Reversible watermarking schemes for audio restoration and echo cancellation

M. Alejandra Menéndez Ortiz, Claudia Feregrino-Uribe, José J. García-Hernández

Technical Report No. CCC-15-007
December 2015

Luis Enrique Erro 1
Santa María Tonantzintla
72840, Puebla, México.

Reversible watermarking schemes for audio restoration and echo cancellation

M. Alejandra Menéndez Ortiz[1], Claudia Feregrino Uribe[1], J. Juan García Hernández[2]

1. Coordinación de Ciencias Computacionales, INAOE
Luis Enrique Erro #1, Sta. Ma. Tonantzintla, Puebla, 72840, México
2. Laboratorio de Tecnologías de Información, CINVESTAV, Unidad Tamaulipas
Parque Científico y Tecnológico TECNOTAM – Km. 5.5 carretera Cd. Victoria-Soto La Marina,
Cd. Victoria, Tamps., 87130, México
E-mail: {m.menendez,cferegrino}@ccc.inaoep.mx,jjuan@tamps.cinvestav.mx

Abstract. Robust reversible watermarking schemes (RWS) allow the reconstruction of a host signal and the extraction of a watermark if no attacks occur, but in the presence of attacks they either: extract the watermark, or reconstruct the host signal. This research focuses on a robust RWS for audio that can both extract the watermark and reconstruct the host audio even when attacks occur. This technical report introduces some efforts that have been done for the construction of such a robust reversible watermarking scheme. An echo cancellation strategy and a self-recovery scheme for audio signals are detailed, some results are presented, the limitations of the schemes, as well as the future work of the investigation are analysed.

Keywords: Reversible watermarking scheme, self-recovery, signal reconstruction, echo-addition

1 Introduction

Digital watermarking schemes insert a secret watermark into a host signal in such a way that it is imperceptible for a human observer but can be recovered using an extraction algorithm. These schemes can be the solution for specific application scenarios, like authentication, copyright protection, fingerprinting, and the like. Nonetheless, watermark insertion produces irreversible modifications. In application scenarios where sensitive imagery is treated (such as deep space exploration, military investigation and recognition, and medical diagnosis [1]), these irreversible modifications cannot be acceptable. *Reversible* watermarking techniques, also known as *invertible* or *lossless* [1] can insert a secret watermark within a carrier signal and later the modifications suffered during insertion can be reversed to obtain the host signal. However, reversible watermarking schemes (RWS) can only reconstruct the host signal if the watermarked version does not suffer attacks. In presence of attacks the reversibility of the scheme is lost, *i.e.* these schemes are fragile by nature.

Robust RWS have been developed to compensate the fragility of reversible watermarking techniques, focusing on two aspects 1) robustness of the watermark and 2) robustness of the signal. Works like those designed by Honsinger *et al.* [2] and De Vleeschouwer *et al.* [3] can reconstruct the original host signal and the secret watermark if no attacks occur; in case of attacks these schemes can extract the secret watermark, but the host signal cannot be recovered. Another type of robust RWS, known in the literature as self-recovery schemes, are the ones presented by Zhang *et al.* [4,5] and Bravo-Solorio *et al.* [6]. In this type of robust RWS, algorithms are able to reconstruct the host image regardless of attacks but they cannot insert a secret watermark. The embedding capacity of these schemes is used for embedding control data necessary to compensate attacks, but no space is left for secret information. Recently some efforts have been made to construct reversible watermarking schemes that can recover both the host signal and the watermarks after the occurrence of attacks [7], but they focus on images and just incipient solutions are proposed.

The aims of this research are directed towards the design of a reversible watermarking scheme with watermark and signal robustness. Although most reversible watermarking schemes have been designed for images, in this work we focus on the design of a scheme for audio signals, because there are applications for audio that require reversible watermarking schemes with watermark and host restoration capabilities. Besides, in digital watermarking most of the research has been on images, rarely exploring solutions for audio, which leaves a broader path for contributions in audio schemes.

However, the design of a scheme with such characteristics is an ambitious task. This scheme should be able to insert data into the signals, the embedded watermarks must resist attacks, and the degradation of the signal after embedding must be lower than a threshold (given by the application). In addition, there is the restriction that the host signals must be perfectly reconstructed in order to be considered a reversible scheme. Therefore, there must be a trade-off between the four properties, namely embedding capacity, robustness, imperceptibility, and reversibility.

In order to construct a reversible watermarking scheme with watermark and signal robustness for audio signals, it is necessary to design an echo-cancellation strategy for its use in application scenarios where additive echo is a possible attack to be considered. Also, it is required to design a self-recovery scheme for audio, that is part of the robust reversible watermarking scheme. This technical report presents some efforts towards the construction of such an echo-cancellation strategy and the design of a self-recovery scheme for audio signals. Section 2 introduces the echo-cancellation strategy developed and presents some results obtained. Section 3 presents the self-recovery scheme designed as part of the robust reversible watermarking scheme being investigated, and the limitations of the scheme are also analysed. The final remarks of this document are stated in Section 4.

2 Echo-cancellation for robust reversible watermarking

Digital watermarking schemes insert a secret watermark into a host signal in such a way that it is imperceptible for a human observer but can be recovered using an extraction algorithm. However, watermark insertion produces irreversible modifications and there are applications where these modifications cannot be acceptable. Such applications are those where sensitive imagery is treated, namely deep space exploration, military investigation and recognition, and medical diagnosis [1], to mention some. *Reversible* watermarking can insert a watermark within a carrier signal and reverse the modifications to obtain the host signal, as long as this watermarked signal is not modified. If the watermarked signal is modified, the embedded watermark cannot be extracted, and the host signal cannot be restored. Because of this characteristic, reversible watermarking schemes (RWS) are fragile.

Robust RWS have been developed to compensate the fragility, and are able to extract the embedded watermark when attacks occur. In the no-attack scenario, these schemes behave like other RWS, *i.e.* the watermark is extracted and the host signal is restored. However, when attacks occur RWS are incapable of restoring the host signal. Other type of watermarking schemes, known in the literature as self-recovery [8], are capable of restoring the signals after attacks. However, self-recovery schemes are incapable of inserting useful payload into the signals, *i.e.* all their embedding capacity is used to insert control data that allows the restoration of the signals after attacks.

A natural desire is to have a RWS that is capable of restoring the signal, and extracting the watermark, *i.e.* a RWS with watermark and signal robustness. Note that in the literature robust RWS refer to reversible watermarking schemes that are only capable of extracting the watermark after attacks.

The work reported in [7] proposes a framework that allows the restoration of host images, and the extraction of the watermarks after a content-replacement attack. This framework uses a fragile reversible stage, and a self-recovery stage to provide the robustness against attacks. The framework works under the assumption that the self-recovery stage is capable of obtaining a perfect restoration of

the image. Current self-recovery schemes that achieve perfect restoration [6,4,5] only deal with content replacement attacks, and are designed considering properties of images, such as their dimensionality.

The objective of this work is the construction of an echo-cancellation process that can be used instead of the self-recovery stage in the framework from [7]. Because the two stages (fragile reversible watermarking, and self-recovery) are independent, this document only addressed the echo-cancellation process.

