High Payload Audio Watermarking Using Sparse Coding with Robustness to MP3 Compression

Mohamed Waleed Fakhr

Electrical and Electronics Department, University of Bahrain
Manama, Bahrain
mfakhr@uob.edu.bh

Abstract. A high payload audio watermarking technique is proposed based on the compressed sensing and sparse coding framework, with robustness to MP3 128kbps and 64kbps compression attacks. The binary watermark is a sparse vector with one non-zero element that takes a positive or negative sign based on the bit value to be encoded. A Gaussian random dictionary maps the sparse watermark to a random watermark embedding vector that is selected adaptively for each audio frame to maximize robustness to the MP3 attack. At the decoder, the Basis Pursuit Denoising algorithm (BPDN) is used to extract the embedded watermark sign. High payloads of (689, 1378 and 2756) bps are achieved with %BER of (0.3%, 0.5% and 1%) and (0.1%, 0.3% and 0.5%) for 64kbps and 128kbps MP3 compression attacks respectively. The signal to embedding noise ratio is kept in the range of 27-30 dB in all cases.

Keywords: Sparse Coding, Compressed Sensing, Audio Watermarking, MP3 Audio, Robust Watermarking, Basis Pursuit Denoising (BPDN).

1 Introduction

Robust watermarking techniques embed a secret code (watermark) within a multimedia file (e.g., MP3 music) where the watermark detection and verification is possible only for the certified authority. The watermark should be imperceptible, with minimal quality degradation (SNR in the vicinity of 30dB), should have high payload and robustness to attacks that do not destroy the original host signal, and should also be secure. This is a challenging problem for audio signals, and in particular, for MP3 audio where MP3 compression/decompression attack is inevitable. A large body of literature has emerged in the last few years dedicated to developing MP3 robust audio watermarking [1-8]. This is in part due to the challenges of the problem, and more importantly, due to the market need for a reliable copyright scheme to control the proliferation of MP3 music over the internet.

Robustness to MP3 attack, high payload and imperceptibility are conflicting requirements. Robustness requires more embedding strength, high payload requires more frequent embedding while imperceptibility requires high signal to embedding noise ratio SNR which is naturally lowered by the former two. Those conflicting criteria can be viewed as a multi-objective optimization problem, where one should

A.I. Awad, A.E. Hassanien, and K. Baba (Eds.): SecNet 2013, CCIS 381, pp. 98–110, 2013.

ideally create an optimal watermark for each audio frame that would maintain the high SNR and be robust to the MP3 attack. And if the high payload is required, the frame length should be in the order of a few milliseconds. Reaching such optimal settings would require the watermark to adapt to the host signal characteristics and presumed attacks. This creates two problems: If the watermark is going to adapt to its host frame, how can this information be conveyed to the decoder? And if this adaptation mechanism is known and is not highly secure then the adversary can reproduce the watermark and eventually remove it.

One way to solve the security issue is to select the watermark embedding vector from a secret random codebook that exists at both the encoder and the decoder. However, the question remains at the detection stage for which watermark to look for within the codebook, since trying all of them at the decoder would lead to many errors. On the other hand, if the codebook is large enough, a search criterion at the encoder can be used to adaptively select the best watermark vector for each audio frame by employing an analysis-by-synthesis paradigm, thus pushing the complexity to the encoder rather than the decoder.

In this paper, a sparse coding watermarking technique that employs an overcomplete dictionary (codebook) is proposed by which the adaptive watermark can be detected seamlessly at the decoder while the security is maintained by the randomness of the codebook [9]. The watermarking technique proposed can be interpreted by the compressed sensing (CS) signal recovery paradigm or, by the sparse coding (SC) paradigm where the L1-minimization is used in both to estimate a sparse vector from its random projection.

At the encoder an algorithm searches for the best watermark vector "atom" in the codebook that is robust to the MP3 coding-decoding attack. Once the best codebook atom is identified, it is embedded in the host frame. At the decoder, the watermark detection is formulated as an L1-norm minimization basis pursuit denoising (BPDN) problem that is solved to estimate the sparse watermark. The rest of the paper is organized as follows. Section 2 briefs the related recent work in MP3 audio watermarking. In section 3, the compressed sensing and sparse coding related theory is discussed and the proposed watermarking idea is introduced. In section 4 the proposed adaptive watermark algorithm is introduced. Related work in CS watermarking is in section 5. Experimental results are detailed in section 6 and conclusions are drawn in section 7.

