A Robust Audio Watermarking Scheme Based on MPEG 1 Layer 3 Compression

David Megías, Jordi Herrera-Joancomartí, and Julià Minguillón

Estudis d'Informàtica i Multimèdia
Universitat Oberta de Catalunya
Av. Tibidabo 39–43, 08035 Barcelona
Tel. (+34) 93 253 7523, Fax (+34) 93 417 6495
{dmegias,jordiherrera,jminguillona}@uoc.edu

Abstract. This paper describes an audio watermarking scheme based on lossy compression. The main idea is taken from an image watermarking approach where the JPEG compression algorithm is used to determine where and how the mark should be placed. Similarly, in the audio scheme suggested in this paper, an MPEG 1 Layer 3 algorithm is chosen for compression to determine the position of the mark bits and, thus, the psychoacoustic masking of the MPEG 1 Layer 3 compression is implicitly used. This methodology provides with a high robustness degree against compression attacks. The suggested scheme is also shown to succeed against most of the StirMark benchmark attacks for audio.

Keywords: Copyright protection, Audio watermarking, Frequency domain methods.

1 Introduction

Electronic copyright protection schemes based on the principle of copy prevention have proven ineffective or insufficient in the last few years (see [1, 2], for example). Pragmatic approaches, like the one adopted for protecting DVDs [3], combine copy prevention with copy detection.

Watermarking is a well-known technique for copy detection, whereby the merchant selling the piece of information (*e.g.* an audio file) embeds a *mark* in the copy sold. From a construction point of view, a watermarking scheme can be described in two stages: mark embedding and mark reconstruction. Since the former determines the mark reconstruction process, the real problem is *where* and *how* the marks should be placed into the product.

Watermarking schemes should provide some basic properties depending on specific applications. Different properties are pointed out in the literature [4, 2, 5, 6] but the most relevant are imperceptibility, capacity and robustness. Imperceptibility, sometimes referred as perceptual quality, guarantees that the mark introduced is imperceptible and then the marked version of the product is not distinguishable from the original one. Capacity measures the amount of information that can be embedded. Such a property is also known as bit rate. Robustness determines the resistance to accidental removal of the embedded mark. All those properties intersect in the sense that an increase in

capacity usually improves robustness but reduces imperceptibility and, reciprocally, an increase in imperceptibility reduces robustness. Hence a trade-off between them must be achieved.

In audio watermarking schemes, the mark embedding process can be performed in different ways, since audio allows multiple manipulations without affecting the perceptual quality. But, since robustness is the most important watermarking property, questions like where and how to place the mark are important issues. In order to maximise imperceptibility, some proposals [7–9] exploit the frequency characteristics of the audio signal to determine the place where the mark should be embedded. Other proposals [10] use echo coding techniques where the mark is encoded by using different delays between the original signal and the echo. Such a technique increases robustness against MPEG 1 Layer 3 audio compression and D/A conversion, but is not suitable for speech signals with frequent silence intervals. Robustness against various signal processing operations is also increased in [11] by dividing the set of the original samples in embedding segments. A more detailed state of the art in audio watermarking can be found in [5].

In this paper we present a novel watermarking scheme for audio. The scheme is based in some sense on the ideas of [12], where a lossy compression algorithm determines where the mark bits are placed. This paper is organised as follows. Section 2 presents the method that describes the new watermarking scheme. Section 3 analyses the properties of the resulting watermarking scheme: imperceptibility, capacity and robustness. Finally, in Section 4, conclusions and some guidelines for further research are outlined.

2 Audio watermarking scheme

The audio watermarking scheme suggested in this paper is inspired in the image watermarking algorithm depicted in [12] in the sense that lossy compression is used in the mark embedding process in order to identify which samples are suitable for marking.

Let the signal S to be watermarked be a collection of Pulse Code Modulation (PCM) samples (for example a RIFF-WAVE[1] file). The aim of the watermarking scheme is to embed a mark into this file in such a way that imperceptibility and robustness of the mark is preserved.

2.1 Mark embedding

Without loss of generality, let S be codified in RIFF-WAVE format. It is well-known that the Human Auditory System (HAS) is sensitive to information in the frequency rather than the time domain. Because of this, the first step of this method is to obtain S_F, the spectrum of S, by applying a Fast Fourier Transform (FFT) algorithm.

