Audio watermarking based on Fibonacci numbers

Mehdi Fallahpour [1] and David Megías [2], Member IEEE

[1] School of Information Technology and Engineering (SITE), University of Ottawa,Ottawa, Canada

E-mail: mfallah@site.uottawa.ca, fallahpour@gmail.com

[2] Estudis d'Informàtica, Multimèdia i Telecomunicació, Internet Interdisciplinary Institute (IN3)

Universitat Oberta de Catalunya, Barcelona, Spain

E-mail: dmegias@uoc.edu

Abstract

This article presents a novel high capacity audio watermarking system to embed data and extract them in a bit-exact manner by changing some of the magnitudes of the FFT spectrum. The key idea is to divide the FFT spectrum into short frames and change the magnitude of the selected FFT samples using Fibonacci numbers. Taking advantage of Fibonacci numbers, it is possible to change the frequency samples adaptively. In fact, the suggested technique guarantees and proves, mathematically, that the maximum change is less than 61% of the related FFT sample and the average error for each sample is 25%. Using the closest Fibonacci number to FFT magnitudes results in a robust and transparent technique. On top of very remarkable capacity, transparency and robustness, this scheme provides two parameters which facilitate the regulation of these properties. The experimental results show that the method has a high capacity (700 bps to 3 kbps), without significant perceptual distortion (ODG is about –1) and provides robustness against common audio signal processing such as echo, added noise, filtering and MPEG compression (MP3). In addition to the experimental results, the fidelity of suggested system is proved mathematically.

***Index Terms*—** Multimedia security, audio watermarking, Fibonacci numbers, Golden ratio

I. INTRODUCTION

In the current information age, with the rapid development of various communication techniques, transferring digital multimedia content becomes more and more usual. However, the illegal copy and distribution of digital multimedia content has also become easier, and a large number of authors' and

publishers' intellectual property copyrights have suffered from violation, which have led to huge damage of their benefits in many applications. Thus, people pay more attention to copyright management and protection nowadays. Embedding secret information, known as watermarks, into multimedia content is considered as a potential solution to copyright infringement [1].

Digital watermarking is a process by which a watermark is hidden or embedded into a media (cover data), for example digital content such as electronic documents, images, audio and video. These embedded data can later be detected or extracted from the marked signal for various applications. There are several applications of audio watermarking including copyright protection, copy protection, content authentication, fingerprinting and broadcast monitoring.

An audio watermarking system may have different properties but must satisfy the following basic requirements:

1. Imperceptibility: The quality of the audio should be retained after adding the watermark. Imperceptibility can be evaluated using both objective and subjective measures.
2. Security: Watermarked signals should not reveal any clues about the watermarks in them. Also, the security of the watermarking procedure must depend on secret keys, but not on the secrecy of the watermarking algorithm.
3. Robustness: The ability to extract a watermark from a watermarked audio signal after various signal processing or malicious attacks.
4. Payload: The amount of data that can be embedded into the host audio signal without losing imperceptibility. For audio signals, data payload refers to the number of watermark data bits that may be reliably embedded within a host signal per unit of time, usually measured in bits per second (bps).

Considering the embedding domain, audio watermarking techniques can be classified into time-domain and frequency-domain methods. In frequency-domain watermarking [2-15], after applying one of the usual transforms such as the Discrete/Fast Fourier Transform (DFT/FFT) [5-7, 11, 12], the Modified Discrete Cosine Transform (MDCT) or the Wavelet Transform (WT) from the signal [8, 10, 13 ,14 ,15], the hidden bits are embedded into the resulting transform coefficients.

In frequency-domain methods, the Fourier transform (FT) is very popular. Among different Fourier transform, the Fast Fourier transform (FFT) is often used due to its reduced computational burden and it has been the chosen transform for the proposed scheme. This transform is also used by different authors, such as in [16], which proposes a multi-bit spread-spectrum audio watermarking scheme based on a

geometric invariant log coordinate mapping (LCM) feature. The watermark is embedded in the LCM feature, but it is actually embedded in the Fourier coefficients which are mapped to the feature via LCM. Consequently, the embedding is actually performed in the FT domain. In Ref. [5, 7, 11, 12], which were proposed by the authors of this paper, the FFT domain is also selected to embed watermarks to take advantage of the translation-invariant property of the FFT coefficients to resist small distortions in the time domain. In fact, using methods based on transforms provides better perceptual quality and robustness against common attacks at the price of increasing the computational complexity with respect to time-domain approaches. Ref. [27] presents a time-spread echo-based audio watermarking scheme with optimized imperceptibility and robustness. Specifically, convex optimization-based finite-impulse-response (FIR) filter design is used to obtain the optimal echo filter coefficients. The desired power spectrum of the echo filter is designed by the maximum power spectral margin (MPSM) and the absolute threshold of hearing (ATH) of the human auditory system (HAS) to ensure the optimal imperceptibility.

In the algorithm suggested in this paper, we select a part of the frequency of FFT spectrum for embedding the secret bits. The selected frequency band is divided into short frames and a single secret bit is embedded into each frame. The largest Fibonacci number that is lower than each single FFT magnitude in each frame must be computed and, depending on the corresponding secret bit to be embedded, all samples in each frame are changed. If the secret bit is “0”, all FFT samples in a frame should be changed to the closest Fibonacci number with even index. If the secret bit is “1”, all FFT samples in a frame should be changed to closest Fibonacci number with odd index.

As mentioned above, the FFT is used to design a scheme in many watermarking systems. To the best of our knowledge, this is the first audio watermarking method based on Fibonacci numbers. Using Fibonacci numbers for embedding the secret bits increases transparency and robustness against attacks, whereas embedding a secret bit into a single FFT sample is usually very fragile. Almost all watermarking methods rely on experimental results to prove the fidelity of watermarking system. However, in this article, in addition to the experimental results, the fidelity of suggested system is proved mathematically.

The experimental results show that this method achieves a high capacity (about 0.7 to 3 kbps), provides robustness against common signal processing attacks (even for strong disturbances) and entails very low perceptual distortion.

The rest of the paper is organized as follows. In Section 2, Fibonacci numbers are presented. Section 3 presents the proposed scheme. Section 4 provides discussion about the fidelity. In Section 5, the experimental results are shown. Finally, Section 6 summarizes the most relevant conclusions of this research.

II. FIBONACCI NUMBERS AND GOLDEN RATIO

The numbers 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, …, known as the Fibonacci numbers, have been named by the nineteenth-century French mathematician Edouard Lucas after Leonard Fibonacci of Pisa, one of the best mathematicians of the Middle Ages, who referred to them in his book Liber Abaci (1202) in connection with his rabbit problem. The Fibonacci sequence has fascinated both amateurs and professional mathematicians for centuries due to their abundant applications and their ubiquitous habit of occurring in totally surprising and unrelated places [17]. In this paper we apply Fibonacci numbers for the first time for audio watermarking.