Many echo cancellation techniques exist [9,10,11,12,13,14,15]; however their application scenario is different from the robust RWS scenario presented here. Existing echo cancellation techniques are designed for teleconferencing scenarios, where a far-end speech is transmitted through a speaker, a distorted (echoed) version of the far-end speech along a near-end speech is received in a microphone, and an echo canceler reduces the far-end echoes with the far-end speech as a reference. On the other hand, there is the robust RWS scenario presented in Fig. 2.1, where an embedding process inserts a secret message (m) into a host signal (x); then an echo is added to the watermarked signal (y) producing a watermarked signal with echo ($\hat{y}$). An echo cancellation method is applied in order to remove the echo and the extraction process recovers a message (m), and restores the host signal (x).

But, as it can be seen from Fig. 2.1, the echo cancellation method does not receive any additional information or any reference signal apart from the signal with additive echo ($\hat{y}$), so the information to restore the signal must be self-contained or the echo cancellation strategy must be designed to compensate modifications, by taking information only from the attacked signal. Therefore, current echo cancellation techniques cannot be used for the robust RWS scenario, and it is clear that a new strategy for echo-cancellation has to be proposed.

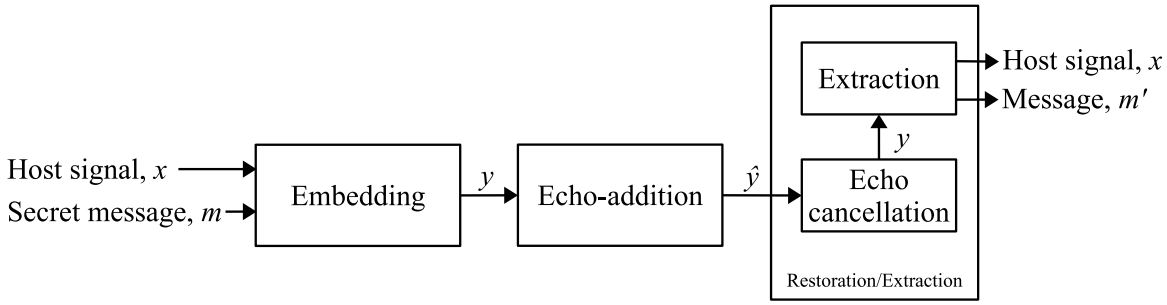

Fig. 2.1: Echo-addition in a RWS with watermark and signal robustness.

The echo-cancellation analyzed in this work is a proof of concept to determine whether such a process can be designed and used in the robust RWS scenario presented. Because of this, the most simple model of additive echo is analyzed; when a suitable strategy for the simplest model is obtained, then the solution can be extended to complex echo models more likely to occur in real-life applications.

2.1 Echo-addition attack

When a sound is produced in a room, for example, the waves reflect on the floor, walls and other objects in the room; humans perceive these reflected waves as echoes. When the reflected wave arrives tenths of milliseconds (ms) after the direct sound, it is heard as a distinct echo [16]. The formula of a signal with an additive echo [17] in the time domain is given by

$$\hat{y}[n] = y[n] + \alpha y[n - \tau], \tag{1}$$

where $\hat{y}[n]$ is the signal with the additive echo, $y[n]$ is the original signal, α is the amplitude factor of the echo and τ is the delay of the echo. In order to cancel the echo, it is necessary to identify the amplitude factor (α) and the delay (τ). The next subsections describe the strategies to propose an echo cancellation process.

Finding the location of the echo In order to locate the position of the echo, *i.e.* the delay (τ), the cepstrum is employed; Fig. 2.2 depicts an audio with additive echo and its cepstrum representation. For discrete time signals, the *cepstrum* is the inverse Fourier transform of the logarithm of the Fourier transform of the signal given by:

$$c[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \ln X[\omega] e^{i\omega n} \mathrm{d}\omega, \tag{2}$$

where $X[\omega]$ is the Fourier transform [18] of the signal, given by:

$$X[\omega] = \sum_{n=-\infty}^{\infty} x[n] e^{-i\omega n}, \tag{3}$$

and $x[n]$ is the input signal. The cepstrum can be used to find periodicity in an audio signal and can be used to find the echo in an audio. Because the cepstrum is symmetric, half of the signal can be ignored and only the first half is necessary to find the position of the echo.

Because of the characteristics of the human auditory system, an echo that occurs less than 60 ms after the beginning of a sound is not audible [19], and those samples can be discarded. The elimination removes initial peaks that correspond to the spectral contribution of the signal without echo. The highest peak in the cepstrum indicates the occurrence of the echo and, therefore, the value of τ.

Determining the amplitude of the echo The cepstrum of a signal with an echo is also used to determine the amplitude α of the echo added. The following development [20,21,17] illustrates the mathematical representation of a signal with an echo in the cepstrum domain. The Fourier transform of an additive echo (eq. 1) is

$$\hat{Y}[\omega] = Y[\omega](1 + \alpha \cdot e^{-i\omega\tau}), \tag{4}$$

where $Y[\omega]$ is the Fourier transform of the signal $y[n]$ and $(1 + \alpha \cdot e^{-i\omega\tau})$ is the spectral contribution of the echo. By applying the logarithmic function to eq. 4, it becomes

$$\ln(\hat{Y}[\omega]) = \ln(Y[\omega]) + \ln(1 + \alpha \cdot e^{-i\omega\tau}). \tag{5}$$

Using the log series expansion,

$$\ln(1 + x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \cdots, \text{ where } -1 < x \leq 1, \tag{6}$$

equation 5 can be expanded as:

$$\begin{aligned} \ln(\hat{Y}[\omega]) = &\ln(Y[\omega]) + \alpha \cdot e^{-i\omega\tau} \\ &- \frac{1}{2}\alpha^2 \cdot e^{-i2\omega\tau} + \frac{1}{3}\alpha^3 \cdot e^{-i3\omega\tau} - \cdots \end{aligned} \tag{7}$$

Finally, to obtain the cepstrum, the inverse Fourier transform is applied to eq. 7:

$$\frac{1}{2\pi}\int_{-\infty}^{\infty}\ln(\hat{Y}[\omega])e^{i\omega n}\mathrm{d}\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty}\ln(Y[\omega])e^{i\omega n}\mathrm{d}\omega + \alpha\delta[n-\tau] - \frac{\alpha^2}{2}\delta[n-2\tau] + \cdots \tag{8}$$

The left side of eq. 8 is known because the cepstrum of the signal with an echo can be calculated. However, an analytical determination of α is hard to find because, in the robust reversible watermarking scenario, the echo cancellation process does not possess the audio without the echo. Therefore, the first term on the right side of eq. 8 cannot be calculated. To simplify the notation, be

$$C[n] = \frac{1}{2\pi}\int_{-\infty}^{\infty}\ln(\hat{Y}[\omega])e^{i\omega n}\mathrm{d}\omega \text{ , and}$$

$$Z[n] = \frac{1}{2\pi}\int_{-\infty}^{\infty}\ln(Y[\omega])e^{i\omega n}\mathrm{d}\omega \text{ ,}$$

then eq. 8 can be re-written as:

$$C[n] = Z[n] + \alpha\delta[n-\tau] - \frac{\alpha^2}{2}\delta[n-2\tau] + \cdots \tag{9}$$

Given that α cannot be found analytically, a strategy to approximate its value had to be proposed. This strategy is based on the assumption that $Z[n]$ can be approximated from $C[n]$. If the approximation of $Z[n]$ is close to the real one, then it can be subtracted from $C[n]$, the peaks of the cepstrum can be isolated and from them, α can be obtained.