In order to determine where the mark bits should be placed, the signal S is compressed using a MPEG 1 Layer 3 algorithm with a rate of R Kbps (tuning parameter) and, then, decompressed again to RIFF-WAVE format. The modified signal, after this

[1] RIFF-WAVE stands for Resource Interchange File Format-WAVEform audio file format.

compression/decompression operation, is called S', and its spectrum S'_F is obtained. Throughout this paper, the Blade codec (**co**mpressor/**dec**ompressor) for the MPEG 1 Layer 3 algorithm has been chosen and, thus, the psychoacoustic model of this codec is implicitly used. Note that audio quality is not an objective of the codec used for this step, since we only need the compression/decompression operation to produce a signal S' which is slightly different from the original S. Hence, any other codec might have been used.

Now, the set of frequencies $F_{\text{mark}} = \{f_{\text{mark}}\}$ suitable for marking are chosen according to the following criteria:

1. All $f_{\text{mark}} \in F_{\text{mark}}$ must belong to the relevant frequencies F_{rel} of the original signal S_F. This means that the magnitude (or modulus) $|S_F(f_{\text{mark}})|$ must be not lower than a given percentage (for example a 2%) of the maximum magnitude of S_F. Therefore, a first set of frequencies $F_{\text{rel}} = \{f_{\text{rel}}\}$ is chosen as:

$$F_{\text{rel}} = \left\{ f \in \left[0, \frac{f_{\max}}{2}\right] : |S_F(f)| \geq \frac{p}{100} |S_F|_{\max} \right\},$$

   where $f_{\max}$ is the maximum frequency of the spectrum, which depends on the sampling rate and the sampling theorem[2], $p \in [0, 100]$ is a percentage and $|S_F|_{\max}$ is the maximum magnitude of the spectrum S_F. Note that the spectrum values in the interval $[f_{\max}/2, f_{\max}]$ are the complex-conjugate of those in $[0, f_{\max}/2]$. The marking frequencies are a subset of these relevant frequencies, *i.e.* $F_{\text{mark}} \subseteq F_{\text{rel}}$.
2. Now, the frequencies to be marked are those which remain "unchanged" after the compression/decompression phase, where "unchanged" means a relative error below a given threshold ε (for example $\varepsilon = 0.05$):

$$F_{\text{mark}} = \{f_1, f_2, \ldots, f_n\} = \left\{ f \in F_{\text{rel}} : \left| \frac{S_F(f) - S'_F(f)}{S_F(f)} \right| < \varepsilon \right\}.$$

Similarly, as done in the image watermarking scheme of [12], a 70-bit stream mark, W ($|W| = 70$), is firstly extended to a 434-bit stream W_{ECC} ($|W_{\text{ECC}}| = 434$) using a dual Hamming Error Correcting Code (ECC). Using dual Hamming binary codes allows us to apply the watermarking scheme as a fingerprinting scheme robust against collusion of two buyers [13]. Finally, a pseudo-random binary stream (PRBS), generated with a cryptographic key k, is added to the extended mark as it is embedded into the original signal.

Once the frequencies in F_{mark} have been chosen, the mark embedding method consists of increasing or decreasing the magnitude of $S_F(f_{\text{mark}})$ in order to embed a '1' or a '0', respectively. The increase or decrease in the magnitude of S_F must be small enough not to be perceptible, but large enough such that the mark can be reconstructed from an attacked signal. The approach of the suggested scheme is to increase or decrease the signal amplitude d dB to embed a '1' or a '0', *i.e.*, if f_{mark} is the frequency at which a bit must be marked, the watermarked signal spectrum will be:

$$\hat{S}_F(f_{\text{mark}}) = \begin{cases} S_F(f_{\text{mark}}) \cdot 10^{d/20} & \text{to embed '1'}, \\ S_F(f_{\text{mark}}) \cdot 10^{-d/20} & \text{to embed '0'}. \end{cases}$$

[2] $f_{\max} = \frac{1}{T_s}$, where T_s is the sampling time.

where the parameter d dB can be tuned. This process is performed for all the frequencies $f_{\text{mark}} \in F_{\text{mark}}$. Note, also, that it is required that n (the number of elements in F_{mark}) should be greater than or equal to the length $|W_{\text{ECC}}|$ of the extended mark (434 in our example). In a typical situation, the mark is embedded tens or hundreds of times all over the spectrum $\hat{S}_F$. In addition, it must be taken into account that the spectrum components in S_F are paired (pairs of complex-conjugate values) and thus the same transformation (adding or subtracting d dB) must be performed to the magnitude $S_F(f_{\text{mark}})$ and to the magnitude of its conjugate. For $f \notin F_{\text{mark}}$ the spectrum of $\hat{S}_F$ is the same as that of S:

$$\hat{S}_F(f) = \begin{cases} S_F(f), & \text{if } f \notin F_{\text{mark}}, \\ S_F(f) \pm d \text{ dB}, & \text{if } f \in F_{\text{mark}}. \end{cases}$$

