Audio Watermarking using Empirical Mode of Decomposition

Mr. A. A. Dhanorkar[1] Dr. S. B. Mohod[2] Prof. D. S. Chandak[3]
[1]P.G Student [2,3]Assistant Professor
[1,2,3]Department of Electrical & Electronic Engineering
[1,2,3]V.Y.W.S PRMCEM Badnera, Maharashtra India

Abstract— In this paper by using the Empirical Mode Decomposition (EMD) developed an algorithm for advance adaptive audio watermarking. The audio signal is classified into frames and each one is decomposed adaptively, by EMD, into intrinsic oscillatory components which is as called Intrinsic Mode Functions (IMFs). The codes are embedded into the extrema of last splitted IMF because last IMFs are most stable frequency component under various attacks and so protects quality of the host signal. A low frequency mode stable under different attacks and preserving audio perceptual quality of the host signal. The data embedding rate of the proposed algorithm is 46.8–51.3b/s. The comparison analysis shows that our method has good performance than watermarking schemes reported recently.

Key words: MATLAB, Steganography and Image and Audio Watermarking

I. Introduction

Earlier the by using invention of cryptography and steganography, it was stimulate to transfer secure information and, therefore need, to achieve secure communication environment we used Some of the techniques employed in early days are writing with an invisible ink, drawing a standard painting with some small modifications, combining two images to create a new image, shaving the head of the messenger in the form of a message, tattooing the message on the scalp and so on [15].

Normally an application is developed by a person or a small group of people and used by many. Hackers are the people who tend to change the original application by modifying it or use the same application to make profits without giving credit to the owner. It is obvious that hackers are more in number compared to those who create. Hence, protecting an application should have the significant priority. Protection techniques have to be efficient, robust and unique to restrict malicious users. The development of technology has increased the scope of steganography and at the same time decreased its efficiency since the medium is relatively insecure. This lead to the development of the new but related technology called „Watermarking‟. Some of the applications of watermarking include ownership protection, proof for authentication, air traffic monitoring, medical applications etc. [1] [5] [12][16]. Watermarking for audio signal has greater importance because the music industry is one of the leading businesses in the world [9].

II. Steganography and Watermarking

A. Steganography

Steganography is evolved from the ancient technique known as the „Cryptography‟. Cryptography protects the contents of the message [15]. On the other hand, steganography is a technique to send information by writing on the cover object invisibly. Steganography comes from the Greek word that means covered writing (stego = covered and graphy = writing) [3]. Here the authorized party is only aware of the existence of the hidden message. An ideal steganographic technique conceals large amount of information ensuring that the modified object is not visually or audibly distinguishable from the original object.The steganography technique needs a cover object and message that is to be transported. It also requires a stego key to recover the embedded message. Users having the stego key can only access the secret message. Another important requirement for efficient steganographic techniques is that, the cover object is modified in a way that the quality is not lost after embedding the message.

B. Watermarking

Watermarking is a technique through which the secure information is carried without degrading the quality of the original signal. The technique consists of two blocks:

- Embedding block
- Extraction block

The system has an embedded key as in case of a steganography. The key is used to increase security, which does not allow any unauthorized users to manipulate or extract data. The embedded object is known as watermark, the watermark embedding medium is termed as the original signal or cover object and the modified object is termed as embedded signal or watermarked data [8].

The embedding block, shown in Figure 1 consists of watermark, original signal (or cover object), and watermarking key as the inputs (creates the embedded signal or watermarked data) [15]. Whereas, the inputs for the extraction block is embedded object, key and sometimes watermark as illustrated in Figure 2 [7].

The watermarking technique that does not use the watermark during extraction process is termed as "blind watermarking". Blind watermarking is superior over other watermarking involving watermark for extraction as watermarked signal and key are sufficient to find the embedded secret information.

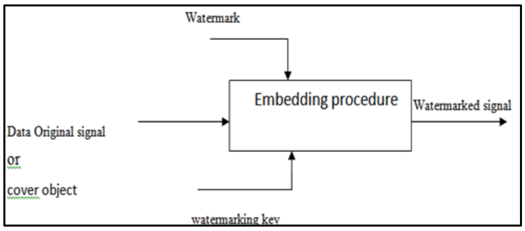

Fig. 1: Digital watermarking embedding.