The equation to produce the sequence of Fibonacci numbers is given below:

$$F_n = \begin{cases} 0, & \text{if } n < 0, \\ 1, & \text{if } n = 1, \\ F_{n-1} + F_{n-2,} & \text{if } n > 1. \end{cases} \quad (1)$$

Fibonacci numbers have very interesting features. One of the most famous ones, which we use in this article, is the ratio between two consecutive Fibonacci numbers [25].

$$F_n = F_{n-1} + F_{n-2}; \quad (2)$$

$$\frac{F_n}{F_{n-1}} = \frac{F_{n-1}+F_{n-2}}{F_{n-1}} = 1 + \frac{F_{n-2}}{F_{n-1}} = 1 + \frac{1}{\frac{F_{n-1}}{F_{n-2}}}; \quad (3)$$

$$\lim_{n\to\infty} \frac{F_n}{F_{n-1}} = \lim_{n\to\infty} \left(1 + \frac{1}{\frac{F_{n-1}}{F_{n-2}}}\right) = 1 + \frac{1}{\lim_{n\to\infty} \frac{F_{n-1}}{F_{n-2}}}; \quad (4)$$

$$\text{If } \lim_{n\to\infty} \frac{F_n}{F_{n-1}} = \varphi\,; \quad \varphi = 1 + \frac{1}{\varphi}; \quad (5)$$

$$\varphi^2 - \varphi - 1 = 0; \quad (6)$$

$$\varphi = \frac{1\pm\sqrt{5}}{2}. \quad (7)$$

As φ is positive, then $\varphi = 1.618$.

In fact, φ is the Golden Ratio which is an irrational number with several curious properties. The Golden Ratio is an irrational number, but not a transcendental one (like π), since it is the solution of a polynomial equation. The Golden Ratio likely obtained its name from the Golden Rectangle, a rectangle whose sides are in the proportion of the Golden Ratio. The philosophy of the Golden Rectangle is an aesthetic one: the ratio is an aesthetically pleasing one and it can be found spontaneously or deliberately turning up in a

great deal of art. Therefore, for instance, the front of the Parthenon can be comfortably framed with a Golden Rectangle. How beautiful the Golden Rectangle is, how often it really does turn up in art, and whether it does really frame the front of the Parthenon, may be largely a matter of interpretation and preference. Each Fibonacci number can be represented by the Golden Ratio [25]. Equation (8) shows how each Fibonacci number is generated by the Golden Ratio.

$$F_n = \frac{\varphi^n - \bar{\varphi}^n}{\sqrt{5}}, \quad (8)$$

where $\bar{\varphi}$ is the negative solution of Eqution (7).

III. PROPOSED SCHEME

Extensive work has been performed over the years in understanding the characteristics of the human auditory system (HAS) and applying this knowledge to audio compression and audio watermarking.

Figure 1 illustrates the range of frequencies and intensities of sound to which the human auditory system responds. The absolute threshold, the minimum level of sound that is detectable by human ear, is strongly dependent on frequency. At the level of pain, sound levels are about six orders of magnitude above the minimal audible threshold. The sound pressure level (SPL) is measured in decibels (dB). Decibels constitute a logarithmic scale, such that each 6 dB increase represents a doubling of intensity. The perceived loudness of a sound is related to its intensity.

Generally, we hear sounds as low as 20 Hz and as high as 20,000 Hz. The frequency of a sound is associated with its pitch. Hearing is best at about 3-4 kHz and sensitivity decreases at higher and lower frequencies, but more so at higher than lower frequencies. Thus, it is clear that, by embedding data in the high frequency band, which is used in the proposed scheme, the distortion will be mostly inaudible and thus more transparency will be obtained [11].

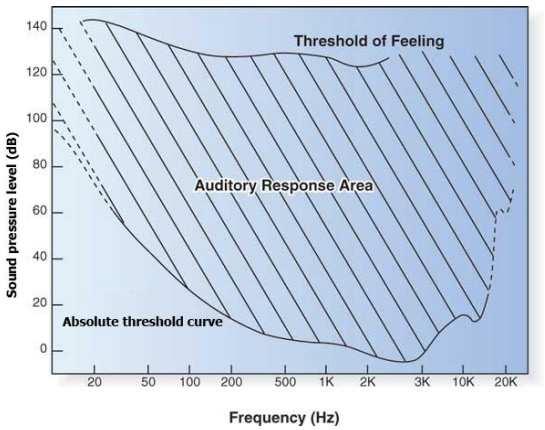

Fig 1. Typical absolute threshold curve of the human auditory response

In the suggested watermarking scheme, we use the following algorithm to embed a watermark logo (secret bit stream) into the FFT coefficients. First of all, the parameters should be adjusted based on the desired capacity, transparency and robustness. The frequency band and frame size are two parameters that set the properties of the proposed watermarking method. The selected frequency band is divided into short frames then each single secret bit of the watermark stream is embedded into all samples of a frame, which makes the method more robust against attacks.

A. TUNING

The suggested system provides two parameters to adjust three properties of the watermarking system. The frequency band, and the frame size (*d)* are the two parameters of this method to adjust capacity, perceptual distortion and robustness. In this scheme, we have general tuning rules which can help us to reach the requirements or to get close to them very fast. The frame size has more effect on robustness, whereas the frequency band has more effect on transparency and capacity. In other words, by increasing the frame size better robustness is achieved. Furthermore, increasing the frequency band leads to better capacity and more distortion.

Note that these parameters allow regulating the ODG between 0 (not perceptible) and –1 (not annoying), with about 650 to 3000 bits per second (bps) of capacity and allowing robustness against MP3-128, which are extremely better than typical requirements.

As most MP3 cut-off frequencies are higher than 16 kHz, the high frequency band, f_h, is set to 16 kHz or lower. Then, to select the frequency band, basically the low frequency band, f_l, should be adjusted. The default value for low frequency band is 12 kHz. Decreasing f_l implies increasing capacity and distortion. Increasing the frame size, d, results in a better robustness, but capacity decreases. The default value for the frame size is $d = 5$.

Fig. 2 shows the flowchart for the selection of the tuning parameters. In the initialization, f_l is 12 kHz, f_h is 16 kHz and d is 5. This flowchart facilitates adjusting the parameters based on the requirements. However, adjusting the parameters based on some demands is very difficult and considering a trade-off between capacity, transparency and robustness is always necessary.