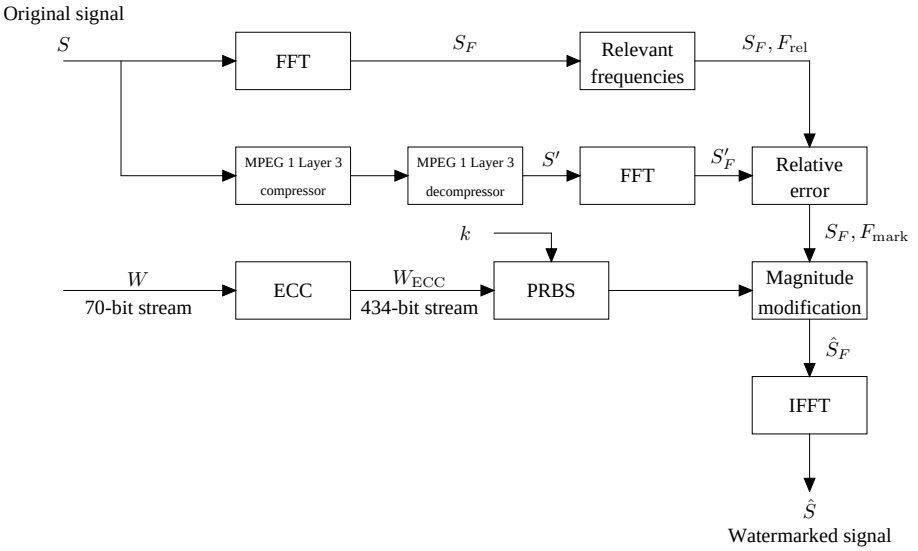

Fig. 1. Mark embedding process

Finally, the marked audio signal is converted to the time domain $\hat{S}$ applying an inverse FFT (IFFT) algorithm. The whole mark embedding process is depicted in the block diagram of Fig. 1. Note that this scheme has been designed to provide with "natural" robustness against compression attacks, since only the frequencies for which the magnitude remains unaltered after compression/decompression, within some specified tolerance (the parameter ε), are chosen for marking. The mark embedding algorithm can be denoted in terms of the following expression:

$$\text{Embed}\left(S, W, \text{parameters} = \{R, p, \varepsilon, d, k\}\right) \rightarrow \left\{\hat{S}, F_{\text{mark}}\right\}$$

2.2 Mark reconstruction

The objective of the mark reconstruction algorithm is to detect whether an audio test signal T is a (possibly attacked) version of the marked signal $\hat{S}$. It is assumed that T is in RIFF-WAVE format. If it were not the case, a format conversion step (for example MPEG 1 Layer 3 decompression) should be performed prior to the application of the reconstruction process.

First of all, the spectrum T_F is obtained applying the FFT algorithm and, then, the magnitude at the potentially marked frequencies $|T_F(f_{\text{mark}})|$, for all $f_{\text{mark}} \in F_{\text{mark}}$, is computed. Note that this method is strictly positional and, because of this, it is required that the number of samples in $\hat{S}$ and T is the same. If there is only a little difference in the number of samples, it is possible to complete the sequences with zeroes. Thus, this methodology cannot be directly applied when resampling attacks occur. In such a case, sampling rate conversion must be performed before the mark reconstruction algorithm can be applied.

When $|T_F(f_{\text{mark}})|$ are available, a scaling step is undertaken in order to minimise the distance of the sequences $|T_F(f_{\text{mark}})|$ and $\left|\hat{S}_F(f_{\text{mark}})\right|$. This scaling is performed to suppress the effect of attacks which modify only a range of frequencies or which scale the PCM signal $\hat{S}$. The following least squares problem is solved:

$$\min_{\lambda} \sum_{f \in F_{\text{mark}}} \left(\left|\hat{S}_F(f)\right| - \lambda \left|T_F(f)\right| \right)^2 .$$

This problem can be solved analytically as follows. Given the vectors

$$\begin{aligned}
\boldsymbol{s} &= \left[\, |S_F(f_1)| \; |S_F(f_2)| \; \ldots \; |S_F(f_n)| \, \right]^{\mathrm{T}}, \\
\hat{\boldsymbol{s}} &= \left[\, \left|\hat{S}_F(f_1)\right| \; \left|\hat{S}_F(f_2)\right| \; \ldots \; \left|\hat{S}_F(f_n)\right| \, \right]^{\mathrm{T}}, \\
\boldsymbol{t} &= \left[\, |T(f_1)| \; |T(f_2)| \; \ldots \; |T(f_n)| \, \right]^{\mathrm{T}},
\end{aligned}$$

where T stands for the transposition operator, it is possible to write the least squares problem in vector form as

$$\min_{\lambda} \left(\hat{\boldsymbol{s}} - \lambda \boldsymbol{t} \right)^{\mathrm{T}} \left(\hat{\boldsymbol{s}} - \lambda \boldsymbol{t} \right),$$

which yields the minimum for:

$$\lambda = \frac{\hat{\boldsymbol{s}}^{\mathrm{T}} \boldsymbol{t}}{\boldsymbol{t}^{\mathrm{T}} \boldsymbol{t}}.$$