2 Audio Watermarking

MP3 is popular because it offers good audio quality with small storage. As a consequence, on-line stores have proliferated because the ease to exchange MP3 files in the Internet. Robust audio watermarking is one way of tracking and copyright management of audio files, where many different approaches have been proposed in the past few years [1-8]. Table 1 summarizes the best MP3-robust techniques among the recent ones for comparison with the results in this paper. Most of the proposed techniques use the quantization index modulation (QIM) and spread spectrum approaches with variation on the embedding domain and exploiting audio characteristics. In summary, acceptable audio watermarking should have an SNR ≥30dB, a bit error rate (BER) ≤1% at MP3 attacks of 128kbps and 64kbp. The main challenge is to meet these specifications with the highest possible payload.

Table 1. MP3 Robust Watermarking Summary

Algorithm	MP3 Quality	SNR	%BER	Payload
Noriega [1] 2010	64kbps	40dB	0.013	230 bps
Bhat [2] 2010	64kbps	30dB	1	196 bps
Dhavale [3] 2011	32kbps	26dB	0.4	1378 bps
Yang [4] 2010	64kbps	30dB	0.02	22 bps
Hamdouni [5] 2012	64kbps	30dB	0.4	100 bps
Ercelebi [8] 2009	128kbps	30dB	0.5	170 bps

3 Proposed CS-SC Watermarking Framework

3.1 Compressed Sensing Related theory

Compressed sensing signal recovery relies on the concept of a sparse domain representation of compressible signals. In the basic formulation by Candès and others [10, 11], if a K-sparse vector x with dimension (N×1) is sampled with a random Gaussian M×N matrix Ω producing a measurement vector y with dimension (M×1) where M < N, then, given the measurement vector and knowing the sampling matrix Ω it is possible to recover the sparse vector x from y as follows:

$$y = \Omega x + e \tag{1}$$

Where e is a measurement noise with variance ε. An exact recovery of the sparse vector x is possible through L1-minimization using the basis pursuit denoising algorithm if Ω satisfies the restricted isometry property (RIP), which is met for Gaussian random orthonormalized matrices and many full of partial transforms [10, 11]. This is done by solving the convex linear optimization problem:

$$minimize \ \|\boldsymbol{x}\|_1 \ s.t. \ \|\boldsymbol{y} - \ \Omega\boldsymbol{x}\|_2 \ \leq \varepsilon \tag{1}$$

Where ε is tolerance, and the following sparsity condition should be satisfied where K is the number of non-zero elements in x:

$$M > \ O\left(K \ Log\left(\frac{N}{K}\right)\right) \tag{3}$$

More recently, there has been an extension to this framework by Laska et. al. [12] in what was named the Justice Pursuit (JP) algorithm explained as follows: If the measurement vector is corrupted by an interference that is sparse in some domain, equation (1) becomes:

$$y = \Omega x + \Phi\beta + e \tag{4}$$

Where Φ is a full or partial transform or random matrix with orthonormal columns and dimensions (M×L) where $L \leq M$ and β is a sparse vector with possibly large amplitude non-zero components of length L and sparsity k. It was shown that both sparse vectors (x and β) can be recovered if the matrices **Ω** and Φ are incoherent and with uncorrelated columns, and the sparsity condition in (3) is now updated to:

$$M > O\left((K + k)\, Log\left(\frac{N+L}{K+k}\right)\right) \tag{5}$$

The recovery algorithm assumes a new sparse vector U=[x β] of size (N+L)×1 and a new matrix $\boldsymbol{\psi}$=[Ω Φ] with size M×(N+L), and (4) is re-written as:

$$\mathrm{y} = \psi\mathrm{U} + \mathrm{e} \tag{6}$$

And the basis pursuit denoising (BPDN) becomes:

$$minimize\ \|\mathbf{U}\|_1 \quad \text{s.t.}\ \|\mathbf{y} - \boldsymbol{\psi}\mathbf{U}\|_2 \leq \varepsilon \tag{7}$$

When the sparse vector U is estimated, its first N elements are those of x and the remaining L elements are those of β [12]. Many fast algorithms have been developed in the literature for solving the basis pursuit denoising, of those, the L1-magic library which uses the primal-dual algorithm [13] is used for all the L1-minimization linear programming algorithms in this paper.

