Audio Watermarking Systems

Design, Implementation and Evaluation of an Echo Hiding Scheme Using Subjective Tests and Common Distortions

http://dx.doi.org/10.3991/ijes.v1i2.3281

Mohamed TARHDA, Rachid ELGOURI, Laamari HLOU
Ibn Tofail University, Kenitra, Morocco

***Abstract*—Today's digital media have made the product (digital content) very flexible and diminished the cost of its distribution. However, it contributes on piracy explosion as digital content can be duplicated and re-distributed at virtually no cost. Watermarking technology appears in order to protect the intellectual property and fight the piracy. It consists on embedding data like copyright labels inside a data source without changing its perceptual quality. In audio domain, watermarking techniques rely on the imperfection of the human auditory system in order to embed data.**

In this paper, we completed a design based on echo hiding technique and implement it in MATLAB. The main idea of this method is to embed data into an original signal by introducing an echo with the appropriate delay. Subjective listening tests reveal that the watermarks are imperceptible. Fidelity tests show that quantity of distortion imposed by watermarks on a signal is small. Robustness tests against common signal processing reveal good responses. The watermark information is always detectable and recoverable.

***Index Terms*—Audio watermarking, echo hiding, imperceptible, robustness, subjective listening test, watermarks.**

I. INTRODUCTION

Steganography and Watermarking are defined as special procedures for embedding signals into digital content [1]. In the first case, the message is hidden for a communication purpose and may be used in military operations. In the second case, it is hidden for commercial purpose and may contain a hidden copyright notice or serial number or even help to prevent unauthorized copying directly [2].

Audio watermarks are special signals embedded into digital audio. These signals are extracted by detection mechanisms and decoded. The embedded data should be inaudible to the human ear and should be statistically undetectable and resistant to malicious manipulations [3].

For audio signals, specifically music recording, various technologies have been developed for applying hidden watermarks [4]. The important requirements of audio watermarking systems are robustness, imperceptibility and an acceptable payload. It can be stated that an ideal watermarking scheme will possess all of these features. However, in practice, a trade-off should be found between robustness, payload and perceptibility [5]. There are many audio watermarking techniques and the choice of the implemented method depends on application.

The general goal of our work is to analyze the efficiency of watermarking technology in communication channels. In other terms, this work is focused on designing an audio watermarking scheme that presents the main following characteristics:

- The embedded information, called watermarks, should be imperceptible.
- It should be detectable. This means that it should be possible to recognize if watermark information is hidden into the signal or not.
- The embedded information should be recoverable.

II. AUDIO WATERMARKING SYSTEMS

In audio domain, watermarking techniques rely on the imperfection of the human auditory system in order to embed data. “Data hiding in audio signals is especially challenging because the human auditory system (HAS) is sensitive and operates over a wide dynamic range. However, there are some “holes” available as it has fairly small differential range. As a result, loud sounds tend to mask out quiet sounds. Additionally, the HAS is unable to perceive absolute phase, only relative phase. Finally, there are some environmental distortions so common as to be ignored by the listener in most cases.” [6].

So, while discussing different techniques of audio watermarking, it should be taken into account the extreme sensitivity of the HAS. It should be also known that a perfect audio watermarking scheme couldn’t be designed. Some of the audio watermarking required characteristics are inversely proportional to the others and vice versa. All methods have limitations [7]. However, depending on the application one can be preferred over the others. In the Table 1, we give an idea about applications of hiding data and the audio watermarking algorithms response to intentional and unintentional attacks.

TABLE 1
COMPARISON SUMMARY BETWEEN THE MOST POPULAR AUDIO WATERMARKING METHODS

	Low bit techniques	**Spread spectrum techniques**	**Phase coding techniques**	**Echo hiding techniques**	**Time scale modification techniques**
Brief Description	The basic idea of this method is to take advantage of the quantisation error that usually derives from digitising the audio signal. It consists on embedding data by replacing the least significant bit of each sampling point by a coded binary string	The basic idea of this method is to spread the watermark data across the entire audible spectrum **[8]**. The watermark is modulated with a pseudo-random sequence called PN sequence	The basic idea of this method is to embed data into signal phase, exploiting the fact that the human auditory system is not sensitive to absolute change of phase.	The basic Idea of this method is to take advantage of the temporal limitations of the human auditory system. The embedding process consists of an echoing of the original signal using a short delay that encodes the information **[9]**.	The basic idea of this method is to compress or expand the time scale of audio track in order to hide data. Thus watermark message is embedded by changing the length of the intervals between salient points of the audio signal **[10]**.
Imperceptibility	Audible noise may be introduced	Watermark may be audible because of its noise addition concept	High quality Imperceptible	Imperceptible. Sometimes, it makes the sound rich.	
Robustness	Poor immunity. Useful only in digital-to-digital environments	Robust to noise attacks		Robust to various attacks	Robust to time scale modification attacks
Data payload	Large capacity 1kbps/1kHz	Moderately low capacity	Low capacity	Moderate capacity	Low embedding rate
Secret key	Use of secret key	Use of secret key	Use of secret key	Use of secret key	Use of secret key
Implementation	Very simple. Easy to implement	Easy to implement	Easy to implement	Complexity in detection because of cepstrum computation	Moderately difficult to implement
Blind/non-blind watermark	Blind watermarking	Blind watermarking	Blind watermarking	Blind watermarking	Blind watermarking
Real time extraction	Slow	Moderately fast	Slow	Moderately fast	Fast
Miscellaneous	Useless in real watermarking applications	If perfect compression scheme exists in future, embedding of pseudo random sequence may be trivial or impossible.		It is attractive for its adjustable parameters.	