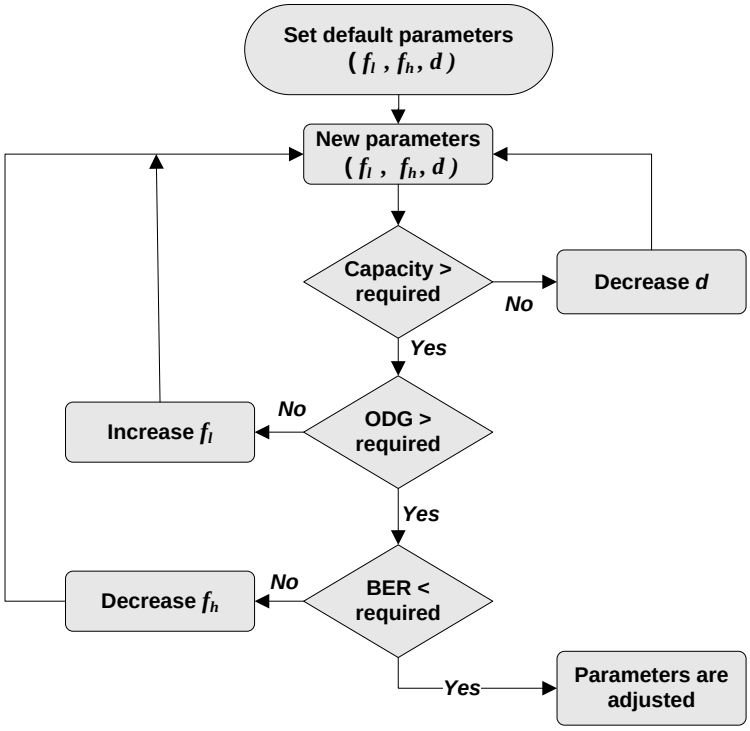

Fig. 2. Flowchart of the tuning process

B. EMBEDDING THE SECRET BITS

The frequency band and the frame size (*d*) are the two required parameters in the embedding process which have to be adjusted according to the requirements. In this section, for simplicity, we do not deal with the regulation of these parameters and just consider them fixed. The effects of these parameters are analyzed in the experimental results part.

For embedding the watermark stream, first the FFT is applied to the audio signal and then, the FFT samples are modified based on Fibonacci numbers and the secret bits. Finally the inverse FFT is applied to generate the marked audio signal. The embedding steps are detailed below.

1. Apply FFT to compute the FFT coefficients of the audio signal. We can use the whole file (for short clips, *e.g.* with less than one minute) or blocks of a given length (*e.g.* 10 seconds) for longer files.
2. Divide the FFT samples in the selected frequency band into frames of size *d*.
3. For all the FFT samples in the current frame, find the largest Fibonacci number $\{fib_{n,i}\}$, the n^{th} Fibonacci number for i^{th} FFT sample, which is lower than the magnitude of the FFT sample $\{f_i\}$. It is worth to mention that we use the following Fibonacci set:

$$F = \{1, 2, 3, 5, 8, 13, 21, 34, 55, \dots\}$$

In the original Fibonacci set there are two ones, one of which is removed in our algorithm.

4. The marked FFT samples $\{f_i'\}$ are obtained by using Equation (9).

$$f_i' = \begin{cases} fib_{n,i}, & \text{if } n \bmod 2 = 0 \text{ and } w_l = 0, \\ fib_{n+1,i}, & \text{if } n \bmod 2 = 1 \text{ and } w_l = 0, \\ fib_{n+1,i}, & \text{if } n \bmod 2 = 0 \text{ and } w_l = 1, \\ fib_{n,i}, & \text{if } n \bmod 2 = 1 \text{ and } w_l = 1. \end{cases} \quad (9)$$

Where $l = \lfloor i/d \rfloor + 1$, w_l is the *l*-th bit of the secret stream and $\lfloor x \rfloor$ denotes the largest integer value lower than or equal to x. Each secret bit is embedded into a suitable frame, in other words, each frame represents a single secret bit.

5. Finally, use the inverse FFT to obtain the marked audio signal.

By enlarging the frequency band, the capacity and distortion increase and robustness decreases. Also, increasing the frame size, strengths the robustness against attacks and reduces the capacity. In addition, the use FFT magnitudes results in better robustness against attacks compared to the use of the real or the imaginary parts only. Fig. 3 provides the flowchart of the embedding algorithm.