3.2 Compressed Sensing (CS) Watermarking Paradigm

The above CS formulation lends itself directly to the watermarking problem. Let the watermark be a sparse vector β of length L and with k non-zero components which take the binary values ±1 based on the required watermark value. In this paper, k=1 (only one non-zero element). For embedding in a measurement vector of length M, a random vector watermark signal W is generated:

$$W = \Phi\,\beta \tag{8}$$

Where Φ is an orthonormalized iid random Gaussian matrix of size (M×L). The host audio signals (music or speech) are known to be compressible signals for which sparse or approximately sparse domains exist. Let the audio frame in the sparse transform domain be x which is assumed to be K-sparse (or approximately sparse) in some transform domain. Going back to the time-domain we take the inverse of the sparsifying transform, and the host signal in time domain is given by:

$$X = \Omega x \tag{9}$$

Where Ω is the inverse of the used transform matrix. Adding (8) and (9) we get the watermarked host signal given by:

$$X^w = \ y = \Omega x + \ \Phi\beta \tag{10}$$

Thus, given the watermarked signal X^w, the BPDN can be used to find both β and x.

3.3 Sparse Coding (SC) Interpretation

Sparse coding with adaptive and non-adaptive overcomplete dictionaries has been used in many applications such as image and audio denoising and classification [14-16]. The basic SC concept models a signal as a sparse decomposition of atoms from an overcomplete dictionary, where the atoms locations and strengths are estimated using the L1-norm minimization. Equation (10) may be generalized as follows:

$$\mathrm{X^w} = \ y \ = \Omega_1 x_1 + \ \Omega_2 x_2 + \cdots + \ \Omega_C x_C + \ \Phi\beta \tag{11}$$

The host signal in time domain is assumed to be a decomposition of C sparse vectors each in a different domain and the matrix Ω is given by $\Omega = [\Omega_1\, \Omega_2 \ldots \Omega_c \ldots \Omega_C]$, where Ω is now of size M×N, with N=C×M and each inverse transform matrix Ω_c is M×M. Equation (11) is written as:

$$X^w = \ y = \psi U \qquad where \qquad \psi = [\Omega \ \ \Phi] \tag{12}$$

Now ψ is the overcomplete dictionary of size M×(N+L), and U is the composite sparse vector its first N elements are those of x and the remaining L elements are those of β.

3.4 Watermark Embedding

Following (10), the watermarked host signal for the $\boldsymbol{i_{th}}$ frame is given by:

$$X_i^w = \Omega x_i + \alpha_i \ * \Phi\ \beta_i \tag{13}$$

Where α_i is an embedding strength which is adaptively adjusted to make the average SNR of the watermarked signal constant as follows*:* Since all watermark vectors (the columns of Φ) are normalized, the SNR is:

$$SNR = 10Log_{10}\frac{\sum_{i=1}^{M} Xi^2}{\alpha^2} \tag{14}$$

In this paper, the SNR is fixed at 30dB, and α is given by:

$$\alpha = 0.025\sqrt{\sum_{i=1}^{M} Xi^2} \tag{15}$$

The watermark sparse vector β_i contains only one non-zero element (the $\boldsymbol{j}_{\mathbf{th}}$ element) with the required watermark sign. Thus, only one vector from the random matrix Φ is used: Φ_j for each audio frame. At the decoder, it is required to estimate the sign of the non-zero element in the β_i vector given the watermarked frame and the matrix Φ.

Having the measurement vector y, we apply the basis pursuit denoising algorithm as in (7) and the watermark sparse vector β is estimated as well as the sparse vector x and the estimated host signal in time domain is obtained by X= Ω x.

