates (64 kbits/s, 128 kbit/s, 160 kbit/s, 224 kbit/s and 320 kbit/s). The Schubert signal is sampled at 32 kHz and the castanet and clarinet signals are sampled at 44 kHz.

Our study indicates that it is possible to reliably detect the watermark when 100 or more blocks of 512 samples of the difference signal are used. Note that this corresponds to 0.8 seconds of the audio signal.

Tables 1, 2, 3, 4, and 5 below gives the probabilities of detection and false alarm for for the final watermark in the following signals: the Schubert signal, a castanet signal, and a clarinet signal. The watermarks were generated with the coding/decoding performed at bit rates of 64, 128, 256, 224, and 320 kbit/s. Note that the probability of detection is 1 or nearly 1 in all cases, and equally important, the probability of false alarm is 0 in all cases.

The detection is also robust when multiple watermarks were present in the audio signal, as shown in Tables 6, 7, and 8. The watermarks were generated with the coding/decoding performed at bit rates of 64, 128, and 160 kbit/s. Note that the probability of detection 1 in all cases, and equally important, the probability of false alarm is 0 in all cases.

We found the that the watermark in the presence of other watermarks as shown in the table was robust to resampling. Specifically, the watermark was still detectable when the sampling rate is doubled and jamming noise is added to the signal.

5. CONCLUSIONS

Our method for the digital watermarking of audio signals extends the previous work on images. Our watermarking scheme consists of a maximal length PN-sequence filtered by the approximate masking characteristics of the HAS and weighted in time, our watermark is imperceptibly embedded into the audio signal and easy to detect by the author thanks to the correlation properties of PN-sequences. Our results show that our watermarking scheme is robust in the presence of additive noise, lossy coding/decoding, resampling, and time-scaling.

REFERENCES

[1] R. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[2] E. Koch, J. Rindfrey, and J. Zhao, “Copyright protection for multimedia data,” *Proceedings of the Intl. Conf. on Digital Media and Electronic Publishing (6-8 December 1994, Leeds, UK).*

[3] B. Macq and J.-J. Quisquater, “Cryptology for digital tv broadcasting,” *Proc. of IEEE*, pp. 944–957, June 1995.

[4] W. Bender, D. Gruhl, and N. Morimoto, “Techniques for data hiding,” *Proc. of the SPIE*, 1995.

[5] I. Cox, J. Kilian, T. Leighton, and T. Shamoon, “Secure Spread Spectrum Watermarking for Multimedia.” Tech. Rep. 95-10, NEC Research Institute, 1995.

[6] A. Choudhury, N. Maxemchuk, S. Paul, and H. Schulzrinne, “Copyright protection for electronic publishing over computer networks,” *IEEE Network*, vol. 9, pp. 12–20, May-June 1995.

[7] J. Brassil, S. Low, N. Maxemchuk, and L. O’Gorman, “Hiding information in document images,” *http://www.research.att.com:80/projects/ecom.html*, 1994.

[8] J. Brassil, S. Low, N. Maxemchuk, and L. O’Gorman, “Electronic marking and identification techniques to discourage document copying,” *IEEE J. Sel. Areas Comm.*, vol. 13, pp. 1495–1504, October 1995.

[9] N. Maxemchuk, “Electronic document distribution,” *ATT Technical Journal*, vol. 73, pp. 73–80, Sept.-Oct. 1994.

[10] S. Low, N. Maxemchuk, J. Brassil, and L. O’Gorman, “Document marking and identification using both line and word shifting,” in *Proc. of INFOCOM ’95: Conference on Computer Communications*, vol. 2, pp. 853–860, 1995.

[11] R. V. Schyndel, A. Z. Tirkel, and C. Osborne, “A digital watermark,” in *Proceedings of ICIP’94*, vol. II, pp. 86–90, November 1994.

[12] K. Matsui and K. Tanaka, “Video steganography: How to secretly embed a signature in a picture,” in *IMA Intellectual Property Project Proceedings 1:1*, January 1994.

[13] K. Tanaka, Y. Nakamura, and K. Matsui, “Embedding secret information into a dithered multi-level image,” in *MILCOM 90: A New Era. 1990 IEEE Military Communications Conference.*

[14] O. Bruyndonckx, J.-J. Quisquater, and B. Macq, “Spatial method for copyright labeling of digital images,” in *Nonlinear Signal Processing Workshop, Thessaloniki, Greece*, pp. 456–459, 1995.

[15] I. Pitas and T. Kaskalis, “Applying signatures on digital images,” in *Nonlinear Signal Processing Workshop, Thessaloniki, Greece*, pp. 460–463, 1995.