In summary, Watermarking technology appeared in order to protect the intellectual property and to fight the piracy. It consists in embedding data like copyright labels inside a data source without changing its perceptual quality. In audio domain, watermarking techniques rely on the imperfection of the human auditory system in order to embed data. There are many audio watermarking methods. The most popular of them are: Low bit method, Spread spectrum method, Phase coding method, Echo hiding method and Time scale modification method. After a comparison between these methods we choose to complete a design based on echo hiding technique and implement it in MATLAB.

III. ECHO HIDING TECHNIQUE

A. Basic idea

Echo hiding method is one of the most popular audio watermarking techniques. It has many interesting applications as copyright protection and data authentication. The basic idea is to hide watermark into echo. In fact, due to human auditory system characteristics, echo is not audible if the delay between the original signal and its echo is below a certain limit. Practically, this limit is about 2.5 milliseconds. Echo hiding is robust to multitude attacks, in particular to lossy data compression algorithms. In addition to this, using redundancy may make the echo more robust to removal attacks [11].

B. Encoding process

In this method we embed the watermark value by changing the delay between the original signal and its echo. The data are hidden by varying three parameters of the echo: initial amplitude, decay rate which is a relative volume of the echo compared to the original signal, and the offset which is the delay between the original signal and the echo (Fig. 1). For simplicity, we choose the case of embedding binary signals. Thus two different delays (called also kernels) and are used to respectively embed binary “zero” and binary “one”. Both delay times are below the threshold at which the human can resolve the echo.

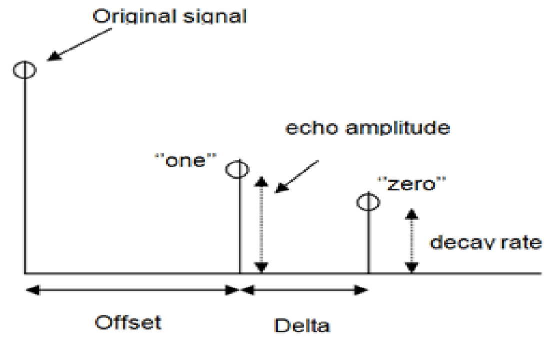

Figure 1. Adjustable parameters

Fig. 2 represents kernels that encode the binary data. These functions are convolved with the original signal in order to create its echoed versions.

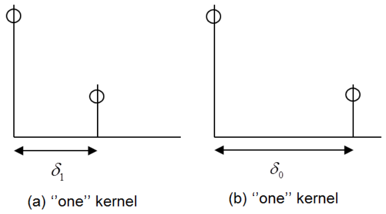

Figure 2. Echo kernels

Fig. 3 provides an echoing example using one kernel. The embedding process consists on convolution operation between the original signal and the chosen kernel. The delay between the original signal and the echo is dependent on which kernel is used.

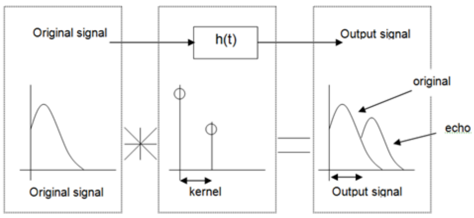

Figure 3. Echoing example

In order to encode more than one bit, the original signal is divided into smaller portions. Then each segment can be echoed with a specific kernel depending on the data we want to embed. Theoretically, each portion is individually convolved with the appropriate system function ("one" kernel or "zero" kernel). Practically, two echoed versions are created: one version is encoding "zero" kernel and the other version is encoding "one" kernel. In order to combine the two signals, two mixer signals are created. Thus, "one" mixer and "zero" mixer are used to attribute the appropriate bit to each portion. Downstream the mixers, we obtain the watermarked signal as if each portion was echoed individually.

C. Decoding process

In order to extract the embedded data, we have to detect the spacing between echoes. Actually, the autocorrelation function, the cepstrum function and the autocepstrum function can be used to separate the original signal from its echo and thus detecting the delay between two echoes. However, these functions have different performances. In our case, we make a comparison between all these methods and we observe that autocepstrum function offers the best results.

Cepstrum analysis is a non-linear signal processing technique with a variety of applications in areas such as speech processing. The term Cepstrum was first introduced by Bogert et al. and has come to be accepted terminology for the inverse Fourier transform of the logarithm of the power spectrum of a signal [12]. The complex Cepstrum for a sequence x is calculated by finding the complex natural logarithm of the Fourier transform of x, then the Inverse Fourier Transform of the resulting sequence. The complex Cepstrum transformation is central to the theory and application of homomorphic systems. This method is used particularly in an echo detection application. In fact, Bogert, Healy and Tukey (1963) observed in their paper "The Quefrency Analysis of Time Series for Echoes: Cepstrum, Pseudoautocovariance, Cross-Cepstrum, and Saphe Cracking." that the logarithm of the power spectrum of a signal, containing an echo has an additive periodic component due to the echo, and thus the Fourier transform of the logarithm of the power spectrum should exhibit a peak at the echo delay [13].

This result will be used in detection of the embedded delay. Before this, we would like to describe mathematically the Cepstrum analysis. This method can be decomposed into a canonical representation consisting of a cascade of three individual systems. These three systems are the Fourier transform, the complex logarithm and the inverse Fourier transform as shown in Fig. 4.

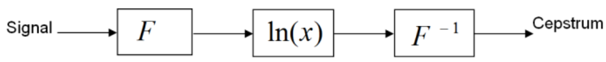

Figure 4. Canonical representation of a Cepstrum

The operational conversion is the result of a basic mathematical property:

First, we consider two signals $x(n)$ and $h(n)$.