Now, each component of $\lambda \boldsymbol{t}$ is divided by the corresponding component of $\boldsymbol{s}$ and the value obtained is compared with $10^{d/20}$ to decide wether a '0', a '1' or a '*' (not identified) might be embedded in this component of $\lambda \boldsymbol{t}$. Let $\boldsymbol{r}_i = \frac{\lambda \boldsymbol{t}_i}{\boldsymbol{s}_i}$:

$$\begin{aligned}
\boldsymbol{r}_i &\in \left[10^{\frac{d}{20}} \left(\frac{100 - q}{100} \right), 10^{\frac{d}{20}} \left(\frac{100 + q}{100} \right) \right] \Rightarrow \hat{\boldsymbol{b}}_i := \text{'1'}, \\
\frac{1}{\boldsymbol{r}_i} &\in \left[10^{\frac{d}{20}} \left(\frac{100 - q}{100} \right), 10^{\frac{d}{20}} \left(\frac{100 + q}{100} \right) \right] \Rightarrow \hat{\boldsymbol{b}}_i := \text{'0'}.
\end{aligned}$$

If none of these two conditions are satisfied, then $\hat{\boldsymbol{b}}_i :=$ '*'. Here $q \in [0, 100]$ is a percentage (*e.g.* $q = 10$) and $\hat{\boldsymbol{b}}_i$ is the i-th component of the vector $\hat{\boldsymbol{b}}$ which contains a sequence of "detected bits". Finally, the PRBS signal is subtracted from the bits $\hat{\boldsymbol{b}}$ to recover the true embedded bits $\boldsymbol{b}$. This operation must preserve unaltered the '*' marks.

Once $\boldsymbol{b}$ has been obtained, it must be taken into account that its length n is (much) greater than the length of the extended mark. Hence, each bit of the mark appears at different positions in $\boldsymbol{b}$. For example, if the length of the extended mark is 434, the first bit should appear at

$$\boldsymbol{b}_1, \boldsymbol{b}_{435}, \boldsymbol{b}_{869}, \ldots, \boldsymbol{b}_{1+434j}, \ldots$$

Some of these bits will be identified as '1', others as '0' and the rest as '*'.

Now a *voting* scheme is applied to decide wether the i-th bit of the mark is '1', '0' or unidentified ('*'). Let n_0, n_1 and n_*, be the number of '0's, '1's and '*'s identified for *the same* mark bit. The simplest approach is to assign to each bit the sign which appears most. For example, if a given mark bit had been identified 100 times with $n_0 = 2$, $n_1 = 47$ and $n_* = 51$, this simple approach would assign a '*' mark to this bit. But, taking into account that any value outside the interval defined above is identified as '*', it is clear that near '1's are identified as '*' although they are much closer to '1' than to '0'. In the reported example, the big difference between the number of '1's and '0's ($47 \gg 2$) can reasonably lead to the conclusion that the corresponding bit can be assigned a '1' with very little probability error, since most of the '*' will probably be near '1's. As a result of this consideration, the voting scheme used in this method ignores the '*' if n_* is not more than twice the difference $|n_1 - n_0|$:

$$\text{bit} := \begin{cases} \text{'*' if } n_* > 2\,|n_1 - n_0|\,, \\ \text{'1' if } n_* \leq 2\,|n_1 - n_0| \text{ and } n_1 > n_0, \\ \text{'0' if } n_* \leq 2\,|n_1 - n_0| \text{ and } n_0 > n_1, \end{cases}$$

A more sophisticated method using statistics might be applied instead of this voting scheme. For instance, an analysis of the distribution of $\boldsymbol{r}_i$ for each bit might be performed. However, the voting procedure described here is simple to implement and fast to execute, which makes it very convenient for real applications.