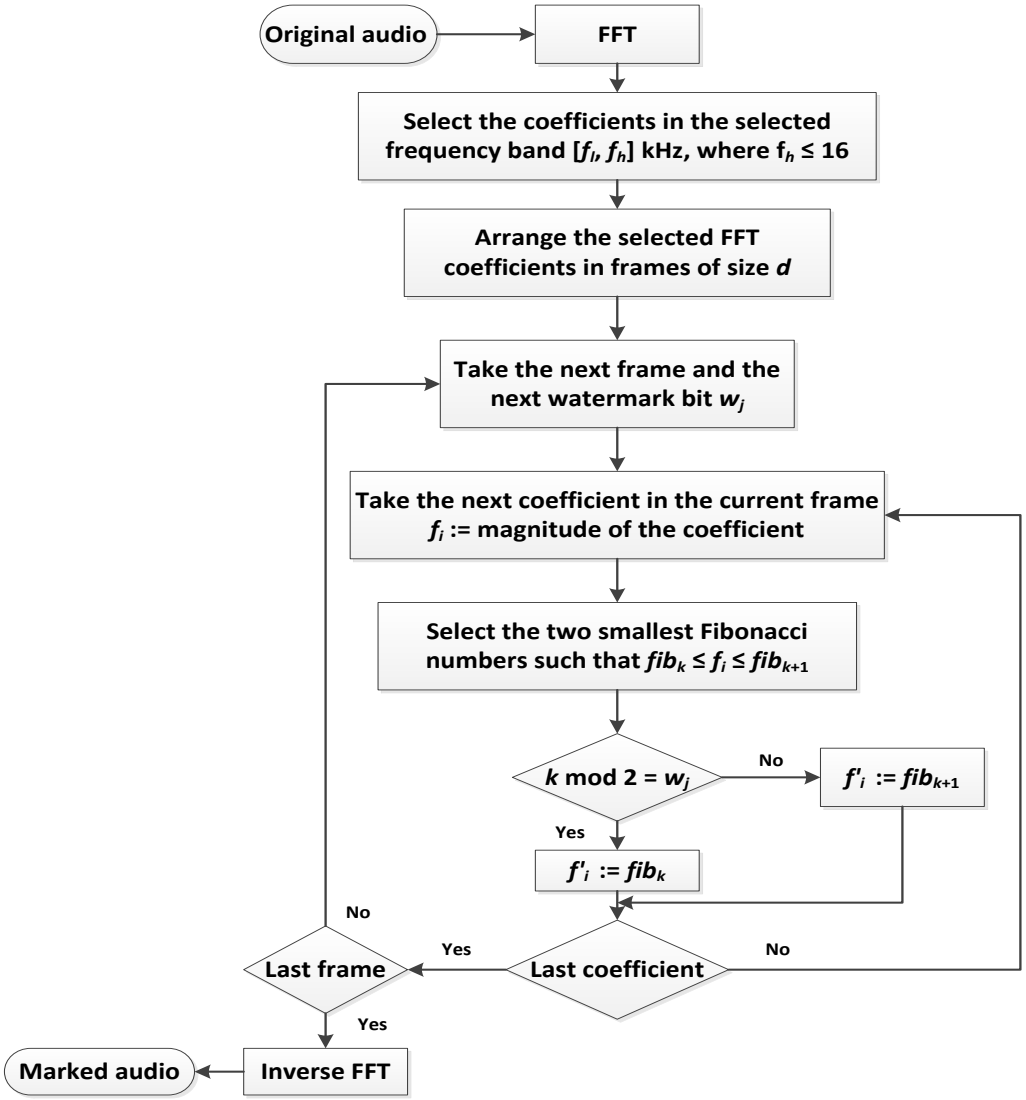

Fig. 3. Flowchart of the embedding algorithm

C. EXTRACTING THE SECRET BITS

The host audio signal is not required in the detection process, and hence, the detector is blind. The detection parameters, the frame size and the frequency band, can be transmitted in a secure way to the detector or standard parameters can be used for all audio signals. The detection process can be summarized in the following steps:

1. Apply the FFT to compute the FFT coefficients of the marked audio signal.
2. Divide the FFT samples in the selected frequency band into frames of size *d*.

3. For each single FFT sample in current frame, find the closest Fibonacci number $\{fib'_{n,i}\}$, the n^{th} Fibonacci number for i^{th} FFT sample, to the magnitude of the FFT sample $\{f_i'\}$. If the FFT sample has the same distance from two Fibonacci numbers, we select the lower Fibonacci number.

   We use $\{1, 2, 3, 5, 8, 13, 21, 34, 55, \dots\}$ as the Fibonacci set.

4. To detect a secret bit in a frame, each sample should be examined to check if it is a zero ("0" embedded) or a one ("1" embedded). Then, depending on the evaluation for all samples in the current frame, a secret bit can be extracted. The watermark bit, w_l' can be extracted by using the following equation:

$$B_i' = \begin{cases} 0, & \text{if } n \bmod 2 = 0, \\ 1, & \text{if } n \bmod 2 = 1. \end{cases} \qquad (10)$$

Where B_i' is the bit extracted from each sample. After getting information about all samples, based on the number of samples which represent "0" or "1" (voting scheme) a secret bit can be extracted for each single frame. If the number of samples identified as "0" is equal to or larger than half the frame size, the extracted bit is "0", otherwise it is "1". For example if the frame size is five and we detect two "0" and three "1", then the extracted secret bit of the frame would be "1".

D. SECURITY

The tuning parameters provide a first level of security in the system. An attacker trying to erase, replace or extract the embedded watermark will not be able to perform these actions if he or she does not know the embedding frequency range and/or the frame size. However, even if an attacker knows or can guess these secret values, the embedded watermark can be further protected with cryptography.

To increase security, a pseudo-random number generator (PRNG) can be used to change the secret bit stream to another stream which makes it more difficult for an attacker to extract the secret information. For example, the embedded bit stream can be constructed as the XOR sum of the real watermark and a pseudo-random bit stream. The seed of the PRNG would be required as a secret key both at the sender and the detector. There are many cryptography techniques [29] that can be used to increase the security of the system. Based on the requirements of the watermarking system, a cryptographic method should be chosen. For example, if we want to increase security, AES encryption is a good choice in terms of complexity.

IV. DISCUSSION

The main idea of using Fibonacci numbers is keeping the modification error in an acceptable range. Here, we prove that the maximum modification error is 60% of the correlated FFT sample in the typical case. Imagine we want to convert the original value of a sample, *s*, to the closest Fibonacci number.

$$F_n \leq s \leq F_{n+1}, \quad (11)$$

where *s* is between two Fibonacci numbers, F_n and F_{n+1}, based on the distance to each it can be converted to closest one.

$$e_1 = s - F_n, \quad (12)$$

$$e_2 = F_{n+1} - s, \quad (13)$$

$$\mathbf{max}\ \boldsymbol{error} = \mathbf{max}\,(\boldsymbol{e_1}, \boldsymbol{e_2}) = (\boldsymbol{F_{n+1}} - \boldsymbol{F_n}) = \boldsymbol{F_{n-1}}. \quad (14)$$

To find the error ratio we need to find the ratio between Fibonacci numbers. Assume that r_n is the ratio between two Fibonacci numbers.

$$r_n = \frac{F_{n+1}}{F_n}\ (n = 1, 2, \dots). \quad (15)$$

$$r_1 = 2, r_2 = 1.5, r_3 = 1.66,\ r_4 = 1.6, r_5 = 1.625, r_6 = 1.615, \dots$$

As shown above, when *n* is very large, r_n is equal to φ. Even when *n* >3 it is very close to φ. Thus we can summarise the max error as below;

$$\mathbf{max\ error} = (\boldsymbol{F_{n+1}} - \boldsymbol{F_n}) = (\boldsymbol{r_n F_n} - \boldsymbol{F_n}) = (\boldsymbol{r_n} - \mathbf{1})\boldsymbol{F_n}. \quad (16)$$

Theorem 1: According to the result presented in Equation (16), the "typical maximum" distortion introduced in the magnitude of a FFT sample using this embedding system is between 0.38 and 0.61.

Proof:

1. If *s* is converted to F_{n+1}

$$\text{max error rate} = \frac{\text{max error}}{F_{n+1}} = \frac{(r_n - 1)F_n}{F_{n+1}} = \frac{(r_n - 1)F_n}{r_n F_n} = \frac{r_n - 1}{r_n}. \quad (17)$$

2. if *s* is converted to F_n

$$\text{max error rate} = \frac{\text{max error}}{F_n} = r_n - 1. \quad (18)$$

Hence, if we assume that the "typical" value is r_n=1.61, the maximum error rate would be between 0.38 and 0.61. Thus the average "typical" maximum error rate would be 0.50. This completes the proof.