$y(n)$ is the convolution result between $x(n)$ and $h(n)$. So:

$$y(n) = x(n) * h(n) \tag{1}$$

Convolution in the time domain is identical to multiplication in the frequency domain, so, using Fourier transform, we obtain:

$$Y(f) = X(f) \cdot H(f) \tag{2}$$

The logarithm of a product is the sum of the individual logarithms. So, we obtain:

$$Ln\big(Y(f)\big) = Ln\big(X(f) \cdot H(f)\big) = Ln\big(X(f)\big) + Ln\big(H(f)\big) \tag{3}$$

Finally, the inverse Fourier transform allows putting the system back into time domain:

$$F^{-1}\left[Ln\big(Y(f)\big)\right] = F^{-1}\left[Ln\big(X(f)\big)\right] + F^{-1}\left[Ln\big(H(f)\big)\right] \tag{4}$$

Using Cepstrums, the autocorrelation of a self-symmetric function can be found by first taking the Cepstrum of the function and then squaring the result. Then we obtain the Autocepstrum. The steps in this process are depicted in Fig. 5 and Fig. 6. Before squaring the Cepstrum, we first take the Fourier transform that places the system in the frequency domain where modifications are linear. After squaring operation, the Inverse Fourier Transform places us back in the time domain. The inverse Fourier transform from step one and the Fourier transform from step two will cancel each other when combined. In the end, we are left with the system shown in Fig. 7 [11].

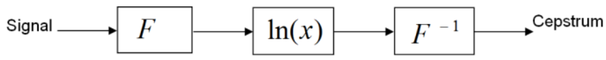

Figure 5. The first step in finding the Autocepstrum is to find the Cepstrum of the signal

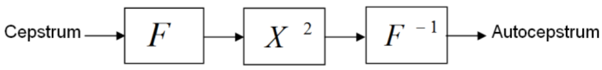

Figure 6. Once we have the Cepstrum, we square it to find the Autocepstrum

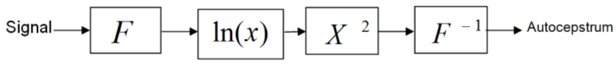

Figure 7. System representation of the autocepstrum

In case of the echo hiding technique, the watermarked signal can be considered as the convolution between the original signal and the kernel. Fig. 8 and Fig. 9 present respectively the Cepstrum and the Autoceptrum of the echo of a speech file, frequency 8 kHz and length 20800 samples (2 seconds):

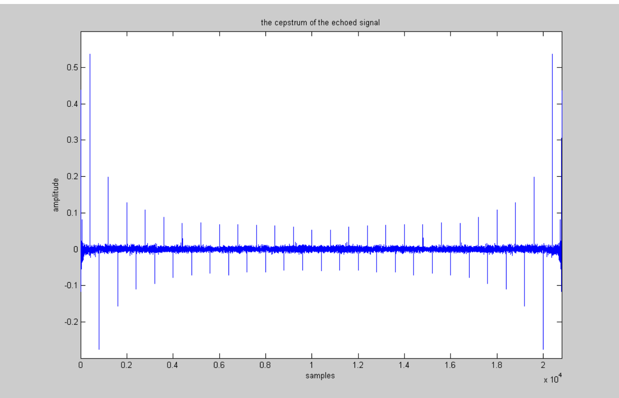

Figure 8. The cepstrum of the echoed signal

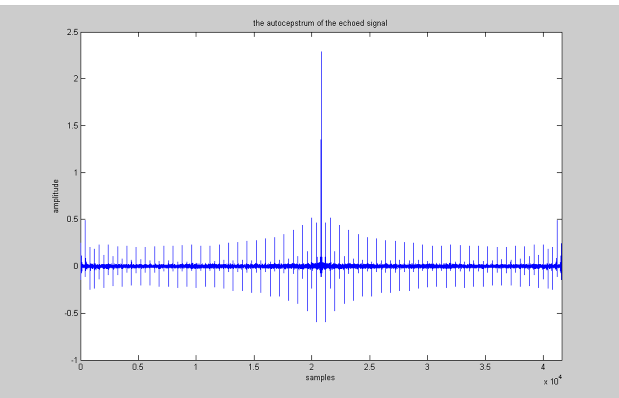

Figure 9. The autocepstrum of the echoed signal

In practice, autocepstrum function offers the best echo detection results. The difference between the first peak and the second one represent the delay between the original signal and its echo.

IV. DESIGN AND IMPLEMENTATION OF ECHO HIDING TECHNIQUE

A. Implementation of the encoder and decoder

Using the basic watermarking system scheme (Fig. 10), we constructed a system that contains two major blocks for embedding process and detection process. The key in our system is the set of global parameters which are necessary in encoding and decoding processes. In addition to this, we construct a third block to evaluate the performance of our design by comparing between encoded data and extracted one. In the following paragraphs, we describe these blocks.

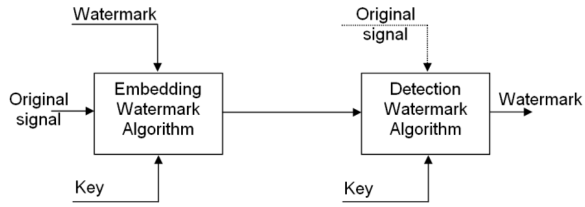

Figure 10. Basic watermarking system

In our system, we suppose that the length of the audio file does not change. Thus, during detection process, the watermarked file has the same length as the original one.

1) Embedding process

The encoding process consist in dividing the original signal into small segments, then echoing each segment with a specific delay according to the information to embed. The needed parameters are length of segments, used delays, attenuation applied to echoes and the information to embed. The information is repeated all over the file. After a pre-processing block, we obtain a matrix of echoed versions of the original signal. Each version corresponds to a specific delay. We also obtain "mixers", which are used to attribute the appropriate bit to each segment. Echoed versions are multiplied with mixers, and then summed up in order to construct the watermarked signal (Fig. 11).