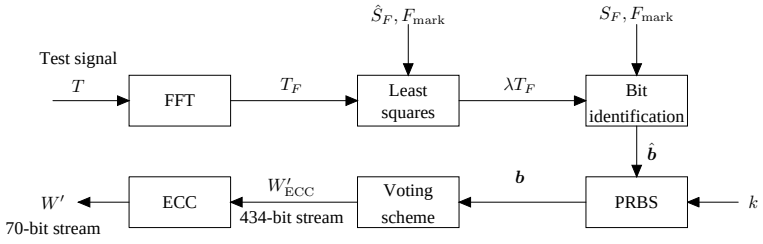

Fig. 2. Mark reconstruction process

As a result of this voting scheme an identified extended mark W'_{ECC} will be available. Finally, W'_{ECC} and the error correcting algorithm are used to recover an identified

70-bit stream mark, W', which will be compared with the true mark W. The whole reconstruction process is depicted in Fig. 2. The mark reconstruction algorithm can be described in terms of the following expression:

$$\text{Reconstruct}\left(T, S, \hat{S}, F_{\text{mark}}, \text{parameters} = \{q, d, k\}\right) \rightarrow \{W', \boldsymbol{b}\}$$

where $\boldsymbol{b}$ is a byproduct of the algorithm which might be used to perform statistical tests.

The proposed scheme is not blind, in the sense that the original signal is needed by the mark reconstruction process. On the other hand, the bit sequence which forms the embedded mark is not needed for reconstruction, which makes this method suitable also for fingerprinting once the mark is properly coded [14].

3 Performance evaluation

As pointed out in Section 1, three main measures are commonly used to assess the performance of watermarking schemes:

Imperceptibility: the extent to which the embedding process leaves undamaged the perceptual quality of the marked object.
Capacity: the amount of information that may be embedded and recovered.
Robustness: the resistance to accidental removal of the embedded bits.

In this section, we test the properties of the proposed scheme presented in Section 2. The scheme described in Section 2 was implemented using a dual binary Hamming code $DH(31, 5)$ as ECC and the pseudo-random generator is a DES cryptosystem implemented in a OFB mode. A 70-bit mark W (resulting in an encoded W_{ECC} with $|W_{\text{ECC}}| = 434$) was included. In order to test the watermarking scheme we have chosen the following parameters for embedding and reconstruction:

- $R = 128$ Kbps, which is the most widely used bit rate in MPEG 1 Layer 3 files.
- $p = 2$, meaning that we only consider relevant those frequencies for which the magnitude of S_F is, at least, a 2% of the maximum magnitude.
- $\varepsilon = 0.05$, which implies that a frequency is considered unchanged after compression/decompression if the magnitude varies less than a 5% (relative error).
- $d = 1$ dB, if lower imperceptibility is required, a lower value can be chosen.
- $q = 10$, *i.e.* a $\pm 10\%$ band is defined about d in order to reconstruct '1's and '0's. This choice is quite conservative, since the '0' and the '1' bands are quite far from each other.

It is worth pointing out that these parameters have been chosen without performing a deep analysis on tuning. Basically, R, p and ε affect the place where the mark bits are embedded; d is related to the imperceptibility of the hidden mark, since it describes how much the spectrum of each marked frequency is disturbed; and, finally, q affects the robustness of the method, since it avoids misidentification of the embedded bits.

To test the performance of the suggested audio watermarking scheme, some of the audio files provided in the Sound Quality Assessment Material (SQAM) page [15] have

been used. The following files have been tested: violoncello (m.p.[3]), trumpet (m.p.), horn (m.p.), glockenspiel (m.p.), harpsichord (arp.[4]), soprano (voice), bass (voice), quartet (voice), English female speech (voice) and English male speech (voice). We have taken only the first ten seconds of each of these files, *i.e.*, 441000 samples, and the mark has been embedded in the left channel only. The glockenspiel file is an especial case, since it has about 5 blank seconds out of 10.

3.1 Imperceptibility

The imperceptibility property determines how much the marked signal $\hat{S}$ differs from the original one S. That is, imperceptibility is concerned with the distortion added with the inclusion of the mark or, in other words, with the audio quality of the marked signal $\hat{S}$ with respect to S. There are several ways to measure audio quality. Here, the signal-to-noise ratio (SNR) and an average SNR (ASNR) are used. The SNR measure determines the power of the noise added by the watermark relative to the original signal, and is defined by

$$\text{SNR} = \frac{\sum_{i=1}^{N} S_i^2}{\sum_{i=1}^{N} \left(S_i - \hat{S}_i\right)^2}$$

where N is the number of samples and S_i ($\hat{S}_i$) denotes the i-th sample of S ($\hat{S}$). Usually, this value is given in dB by performing the operation $10 \log_{10}(\text{SNR})$. Another measure usual in audio quality assessment is an average of the SNR computed taking sample blocks of some length. A typical choice is to consider pieces of 4 ms, which, with a sampling rate of 44100 Hz, means 176 samples. The SNR of all these pieces is computed, and the average for all the sample blocks is obtained. The ASNR measure is often given in dB. The measure used in this paper does not take into account the Human Auditory System (HAS) and, thus, all frequencies are equally weighted.