In order to estimate the cepstrum of the audio without echo, each of the peaks in the cepstrum are selected and a window of samples around the peak is constructed. The value of the peak is discarded and a new value is calculated by interpolation from the samples around. Fig. 2.2 presents the cepstrum of an audio with additive echo (blue line) and and the cepstrum with the interpolated peaks (green line). The window of samples is constructed selecting N samples around the peak (where N is an arbitrary number), a new value is interpolated from that window, then it is substituted by the peak and this process is repeated for all the peaks in the cepstrum.

Be $Z_{\text{approx}}[n]$ the interpolated version of the cepstrum, which is also an approximation of $Z[n]$. Because $Z_{\text{approx}}[n] \cong Z[n]$, the δ peaks from eq. 9 can be isolated as:

$$C[n] - Z_{\text{approx}}[n] = \alpha\delta[n-\tau] - \frac{\alpha^2}{2}\delta[n-2\tau] + \cdots \tag{10}$$

Be $C'[n] = C[n] - Z_{\text{approx}}[n]$ to maintain a simplified notation. If $C'[n]$ is evaluated for $n = \tau, 2\tau, 3\tau, \cdots$, eq. 10 can be rearranged as:

$$\alpha - \frac{\alpha^2}{2} + \cdots - \frac{\alpha^m}{m} = C'[\tau] + C'[2\tau] + \cdots + C'[m\tau], \tag{11}$$

where m is the polynomial order where eq. 11 is truncated. Because the values of $C'[n]$ are known constants, $k = C'[\tau] + C'[2\tau] + \cdots + C'[m\tau]$ and eq. 11 can be expressed as:

$$\alpha - \frac{\alpha^2}{2} + \cdots - \frac{\alpha^m}{m} - k = 0 \tag{12}$$

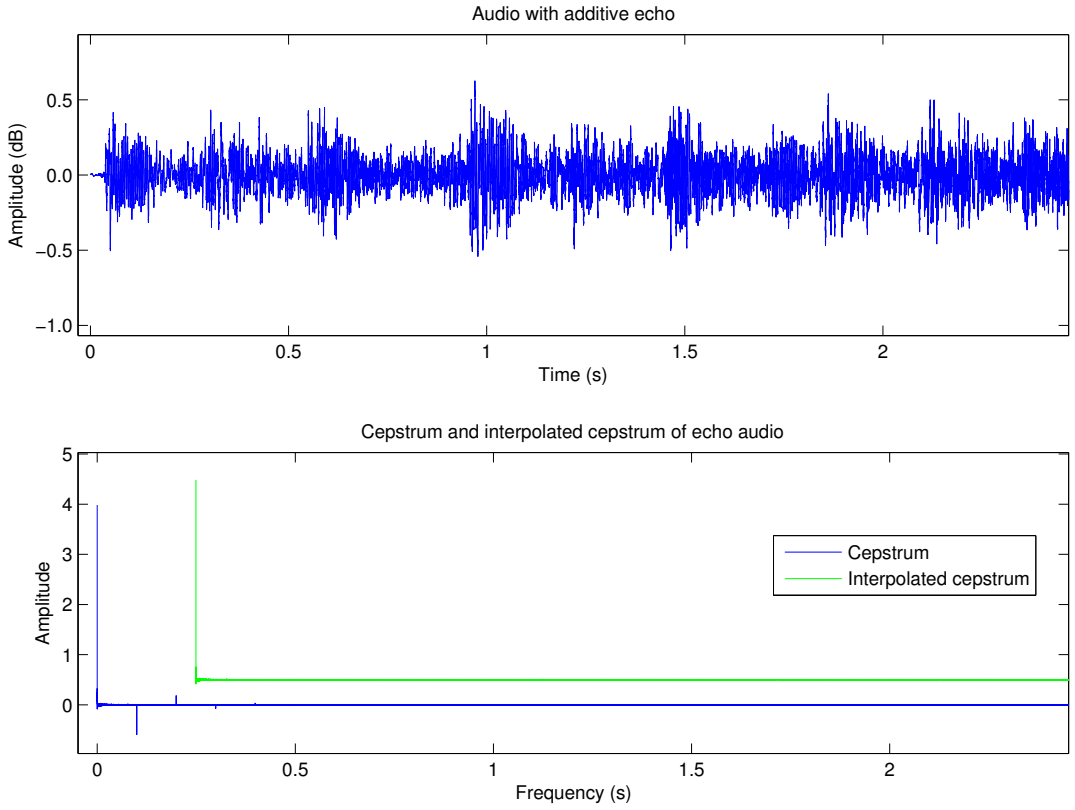

Fig. 2.2: Audio with additive echo, its cepstrum representation and the interpolated cepstrum.

The polynomial given by eq. 12 is solved, obtaining m roots. From these roots, the complex ones are discarded and a positive real root is selected as the value of α. With the calculated values for τ and α, the additive echo can be inversed in order to remove it. The experimental results obtained with the strategy described are presented below.

2.2 Experimental results

The purpose of these experiments was to test the performance of the proposed echo cancellation method. To do so, a set of 10 audio signals of 5 genres, *viz.* jazz, orchestra, rock, pop, and vocal was selected. The addition of an echo was performed by implementing eq. 1 with controlled parameters α and τ. The metrics used to measure the quality of the audio signals were the Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE), and Perceptual Evaluation Audio Quality (PEAQ) given by the Objective Difference Grade (ODG). The mean (μ), standard deviation (σ), minimum and maximum of the PSNR, MSE, and ODG for the 10 audio signals in each genre were calculated. Table 1 presents the quality results for the audio signals with additive echo, with $\alpha = 0.6$ and $\tau = 0.1$; these values were arbitrarily selected from the possible range of values for $\alpha \in [0, 1]$ and $\tau \in [0.06, T]$, where T is the length of the audio in seconds. The mean PSNR values of the signals with echo are low, and along with the mean ODG values that are around -4 indicate a very annoying degradation of the echo signals with respect to the host signals.

In order to remove the echo, first the value of τ was determined with the cepstrum, as previously explained. Then, an approximated value of α was calculated following the strategy mentioned above. To obtain $Z_{\text{approx}}(n)$, a *spline* interpolation was used to remove the peaks from the cepstrum, and the polynomial to approximate α was truncated at $m = 5$. The spline interpolation and the value for

Table 1: PSNR, MSE, and ODG results for echo signals ($\alpha = 0.6$, $\tau = 0.1$).

Genre	PSNR (dB)				MSE				ODG			
	μ	σ	Min	Max	μ	σ	Min	Max	μ	σ	Min	Max
Jazz	21.9	1.7	19.0	25.6	6.99×10^{-3}	2.63×10^{-3}	2.75×10^{-3}	1.26×10^{-2}	-3.7	0.2	-3.9	-3.2
Orchestra	25.3	6.3	17.3	35.0	6.05×10^{-3}	6.39×10^{-3}	3.16×10^{-4}	1.88×10^{-2}	-3.6	0.4	-3.8	-2.7
Rock	13.9	2.3	11.1	17.3	4.58×10^{-2}	2.28×10^{-2}	1.88×10^{-2}	7.72×10^{-2}	-3.3	0.4	-3.8	-2.8
Pop	17.9	2.5	12.7	20.8	1.93×10^{-2}	1.35×10^{-2}	8.34×10^{-3}	5.35×10^{-2}	-3.8	0.2	-3.9	-3.4
Vocal	17.7	2.0	15.5	22.2	1.83×10^{-2}	6.94×10^{-3}	5.98×10^{-3}	2.83×10^{-2}	-3.8	0.0	-3.9	-3.7

m were determined experimentally; 5 interpolators were tested and spline presented better results; approximations for α with values of $m \in [1, 10]$ were tested and $m = 5$ produced approximations closer to the real α. The mean values are very close to the real value of α used and their standard deviation is very small, which means that all the α approximations are very close to the mean. Fig. 2.3 presents the distribution of approximated α values, where it can be seen that the mean values are very close to the real value of α.