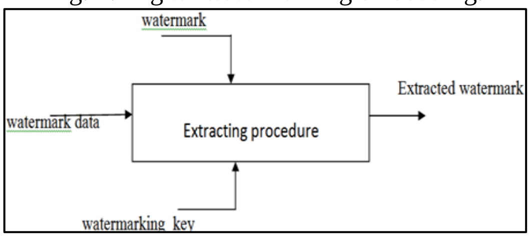

Fig. 2: Digital watermarking extraction

III. MODERN AUDIO WATERMARKING METHODS

The principles of commercial watermarking systems are secret and can only be deduced from the promotional material. In some cases, the published results of tests on specific commercial systems have put into the public significant insights into the operating principles of those systems. Very brief summary is written here. The principles of watermarking can be divided into three categories, according to domains in which the transformation is performed. Some of time-domain method.

A. *Time Domain Method*

1) *Adding Noise*

The watermark may be added to the audio stream simply as a low-level signal. The most common way of implementing such a system is by using spread spectrum modulation; however, other modulation systems are possible

2) *Adding Echoes*

It has been long established that the human auditory system is insensitive to 'echoes'. This was firs established by Mr. Haas in early 1950´s [6]. On a short time-scale, around 20ms, a repetition of the signal is not perceptible with the amplitude somewhat lower than the original signal.

3) *Modifying phase*

An Audio is said to be phase insensitive, in that the Human Auditory System (HAS) is not aware of either absolute or relative phase of the signal. That is why phase encoding has been introduced for purpose of watermarking. This method is included under "time-domain methods", but it requires the use of Fourier transformation for its implementation.

B. *Frequency-Domain Methods*

1) *Adding modulated carriers*

At least one modern watermarking system uses the principle of adding modulated carrier frequencies to an audio signal. To be successful, such a system needs to have a system for calculating and making use of psycho-acoustic audibility thresholds. On the other hand, such a system will not be probably very secure.

2) *Adding noise in a transform domain*

The watermarked data can simply be added to the coding coefficients as in any other additive system working in a transform domain. This method has also an alternative in modifying of coding coefficients by biased error distribution. Both are used in MPEG, AC3, etc. In all of mentioned methods, account may be taken of the HAS sensitivities (temporal of frequency masking techniques have to considered) to embed the watermark signal in the balance between added watermark energy and perceptibility. Any practicable watermarking system represents a workable compromise between these requirements.

C. *Spread-Spectrum Modulation Method*

This is one of well-known watermarking techniques. The method is based on spreading the watermarked signal over the entire audible frequency spectrum so that it approximates white noise, at power level as to be inaudible (below the ambient noise) [4]. A pseudorandom sequence called chip is used to modulate a carrier wave which creates the signal spread watermark code, as shown in Figure. The message bit rate is always less then chip rate. Generally, the watermark extractor may have two purposes: deciding whether the tested signal contains a watermark and extracting a message that watermark may carry.

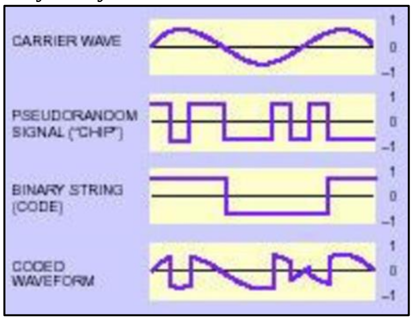

Fig. 3: Tested signal

IV. PROPOSED WATERMARKING ALGORITHM

The idea of the proposed watermarking method is to hide into the original audio signal a watermark together with a Synchronized Code (SC) in the time domain. The input signal is first segmented into frames and EMD is conducted on every frame to extract the associated IMFs Then a binary data sequence consisted of SCs and informative watermark bits is embedded in the extrema of a set of consecutive last-IMFs. A bit (0 or 1) is inserted per extrema. Since the number of IMFs and then their number of extrema depend on the amount of data of each frame, the number of bits to be embedded varies from last-IMF of one frame to the following. Watermark and SCs are not all embedded in extrema of last IMF of only one frame. In general the number of extrema per last-IMF (one frame) is very small compared to length of the binary sequence to be embedded.