In Table 1, the SNR and ASNR measures obtained for the ten benchmark files are shown. The SNR measures are about 19 dB whereas the ASNR measures are about 20 dB. This means that power of the noise introduced by watermarking is roughly 0.01 times the power of the original signal, which is quite satisfying and might even be improved (reduced) by choosing proper tuning parameters.

Obviously, the parameter d only affects the imperceptibility of the watermark, since it determines to which extent the spectrum of the marked signal $\hat{S}$ is modified with respect to the original signal S. Hence, by reducing d, to say 0.5 dB, the imperceptibility of the mark would increase, though it will be more easily removed. The parameters R, p and ε determine how many frequencies are chosen for watermarking and, thus, they also affect the imperceptibility of the mark. The larger the number of marked frequencies is, the more perceptible the mark becomes. This establishes a link between the imperceptibility and the capacity of the watermarking system. Hence a tradeoff

[3] "m.p." stands for "melodious phase".

[4] "arp." stands for "arpegio".

Table 1. Capacity and imperceptibility

SQAM file	Marked bits	Capacity (bits)	SNR (dB)	ASNR (dB)
violoncello	4477	722	18.92	20.91
trumpet	3829	617	18.83	19.84
horn	1573	253	18.96	21.10
glockenspiel	1258	202	25.78	29.75
harpsichord	3874	624	21.25	22.84
soprano	5042	813	19.47	21.59
bass	15763	2542	19.02	20.08
quartet	13548	2185	19.22	20.36
female speech	10677	1722	19.57	21.84
male speech	9359	1509	19.44	21.49

between imperceptibility and capacity must be achieved. Note, also, that capacity is related to robustness, since an increase in the number of times the mark is embedded into the signal results in decreasing the probability of losing the mark.

3.2 Capacity

The capacity of the watermarking scheme is determined by the parameters R, p and ε used in the embedding process. Since the marked frequencies are chosen according to the difference of S and S' (the compressed/decompressed signal), it is obvious that the rate R is a key parameter in this process. The percentage p determines which frequencies are significant enough to be taken into account and, thus, this is a relevant parameter as capacity is concerned. Finally, the relative error ε are used to measure wether two spectral values of S and S' are equal, which also affects the number of marked frequencies.

In Table 1, the capacity of the suggested scheme for the ten benchmark files is displayed. We have considered that the true capacity is not the number of marked bits (the second column), since the extended watermark is highly redundant: 70 bits of information plus 364 bits of redundancy. Hence only $70/434$ of the marked bits are the true capacity (third column). However, this redundancy is relevant to the robustness of the method, as it allows to correct errors once the extended mark W'_{ECC} is recovered. Note, also, that 10 seconds of music are enough to allow for, at least, 3 times the mark. If 3-minute files were marked using this method, the capacity of method would be between 3652 bits (or 52 times a 70-bit mark) plus the redundancy for the glockenspiel file and 45763 bits (or 653 times a 70-bit mark) plus the redundancy for the quartet file. It must be taken into account that the glockenspiel file is an especial case, since it only contains 5 seconds of music.

3.3 Robustness assessment

The *robustness* of the resulting scheme has been tested using the StirMark benchmark for audio [16], version 0.2. Some of the attacks in this benchmark can not be evaluated

for the watermarking scheme presented in this paper, since the current version of our watermarking scheme does not allow for a large difference between the number of samples of the marked ($\hat{S}$) and the attacked (T) signals. In addition, only the left channel has been marked in the experiments, thus stereo attacks do not apply here either. The attacks considered for this test are summarised in Table 2.

Table 2. Attacks described in the StirMark benchmark for audio

Name	Number	Name	Number	Name	Number
AddBrumm	1—11	AddDynnnoise	12	Addnoise	13—17
AddSinus	18	Amplify	19	Compressor	20
Echo	21	Exchange	22	FFT_HLPass	23
FFT_Invert	24	FFT_RealReverse	25	FFT_Stat1	26
FFT_Test	27	FlippSample	28	Invert	29
LSBZero	30	Normalise	31	RC-HighPass	32
RC-LowPass	33	Smooth	34	Smooth2	35
Stat1	36	Stat2	37	ZeroCross	38

According to this table, thirty-eight different attacks are performed. The attack AddFFTNoise with default parameters destroys the audio file (it produces no sound) and, thus, no results are available for this attack. Future versions of the watermarking scheme should cope with stereo attacks (ExtraStereo and VoiceRemove) and attacks which modify the number of samples in a significant way (CutSamples, ZeroLength, ZeroRemove, CopySample and Resampling), but the current version of the watermarking scheme proposed here cannot cope with them.