In the clean situation with no MP3 attack or other attacks, both the watermark location and sign are recovered perfectly with strong enough embedding strength (with SNR less than 70dB) and the number of columns in Φ is small. However, with the MP3 compression followed by decompression, more than one non-zero value usually appears in the recovered watermark and the sign of the correct location may be flipped. This is because the MP3 compression-decompression attack is highly nonlinear. Figure 1 illustrates a typical case where L=8 for the same audio frame and watermark, with and without the MP3 attack.

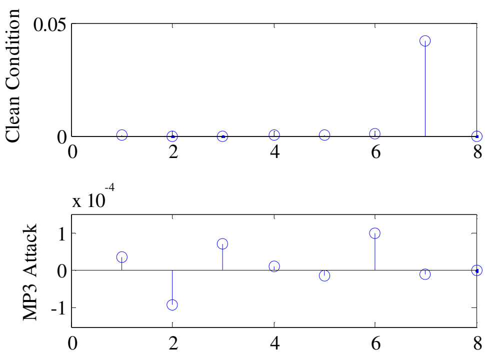

Fig. 1. Estimated sparse Watermark vector in MP3 attack

3.5 Watermark Extraction

Therefore, at the decoder, the largest non-zero value is found, and its sign is taken as the estimated watermark. This arrangement works well for small payloads and 128kbps MP3 quality. However, as shown in figure 1, it does not work well for 64kbps MP3 attack and for larger payloads as the correct spike may shift and more non-zero spikes appear. Moreover, it is very difficult to model the MP3 compression effect analytically on the recovered watermark due to the nonlinear and non-stationary behaviour of the MP3 attack.

4 Adaptive Watermark Selection Algorithm

Since there are L columns (atoms) in the Φ matrix it would be optimal to select the atom Φ_j for each new audio frame that is most resistant to the MP3 attack. In fact, the proposed algorithm selects Φ_j such that when embedded as the watermark vector and the MP3 compression-decompression attack is applied, the watermark recovery algorithm finds the correct watermark sign. This idea bears some similarity to the analysis-by-synthesis framework in speech coding in the codebook search for the best excitation.

The proposed algorithm works as follows: For each new audio frame it first selects the atom with the smallest dot product with the audio frame $X_i^T . \Phi_j$ (e.g., most orthogonal), and uses it for initial embedding. A super-frame (SF) is constructed which has the watermarked frame in the middle with the preceding 60ms and the following 60ms concatenated to it. This SF goes through the MP3 compression-decompression attack then the watermarked frame in the middle, which is now distorted, is extracted. The watermark recovery algorithm is applied. If the sign is correct the next audio frame is processed. If not, an exhaustive search over the L atoms is done, whereby each atom is embedded, the SF is created and the MP3 attack is applied.

The algorithm selects the atom which produces the correct sign and the highest ratio R_k between the largest correct sign and the largest wrong sign. If none of the L atoms produced a solution, the search is repeated once with an increase in embedding strength $\alpha_i' = 1.5 * \alpha_i$.

The algorithm continues until all frames have been watermarked. The algorithm steps are shown in table 2 below.

4.1 Adaptive Watermark Atom Selection Algorithm Steps

Input: Frame X_i, dictionary $\psi = [\Omega\ \Phi\ ]$

for i=1:I (frames)

Calculate α_i as in (15)

$\Phi = [\Phi_1\ \Phi_2\ \Phi_3 .... \Phi_j\ ....\Phi_{J-1}\ \Phi_J\ ]$ (Φ_j column vector atoms)

Find the column vector Φ_j for which $X_i^T . \Phi_j$ is minimum

Embed: $X_i^w = X_i + \alpha_i * \beta_k * \Phi_j$ ($\beta_k = \pm 1 = \boldsymbol{w_i}$)

Create super-frame $X_i^s = [X_{i-m}\ .... X_i^w\ .... X_{i-m}\ ]$

Apply the MP3 compression on X_i^s

Apply MP3 decompression and extract the watermarked frame: $X_i^{w'}$

Solve the BPDN to get the watermark sparse vector β

Find the largest element in β

$$l = argmax(abs(\beta))$$
$$w^1 = sign(\beta(l))$$

$if\ w^1 = \boldsymbol{w_i}$ Go to 1

else {

for j=1:L (Atoms in Φ)

for each Φ_j apply steps from (5-9), store the sign w_j^1 and the ratio R_j

Exclude the ones with wrong sign, sort the ones with the correct sign and select Φ_j with the largest R_j