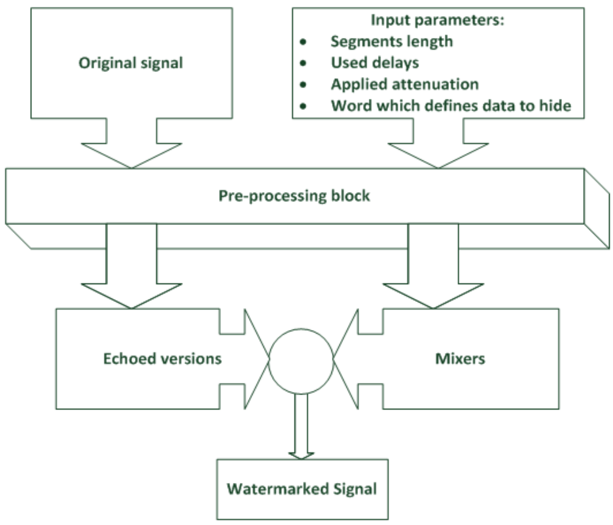

Figure 11. Embedding process scheme

2) *Extracting process*

In the extracting process, we insert the parameters: segments length, pattern (specific delay used to detect if there is watermark in the analysed file or not), used delays and the interval for searching the peak corresponding to the embedded delay. The detection process contains the following steps (Fig. 12):

- First, we look for the embedded pattern in the specific segments. If the pattern is detected then we decide that there are watermarks in the analysed file, otherwise not.
- If the pattern is detected with success, we look for the embedded delays in all segments. So we obtain long sequence of detected delays. This sequence contains the watermark information embedded several times.
- As we know the length of the embedded information, we estimate the embedded information by comparison. So even if some portions of the watermarked signal are destroyed, we can extract the embedded information.

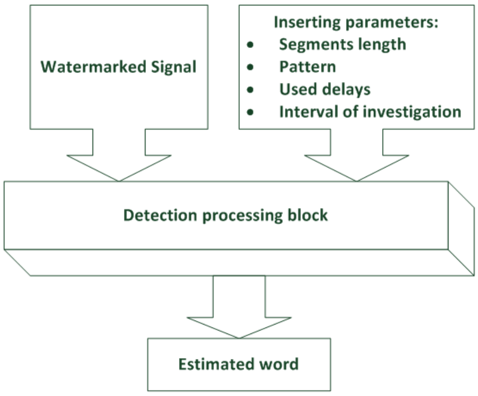

Figure 12. Detection process scheme

3) *Evaluation process*

This block is used to evaluate detection results. It allows us to check the bit error rate and the segment error rate (Fig. 13). By bit error rate we mean the percentage of erroneous detected delays. The embedded sequence can be divided into several portions that present the same embedded information but repeated. In detection process, we call segment error rate the percentage of erroneous detected portions.

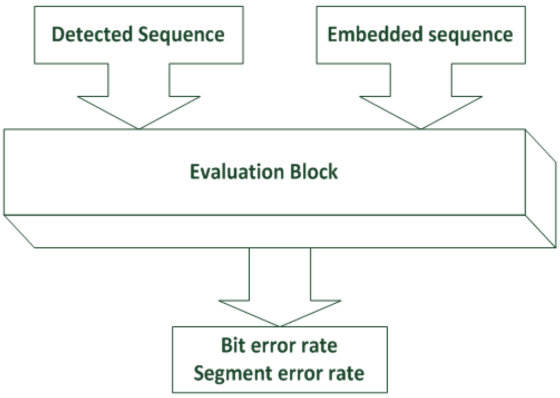

Figure 13. Evaluation process scheme

B. *Preliminary tests*

1) *Testing threshold of perceptibility distortion*

This has for goal to check the threshold of perceptibility of the echo. It was performed over a speech file, frequency 8 kHz and length 20800 samples (2 seconds). The whole file was delayed with the same delay.

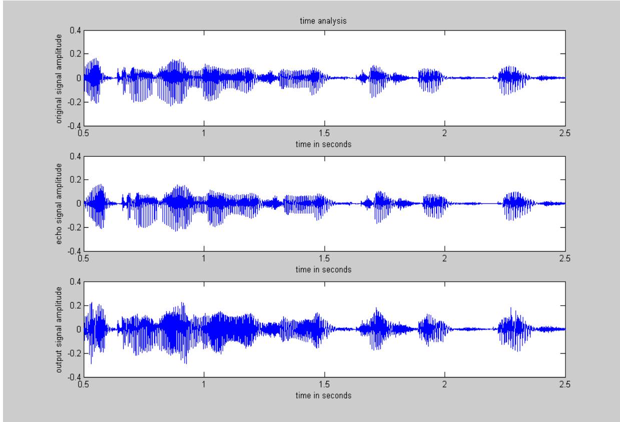

Figure 14. Time analysis of the original signal, its echo, and the output signal (the watermarked signal)

When the delay between the original and its echo becomes weak, the output signal shape tends to be like the original one (Fig.14). In this test, we find that the echo becomes imperceptible when the input delay is less than or equal to 2,5 milliseconds. When the echo delay is between 2,5 and 4 milliseconds, the perception looks like a change in the timbre of the sound, usually called coloration.

2) *Testing the effect of file type and the performance of used functions*

In this step the echo is not attenuated and the whole length of the signal is used. Downstream tests, we notice that the autocorrelation gives the worst results while results given by Cepstrum and Autocepstrum functions are more interesting. The Autocepstrum presents peaks amplitude higher than Cepstrum's ones. Furthermore, the detection results depend on the spectral characteristics of the used file. We find the worst results for classical music. In this kind of music, instruments operate in narrow spectral bands and the sound is very clear. Perhaps, this explains why the distortion is easy to detect in classical music.