In order to test the robustness of the suggested watermarking scheme against these 38 attacks, a correlation measure between the embedded mark W and the identified mark W' is used. Let W_i and W'_i be, respectively, the i-th bit of W and W', hence

$$\beta_i = \begin{cases} 1, & \text{if } W_i = W'_i \\ -1, & \text{if } W_i \neq W'_i \end{cases}$$

is defined. Now, the correlation is computed, taking into account the β_i for all the $|W|$ bits (70 in our case) of the mark, as follows:

$$\text{Correlation} = \frac{1}{|W|} \sum_{i=1}^{|W|} \beta_i.$$

This measure is 1 when all the $|W|$ bits are correctly recovered ($W = W'$) and it is -1 when all the $|W|$ bits are misidentified. A value of about 0 is expected when 50% of the bits are correctly recovered, as it would occur if the mark bits were reconstructed randomly. In the StirMark benchmark test, we have considered that the watermarking scheme survives an attack **only if the correlation is exactly 1**, *i.e.* only if all the 70 bits of the mark are correctly recovered.

Table 3. Survival of the mark to the StirMark test

Number	Survival ratio	Number	Survival ratio	Number	Survival ratio
1	10/10	2	10/10	3	10/10
4	10/10	5	10/10	6	10/10
7	10/10	8	10/10	9	10/10
10	10/10	11	10/10	12	6/10
13	10/10	14	10/10	15	10/10
16	10/10	17	8/10	18	6/10
19	10/10	20	10/10	21	0/10
22	10/10	23	3/10	24	10/10
25	10/10	26	0/10	27	0/10
28	0/10	29	10/10	30	10/10
31	10/10	32	1/10	33	10/10
34	9/10	35	9/10	36	10/10
37	10/10	38	1/10		

In Table 3 the survival of the mark against the StirMark benchmark attacks is displayed. The relation between the attack number and the name given in the StirMark benchmark is given in Table 2. Each attack has been performed to the ten files of the SQAM corpus reported above. Hence, the results are shown in a $x/10$ ratio since the total number of files is 10. As remarked above, the mark is considered to be recovered only if all the 70 bits are correctly reconstructed.

The results of Table 3 show that only 7 of the 38 attacks of the StirMark benchmark performed in this paper cause serious damage to the embedded mark. The attacks with survival ratios of 6/10 or above produce good correlation values in the non-survived cases, which suggests that better results might arise with an appropriate tuning of the watermarking scheme. The non-survived attacks are the following: 21 (Exchange), 23 (FFT_HLPass), 26 (FFT_Stat1), 27 (FFT_Test), 28 (Flipp_Sample), 32 (RC_Highpass) and 38 (ZeroCross). It must be remarked that most of these attacks produce significant audible damage to the signal and would not be considered acceptable under the most usual situations, especially for music files.

Finally, a set of MPEG 1 Layer 3 compression attacks (using the Blade codec) have been carried out to the marked soprano SQAM file in order to test the robustness of the suggested watermarking scheme against compression. Since the rate used for watermarking is $R = 128$ Kpbs, it was expected that the scheme is able to overcome compression attacks with bit rates of 128 Kbps and higher.

Table 4 displays the correlation values obtained for the MPEG 1 Layer 3 compression attacks for several bit rates, from 320 Kbps to 32 Kbps. This table shows that the watermarking scheme suggested here is not only robust for all bit rates greater than or equal to 128 Kpbs, as expected, but also to rates 112 and 96 Kbps, which are more

Table 4. MPEG 1 Layer 3 compression attacks

Bit rate (Kbps)	320	256	224	192	160	128	112
Compression ratio	4.41:1	5.51:1	6.30:1	7.35:1	8.82:1	11.03:1	12.60:1
Correlation	1	1	1	1	1	1	1
Bit rate (Kbps)	96	80	64	56	48	40	32
Compression ratio	14.70:1	17.64:1	22.05:1	25.20:1	29.30:1	35.28:1	44.10:1
Correlation	1	0.97	0.97	0.94	0.83	0.80	0.49

compressed than the rate used for watermarking (128 Kbps). In addition, the correlation value is very close to 1 even for rates 80, 64 and 56 Kbps. Of course, better robustness against compression attacks might be achieved by choosing a different rate for watermarking, for example $R = 64$ Kbps.

4 Conclusions and further research

This paper presents a watermarking method which uses MPEG 1 Layer 3 compression to determine the position of the embedded mark. The main idea of the method, borrowed from the image watermarking scheme of [12], is to find the frequencies for which the spectrum of the original signal is not modified after compression. These frequencies are used to embed the mark bits by adding or subtracting a given parameter to the magnitude of the spectrum. The method is complemented with an error correcting code and a pseudo-random binary signal to increase robustness and to avoid collusion of two buyers. Thus, this watermarking approach is also suitable for fingerprinting.