[16] E. Koch and J. Zhao, “Towards robust and hidden image copyright labeling,” in *Nonlinear Signal Processing Workshop, Thessaloniki, Greece*, pp. 452–455, 1995.

[17] F. Boland, J. O. Ruanaidh, and C. Dautzenberg, “Watermarking digital images for copyright protection,” *IEE Intl. Conf. on Image Proc. and Its Applications, Edinburgh, 1995.*

[18] D. Gruhl, N. Morimoto, and W. Bender, “The data hiding homepage,” *http://nif.www.media.mit.edu/DataHiding/index.html*, vol. 21, no. 2, pp. 120–126, 1978.

[19] J. Gruber, “Smart paper,” *Wired*, vol. 2, December 1994.

[20] A. Lippman, “Receiver-compatible enhanced definition television system.” U. S. Patent 5,010,405, 1991.

[21] E. Adelson, “Digital Signal Encoding and Decoding Apparatus.” U. S. Patent 4,939,515, 1990.

[22] J. Johnston and K. Brandenburg, “Wideband coding–perceptual considerations for speech and music,” in *Advances in Speech Signal Processing* (S. Furui and M. M. Sondhi, eds.), New York: Dekker, 1992.

[23] P. Noll, “Wideband speech and audio coding,” *IEEE Comm. Mag.*, pp. 34–44, November 1993.

[24] “Codage de l’image animee et du son associe pour les supports de stockage numerique jusqu’a environ 1,5 mbit/s,” tech. rep., ISO/CEI 11172, 1993.

[25] N. Moreau, *Techniques de Compression des Signaux.* Masson, 1995.

[26] S. Haykin, *Communication Systems, 3rd Edition.* John Wiley and Sons, 1994.

[27] R. Dixon, *Spread Spectrum Systems.* John Wiley and Sons, 1976.

[28] “Mpeg audio-layer 3 encoder/decoder software,” tech. rep., Fraunhofer-IIS 1994, 1995, 1994-1995. http://www.iis.fhg-de.

Table 1. Detection of watermark – coding/decoding at 64 kbit/s.

Audio Signal	*Schubert*	*castanets*	*clarinet*
Detection Thresh.	0.74	0.525	0.62
P_{detect}	1	1	1
$P_{falsealarm}$	0	0.0008	0.0147
# of different noises	2784	2385	2014

Table 2. Detection of watermark – coding/decoding at 128 kbit/s.

Audio Signal	*Schubert*	*castanets*	*clarinet*
Detection Thresh.	0.74	0.525	0.62
P_{detect}	0.9996	1	1
$P_{falsealarm}$	0	0	0
# of different noises	3784	2385	2014

Table 3. Detection of watermark – coding/decoding at 160 kbit/s.

Audio Signal	*Schubert*	*castanets*	*clarinet*
Detection Thresh.	0.74	0.525	0.62
P_{detect}	1	1	1
$P_{falsealarm}$	0	0	0
# of different noises	2784	2385	2014

Table 4. Detection of watermark – coding/decoding at 224 kbit/s.

Audio Signal	*Schubert*	*castanets*	*clarinet*
Detection Thresh.	0.74	0.525	0.62
P_{detect}	0.9993	1	1
$P_{falsealarm}$	0	0	0
# of different noises	2784	2385	2014

Table 5. Detection of watermark – coding/decoding at 320 kbit/s.

Audio Signal	*Schubert*	*castanets*	*clarinet*
Detection Thresh.	0.74	0.525	0.62
P_{detect}	0.9993	1	1
$P_{falsealarm}$	0	0	0
# of different noises	2784	2385	2014

Table 6. Multiple watermark detection – coding/decoding at 64 kbit/s.

Audio Signal	*Watermark a*	*Watermark b*
Detection Thresh.	0.71	0.50
P_{detect}	0.9968	1
$P_{falsealarm}$	0.0016	0
# of different noises	1242	1242

Table 7. Multiple watermark detection – coding/decoding at 128 kbit/s.

Audio Signal	*Watermark a*	*Watermark b*
Detection Thresh.	0.71	0.50
P_{detect}	1	1
$P_{falsealarm}$	0.0016	0
# of different noises	1242	1242

Table 8. Multiple watermark detection – coding/decoding at 160 kbit/s.

Audio Signal	*Watermark a*	*Watermark b*
Detection Thresh.	0.71	0.50
P_{detect}	1	1
$P_{falsealarm}$	0.0016	0.000852
# of different noises	1242	1242