3) *Testing attenuation effect*

Here, we use the whole length of the original signal but we change the attenuation applied to the echo, and we change the input delay. We notice that the attenuation is acceptable above 0.5. Under this value, detection of the delay is very often erroneous.

4) *Testing length effect*

Now, we fix the attenuation in the value of 0.5 and we observe the effect of the delay used during detection step. The frequency of the signal is 8000 kHz and the delays are equal or less than 1.3 millisecond. In fact, we remark that autocepstrum method offers the best result. Thus, we have a good result for the length of 50 samples, i.e. 6.25 milliseconds, with no error rate; however we remark that this rate is not stable. It is about 17.5% when the tested length is between 55 and 75 samples and it becomes about 50% it is between 80 and 105 samples. Similar results are obtained for the attenuation value 0.6 with some improvement for 55 to 75 samples whose rate error is 7.5% for attenuation value 0.7, but in this time the error rate for 55 to 75 samples is only 2.5%. These results can be explained by checking spectral characteristics of the signal. In fact, the original signal is a speech file. So the distribution of the energy is not monotonous and it is characterized by a great variety of its spectral shape. Thus, when the length of the signal is increased it does not mean that the result of the autocorrelation will necessarily be better.

In summary, we conclude:

- The performance of the detection depends on the type of files. For example the best results were obtained for noise-like files and the worst ones were observed for classical music.
- Autocepstrum function gives the best results in decoding process.
- Applying attenuation to echo made higher the perceptibility quality but it provided worse results in detection.
- In general, increasing the length of the watermarked sequence in detection process enhanced the echo detection.

V. TESTS AND SIMULATIONS

Here, we embedded information of 8 bits in different types of audio files. Then, we made several tests to check the correctness of our system, its performance and its robustness against some common signal processing operations.

A. Testing perceptibility

In order to test the imperceptibility of a watermarking technique, several methods have been discussed in literature. Here, we choose to do the perceptibility test of our system based on the ITU-R BS 1116-1 recommendations [14].

1) Subjective listening test: Presentation

Here, we introduce the method called the double-blind triple-stimulus with hidden stimulus. This method has been found to be especially sensitive, stable and to permit accurate detection of small impairments [14].

In our case, we use 16 different audio files. The first three trials were considered as a training phase that accustoms the listeners to the test. Two types of listeners were differentiated throughout the test:

- Professional listeners who are involved in audio watermarking techniques and are familiarized with distortions caused by imbedding data into audio files.
- Normal listeners with no special knowledge in music.

During the test, the listeners must give for each one of the stimuli B and C a grade from 1 to 5, according to the following table:

TABLE 2
ITU-R FIVE-GRADE IMPAIRMENT SCALE

Impairment	Grade
Imperceptible	5.0
Perceptible, but not annoying	4.0
Slightly annoying	3.0
Annoying	2.0
Very annoying	1.0

The watermarked signal is imperceptible when score given to it is 5.0 and it's very annoying for the score 1.0 (Table 2).

The aim of the test is to see whether the watermarked files keep good quality comparing to the original files or not. We deliberately choose to test the perceptibility of our algorithm using not optimum parameters. It makes confident that our algorithm will work better in the optimal cases. For these reasons, our tests have some constraints:

- In all files, we insert information using five different delays. Four delays are used for embedding watermarks and one is used as a pattern. The data rate is 26 embedded delays. The watermark is repeated in the whole track.
- The embedded information is the same in all tracks. The delays have the same value in seconds. Thus, we choose standard delays for files without taking into account their spectral characteristics. This explains why in some tracks the distortion caused by watermarks is obvious while it is totally imperceptible for others.
- The minimum value of the used delay is 1millisecond and the maximum value is 2.5 milliseconds (limit of coloration).
- The watermark is inserted since the first sample. In fact, when we begin inserting watermarks after some milliseconds after the beginning of the file, the perceptibility is improved.

In order to draw a relatively reliable conclusion the size of the listening panel is chosen as 10 subjects: 3 subjects are professional listeners and 7 subjects are normal listeners. The sequences can be played repeatedly for as long as the test person wants.

The results of the listening tests are presented according to the so-called subjective difference grade (SDG) [15] (table 3). It is calculated by subtracting the score assigned to the actual hidden reference signal from the score assigned to the actual coded signal:

$$SDG = Score_{Signal\_Under\_Test} - Score_{Reference\_Signal} \quad (5)$$

The following table shows the SDG value according to impairments:

TABLE 3
SUBJECTIVE DIFFERENCE GRADE

Impairment	Subjective Difference Grade (SDG)
Imperceptible	0.0
Perceptible, but not annoying	-1.0
Slightly annoying	-2.0
Annoying	-3.0
Very annoying	-4.0

2) Subjective listening test: Results

As it was expected, there is a difference in perceptibility tests results between the two types of listeners. For normal listeners, the watermarks are in general not perceptible. And for professional listeners, the watermarks are perceptible but not annoying. In addition to this, the results depend on the type of file. Table 4 presents these results:

TABLE 4
SUBJECTIVE LISTENING TESTS RESULTS

	Professional listeners		Normal listeners	
	Average	Interpretation	Average	Interpretation
Classical1	-1,83	Perceptible but not annoying	-0,6	Imperceptible
Speech	-1	Slightly perceptible	-0,74	Imperceptible
Blues1	-2,03	Slightly annoying	-0,94	Imperceptible
Classical2	-1,7	Perceptible but not annoying	-0,47	Imperceptible
Speech	0	Imperceptible	-0,17	Imperceptible
Country	-1,33	Perceptible but not annoying	-0,31	Imperceptible
Folk1	-1,5	Perceptible but not annoying	-1,42	Perceptible but not annoying
Folk2	-1,16	Perceptible but not annoying	-0,95	Imperceptible
Blues1	0	Imperceptible	-0,31	Imperceptible
Pop1	-1	Slightly perceptible	-0,85	Imperceptible
Dance	-2	Slightly annoying	-0,51	Imperceptible
Speech	-1,5	Perceptible but not annoying	-0,45	Imperceptible
Pop2	-2,33	Slightly annoying	-0,91	Imperceptible

Fig. 15 shows perceptibility test results. They present for all files the minimum and the maximum value given by listeners and the average value. The first graphic presents normal listeners results. Fig. 16 presents professional listeners results.