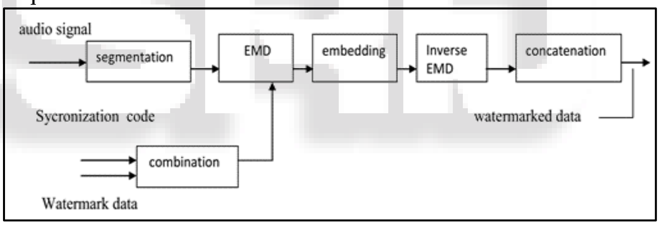

Fig. 4: Watermark embedding using EMD

A. *Watermark Embedding*

Before embedding, SCs are combined with watermark bits to form a binary sequence. Watermark embedding can be detailed as follows 2). Basics of our watermark embedding are shown in Fig. 3 and detailed has follows:

- Step 1: Split original audio signal into frames.
- Step 2: Decompose each frame into IMFs.
- Step 3: Embed times the binary sequence into extrema of the last IMF by QIM.

$$e_i^* = \begin{cases} \left\lfloor e_i^*/s \right\rfloor .s + sgn\left(\frac{3s}{4}\right) & if\ m_i = 1 \\ \left\lfloor e_i^*/s \right\rfloor .s + sgn\left(\frac{s}{4}\right) & if\ m_i = 0 \end{cases}$$

Where m_i and e_i^* are the extrema of the host audio signal and the watermarked signal respectively. Sign function is equal to "+" if is a maxima, and "" if it is a minima. Denotes the floor function, and S denotes the embedding strength chosen to maintain the inaudibility constraint.

- Step 4: Reconstruct the frame using modified and concatenate the watermarked frames to retrieve the watermarked signal.

V. MATLAB

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include

- Math and computation
- Algorithm development
- Data acquisition
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including graphical user interface building.

MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar non interactive language such as C or FORTRAN. The name MATLAB stands for matrix laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects. Today, MATLAB engines incorporate the LAPACK and BLAS libraries, embedding the state of the art in software for matrix computation.

MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. In industry, MATLAB is the tool of choice for high- productivity research, development, and analysis. MATLAB features a family of add-on application-specific solutions called toolboxes. Very important to most users of MATLAB, toolboxes allow you to learn and apply specialized technology. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems. Areas in which toolboxes are available include signal processing, control systems, neural networks, fuzzy logic, wavelets, simulation, and many others.

A. *The MATLAB System*

The MATLAB system consists of five main parts:

B. *Development Environment*

This is the set of tools and facilities that help you use MATLAB functions and files. Many of these tools are graphical user interfaces. It includes the MATLAB desktop and Command Window, a command history, an editor and debugger, and browsers for viewing help, the workspace, files, and the search path.

C. *The MATLAB Mathematical Function:*

This is a vast collection of computational algorithms ranging from elementary functions like sum, sine, cosine, and complex arithmetic, to more sophisticated functions like matrixinverse, matrix eigen values, Bessel functions, and fast Fourier transforms.

D. *The MATLAB Language*

This is a high-level matrix/array language with control flow statements, functions, data structures, input/output, and object-oriented programming features. It allows both "programming in the small" to rapidly create quick and dirty throw-away programs, and "programming in the large" to create complete large and complex application programs.

E. *Graphics*

MATLAB has extensive facilities for displaying vectors and matrices as graphs, as well as annotating and printing these graphs. It includes high-level functions for two-dimensional and three-dimensional data visualization, image processing, animation, and presentation graphics. It also includes low-level functions that allow you to fully customize the appearance of graphics as well as to build complete graphical user interfaces on your MATLAB applications.

F. *The MATLAB Application Program Interface (API)*

This is a library that allows you to write C and Fortran programs that interact with MATLAB. It includes facilities for calling routines from MATLAB (dynamic linking), calling MATLAB as a computational engine, and for reading and writing MAT-files.