End loop over j. if an atom is found Go to 1

else{

Do steps (11 to 14) with $\alpha_i' = 1.5 * \alpha_i$

end

end

end

5 Related Work in CS Watermarking

In 2007 Shiekh and Baraniuk [17] proposed a transform domain watermarking model based on compressed sensing for image watermarking assuming the host signal to be sparse in the DCT domain and the watermark was a randomly spread binary sequence. The work proposed here, in contrast to their work, assumes that the watermark is sparse and that the host can be sparsely coded by an overcomplete dictionary. In 2009 Tagliasacchi et. al. [18, 19] proposed a hash based tampering detection algorithm for audio and image data. If the tampering is sparse enough, it can be localized by solving the CS decoding problem. Their work is more oriented towards content authentication while the work here is focusing on robust watermarking. In [20, 21], the author of this paper has proposed a compressed sensing robust watermarking technique and compared it to the technique in [17]. In [20, 21], the watermark random vector was fixed and not adaptive. Thus, robustness of watermark was obtained however, with relatively low embedding rate that did not exceed 11bps for 64kbps MP3.

6 Experimental Results

The proposed algorithm was tested on parts of 3 MP3 songs used in [1, 20, 21], with 60 seconds from each, and the averaged results are shown here. The original MP3 are 128kbps mono, sampled at 44,100Hz. The MP3 attacks on the watermarked audio are 64kbps and 128kbps, and the payloads used are 689bps, 1378bps and 2756bps corresponding to audio frames of length 64, 32 and 16 samples respectively. The signal part of the dictionary is a concatenation of three inverse transforms each is M×M, namely, discrete cosine transform, Walsh-Hadamard transform and a Karhunen-Loeve transform trained on 60 seconds drawn from the 3 songs. Since the audio files are originally in MP3 128kbps format, each file is converted to the “WAV” format to do the watermark embedding, then the MP3 attack is applied either with a 128kbps or 64kbps compression. In the latter case, we get half the number of original samples of the 128kbps MP3 original file and an interpolation of order 2 is used to get the same number of samples again.

Figures 2, 3 and 4 summarize the results for the three payloads tested in the 64kbps MP3 attack, where the "No MP3" and "MP3" attack cases are with the chosen watermark atom that is most orthogonal to the audio frame. The "Proposed" is with the adaptive atom selection algorithm. Figures 5, 6 and 7 correspond to the 128kbps MP3 attack for the three payloads tested. Tables 2 and 3 summarize the best obtained results. The results show that high payloads of 689bps, 1378bps and 2756bps with acceptable %BER are obtained using the proposed algorithm with the 128kbps and 64kbps MP3 attacks. Without the proposed algorithm, the %Success is around (50%-65%), in the 64kbps, and (55%-80%) in the 128kbps MP3 attacks respectively, and gets worse with increasing L since more atoms are competing in the decoding stage. This increase in L becomes an advantage for the proposed algorithm since more L gives the algorithm a larger search space to find the best atom which is evident from figures 2 to 7 where the %Success approaches 100% with increasing *L*. It is to be noted that the proposed algorithm pushes the complexity to the encoder side due to the best atom search phase. The algorithm complexity depends on the %success rate of the baseline watermark, and also depends on the number of atoms in the dictionary L, and on the complexity of the BPDN algorithm used.