Note that this "typical maximum" distortion may be exceeded for some small values of *n*. However, for the vast majority of cases, the maximum distortion would be below 50% of the original magnitude. FFT samples will typically have high values but if they are less than 3, r_n would be equal to 2 or 1.5. Thus in the worst case, which occurs rarely, the maximum error would be between 0.66 and 1.

If the FFT samples have uniform distribution, in other words, all values has equal probability the average error rate is 0.25. It means average change for each FFT sample is 25%. This fact has a remarkable effect as imperceptibility is taken into account.

The good point about Fibonacci numbers is that the distance between 0s and 1s automatically adapts to the magnitude of the FFT coefficient to be modified. For example for a coefficient with magnitude 1.8, we would choose 1 or 2 to embed 1 or 0, respectively, but, with a magnitude of 6.3, we would choose between 5 (to embed a 0) and 8 (to embed a 1). The distance is adapted taking into account the magnitude of the coefficient to be changed. This is a good way to obtain a convenient trade-off between transparency and robustness. This does not happen, for example, in Quantization Index Modulation [26]. In that case, the distance between zeroes and ones is uniform and equal to the quantization step.

However, this is not the only method to obtain an "exponential-like spacing" of the marked coefficients. Assume that, instead of Fibonacci numbers, we use another sequence defined as follows[1]:

$$F_n = \lfloor k^n \rfloor, \quad \text{for } n = 1, 2, \dots , \qquad (19)$$

where

$$\frac{F_{n+1}}{F_n} \approx k, \quad \text{for } n = 1, 2, \dots \quad (20)$$

We have the following options for *k*:

1- If $|k| < 1$, the generated sequence is formed by numbers lower than one, which is not suitable for this system.
2- If $|k| > 2$, the increase rate is too high and this is not practical for the suggested method.
3- If $1<|k|<2$, the results are practical. Table I presents these sequence when *k* is 1.3, 1.5, 1.7, and 1.9. This table shows that, when *k* is 1.5 or 1.7, the sequence is close to that Fibonacci numbers.

[1] We select integer numbers only for a fair comparison with the Fibonacci sequence.

As proved in the discussion part, when n >3, the ratio between the Fibonacci numbers is very close to φ (Golden ratio). In other words if $k = \varphi$, the generated sequence will be very almost identical to the Fibonacci numbers.

TABLE I. Series with different k

n	1	2	3	4	5	6	7	8	9	10
$F_n\ (k = 1.3)$	1	2	3	4	6	8	10	13	17	23
$F_n\ (k = 1.5)$	1	2	3	5	7	11	17	25	38	57
Fibonacci	**1**	**2**	**3**	**5**	**8**	**13**	**21**	**34**	**55**	**89**
$F_n\ (k = 1.7)$	1	2	4	8	14	24	41	69	118	201
$F_n\ (k = 1.9)$	1	3	6	13	24	47	89	169	322	613

We can observe that Fibonacci numbers are somehow in the "centre" of the valid sequences. The spacing between the values of the marked coefficients is not either too small or too large for the Fibonacci sequence. Intuitively, this can have a convenient effect as the trade-off between the watermarking properties (capacity, transparency and robustness) is concerned. However, whether the Fibonacci sequence has some advantages to other choices of sequences with a similar behaviour (like the other ones given in Table I) or not must be checked experimentally. Section IV.B is devoted to showing that the Fibonacci numbers are a particularly good choice in terms of the trade-off between the watermarking properties.

V. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed method and to consider the applicability of the scheme in a real scenario, the album *Rust* by No, Really [18] have been used. All audio clips are sampled at 44.1 kHz with 16 bits per sample and two channels. The experiments have been performed and presented for each channel of the audio signals separately.

A. TRANSPARENCY, CAPACITY AND ROBUSTNESS

We provide imperceptibility results both as SNR and ODG where ODG = 0 means no degradation and ODG = –4 means a very annoying distortion. The SNR is provided only for comparison with other works, but ODG is a more appropriate measurement of audio distortions, since it is assumed to provide an accurate model of the subjective difference grade (SDG) results which may be obtained by a group of human listeners. The SNR results are computed using the whole (original and marked) files, whereas the

ODG results are provided using the advanced ITU-R BS.1387 standard [19] as implemented in the Opera software [20] (the average of measurements taken in frames of 1024 samples).

In addition to the ODG, we have also obtained subjective quality measurements [30, 31]. Subjective listening tests are necessary to perceptual quality assessment, since the final judgment is made by human acoustic perception. For the subjective listening tests, five participants (three men and two women) were selected with the original and the marked audio signals, and were asked to report the dissimilarities between the two signals, using a five-point subjective grade (SDG): SDG = 5 means excellent quality, SDG = 4 is good and SDG = 1 means bad. The output of the subjective tests is often an average of the quality ratings called Mean Opinion Score (MOS). The subjective experimental results (Table II) show that the perceived quality of marked signal is good (greater than 3.5 in all cases). According to these outputs, we can confirm convenient imperceptibility of the watermark in the audio signals.

Table II shows the perceptual distortion, payload and BER under the MP3 compression attack with different bit rates. Note that different values for parameters are used to achieve a different trade-off between capacity, transparency and robustness, as usual for all watermarking systems. For example, for "Go", a frame size d = 1 and a wide frequency band, the results show a high capacity of 3075 bits per second (bps), and a bit error rate (BER) equal to 0.11 after the MP3-128 attack is applied. If we increase the frame size to 3, the BER decreases to 0.03. Also, shifting the frequency band from 13–16 KHz to 12–15 KHz increases distortion and robustness.

Table II. Results of 5 mono signals

Audio File	*Time (m:sec)*	*Frame size*	*Frequency band (KHz)*	*SNR (dB)*	*MP3 Attack*		*ODG of marked*	*SDG of marked*	*Payload (bps)*
					rate	*BER*			
Beginning of the End	3:16	1	14 – 16	58.1	128	0.00	–0.95	4.0	2050
		1	14 – 16	58.1	80	0.03	–0.95	4.0	2050
		1	14 – 16	58.1	64	0.09	–0.95	4.0	2050
		1	13 – 16	55.6	128	0.00	–1.10	3.6	3075
		1	13 – 16	55.6	80	0.05	–1.10	3.6	3075
		1	15 – 16	61.6	128	0.00	–0.5	4.2	1025
		1	15 – 16	61.6	80	0.03	–0.5	4.2	1025
Do You Know Where Your …	2:31	3	14 – 16	42.9	128	0.12	–0.31	4.4	683
		3	12 – 16	36.9	128	0.13	–0.88	4.0	1366
Go	1:51	3	13 – 16	44.5	128	0.03	–0.61	4.2	1024
		1	12 – 15	35.9	128	0.11	–0.97	3.6	3075
Stop Payment	2:09	1	13 – 16	50.0	128	0.09	–0.65	4.0	3075
		1	14 – 16	52.2	128	0.11	–0.29	4.4	2050
Thousand Yard Stare	3:57	1	14 – 16	53.5	128	0.00	–0.55	4.2	2050
		1	14 – 16	53.5	80	0.09	–0.55	4.2	2050
		1	13 – 16	51.9	128	0.0	–0.84	3.8	3075

		1	13 – 16	51.9	96	0.07	–0.84	3.8	3075
		1	13 – 16	51.9	80	0.11	–0.84	3.8	3075

In principle, there is not a direct relationship between the embedding bit rate (or the payload of the method) and the detection rate (BER) after attacks, or the transparency. A discussion on the effect of the tuning parameters is provided next to illustrate the guidelines provided in Section III.A.