1) *Matlab Working Environment*

a) MATLAB Desktop

Matlab Desktop is the main Matlab application window. The desktop contains five sub windows, the command window, the workspace browser, the current directory window, the command history window, and one or more figure windows, which are shown only when the user displays a graphic. The command window is where the user types MATLAB commands and expressions at the prompt (>>) and where the output of those commands is displayed. MATLAB defines the workspace as the set of variables that the user creates in a work session. The workspace browser shows these variables and some information about them. Double clicking on a variable in the workspace browser launches the Array Editor, which can be used to obtain information and income instances edit certain properties of the variable. The current Directory tab above the workspace tab shows the contents of the current directory, whose path is shown in the current directory window. For example, in the windows operating system the path might be as follows: C:\MATLAB\Work, indicating that directory “work” is a subdirectory of the main directory “MATLAB”; WHICH IS INSTALLED IN DRIVE C. clicking on the arrow in the current directory window shows a list of recently used paths. Clicking on the button to the right of the window allows the user to change the current directory.

MATLAB uses a search path to find M-files and other MATLAB related files, which are organize in directories in the computer file system. Any file run in MATLAB must reside in the current directory or in a directory that is on search path. By default, the files supplied with MATLAB and math works toolboxes are included in the search path. The easiest way to see which directories are on the search path. The easiest way to see which directories are soon the search path, or to add or modify a search path, is to select set path from the File menu the desktop, and then use the set path dialog box. It is good practice to add any commonly used directories to the search path to avoid repeatedly having the change the current directory.

The Command History Window contains a record of the commands a user has entered in the command window, including both current and previous MATLAB sessions. Previously entered MATLAB commands can be selected and re-executed from the command history window by right clicking on a command or sequence of commands. This action launches a menu from which to select various options in addition to executing the commands. This is useful to select various options in addition to executing the commands. This is a useful feature when experimenting with various commands in a work session.

b) MATLAB Editor to create M-Files

The MATLAB editor is both a text editor specialized for creating M-files and a graphical MATLAB debugger. The editor can appear in a window by itself, or it can be a sub window in the desktop. M-files are denoted by the extension .m, as in pixel up m The MATLAB editor window has numerous pull-down menus for tasks such as saving, viewing, and debugging files. Because it performs some simple checks and also uses color to differentiate between various elements of code, this text editor is recommended as the tool of choice for writing and editing M- functions. To open the editor, type edit at the prompt opens the M-file file name. m in an editor window, ready for editing. As noted earlier, the file must be in the current directory, or in a directory in the search path.

VI. Simulation Results

A. *Effectiveness of Embedded watermark*

To show the effectiveness of our scheme, simulations are performed on audio signals including pop, jazz, rock and classic sampled at 44.1 kHz. The embedded watermark, W, is a binary logo image of size bits (Fig. 5.1). We convert this 2D binary M*N=43*48=1632 image into 1D sequence in order to embed it into the audio signal. The C used is a 16 bit Barker sequence 1111100110101110. Each audio signal is divided into frames of size 64 samples and the threshold is set to 4. The value is fixed to 0.98. These parameters have been chosen to have a good compromise between imperceptibility of the watermarked signal, payload and robustness. Fig. Shows a portion of the pop signal and its watermarked version. This figure shows that the watermarked signal is visually indistinguishable from the original one.

Perceptual quality assessment can be performed using subjective listening tests by human acoustic perception or using objective evaluation tests by measuring the SNR and Objective Difference Grade (ODG). In this work we use the second approach. ODG and SNR values of the four watermarked signals are reported in Table. The SNR values are above 20 dB showing the good choice of value and confirming to IFPI standard. All ODG values of the watermarked audio signals are between and 0 which demonstrates their good quality.

B. *Robustness Test*

To assess the robustness of our approach, different attacks are performed

1) Noise

White Gaussian Noise (WGN) is added to the watermarked signal until the resulting signal has an SNR of 20 dB.

2) Filtering

Filter the watermarked audio signal using Wiener filter.

3) Cropping

Segments of 512 samples are removed from the watermarked signal at thirteen positions and subsequently replaced by segments of the watermarked signal contaminated with WGN

4) Resampling

The watermarked signal, originally sampled at 44.1 kHz, is re-sampled at 22.05 Hz and restored back by sampling again at 44.1 kHz.