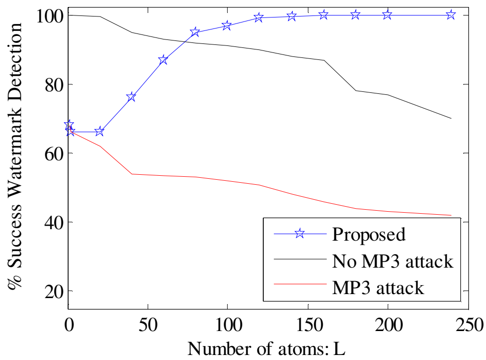

Fig. 2. M=64 (Payload 689bps), 64kbps MP3

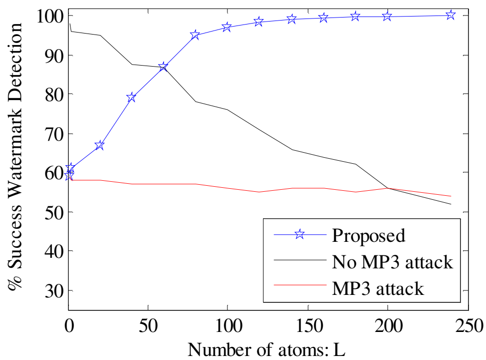

Fig. 3. M=32 (Payload 1378bps), 64kbps MP3

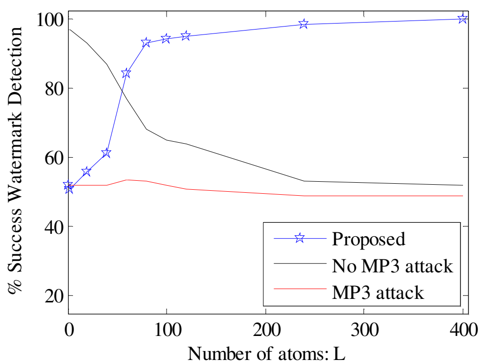

Fig. 4. M=16 (Payload 2756bps), 64kbps MP3

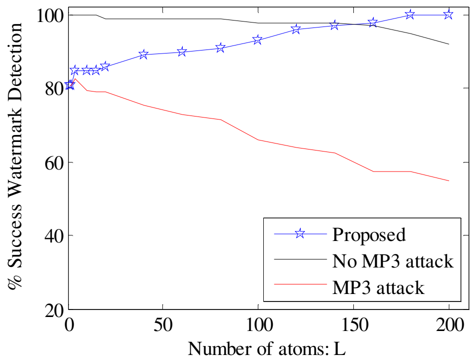

Fig. 5. M=64 (Payload 689bps), 128kbps MP3

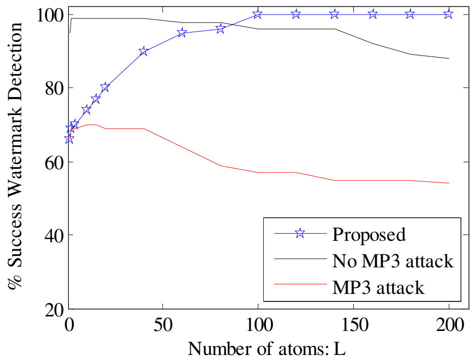

Fig. 6. M=32 (Payload 1378bps), 128kbps MP3

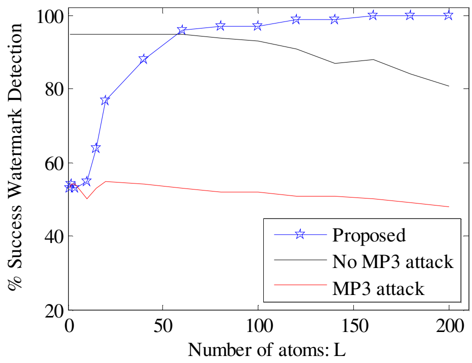

Fig. 7. M=16 (Payload 2756bps), 128kbps MP3

Table 2. Proposed Algorithm Results with 64kbps MP3 attack

MP3 Quality	**SNR**	**%BER**	**Frame length**	**Payload**	**#Atoms**
64kbps	30dB	0.3%	64 samples	689 bps	240
64kbps	30dB	0.5%	32 samples	1378 bps	240
64kbps	27dB	1.0%	16 samples	2756 bps	400

Table 3. Proposed Algorithm Results with 128kbps MP3 attack

MP3 Quality	**SNR**	**%BER**	**Frame length**	**Payload**	**#Atoms**
128kbps	30dB	0.1%	64 samples	689 bps	200
128kbps	30dB	0.3%	32 samples	1378 bps	200
128kbps	28dB	0.5%	16 samples	2756 bps	200