The embedding bit rate can be increased in two ways, namely, choosing a wider frequency band for embedding, or decreasing the frame size (*d*). In the first case, depending of the attack, it is possible to increase the embedding bit rate without a significant change in the BER. However, some attacks are very sensitive with respect to the frequency range used for embedding. In Table II, we can see that, for the file "Beginning of the End", changing the frequency band from 13–16 to 15–16 kHz reduces the capacity and increases the BER for the MP3-80 attack. As *d* is concerned, a decrease in this parameter increases the embedding bit rate but decreases the robustness (BER). This can also be observed in Table II.

As the trade-off between capacity and transparency is concerned, Table II also shows that enlarging the frequency band decreases transparency. Regarding the frame size, usually an increase in *d* (which reduces the embedding bit rate) results in lower transparency.

Table III illustrates the effect of several common attacks, provided by the Stirmark Benchmark for Audio (SMBA) v1.0 [21], on ODG and BER for the two selected audio test files. The tuning parameters were selected for each signal, here frequency band is 14–16 KHz and the frame size is equal to one yielding a capacity of 2050 bps. Then the embedding method was applied, the SMBA software was used to attack the marked files and, finally, the detection method was applied for the attacked files. The ODG in Table III is calculated between the marked and the attacked-marked files. The parameters of the attacks are defined according to the description provided in the SMBA web site [21]. For example, in AddBrumm, 1–4k shows the strength and 1–5k shows the frequency. This row reports that any value in the range 1–4k for the strength and 1–5k for the frequency can be used without any significant change in BER. It can be seen that the proposed scheme produces excellent robustness against all these attacks (BER close to zero) even if the attacks significantly distort the audio files (even for ODG lower than –3).
In addition to common attacks form Stirmark Benchmark for Audio (SMBA) v1.0 [21], we consider a well-known synchronisation technique in the time domain [28]. In [28], the 16-bit Barker code "1111100110101110" is embedded by modifying the average of a few consecutive samples. The advantage of this synchronisation marks is that the search can be performed in the time domain, without

computing any kind of transform. The effect of embedding this synchronisation marks on ODG and BER is included in Table III. Apart from [28], a more recent synchronisation technique in the time domain is presented in [32].

The desynchronisation attack affects the location of the watermark, which causes the location of samples move forward or backward. The suggested system is fragile against desynchronisation attack unless synchronisation marks are also embedded. Once the synchronisation technique of [28] (or [32]) is used, the suggested scheme is robust against desynchronisation. The robustness against desynchronisation can be further increased by combining time-domain and frequency-domain synchronisation marks, as proposed in [33].

TABLE III. Robustness test results for two selected files

Attack name	Beginning of the End			Thousand Yard Stare		
	ODG of attacked file	*parameters*	*BER*	ODG of attacked file	*parameters*	*BER*
AddBrumm	–3.3	1–4k, 1–5k	0.0	–2.3	1–4k, 1–4k	0.0
AddDynNoise	–0.8	1	0.11	–0.6	1	0.0
AddNoise	–1.9	1–30	0.01	–0.6	1–1000	0.0
AddSinus	–1.75	1–5k, 1–5k	0.0	–1.3	1–5k, 1–5k	0.0
Amplify	–0.25	60–140	0.0	–0.3	60–140	0.0
BassBoost	–3.7	0–40,0–50	0.0	–3.9	0–60,0–60	0.0
Echo	–2.6	3	0.01	–2.5	3	0.0
FFT_RealReverse	–3.6	2	0.0	–3.8	2	0.0
FFT_Stat1	–0.2	2	0.0	–0.4	2	0.0
Invert	–3.8	–	0.00	–3.7	–	0.0
LSBZero	–0.1	–	0.0	–0.2	–	0.0
RC_HighPass	–3.1	0–18k	0.0	–3.6	0–18k	0.0
RC_LowPass	–0.8	8k–20k	0.0	–0.9	8k–20k	0.0
Stat1	–0.3	–	0.0	–0.8	–	0.23
Synchronisation	–0.1	–	0.01	–0.1	–	0.0

B. TRADE-OFF USING DIFFERENT SEQUENCES

All series generated by Equation (19) can be used by the suggested system but they provide different transparency and robustness. Table IV illustrates transparency in terms of SNR and ODG, and also robustness against MP3 compression. Increasing *k* reduces the quality of marked file, since the numbers for replacing the magnitudes FFT coefficients of the audio file become more spaced. For example, according to Table I, for numbers between 1 and 25, if we use F_n $(k = 1.3)$, we have 10 elements to represent the coefficients. However, if we use F_n $(k = 1.9)$, there are just five elements to represent the same range. It is obvious that increasing *k* reduces the audio quality. On the other hand, increasing *k*, increases robustness. When *k* is larger, the differences between the elements of the sequence are larger and, thus, the noise and other modifications do not damage the watermark. For example for the sequence F_n $(k = 1.3)$, if the extracted modified magnitude for a coefficient is 7, we can not distinguish if the original value was 6 or 8 and thus, we can not always extract the correct watermark bit corresponding to this coefficient. However, for F_n $(k = 1.9)$, if the extracted magnitude is 7, it is very likely that the original value was 6. Even if it is 8, we can detect 6 and then extract the correct watermark bit. Therefore increasing *k* increases the robustness.

The popular challenge of watermarking is considering a trade-off based on the requirements. Obtaining ODG values in [–1, 0] or, in other words, [perceptible but not annoying, imperceptible] and the best BER against MP3 compression, leads us to use Fibonacci numbers. In fact, Fibonacci numbers provide the best trade-off for transparency and robustness as illustrated in Table IV. The reason for this is that Fibonnacci numbers are in the centre of the sequences generated with Equation (19) for *k* in (1, 2), producing a convenient trade-off in many different situations. It may happen that, for some specific file, a better trade-off is obtained with a different value of *k*, but the central position of the Fibonacci sequence, as shown in Table IV, leads to a convenient trade-off in many situations.