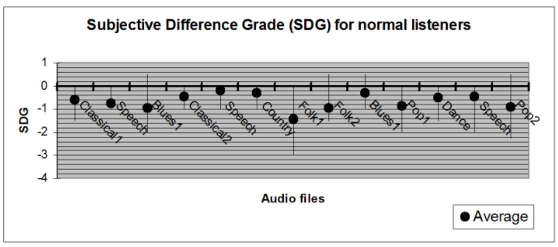

Figure 15. Subjective difference grade (SDG) for normal listeners

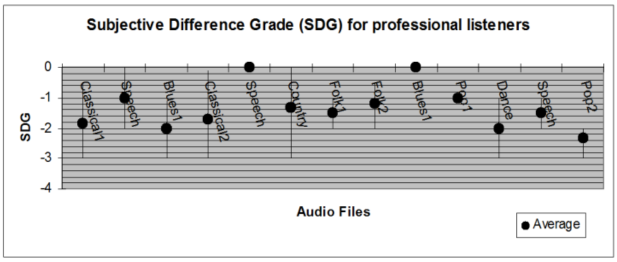

Figure 16. Subjective difference grade (SDG) for professional listeners

It is apparent from the results (Fig. 15 and Fig. 16) that our system shows imperceptible quality for listeners. Even sometimes, watermarked files can be perceived to be better than original ones (cases of blues1, blues2 and folk2)

B. Measuring fidelity

Tests to measure the amount of degradation caused to original signal are important to assure high quality of the watermarked signal. In general, content owners are more concerned with the degradation of the watermarked signal than users as they have access to both original signal and watermarked one. Hence, we talk about fidelity, which refers to the similitude between the original signal and the watermarked signal. Thus measuring fidelity consists on measuring the distortion induced by watermarks [16].

Many difference metrics have been discussed in the literature, but the most common is the signal to noise ratio. By noise, we mean the noise induced by inserting data into the original signal. This signal to noise ratio is generally expressed in decibels as follow:

$$SNR(dB) = 10 \cdot \log_{10}\left(\frac{\sum_{n}^{N} A_n^2}{\sum_{n}^{N} (A_n - A'_n)^2} \right) \qquad (6)$$

Where corresponds to the sample of the original audio file, and to the sample of the watermarked signal.

Ref. [17] mention that one could expect to have perceptible noise distortion for SNR values of 35 dB. In table 5, we notice that SNR value is less than 8 dB for all audio files. Thus, the quantity of distortion that a watermark imposes on a signal is tolerable.

TABLE 5
SIGNAL TO NOISE RATIO (SIGNAL IS THE ORIGINAL SIGNAL & NOISE IS THE NOISE DUE TO WATERMARKS INSERTION)

	SNR in dB
Country	6.22
Salsa	6.24
Blues	5.68
Folk1	6.03
Folk2	5.74
Mer Calme	4.89
Mer Forte	5.14
Pop1	7.11
Pop2	6.05
Dance	7.71
Classical	6.27
Speech	4.27

C. Measuring robustness

Watermarks have to be able to withstand a series of signal operations that are performed either intentionally or unintentionally on the cover signal and that can affect the recovery process.

In our case, we tested our algorithm for some common signal processing operations. To do this, we first embed data into several kinds of audio files. Second, we check the bit error rate and the segment error rate. By bit error rate, we mean the ratio of incorrect extracted bits to the total number of the embedded bits. Segment error rate is the ratio of the incorrect extracted watermark information to the total number of the embedded words. Third, we perform the chosen signal processing operations. We finally proceed to the extraction of the watermark. If watermark recovery is successful, then we calculate the bit error rate and the segment error rate (Fig. 17).

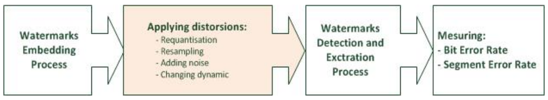

Figure 17. Performing robustness tests

To make the test more efficient, the embedded watermark is randomly generated. Furthermore, the test is repeated several times to avoid the case where information is recovered by chance. In the current project the chosen tests are the common processing operations. These kinds of attacks are known as signal diminishment attacks. Nowadays, they can be performed by anyone since the expansion of the Internet makes audio tools available to anyone even without an extensive knowledge of digital signal processing techniques. Table 6 shows the detection results before performing the chosen signal processing operations.

TABLE 6
WATERMARKS DETECTION RESULTS BEFORE PERFORMING ROBUSTNESS TESTS

Audio files	Bit error rate	Segment error rate	Watermarks
Folk1	1.42	6	Detectable & recoverable
Folk2	1.9	4	Detectable & recoverable
Mer Calme	0.95	0	Detectable & recoverable
Mer Forte	0.95	0	Detectable & recoverable
Pop1	2.85	8	Detectable & recoverable
Pop 2	0.47	0	Detectable & recoverable
Dance	15.7	42	Detectable & recoverable
Classical	14.7	38	Detectable & recoverable
Country	14.2	40	Detectable & recoverable
Salsa	5.7	20	Detectable & recoverable
Speech	15.2	40	Detectable & recoverable
Blues	19.6	43	Detectable & recoverable

The bit error rate is calculated using the whole length of the signal, but the segment error rate is calculated in taking into consideration only the length that corresponds to the completed embedded word. This explains why sometimes the segment error rate is null even if the bit error rate is not.