5) Compression

(64 kb/s and 32 kb/s)UsingMP3, the watermarked signal is compressed and then decompressed.

6) Requantization

The watermarked signal is re-quantized down to 8 bits/ sample and then back to 16 bits/sample. Table II shows the extracted watermarks with the associated and values for different attacks on pop audio signal. values are all above 0.9482 and most values are all below 3%. The extracted watermark are visually similar to the original watermark. These results shows the robustness of watermarking method for pop audio signal. Even in the case of WGN attack with SNR of 20 dB, our approach does not detects any error. This is mainly due to the insertion of the watermark into extrema. In fact low frequency sub band has high robustness against noise addition [3],[4].This is robustness is due to the fact hat even the perceptual characteristics of individual audio files vary, the EMD decomposition adapts to each one. Table IV shows comparison results in terms of payload and robustness to P3 compression attack of our method to nine recent watermarking schemes.

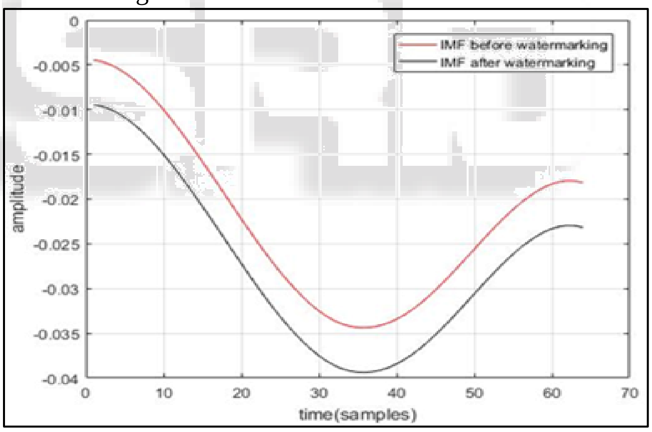

Fig. 5: IMF behavior before and after watermarking.

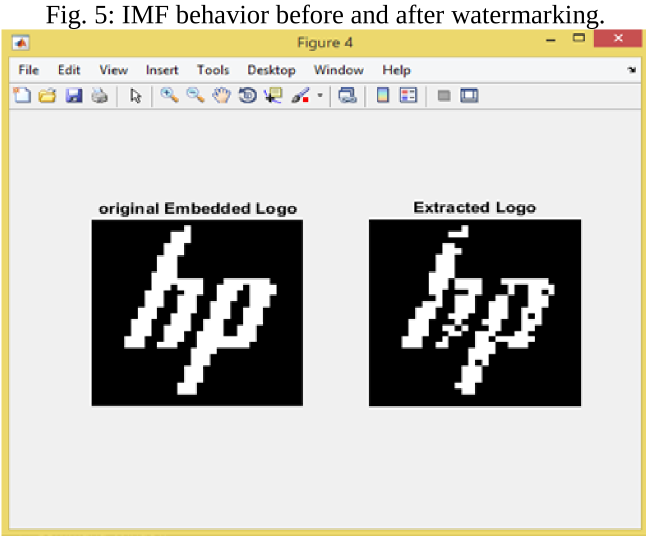

Fig. 6: Resampling attack on signal

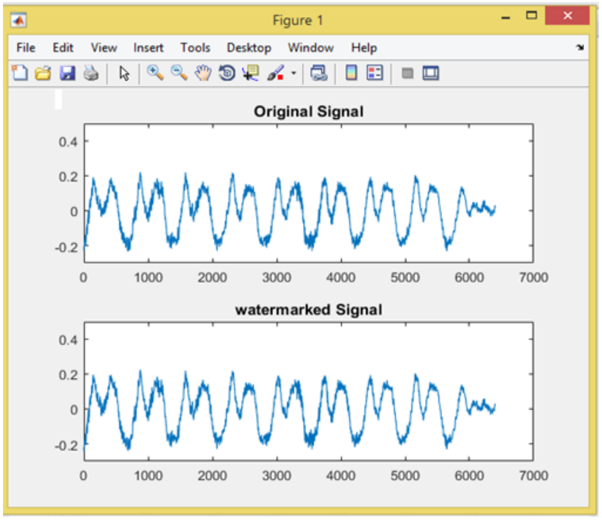

Fig. 7: Original and watermarked signal

VII. CONCLUSION

In this paper the proposed approach of EMD for anew adaptive watermarking .By a very low frequency mode (last IMF) watermarked is embedded, thus achieving better performance against various attacks. Watermark is associated with synchronization codes and thus the synchronized watermark has the ability to resist shifting and cropping. Based on QIM Data bits of the synchronized watermark are embedded in the at extrema of the last IMF of the audio signal .Extensive simulations over different audio signals indicate that the proposed watermarking.