TABLE IV. Transparency and robustness for different sequences

Audio File	*Sequence*	*Frame size*	*Frequency band (kHz)*	*SNR (dB)*	*ODG of marked*	*BER of MP3 Attack*		
						MP3-128	*MP3-112*	*MP3-96*
Do You Know Where Your …	F_n $(k = 1.3)$	3	8 – 14	34.62	–0.40	0.28	0.36	0.44
	F_n $(k = 1.5)$	3	8 – 14	31.22	–0.74	0.13	0.23	0.34
	Fibonacci	3	8 – 14	30.6	–0.82	0.10	0.18	0.28
	F_n $(k = 1.7)$	3	8 – 14	28.8	–1.23	0.07	0.15	0.23
	F_n $(k = 1.9)$	3	8 – 14	25.3	–1.39	0.05	0.13	0.21
Go	F_n $(k = 1.3)$	3	10 – 14	39.8	–0.68	0.25	0.31	0.38
	F_n $(k = 1.5)$	3	10 – 14	38.43	–0.86	0.12	0.22	0.32
	Fibonacci	3	10 – 14	37.7	–0.96	0.11	0.18	0.28
	F_n $(k = 1.7)$	3	10 – 14	36.00	–1.32	0.09	0.15	0.24
	F_n $(k = 1.9)$	3	10 – 14	33.8	–1.48	0.07	0.12	0.18

C. COMPARISON

The suggested system in this article has been compared with several recent audio watermarking schemes. Each watermarking system has different properties. Because of this, it is hard to establish a fair comparison of the proposed scheme with some audio watermarking schemes. Therefore, we have chosen a few recent and relevant robust audio watermarking schemes in the literature. Table V provides a comparison between the proposed watermarking algorithm and several recent audio watermarking strategies robust against the MP3 attack. We can classify watermarking methods in two groups.

1- **LOW CAPACITY:**

These methods [2, 3, 4, 9, 16, 22, 23, 24] provide low capacity, payload about a few hundred bits per second, usually acceptable transparency and robust against some attacks. Ref. [2] proposes a very robust, very low capacity and high distortion scheme. Ref. [4] measures distortion through the mean opinion score (MOS), which is a subjective measurement, and achieves transparency between imperceptible and perceptible but not annoying, MOS = 4.7. The scheme described in [4] and the system proposed in this paper lead to high capacity and low distortion but they are not as robust as the low-capacity method described in [2]. Ref. [22] proposes a method based on cochlear delay characteristics that is robust against MPEG compression and resampling. Ref. [23] presents a very robust scheme against resampling and compression but it has a very low capacity (7–30 bps). The quality of marked signal is source-dependent. *I.e.* for some audio signals the quality of the marked signal is good and for others is significantly low. Speech applications and codecs are considered in [24]. In [24], the distortion introduced to the marked signal is slightly annoying, capacity is very low and robustness is achieved against compression attacks. Recently, Ref. [16] presents a very fast method which uses the Fourier transform. The embedding capacity is low, 64 bits per second, but the scheme is very robust against several attacks.

2- **HIGH CAPACITY**

These methods [5, 6, 10, 11, 12] provide high capacity solutions for audio watermarking. Payload is about a few thousands bps, transparency and robustness against different attacks are properties of these systems. Ref. [5], which was proposed by the authors of this paper, has a remarkable performance in the different properties, but the scheme proposed in this paper can manage the needed properties better, since there are two useful adjustable parameters. The methods [6, 10, 11, 12], also proposed by the authors of this paper, have high capacity but they are not too robust against attacks compared with the scheme proposed in this paper. In [6], only the MP3 attack was evaluated.

In fact, the most valuable achievement of the proposed scheme is robustness against very difficult attacks such as Echo, filtering and noises. Ref. [12] and the scheme suggested in this paper are robust against MP3-64, but Ref. [12] provides 512 bps, half the capacity of the proposed scheme.

This comparison proves the superiority in both capacity and imperceptibility of the suggested method with respect to other techniques in the literature, and in robustness as schemes with similar capacity/imperceptibility are concerned. This is particularly relevant, since the proposed scheme can embed much more information and, at the same time, introduces less distortion in the marked file.

TABLE V: Comparison of different watermarking algorithms

Algorithm	Capacity (bps)	Imperceptibility in SNR (dB)	Imperceptibility (ODG)
[2]	2	42.8 to 44.4	–1.66 < ODG < –1.88
[3]	4.3	29.5	Not reported
[4]	689	Not reported	Not reported
[24]	8	Not reported	–3 < ODG < –1
[16]	64	30 –45	–1< ODG
[9]	2.3	Not reported	Not reported
[22]	4–512	Not reported	–1 < ODG
[23]	7–30	Not reported	Not reported
[5]	3 k	30.55	–0.6
[6]	2 k – 6 k	Not reported	–0.6 < ODG < –1.7
[10]	11 k	30	–0.7
Proposed	683 to 3 k	35 to 61	–0.3 < ODG < –1.1

In short, the proposed scheme achieves higher capacity if we compare it to methods with similar robustness and imperceptibility, and more robustness and imperceptibility if we compare it to methods with similar capacity.

VI. CONCLUSIONS

In this article, a high-capacity transparent watermarking system for digital audio, which is robust against common audio signal processing attacks and the Stirmark Benchmark for audio, is presented. The suggested method guarantees that the maximum change of each FFT sample is less than 61% (for typical values of FFT samples) and the average error for each sample is 25%. The frame size and the selected frequency band are the two adjustable parameters of this system that determine the capacity, the perceptual distortion and the robustness trade-off of the system accurately. Furthermore, the suggested scheme is blind, since it does not need the original signal for extracting the hidden bits. The experimental results show that this method has a high capacity (700 bps to 3 kbps) without significant perceptual distortion (ODG about –1) and provides robustness against common signal processing attacks such as echo, added noise, filtering or MPEG compression (MP3) even with rates as low as 64 kbps. In addition, the proposed method clearly overcomes the robustness results of recent methods that can be compared with it in terms of capacity.

REFERENCES

[1] H. J. Kim, "Audio watermarking techniques," in Proc. Pacific Rim Workshop Digital Steganography, 2005, pp. 1–17.

[2] S. Xiang, H.J. Kim, J. Huang, "Audio watermarking robust against time-scale modification and MP3 compression," *Signal Processing*, Vol.88 n.10, pp.2372-2387, October, 2008.