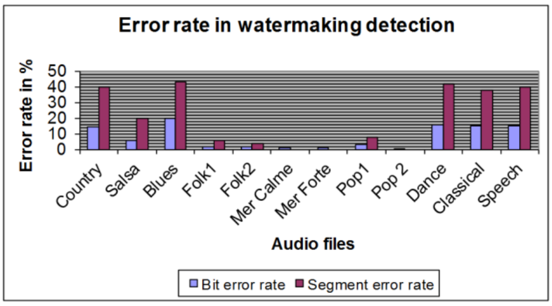

Figure 18. Error rate in watermarks detection before performing robustness tests

In general, we notice that audio files, which have noise-like characteristics, present small bit error rate (Fig. 18).

1) Dynamics

These operations change the loudness profile of the audio signal. The most basic way of performing this consists in increasing or decreasing the loudness directly. More complicated operations include limiting, expansion and compression, as they constitute non-linear operations that are dependent on the audio cover.

Fig. 19 presents detection results depending on the coefficient applied in order to change the audio file's dynamics. In general, the change of dynamics does not affect watermark detection seriously. The bit error rate is almost stable for all files. The watermark is always detectable and recoverable.

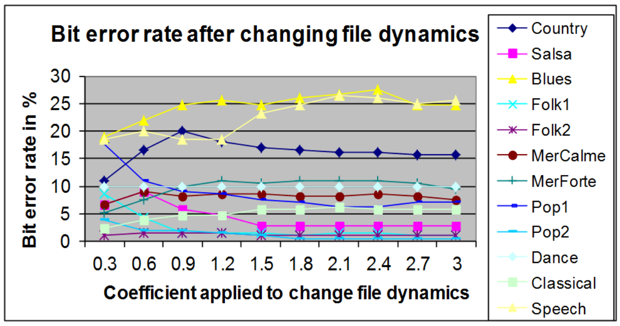

Figure 19. Error rate in watermarks detection after changing dynamics tests

2) Noise

Noise can be added in order to remove a watermark. This noise can even be imperceptible, if it is shaped to match the properties of the cover signal. Sometimes noise will appear as the product of other signal operations, rather than intentionally.

The following graphic shows detection results after adding white Gaussian noise (AWGN). We notice that blues, speech, and country music are the most affected by noise, but in general the bit error rate doesn't augment very much (Fig. 20). We note also that watermark information is always detectable and recoverable.

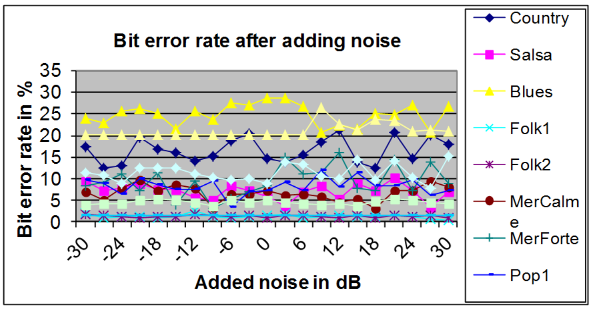

Figure 20. Bit error rate after adding noise tests

3) *Requantization*

The quantization of a signal may introduce errors in original signal value. The purpose of this test is to see the effect of the re-quantization on watermarks detection. Table 7 shows the results. In general, the bit error rate augments but the embedded information is still detectable and recoverable.

TABLE 7
WATERMARKS DETECTION RESULTS AFTER RE-QUANTIZATION TESTS

	bit error rate		segment error rate		watermarks
	Before test	After test	Before test	After test	
Country	14.2	19	40	39	Detectable & recoverable
Salsa	5.7	10	20	32	Detectable & recoverable
Blues	19.6	30	43	49	Detectable & recoverable
Folk1	1.42	1.9048	6	6	Detectable & recoverable
Folk2	1.9	1.9048	4	6	Detectable & recoverable
MerCalme	0.95	10.476	0	38	Detectable & recoverable
MerForte	0.95	2.8571	0	10	Detectable & recoverable
Pop1	2.85	9.0476	8	32	Detectable & recoverable
Pop2	0.47	1.9048	0	4	Detectable & recoverable
Dance	15.7	12.857	42	39	Detectable & recoverable
Classical	14.7	14.286	38	39	Detectable & recoverable
Speech	15.2	21.25	40	45	Detectable & recoverable

D. *Resampling*

This test consists in changing the frequency of the signal. Here, we choose to divide the original frequency by two. Table 8 shows the watermark detection results. In general the bit and segment error rate augment but watermarks are always detectable and recoverable.

TABLE 8
WATERMARKS DETECTION RESULTS AFTER RESAMPLING TESTS

	bit error rate		segment error rate		watermarks
	Before test	After test	Before test	After test	
Country	14.2	17	40	36	Detectable & recoverable
Salsa	5.7	5.7143	20	22	Detectable & recoverable
Blues	19.6	20.238	43	40	Detectable & recoverable
Folk1	1.42	1.4286	6	4	Detectable & recoverable
Folk2	1.9	1.4286	4	2	Detectable & recoverable
Mer Calme	0.95	8.0952	0	32	Detectable & recoverable
Mer Forte	0.95	9.5286	0	20	Detectable & recoverable
Pop1	2.85	8.0952	8	28	Detectable & recoverable
Pop2	0.47	1.4286	0	0	Detectable & recoverable
Dance	15.7	9.5238	42	30	Detectable & recoverable
Classical	14.7	4.7619	38	16	Detectable & recoverable
Speech	15.2	20.571	40	38.333	Detectable & recoverable

VI. CONCLUSION

In this paper, current state-of-the-art watermarking schemes are briefly summarized. The echo hiding technique was chosen for implementation because of its simplicity and high resistance to common signal processing in comparison with other techniques. Thus, we made an algorithm that allows embedding data into audio files. The redundancy of the watermark information makes the algorithm robust, and the embedded data can usually be recovered from just a small portion of the watermarked file. Our algorithm achieves the goals, and we summarize them as follows:

- The embedded information is imperceptible. In subjective listening tests, two types of listeners were differentiated: professional listeners, which are familiar with the audio watermarking domain, and normal listeners with no special knowledge in music. Perceptual tests show that watermarks are imperceptible for normal listeners. For professional listeners watermarks can sometimes be perceptible but not annoying.
- The watermark information is detectable. By using pattern, we can recognize if our watermarks are hidden into the signal or not.
- The embedded information is recoverable. The recovery of the embedded data is possible even if only fragments of the host signal are available.