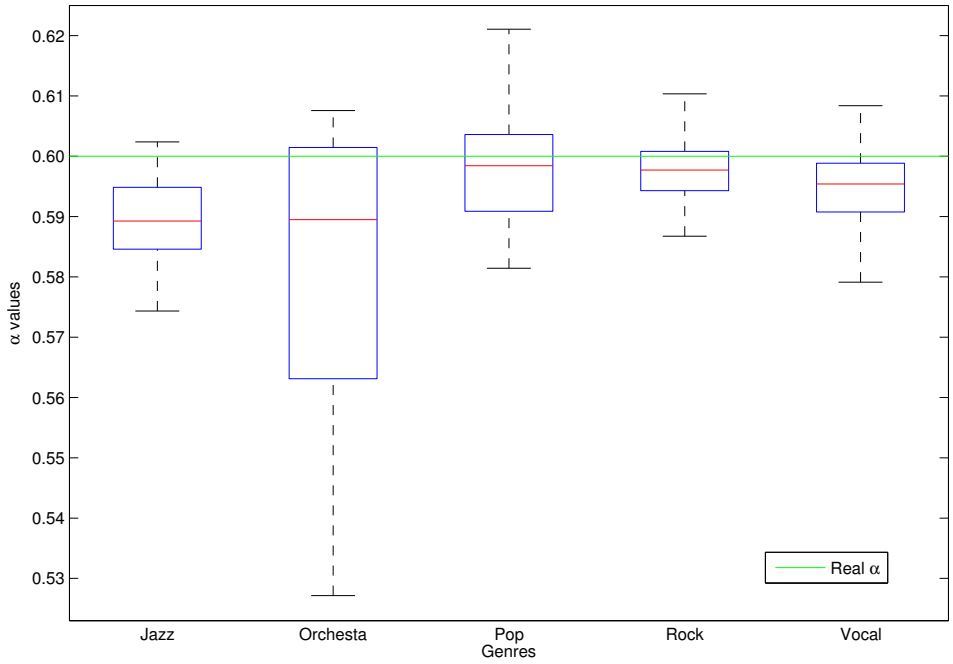

Fig. 2.3: Distribution of approximated α values.

After obtaining the α approximations, the echo was canceled with the calculated value of τ and the mean of approximated α values. Table 2 presents the quality results for audio signals without echo (echo-less). As it can be seen in these results, the mean PSNR values are high, which means more similarity between the original and echo-less signals; the mean MSE values are low, which indicate small differences; and the mean ODG values are very close to 0 and in most cases even above 0, which means that although the echo-less signals contain artifacts, these are inaudible. The standard deviations for the three metrics are relatively small, so the dispersion from the mean value is also small.

Table 2: PSNR, MSE, and ODG results for echo-less signals.

Genre	PSNR (dB)				MSE				ODG			
	μ	σ	Min	Max	μ	σ	Min	Max	μ	σ	Min	Max
Jazz	58.3	6.5	52.5	75.1	2.45×10^{-6}	1.71×10^{-6}	3.07×10^{-8}	5.56×10^{-6}	0.1	0.1	−0.1	0.2
Orchestra	59.8	17.9	23.8	92.2	4.21×10^{-4}	1.32×10^{-3}	6.04×10^{-10}	4.18×10^{-3}	−0.3	1.2	−3.6	0.2
Rock	60.3	9.2	49.8	80.3	2.94×10^{-6}	3.55×10^{-6}	9.38×10^{-9}	1.04×10^{-5}	0.2	0.0	0.2	0.2
Pop	58.0	11.3	47.0	85.0	5.94×10^{-6}	6.72×10^{-6}	3.14×10^{-9}	1.99×10^{-5}	0.0	0.1	−0.2	0.2
Vocal	59.0	8.5	45.3	72.7	5.42×10^{-6}	9.31×10^{-6}	5.37×10^{-8}	2.95×10^{-5}	0.1	0.1	−0.2	0.2

3 Self-recovery for audio restoration

Technologies that allow to share and modify digital content arise rapidly in recent years, many of these technologies facilitate the modification of digital images, videos and audio. However, there are cases where owners do not wish unauthorized modifications of their content. Fragile watermarking was devised as a means to authenticate digital contents and in some applications, for tamper localization. Once the schemes were capable of identifying the positions where tampering had occur, a natural desire was to somehow restore the tampered regions and with this idea self-recovery schemes arose. The scheme proposed by Fridrich and Goljan [22] was the first to introduce the idea of self-embedding the image to restore the tampered regions.

To date, many schemes designed for images have been proposed, some deal with images in the spatial domain and focus on resisting content replacement attacks as [23,24]; other works deal with signal processing or cropping attacks as the schemes proposed in [25,26,27]. A few self-recovery schemes for video signals have also been proposed in [28,29,30,31]. There are self-recovery schemes for images that even achieve perfect restoration of the tampered content, provided that the attacked areas are small [4,32,33].

However, schemes for audio signals only deal with authentication and tamper localization [34,35,36,37,38]. A functional self-recovery scheme for audio signals has not been proposed yet. Some efforts done towards the construction of such a self-recovery scheme are presented and the difficulties that a scheme that deals with the characteristics of audio signals would have to overcome are also stated.

3.1 Proposed method

Self-recovery watermarking schemes arose with the idea of restoring the missing areas besides identifying the tampered regions. Although every scheme uses a different strategy, the general ideas for the encoding and decoding processes are as follows. The encoding process calculates reference bits and check bits from the image; reference bits are a reduced version of the image itself (calculated by compressing or obtaining a descriptive representation of the image), check bits are the result of feeding regions of the image to a hash function; both reference bits and check bits are scattered through the image for embedding, obtaining in this manner the watermarked image. The decoding process receives an image and extracts a watermark, from which the extracted reference bits and check bits are obtained; the extracted check bits are compared against the check bits calculated from the received image to identify the tampered regions; by using the reference bits from non-tampered regions of the image, the tampered reference bits can be restored, and with both the non-tampered and restored reference bits the tampered areas of the image can be recovered.

The strategies for schemes presented so far are designed for images and deal with 2D signals. Although there are fragile watermarking schemes for audio signals (1D signals), they only solve the authentication problem and in some cases they achieve tamper detection; however, a self-recovery

scheme suitable for audio has not yet been proposed. The efforts made for constructing such a self-recovery scheme are reported in this work.