[3] M. Mansour and A. Tewfik, "Data embedding in audio using time-scale modification," *IEEE Trans. Speech Audio Process.*, Vol. 13, no. 3, pp. 432–440, 2005.

[4] J. J. Garcia-Hernandez, M. Nakano-Miyatake and H. Perez-Meana, "Data hiding in audio signal using Rational Dither Modulation", *IEICE Electron. Express*, Vol. 5, No. 7, pp.217-222, 2008.

[5] M. Fallahpour, D. Megías, "High capacity audio watermarking using FFT amplitude interpolation" *IEICE Electron. Express*, Vol. 6, No. 14, pp. 1057-1063, 2009.

[6] M. Fallahpour, D. Megías, "High Capacity Method for Real-Time Audio Data Hiding Using the FFT Transform", Advances in Information Security and Its Application, Springer-Verlag. pp. 91-97, 2009.

[7] M. Fallahpour, D. Megías, "Robust high-capacity audio watermarking based on FFT amplitude modification" *IEICE Trans on Information and Systems*, Vol.E93-D, No. 01, pp.87-93, Jan. 2010.

[8] M. Fallahpour, D. Megías, "DWT–based high capacity audio watermarking", *IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences*, Vol. E93-A, No. 01, pp. 331-335, Jan. 2010.

[9] W. Li, X. Xue, "Content based localized robust audio watermarking robust against time scale modification" *IEEE Trans. Multimedia*, Vol. 8, No. 1, pp. 60-69, Feb. 2006.

[10] M. Fallahpour, D. Megías, "High capacity audio watermarking using the high frequency band of the wavelet domain" *Multimedia Tools and Applications, Springer* Vol. 52, pp. 485-498, 2011.

[11] M. Fallahpour, D. Megías. "Secure logarithmic audio watermarking scheme based on the human auditory system". Multimedia Systems. ISSN.0942-4962. DOI: 10.1007/s00530-013-0325-1, 2013. link

[12] M. Fallahpour, D. Megías, "FFT-based robust high-capacity audio watermarking" *International Journal of Innovative Computing, Information and Control*, Vol.8, No.4, April 2012.

[13] S.T. Chen, G.D. Wu, H.N. Huang, "Wavelet-domain audio watermarking scheme using optimisation-based quantisation", IET Signal Process, Vol. 4, Iss. 6, pp. 720 –727, 2010.

[14] N. K. Kalantari, M. A. Akhaee, M. Ahadi, H. Amindavar, "Robust Multiplicative Patchwork Method for Audio Watermarking", IEEE Transactions on Audio, speech, and language processing, Vol. 17, No. 6, pp. 1133–1141, August 2009.

[15] S.T. Chen, H.N. Huang, C.J. Chen, G.D. Wu, "Energy-proportion based scheme for audio watermarking", IET Signal Process., Vol. 4, Iss. 5, pp. 576–587, 2010.

[16] X. Kang, R. Yang, J. Huang, "Geometric Invariant Audio Watermarking Based on an LCM Feature", IEEE Transactions On Multimedia, Vol. 13, No. 2, pp. 181–190, April 2011.

[17] G.E.Bergum et al (eds.) *Applications of Fibonacci Numbers*, Volume 4. 1991.

[18] No, Really, "Rust". http://www.jamendo.com/en/album/7365. (Checked on July 17, 2014).

[19] T. Thiede, W. C. Treurniet, R. Bitto, C. Schmidmer, T. Sporer, J. G. Beerens, C. Colomes, M. Keyhl, G. Stoll, K. Brandenburg, and B. Feiten, "PEAQ - The ITU Standard for Objective Measurement of Perceived Audio Quality," *Journal of the AES*, vol. 48(1/2), pp. 3–29, 2000.

[20] OPTICOM OPERA software site. http://www.opticom.de/products/opera.html.

[21] Stirmark Benchmark for Audio. http://wwwiti.cs.uni-magdeburg.de/~alang/smba.php.

[22] M. Unoki, D. Hamada, "Method of digital-audio watermarking based on cochlear delay characteristics", International Journal of Innovative Computing, Information and Control Vol. 6, No. 3(B), pp. 1325–1346, March 2010.

[23] K. Kondo, K. Nakagawa, "A digital watermark for stereo audio signals using variable inter-channel delay in high-frequency bands and its evaluation", International Journal of Innovative Computing, Information and Control Vol. 6, No. 3(B), pp. 1209–1220, March 2010.

[24] A. Nishimura, "Audio data hiding that is robust with respect to aerial transmission and speech codecs", International Journal of Innovative Computing, Information and Control Vol. 6, No. 3(B), pp. 1389–1400, March 2010.

[25] R. A. Dunlap "The Golden Ratio and Fibonacci Numbers. 1997." World Sci. Pub. Co., NJ.

[26] Chen, Brian, and Gregory W. Wornell. "Quantization index modulation: A class of provably good methods for digital watermarking and information embedding." *Information Theory, IEEE Transactions on* 47.4 (2001): 1423-1443.

[27] Guang Hua, Jonathan Goh, Vrizlynn L. L. Thing, “Time-Spread Echo-Based Audio Watermarking With Optimized Imperceptibility and Robustness”, IEEE/ACM Transactions on Audio, Speech and Language Processing, Volume 23 Issue 2, Pages 227-239, February 2015.

[28] X.Y. Wang and H. Zhao. A novel synchronization invariant audio watermarking scheme based on DWT and DCT. IEEE Transactions on Signal Processing, 54(12):4835–4840, December 2006.

[29] J. Katz, Y. Lindell, “Introduction to Modern Cryptography: Principles and Protocols”, *Chapman & Hall/CRC Cryptography and Network Security Series,* CRC Press, 2007.

[30] M. Unoki, K. Imabeppu, D. Hamada, A. Haniu, and R. Miyauchi, “Embedding limitations with digital-audio watermarking method based on cochlear delay characteristics,” *Journal of Information Hiding and Multimedia Signal Processing*, vol. 2, No. 1, pp. 1-23, January 2011.

[31] S. Wang and M. Unoki, “Speech Watermarking Method based on Formant Tuning,” *IEICE Trans. INF. & SYST.*, Vol. E98-D, No. 1, pp. 29-37, Jan. 2015.

[32] D. Megías, J. Serra-Ruiz and M. Fallahpour, “Efficient self-synchronised blind audio watermarking system based on time domain and FFT amplitude modification”. *Signal Processing*, vol. 90, No. 12, pp. 3078-3092. Dec. 2010.

[33] D. Megías, "PCT patent application PCT/EP2013/074971 - Method and apparatus for embedding and extracting watermark data in an audio signal", November 28, 2013.