The performance of the suggested schemes has been evaluated for the SQAM file corpus using three measures: imperceptibility, capacity and robustness. We have shown that (without tuning) the power of the embedded watermark is about 0.01 times that of the original signal. As capacity is concerned, for typical 3-minute music files, the mark can be repeated hundreds of times within the marked signal. Finally, robustness has been tested by performing the applicable attacks in the StirMark benchmark and also MPEG 1 Layer 3 attacks. The suggested scheme has been shown to be robust against most of the StirMark attacks and to compression attacks with compression ratios larger than that used for watermarking.

There are several directions to further the research presented in this paper. Part of the future research will be focused on the parameters of the scheme, since some guidelines to tune these parameters must be suggested. In addition, the watermarking scheme must be adapted to stereo files by marking both the left and the right channels appropriately. It is also required that the watermarking scheme is able to cope with attacks which modify the number of samples in the attacked signal in a significant way. The use of filters which model the HAS to measure imperceptibility is another research topic. Finally, the possibility to work with blocks of samples instead of using the whole file should be addressed.

Acknowledgements

This work is partially supported by the Spanish MCYT and the FEDER funds under grant no. TIC2001-0633-C03-03 STREAMOBILE.

References

1. Petitcolas, F., Anderson, R., Kuhn, M.: Attacks on copyright marking systems. In: 2nd Workshop on Information Hiding. LNCS 1525, Springer-Verlag (1998) 219–239
2. Petitcolas, F., Anderson, R.: Evaluation of copyright marking systems. In: Proceedings of IEEE Multimedia Systems'99. (1999) 574–579
3. Bell, A.: The dynamic digital disk. IEEE Spectrum **36** (1999) 28–35
4. Swanson, M., Kobayashi, M., Tewfik, A.: Multimedia data-embedding and watermarking technologies. In: Proceedings of the IEEE. Volume 86(6)., IEEE Computer Society (1998) 1064–1087
5. Swanson, M.D.; Bin Zhu; Tewfik, A.: Current state of the art, challenges and future directions for audio watermarking. In: Proceedings of IEEE International Conference on Multimedia Computing and Systems. Volume 1., IEEE Computer Society (1999) 19–24
6. Voyatzis, G.; Pitas, I.: Protecting digital image copyrights: a framework. IEEE Computer Graphics and Applications **19** (1999) 18–24
7. Cox, I.J., Kilian, J., Leighton, T., Shamoon, T.: Secure spread spectrum watermarking for multimedia. IEEE Transactions on Image Processing **6** (1997) 1673–1687
8. M.D. Swanson, B. Zhu, A.T., Boney, L.: Robust audio watermarking using perceptual masking. Elsevier Signal Processing, Special Issue on Copyright Protection And Access Control **66** (1998) 337–335
9. W. Kim, J.L., Lee, W.: An audio watermarking scheme robust to mpeg audio compression. In: Proc. NSIP. Volume 1., Antalya, Turkey (1999) 326–330
10. D. Gruhl, A.L., Bender, W.: Echo hiding. In: Proceedings of the 1st Workshop on Information Hiding. Number 1174 in Lecture Notes in Computer Science, Cambridge, England, Springer Verlag (1996) 295–316
11. Bassia, P., Pitas, I., Nikolaidis, N.: Robust audio watermarking in the time domain. IEEE Transactions on Multimedia **3** (2001) 232–241
12. Domingo-Ferrer, J., Herrera-Joancomartí, J.: Simple collusion-secure fingerprinting schemes for images. In: Proceedings of the Information Technology: Coding and Computing ITCC'2000, IEEE Computer Society (2000) 128–132
13. Domingo-Ferrer, J., Herrera-Joancomartí, J.: Short collusion-secure fingerprinting based on dual binary hamming codes. Electronics Letters **36** (2000) 1697–1699
14. Boneh, D., Shaw, J.: Collusion-secure fingerprinting for digital data. In: Advances in Cryptology-CRYPTO'95. LNCS 963, Springer-Verlag (1995) 452–465
15. Purnhagen, H.: SQAM - Sound Quality Assessment Material (2001) `http://www.tnt.uni-hannover.de/project/mpeg/audio/sqam/`.
16. Steinebach, M., Petitcolas, F., Raynal, F., Dittmann, J., Fontaine, C., Seibel, S., Fates, N., Ferri, L.: Stirmark benchmark: audio watermarking attacks. In: Proceedings of the Information Technology: Coding and Computing ITCC'2001, IEEE Computer Society (2001) 49–54