The objective of this research is to propose a self-recovery scheme for audio signals that is capable of perfect restoration after a watermarked signal is submitted to a content replacement attack. There are only three self-recovery schemes for images that achieve perfect restoration [4,32,33]; one of these methods was selected as a baseline and its general ideas were considered for the construction of the self-recovery scheme for audio presented in this work. The work from [4] was selected because it was the first methods to obtain perfect restoration, and the other two schemes use very similar ideas to achieve the same goal.

One of the greatest challenges with self-recovery for audio is the distortion caused by the embedding process. The goal applications where this scheme is to be used require audio signals with a transparency over -2 ODG. The objective difference grade (ODG) is the transparency metric recommended by ITU-R B.S.1387 and implemented in [39]. Because of this transparency restriction, a strategy to reduce perceptual impact had to be devised by the use of the integer Discrete Cosine Transform (intDCT) domain for embedding and extraction of the watermark. The use of this domain was inspired by the work of [40] where a reversible watermarking scheme for audio in the intDCT domain is proposed.

intDCT transform

The intDCT domain is used both for embedding and extraction of the watermark. An audio signal in the time domain is transformed to the intDCT domain using the fast intMDCT algorithm proposed by [41]. The intDCT of an N-poing audio signal $x[n]$ is defined as:

$$\mathbf{X} = C_N^{IV} \cdot \mathbf{x}, \tag{13}$$

where $\mathbf{X}$ are the intDCT coefficients of $\mathbf{x}$, and

$$\mathbf{x} = x[n]_{\{n=0,1,\cdots,N-1\}},$$
$$\mathbf{X} = X[m]_{\{m=0,1,\cdots,N-1\}}.$$

C_N^{IV} is the transform matrix, defined as:

$$C_N^{IV} = \sqrt{\frac{2}{N}}\left[\cos\left(\frac{(m+1\div 2)(n+1\div 2)\pi}{N}\right)\right], \tag{14}$$

where $m = 0, 1, \cdots, N-1$ and $n = 0, 1, \cdots, N-1$. Because C_N^{IV} is an orthogonal matrix, the inverse intDCT transform is given by:

$$\mathbf{x} = C_N^{IV} \cdot \mathbf{X}. \tag{15}$$

Encoding process.

Because of the dimensionality of audio signals, it is difficult to process them as a whole, as it is done in schemes for images; so the self-recovery strategy has to process windows of samples. For an audio signal of size L, select a window of samples with length L_w, there are a total of $\lfloor L \div L_w \rfloor$ windows for the signal. To increase the accuracy of tamper detection, and for implementation purposes, each window being processed is divided in segments of length L_s, for each window, there are a total of $\lfloor L_w \div L_s \rfloor$ segments.

Reference bits generation. In this step, the bits that will be used to restore the signal are generated. When working with audio signals with CD quality, each sample is represented by 16 bits; this amount

of information cannot be embedded within the signal and it must be reduced. So, the samples in each window are re-quantized to 8 bits per sample, obtaining $8 \times L_w$ bits per window. Pseudo-randomly permute those bits, based on the secret key, and divide them into $L_w \div n_g$ groups, where n_g is a value power of 2; each group contains $n_b = n_g \times 8$ bits. Denote the bits in a group as $b_t(1), b_t(2), \cdots, b_t(n_b)$ where $t = 1, 2, \cdots, L_w \div n_g$. For each group, calculate $n_{rb} = n_b \div (8 \times \text{compRat})$ reference bits $r_t(1), r_t(2), \cdots, r_t(n_{rb})$ where 'compRat' is the compression ratio of the bits, *i.e.* compRat $= 2$ will keep $\frac{1}{16}$ of the $8 \times L_w$ original bits, compRat $= 4$ will keep $\frac{1}{32}$ of the $8 \times L_w$ original bits, and so on. The reference bits are calculated in the following way:

$$\begin{bmatrix} r_t(1) \\ r_t(2) \\ \vdots \\ r_t(n_{rb}) \end{bmatrix} = A_t \cdot \begin{bmatrix} b_t(1) \\ b_t(2) \\ \vdots \\ b_t(n_b) \end{bmatrix}, t = 1, 2, \cdots, \frac{L_w}{n_g}, \tag{16}$$

where A_t are pseudo-random binary matrices of size $n_{rb} \times n_b$, the matrices A_t are calculated based on the secret key and the arithmetic in eq. 16 is modulo-2. For this implementation, $n_g = 256$ is set to construct $L_w \div 256$ bit-groups, each group contains also $n_b = 2048$ bits; the compression ratio is set to 4, which means that there are $n_{rb} = 64$ reference bits per group, the A_t matrices have sizes of 64×2048, and a total of $L_w \div 4$ reference bits are obtained. The final reference bits are pseudo-randomly permuted based on the secret key.

Check bits generation. This step calculates the check bits that will be used to identify the segments of the signal where tampering occurs. Because any modification in the intDCT domain affects all the time domain samples in a segment of audio, there is no way of knowing, just from the time-domain representation of a signal, which samples carry watermark information and which samples do not. For this reason, the check bits are obtained from the intDCT coefficients. For each segment in the window, calculate its forward intDCT transform. From the intDCT coefficients, collect the values of the coefficients that will not be modified during the embedding process, also collect the reference bits that correspond to the segment. Feed the non-modified intDCT coefficients and corresponding reference bits to a hash function that produces 256 hash bits per segment. There are a total of $256 \times \frac{L_w}{L_s}$ hash bits per window. Pseudo-randomly permute the hash bits from the whole window, using the secret key to determine the order. To reduce the number of check bits, divide the hash bits into $L_w \div 4$ subsets, then calculate a modulo-2 sum of the 4 hash bits in each subset, the sum will produce 64 check bits per segment and $64 \times \frac{L_w}{L_s}$ check bits per window.

Embedding. In this final step of the encoding process, the watermark bits to be embedded in each segment are obtained from the reference bits and the check bits previously generated. The watermark bits for each segment are obtained by concatenating $L_s \div 4$ reference bits with the corresponding 64 check bits of the segment to produce the watermark, denoted as $w[k]$, where $k = \{1, 2, \cdots, K\}$, and K is the size of the watermark. In this case $K = L_s \div 4 + 64$. The insertion of the watermark is done through prediction-error expansion (PEE) in the intDCT domain, in a similar fashion as in the scheme by [40]. Because of the characteristics of the intDCT transform explained above, it is necessary to consider that coefficients in odd positions are more similar to other coefficients in odd positions, and coefficients in even positions are more similar to other coefficients in even positions. The prediction value of the i^{th} coefficient, denoted by $\hat{X}[i]$ is calculated as:

$$\hat{X}[i] = \left\lfloor \frac{X[i-2] - X[i-4]}{2} \right\rfloor, \tag{17}$$

and the prediction-error, denoted as p, is given by:

$$p = X[i] - \hat{X}[i], \tag{18}$$

where $i = \{\text{lowBand} + K - 1, \text{lowBand} + K - 2, \cdots, \text{lowBand}\}$, 'lowBand' is an offset value used to find an appropriate region of frequencies to be modified, in such a way that the distortion in the time domain signal is reduced to an acceptable ODG threshold; the calculation of this offset value is detailed below.