VIII. FUTURE SCOPE

This method can be effectively used in sending hidden text messages over audio signals. With the increasing use of social media applications that allow us to send voice messages, our technique can be collaborated with such applications so as to send secret text messages within the voice message in real time

REFERENCES

[1] I. J. Cox and M. L. Miller, "The first 50 years of electronic watermarking," J. Appl. Signal Process., vol. 2, pp. 126–132, 2002.

[2] L. Boney, A. H. Tewfik, and K. N. Hamdy, "Digital watermarks for audio signals," in Proc. 3rd IEEE Int. Conf. Multimedia Computing and Systems, 1996, pp. 473–480.

[3] S. Wu, J. Huang, D. Huang, and Y. Q. Shi, "Efficiently self-synchronized audio watermarking for assured audio data transmission," IEEE Trans. Broadcasting, vol. 51, no. 1, pp.69–76, Mar. 2005.

[4] V. Bhat, K. I. Sengupta, and A. Das, "An adaptive audio watermarking based on the singular value decomposition in the wavelet domain," Digital Signal Process., vol. 2010, no. 20, pp. 1547–1558, 2010.

[5] D. Kiroveski and S. Malvar, "Robust spread-spectrum audio watermarking," in Proc. ICASSP, 2001, pp. 1345–1348.

[6] N. E. Huang et al., "The empirical mode decomposition and Hilbert spectrum for nonlinear and non-stationary time series analysis," Proc. R. Soc., vol. 454, no. 1971, pp. 903–995, 1998.

[7] D.Kiroveski and H. S. Malvar, "Spread-spectrum watermarking of audio signals," IEEE Trans. Signal Process., Vol. 51, No. 4, Pp.1020–1033, Apr. 2003

[8] K. Khaldi and A. O. Boudraa, "On signals compression by EMD," Electron. Lett, vol. 48,no. 21, pp. 1329–1331, 2012.

[9] K. Khaldi, M. T.-H. Alouane, and A. O. Boudraa, "Voiced speech enhancement based on adaptive filtering of selected intrinsic mode functions," J. Adv. in Adapt. Data Anal., vol. 2, no. 1, pp. 65–80, 2010.

[10] L. Wang, S. Emmanuel, and M. S. Kankanhalli, "EMD and psychoacoustic model based watermarking for audio," in Proc. IEEE ICME, 2010, pp. 1427–1432.

[11] A. N. K. Zaman, K. M. I. Khalilullah, Md. W. Islam, and Md. K. I. Molla, "A robust digital audio watermarking algorithm using empirical mode decomposition," in Proc. IEEE CCECE, 2010, pp. 1–4.

[12] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "A secure, robust watermark for multimedia," LNCS, vol. 1174, pp. 185–206, 1996.

[13] B. Chen and G. W. Wornell, "Quantization index modulation methods for digital watermarking and information embedding of multimedia," J. VLSI Signal Process. Syst., vol. 27, pp. 7–33, 2001.

[14] W.-N. Lie and L.-C. Chang, "Robust and high-quality time-domain audio watermarking based on low frequency amplitude modification," IEEE Trans. Multimedia, vol. 8, no. 1, pp. 46–59, Feb. 2006.

[15] I.-K. Yeo and H. J. Kim, "Modified patchwork algorithm: A novel audio watermarking scheme," IEEE Trans. Speech Audio Process., vol.11, no. 4, pp. 381–386, Jul. 2003.

[16] D. Kiroveski and H. S. Malvar, "Spread-spectrum watermarking of audio signals," IEEE Trans. Signal Process., vol. 51, no. 4, pp.020–1033, Apr. 2003.