Furthermore, the system presents good responses against usual signal processing operations. Different tests were performed over our algorithm. The results are interesting and promising. We test the performance and robustness of our system over common signal processing operations. On one hand, the bit error rate stays almost stable after changing dynamics or adding noise. On the other hand, resampling and re-quantization affect very slightly the detection performance. The best results are obtained for noise-like files and the worst ones are observed in classical music. In all cases, watermark information is detectable and recoverable. However, the error rate remains higher and makes the use of the system in the real applications limited. There are some solutions to this problem such as the use of multiple echoes or spreading echo in time domain. Thus, our project has several advantages and can be viewed as positive first step through a professional audio watermarking system.

REFERENCES

[1] Dora M. Ballesteros L, Juan M. Moreno, "*On the ability of adaptation of speech signals and data hiding*", Journal: Expert Systems with Applications, Volume 39 Issue 16, November, 2012, Pages 12574-12579

[2] Y. Cassuto, M. Lustig, G. Leifman, T. Mizrahi, E. Borenstein, S. Mizrachi, N. Peleg, "*Real Time Implementation for Digital Watermarking in Audio Signals Using Perceptual Masking*", paper for the 3rd European DSP Education and Research Conference, ESIEE, Noisy Le Grand, Paris (September 2000)

[3] Hyoung J. K. ,"*Audio Watermarking Techniques*", Department of Control and Instrumentation Engineering, Kangwon National University, Chunchon 200-701, Korea 2001

[4] Djebbar, F., Ayad, B., Hamam, H., & Abed-Meraim, K. "*A view on latest audio steganography techniques*", the innovations in information technology international conference on (IIT), 25–27 April 2011

[5] Arnold M., "*Audio Watermarking: Features, Applications and Algorithms*," Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2000), New York, Volume 2, July 2000, Pages 1013-1016

[6] Bender W., Gruhl D., Morimoto N., Lu A., "*Techniques for data hiding*", IBM Systems Journal, Vol. 35, 1996.

[7] Goenka, M. K. V., & Patil, M. P. K,"*Overview of Audio Watermarking Techniques*", International Journal of Emerging Technology and Advanced Engineering, SSN 2250-2459, Volume 2, Issue 2, February 2012

[8] Hartung, Frank, Su, Jonathan K., et GIROD, Bernd. "*Spread Spectrum Watermarking: Malicious Attacks and counterattacks*". Conference Security and Watermarking of Multimedia Contents, Proc SPIE, January 1999, vol. 3657, p. C1

[9] Brickman Adam., "*Literature Survey on Audio Watermarking*", EE381K-Multidimensional Signal Processing, March, 2003, vol. 3.

[10] Mansour M. F., and Tewfik A. H., "*Audio Watermarking by Time-Scale Modification,*" IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'01), Salt Lake City, May 2001. Pages 1353-1356.

[11] Bender W., Gruhl D., and Lu A., "*Echo Hiding*", Book Information Hiding - First International Workshop Cambridge, U.K., Springer Berlin Heidelberg,1996. Pages 295-315.

[12] Rabiner L. R., and Schafer R. W., Book "*Digital Processing of Speech Signals*", Prentice Hall, Englewood Cliffs, NJ, 1979.

[13] A.V. Oppenheim and R.W. Schafer, Book "*Discrete-Time Signal Processing*", Prentice Hall, Englewood Cliffs, NJ, Vol. 5. 1989

[14] International Telecommunication Union Recommendations ITU-R BS.1116-1, "*Methods for the subjective assessment of small impairments in audio systems including multichannel sound systems*", Approved in 1997-10

[15] Arnold M., "*Subjective and Objective Quality Evaluation of Watermarked Audio Tracks*", Proceeding of the Second International Conference on WEB Delivering of Music, Germany, 2002, Pages 161-167

[16] Gordy J. D., and Bruton L. T., "*Performance Evaluation of Digital Audio Watermarking Algorithms*" Circuits and Systems conference, Proceedings of the 43rd IEEE Midwest Symposium on. IEEE, 2000. Volume 1, Pages 456-459.

[17] Petitcolas, F. A., & Anderson, R. J. *Evaluation of copyright marking systems*. In Multimedia Computing and Systems, 1999. IEEE International Conference July, 1999, Vol. 1, pp. 574-579.

AUTHORS

Mohamed TARHDA is with Electrical Engineering and Energy Systems Laboratory at Faculty of Science, University IBN TOFAIL Kenitra Morocco (e-mail: tarhdamo@yahoo.fr)

Rachid ElGOURI is Professor Assistant with Electrical Engineering and Energy Systems Laboratory at Faculty of Science, University IBN TOFAIL Kenitra Morocco (e-mail: elgouri.rachid@yahoo.fr).

Laamari HLOU is Professor with Electrical Engineering and Energy Systems Laboratory at Faculty of Science, University IBN TOFAIL Kenitra Morocco (e-mail: hloul@yahoo.com)

Submitted 23 October 2013. Published as re-submitted by the authors 02 November 2013.