The prediction-error p is expanded and a watermark bit is embedded as:

$$p_w = 2 \times p + w[k]. \tag{19}$$

The watermarked intDCT coefficients are obtained by:

$$Y[i] = \hat{X}[i] + p_w. \tag{20}$$

Offset optimization. As it was previously mentioned, one of the greatest challenges on the design of a self-recovery scheme for audio is the reduction of perceptual impact produced in the encoding process. It is crucial to find a strategy that allows the insertion of the required payload, and still maintains an acceptable transparency for practical applications. The strategy devised in this work is the insertion and extraction in the intDCT domain; however, for this strategy to be successful in terms of transparency, an adequate region of frequencies has to be identified. This adequate region of frequencies is one that contains enough energy, so the modifications caused by embedding do not affect significantly the transparency of the watermarked audio signals. To do so, an offset value, denoted as 'lowBand' is used to find the best region for embedding. A band with the first 10 frequencies is always kept intact, and the 'lowBand' offset is incremented until the the last possible frequency is reached, *i.e.* $\text{lowBand} = \{10, 11, \cdots, Ls - (Ls \div 4 + 64)\}$. This search could be done exhaustively, *i.e.* with increments of 1; however, to reduce the computational cost bigger steps can be used. For the first phase of experimental results, a 'lowBand' step of 30 was used.

Security layer. In a speech restoration scenario, a framing person could be interested on rendering impossible to restore the original speech from the tampered speech. In a music censorship scenario, a customer could desire to restore the original uncensored version of the song without paying the corresponding fee. Both of these things can be done if the secret key used to disperse the reference and check bits can be predicted. If a small key-space is used, a brute force algorithm could find the key. With this key, the reference bits that correspond to a certain region of the speech signal can be found in the rest of the signal, by eliminating those reference bits the original speech would not be restored. In the other scenario, if the secret key is predicted, a customer can restore the uncensored song without payment of the fee. Because of this, a big enough key-space is necessary; a key as the one used by the Advanced Encryption Standard (AES) is recommended, *i.e.* a symmetric key of 256 bits.

Decoding process.

As in the encoding process, an audio signal of size L is divided into windows of samples of length L_w, and each window is further divided in segments of size Ls. The decoding process is applied to each of the $L \div L_w$ windows and the general steps are detailed below. The watermark is extracted from the intDCT coefficients of each segment, after extraction from all the segments the reference bits and check bits of the window can be reconstructed using the secret key. The extracted check bits are compared against the check bits obtained from the received signal to detect the tampered regions.

The reference bits and the sample values from non-tampered regions are used to restore the tampered samples.

Watermark extraction. First, each window of signal samples is divided into segments of length L_s as in the encoding process, the intDCT coefficients of each segment are obtained and the PEE extraction process is applied in the following way. The prediction value $\hat{X}[i]$ is calculated as:

$$\hat{X}[i] = \left\lfloor \frac{X[i-2] + X[i-4]}{2} \right\rfloor, \tag{21}$$

and the expanded prediction-error is given by:

$$p_w = Y[i] - \hat{X}[i], \tag{22}$$

where $i = \{\text{lowBand}, \text{lowBand} + 1, \cdots, \text{lowBand} + K - 1\}$. Note that the same 'lowBand' value used for embedding is required for the extraction of the watermark. The original prediction-error p is obtained by:

$$p = \left\lfloor \frac{p_w}{2} \right\rfloor, \tag{23}$$

and the watermark bit $w[k]$ is extracted as:

$$w[k] = p_w - 2 \times p. \tag{24}$$

The original intDCT coefficients are restored by:

$$X[i] = \hat{X}[i] + p. \tag{25}$$

The original sample values in the time domain are obtained by applying the inverse intDCT transform to the restored intDCT coefficients. The watermark extracted from each segment is divided into reference bits and check bits, all the reference and check bits of the window are obtained when the watermarks of all the segments are extracted.

Tampered segment identification. The check bits extracted in the previous step are compared against the check bits calculated from the extracted reference bits and the restored sample values from the previous step. The consistency between these check bits is the criteria for judging a segment as "non-tampered" or "tampered".

To calculate the check bits from the received signal, the non-modified intDCT coefficients of each segment are collected, along with the reference bits that correspond to that segment. All these values are fed to the same hash function to obtain 256 hash bits per segment, the $256 \times \frac{L_w}{L_s}$ hash bits are pseudo-randomly permuted in the same way as in the encoding process, and the hash bits are divided into $L_w \div 4$ subsets, as in the encoding process, calculate a modulo-2 sum of the 4 bits in each subset to obtain $64 \times \frac{L_w}{L_s}$ "calculated check bits".

The 64 calculated check bits are compared against the extracted check bits. Denote the number of extracted check bits in a segment as N_E, and be N_F the number of extracted check bits different to their corresponding calculated check bits, where $N_F \leq N_E$. For a given N_E, an integer T is found, such that it satisfies:

$$\sum_{l=0}^{T} P_T, N_F(l) < 10^{-9}, \tag{26}$$

and

$$\sum_{l=0}^{T+1} P_T, N_F(l) \geq 10^{-9}. \tag{27}$$

If $N_F > T$ then the segment is regarded as "tampered" and "non-tampered" otherwise. The probability of falsely identifying a tampered segment as a non-tampered one is less than 10^{-9}.

Signal restoration. In this final step, the original sample values from "tampered" segments are restored. Mark the reference bits and sample values from each tampered segment as 'NaN' values to facilitate its differentiation from reference bits and samples from non-tampered segments in the next steps. The vectors and matrices from eq. 16 are recalculated with the extracted reference bits and the interim restored signal obtained so far (the time-domain signal obtained after watermark extraction). Because the received signal is quantized at 16 bps, it has to be re-quantized to 8 bps to construct the b_t vectors; note that each 'NaN' in the interim restored signal is traduced to 8 'NaN' values at the 8-bit representation of the signal.

The $8 \times L_w$ bits of the binary representation of the signal are divided into $Lw \div n_g$ groups as in the encoding process, each group contains $n_b = n_g \times 8$ bits. The number of reliable reference bits in each bit-group, denoted as n_t, may be less than the original n_{rb} reference bits from encoding ($n_{rb} = 64$). Equation 16 implies that:

$$\begin{bmatrix} r_t(s_1) \\ r_t(s_2) \\ \vdots \\ r_t(s_{n_t}) \end{bmatrix} = A_t^{(R)} \cdot \begin{bmatrix} b_t(1) \\ b_t(2) \\ \vdots \\ b_t(n_b) \end{bmatrix}, t = 1, 2, \cdots, \frac{L_w}{n_g}. \tag{28}$$

The r_t vectors contain the reliable extracted reference bits and $A_t^{(R)}$ is a matrix that has all the rows from A_t that correspond to the reliable extracted reference bits, *i.e.* all the rows in r_t with 'NaN' values are removed and the same rows from A_t are removed to obtain $A_t^{(R)}$. On the other side of eq. 28, the n_b bits in each bit-group contain two types of bits: 1) the missing bits from "tampered" segments, and 2) the recovered bits from other positions. The assumption of this restoration strategy relies on the fact that, if a small region of the signal was tampered, then the missing bits in each b_t are small (because those missing bits are dispersed throughout different bit-groups) and do not affect the restoration.

In this way, the reliable reference bits and the non-missing bits in the b_t groups can provide enough information to recover the original values of missing bits. Let be $B_{t,1}$ a column vector that corresponds to the missing bits from b_t, and $B_{t,2}$ a column vector that corresponds to the recovered bits in b_t, *i.e.* $B_{t,1}$ is a column vector that corresponds to the rows in b_t that contain 'NaN' values and $B_{t,2}$ is a column vector that corresponds to the rows in b_t with values different to 'NaN'. Then eq. 28 can be reformulated as:

$$\begin{bmatrix} r_t(s_1) \\ r_t(s_2) \\ \vdots \\ r_t(s_{n_t}) \end{bmatrix} - A_t^{(R,2)} \cdot B_{t,2} = A_t^{(R,1)} \cdot B_{t,1}, t = 1, 2, \cdots, \frac{L_w}{n_g}, \tag{29}$$

where $A_t^{(R,1)}$ is a matrix constructed from the columns of $A_t^{(R)}$ that correspond to the missing bits in b_t, and $A_t^{(R,2)}$ is a matrix constructed from the columns of $A_t^{(R)}$ that correspond to the recovered bits in b_t. From eq. 29, the left side and the matrix $A_t^{(R,1)}$ are known variables, so only $B_{t,1}$ has to be found. Let n_{mb} be the number of elements in $B_{t,1}$, then the size of the matrix $A_t^{(R,1)}$ is $n_t \times n_{mb}$. A number of

n_{mb} unknowns are solved according to the n_t equations in the binary system, so the idea is to solve eq. 29 for $B_{t,1}$, therefore obtaining the missing bits. With those missing bits, the 8-bit representation of the signal can be restored and a re-quantization must be done again to restore the 16-bit representation of the audio signal. The samples restored after watermark extraction that correspond to non-tampered regions are kept for the 16-bit representation of the signal. Only the samples from tampered regions are re-quantized from the 8-bit restored samples to construct the final restored 16-bit signal.

Limitations.

The results presented in the previous section are encouraging because the biggest challenge on self-recovery for audio, namely embedding transparency, has been solved, at least for the -2 ODG threshold imposed by the applications. However, this quality is limited by the embedded payload. The current strategy inserts $L_s \div 4 + 64$ bits per segment, which is equivalent to 0.02 bits per second (bps). If the payload is increased to 0.03 bps, the embedding distortions are unacceptable for the target applications, *i.e.* the ODG values fall under -2, seriously compromising the practicability of the scheme. An strategy that used a payload of 0.01 bps was also tested; however, the false positive detection of tampered segments was too high, and the scheme could not properly restore the real tampering caused by the attacks. So, it can be seen that there is a very strict balance between payload capacity, embedding transparency, and restoration capability. On the one hand, the increase of payload improves the restoration capability but reduces transparency. On the other hand, reduction of payload increases transparency but reduces the restoration capability.

On regard of the restoration capabilities of the proposed scheme, the results in the previous section show that, in general terms, the scheme can restore signals that were tampered with a maximum of 0.6%. However, these results also suggest that the restoration for different percentages of attacks greatly depends on the characteristics of the audio signals, *i.e.* higher energy signals allow better restorations because the restored samples do not have such a big contrast against the rest of the samples, as opposed to low energy signals, where even after restoration the contrast among samples is notorious. However, the limitation on restoration capability and the percentages that the scheme can tolerate are still being investigated.

In the proposed self-recovery scheme, the problem of overflow that can occur when embedding the watermark is addressed in a pre-processing of the host audio signals, where their volume is reduced by applying a division by 2, or a right-bit shift. Reversible watermarking techniques use more sophisticated strategies to deal with the underflow and overflow, the most common one being the construction of a location map that indicates when these problems occur. However, the inclusion of a location map would require to increase the payload that is embedded into the signals, and as already mentioned, a slight increase on the payload produces unacceptable transparency.

The content replacement attack is the only one being considered for this scheme, in part for the applications where it is desired to be used, but also because if other attacks are considered, then other strategies would have to be included in the scheme. If these other strategies require the insertion of more bits into the signals, the scheme would not fulfil the transparency requirement with the current embedding strategy in the intDCT domain. Even with a content replacement attack that increases or reduces the number of samples in the signal, it would be necessary to design a synchronization strategy to detect the positions where the increase or reduction of samples took place and to be able to find the position of the next watermarked window. This is an initial effort on the construction of a self-recovery scheme for audio, and other attacks will be investigated in future stages of the research.

4 Conclusions and future work

In this technical report some efforts for designing watermarking schemes for audio that allow the restoration of the signals after two types of attacks are applied are addressed. These attacks are echo-addition and content-replacement.

An echo-cancellation strategy was proposed that allows the elimination of a one-tap echo added to audio signals with fair quality; however the reduction of the error after echo cancellation is desired. Future efforts are being oriented towards the reduction of these errors and to expand this solution for multi-tap echoes.

Also, a self-recovery watermarking scheme for audio signals is presented. This self-recovery scheme allows the identification of tampering in segments of audio signals and allows the reconstruction of those regions to a certain extent. Future efforts are oriented towards the improvement of the restoration capabilities of the scheme and the increase on embedding capacity.

The final objective of this research is the construction of a reversible watermarking scheme for audio signal with watermark and signal robustness, in the sense that this scheme is capable of restoring the original signal and extracting the secret watermark even in the presence of attacks.

References

1. R. Caldelli, F. Filippini, and R. Becarelli, "Reversible Watermarking Techniques: An Overview and a Classification," *EURASIP Journal of Information Security*, vol. 2010, pp. 1–19, Jan. 2010.
2. C. W. Honsinger, P. W. Jones, M. Rabbani, and J. C. Stoffel, "Lossless recovery of an original image containing embedded data." US Patent, August 2001. US Patent 6,278,791.
3. C. De Vleeschouwer, J.-F. Delaigle, and B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *Multimedia, IEEE Transactions on*, vol. 5, no. 1, pp. 97–105, 2003.
4. X. Zhang and S. Wang, "Fragile Watermarking With Error-Free Restoration Capability," *IEEE Transactions on Multimedia*, vol. 10, no. 8, pp. 1490–1499, 2008.
5. X. Zhang, S. Wang, Z. Qian, and G. Feng, "Reference Sharing Mechanism for Watermark Self-Embedding," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 485–495, 2011.
6. S. Bravo-Solorio, C.-T. Li, and A. Nandi, "Watermarking method with exact self-propagating restoration capabilities," in *IEEE International Workshop on Information Forensics and Security (WIFS), 2012*, pp. 217–222, 2012.
7. A. Menendez-Ortiz, C. Feregrino-Uribe, and J. J. Garcia-Hernandez, "Reversible image watermarking scheme with perfect watermark and host restoration after a content replacement attack," in *The 2014 International Conference on Security and Management (SAM'14)*, vol. 13, pp. 385–391, July 2014.
8. S. Bravo-Solorio and A. K. Nandi, "Secure fragile watermarking method for image authentication with improved tampering localisation and self-recovery capabilities," *Signal Processing*, vol. 91, no. 4, pp. 728–739, 2011.
9. D. Messerschmitt, "Echo Cancellation in Speech and Data Transmission," *Selected Areas in Communications, IEEE Journal on*, vol. 2, pp. 283–297, Mar 1984.
10. A. Gilloire and M. Vetterli, "Adaptive filtering in subbands with critical sampling: analysis, experiments, and application to acoustic echo cancellation," *Signal Processing, IEEE Transactions on*, vol. 40, pp. 1862–1875, Aug 1992.
11. M. Sondhi, D. Morgan, and J. Hall, "Stereophonic acoustic echo cancellation-an overview of the fundamental problem," *Signal Processing Letters, IEEE*, vol. 2, pp. 148–151, Aug 1995.
12. J. Benesty, D. Morgan, and M. Sondhi, "A better understanding and an improved solution to the specific problems of stereophonic acoustic echo cancellation," *Speech and Audio Processing, IEEE Transactions on*, vol. 6, pp. 156–165, Mar 1998.
13. T.-A. Vu, H. Ding, and M. Bouchard, "A survey of double-talk detection schemes for echo cancellation applications," *Canadian Acoustics*, vol. 32, no. 3, pp. 144–145, 2004.
14. S. Y. Lee and N. S. Kim, "A Statistical Model-Based Residual Echo Suppression," *Signal Processing Letters, IEEE*, vol. 14, pp. 758–761, Oct 2007.
15. B. Panda, A. Kar, and M. Chandra, "Non-linear adaptive echo supression algorithms: A technical survey," in *International Conference on Communications and Signal Processing (ICCSP), 2014*, pp. 076–080, April 2014.
16. J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in network and acoustic echo cancellation*. Springer, 2001.
17. A. Oppenheim and R. Schafer, "From frequency to quefrency: a history of the cepstrum," *IEEE Signal Processing Magazine*, vol. 21, pp. 95–106, Sept 2004.

18. R. W. Schafer, "Homomorphic Systems and Cepstrum Analysis of Speech," in *Springer Handbook of Speech Processing* (J. Benesty, M. M. Sondhi, and Y. Huang, eds.), ch. 9, pp. 161–180, Springer-Verlag Berlin Heidelberg, 2008.
19. M. Talbot-Smith, *Sound Engineering Explained*. Elsevier, 2002.
20. R. Kemerait and D. Childers, "Signal detection and extraction by cepstrum techniques," *IEEE Transactions on Information Theory*, vol. 18, pp. 745–759, Nov 1972.
21. G.-H. Jain, "Digital Signal Processing of Multi-Echo Interference for Angle Modulated Signal," Master's thesis, Texas Tech University, 1982.
22. J. Fridrich and M. Goljan, "Protection of digital images using self embedding," in *Symposium on Content Security and Data Hiding in Digital Media*, Newark, NJ, USA, 1999.
23. K. Hung and C.-C. Chang, "Recoverable Tamper Proofing Technique for Image Authentication Using Irregular Sampling Coding," in *Autonomic and Trusted Computing* (B. Xiao, L. Yang, J. Ma, C. Muller-Schloer, and Y. Hua, eds.), vol. 4610 of *Lecture Notes in Computer Science*, pp. 333–343, Springer Berlin Heidelberg, 2007.
24. S.-S. Wang and S.-L. Tsai, "Automatic image authentication and recovery using fractal code embedding and image inpainting," *Pattern Recognition*, vol. 41, no. 2, pp. 701–712, 2008.
25. X. Zhu, A. T. Ho, and P. Marziliano, "A new semi-fragile image watermarking with robust tampering restoration using irregular sampling," *Signal Processing: Image Communication*, vol. 22, no. 5, pp. 515–528, 2007.
26. S. Bravo-Solorio, C.-T. Li, and A. Nandi, "Watermarking with lowembedding distortion and self-propagating restoration capabilities," in *19th IEEE International Conference on Image Processing (ICIP), 2012*, pp. 2197–2200, 2012.
27. H. He, F. Chen, H.-M. Tai, T. Kalker, and J. Zhang, "Performance Analysis of a Block-Neighborhood-Based Self-Recovery Fragile Watermarking Scheme," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 1, pp. 185–196, 2012.
28. B. Mobasseri, "A spatial digital video watermark that survives MPEG," in *International Conference on Information Technology: Coding and Computing, 2000.*, pp. 68–73, 2000.
29. M. U. Celik, G. Sharma, A. M. Tekalp, and E. S. Saber, "Video authentication with self-recovery," in *Electronic Imaging 2002*, pp. 531–541, International Society for Optics and Photonics, 2002.
30. A. M. Hassan, A. Al-Hamadi, Y. M. Y. Hasan, M. A. A. Wahab, and B. Michaelis, "Secure Block-Based Video Authentication with Localization and Self-Recovery," *World Academy of Science, Engineering and Technology*, vol. 2009, pp. 69–74, September 2009.
31. Y. Shi, M. Qi, Y. Yi, M. Zhang, and J. Kong, "Object based dual watermarking for video authentication," *Optik - International Journal for Light and Electron Optics*, vol. 124, no. 19, pp. 3827–3834, 2013.
32. X. Zhang, S. Wang, Z. Qian, and G. Feng, "Reference Sharing Mechanism for Watermark Self-Embedding," *IEEE Transactions on Image Processing*, vol. 20, no. 2, pp. 485–495, 2011.
33. S. Bravo-Solorio, C.-T. Li, and A. Nandi, "Watermarking method with exact self-propagating restoration capabilities," in *IEEE International Workshop on Information Forensics and Security (WIFS), 2012*, pp. 217–222, 2012.
34. E. Gómez, P. Cano, L. Gomes, E. Batlle, and M. Bonnet, "Mixed Watermarking-Fingerprinting Approach for Integrity Verification of Audio Recordings," 2002.
35. M. Steinebach and J. Dittmann, "Watermarking-based digital audio data authentication," *EURASIP Journal on Advances in Signal Processing*, vol. 2003, pp. 1001–1015, Jan. 2003.
36. T. Xu, X. Shao, and Z. Yang, "Multi-watermarking Scheme for Copyright Protection and Content Authentication of Digital Audio," in *Advances in Multimedia Information Processing - PCM 2009* (P. Muneesawang, F. Wu, I. Kumazawa, A. Roeksabutr, M. Liao, and X. Tang, eds.), vol. 5879 of *Lecture Notes in Computer Science*, pp. 1281–1286, Springer Berlin Heidelberg, 2009.
37. H. Wang and M. Fan, "Centroid-based semi-fragile audio watermarking in hybrid domain," *Science China Information Sciences*, vol. 53, no. 3, pp. 619–633, 2010.
38. M.-Q. Fan, P.-P. Liu, H.-X. Wang, and H.-J. Li, "A semi-fragile watermarking scheme for authenticating audio signal based on dual-tree complex wavelet transform and discrete cosine transform," *International Journal of Computer Mathematics*, vol. 90, no. 12, pp. 2588–2602, 2013.
39. T. Thiede, W. C. Treurniet, R. Bitto, C. Schmidmer, T. Sporer, J. G. Beerends, and C. Colomes, "PEAQ - The ITU Standard for Objective Measurement of Perceived Audio Quality," *Journal of the Audio Engineering Society*, vol. 48, no. 1/2, pp. 3–29, 2000.
40. Q. Chen, S. Xiang, and X. Luo, "Reversible Watermarking for Audio Authentication Based on Integer DCT and Expansion Embedding," in *Digital Forensics and Watermaking* (Y. Shi, H.-J. Kim, and F. Pérez-González, eds.), vol. 7809 of *Lecture Notes in Computer Science*, pp. 395–409, Springer Berlin Heidelberg, 2013.
41. H. Haibin, S. Rahardja, Y. Rongshan, and L. Xiao, "A fast algorithm of integer MDCT for lossless audio coding," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04).* , vol. 4, pp. IV–177–IV–180, May 2004.