7 Conclusion and Future Work

A compressed sensing based, sparse coding watermarking framework is proposed where the watermark is a sparse vector with one non-zero element that takes the required sign of the encoded bit. The sparse coding technique uses an overcomplete dictionary that is a concatenation of a signal dictionary and a random Gaussian dictionary. The random dictionary is used to map the sparse watermark vector to a random watermark embedding vector. An adaptive watermarking algorithm is proposed where an atom from the random dictionary is selected for each new audio frame such that the (BPDN) decoding is robust to a single MP3 compression-decompression attack. The proposed technique is tested on three songs where the watermarked signals went through 64kbps and 128kbps MP3 attacks and the signal to embedding noise ratio (SNR) was kept above 27dB in all cases, and the watermarking effect was negligible.

High payloads (689, 1378 and 2756 bps) are achieved with acceptable quality and bit error rates. It is to be noted that the proposed algorithm is robust only to a single MP3 compression-decompression attack. Experimental results showed that repeating the attack degrades the watermark detection significantly. Also, the proposed algorithm was not tested under other attacks such as additive noise, synchronization, collusion, cropping/swapping and time-scale modification attacks.

Future work includes making the proposed algorithm robust to multiple MP3 compression-decompression attacks, and to other types of attacks. One direction is to synthesize a watermark random vector for each frame using a multi-objective optimization algorithm, where the watermark random vector is learned for each new frame. Another direction is to have atoms with different spectral characteristics, and choose the one that is least affected by the spectral distortion suffered by the MP3 attack. Finally, the proposed framework can be directly applied to Image and video watermarking with the emphasis on the robustness to JPEG and different video compression standards.

References

1. Noriega, R.M., Nakano, M., Kurkoski, B., Yamaguchi, K.: High Payload Audio Watermarking: toward Channel Characterization of MP3 Compression. Journal of Information Hiding and Multimedia Signal Processing 2(2), 91–107 (2011)
2. Vivekananda, B.K., Indranil, S., Abhijit, D.: An Audio Watermarking Scheme using Singular Value Decomposition and Dither-Modulation Quantization. Multimedia Tools and Applications Journal 52(2-3), 369–383 (2011)
3. Dhavale, S.V., Deodhar, R.S., Patnaik, L.M.: Walsh Hadamard Transform Based Blind Watermarking for Digital Audio Copyright Protection. In: Das, V.V., Thankachan, N. (eds.) CIIT 2011. CCIS, vol. 250, pp. 469–475. Springer, Heidelberg (2011)
4. Yang, H., Bao, D., Wang, X., Niu, P.: A Robust Content Based Audio Watermarking using UDWT and Invariant Histogram. Multimedia Tools and Applications Journal (November 2010)
5. El Hamdouni N., Adib A., Labri S., Torki M.: A Blind Digital Audio Watermarking Scheme Based on EMD and UISA Techniques. Multimedia Tools and Applications Journal (January 2012)
6. Tewari, T.K., Saxena, V., Gupta, J.P.: Audio Watermarking: Current State of Art and Future Objectives. International Journal of Digital Content Technology and Applications 5(7), 306–313 (2011)
7. Datta, K., Gupta, I.S.: Partial Encryption and Watermarking Scheme for Audio Files with Controlled Degradation of Quality. Multimedia Tools and Applications, Journal (2012)
8. Ercelebi, E., Batakci, L.: Audio watermarking Scheme Based on Embedding Strategy in Low Frequency Components with a Binary Image. Digital Signal Processing 19(2), 265–277 (2009)
9. Orsdemir, A., Altun, H.O., Sharma, G., Bocko, M.F.: On the Security and Robustness of Encryption via Compressed Sensing. In: IEEE Military Communication Conference MILCOM 2008, pp. 1–7 (2008)
10. Candès, E., Tao, T.: Decoding by Linear Programming. IEEE Transaction on Information Theory 51(12), 4203–4215 (2005)
11. Candès, E., Randall, P.: Highly Robust Error Correction by Convex Programming. IEEE Transaction on Information Theory 54(7) (2006)
12. Laska, J., Davenport, M., Baraniuk, R.: Exact Signal Recovery from Sparsely Corrupted Measurements through the Pursuit of Justice. In: Asilomar Conf. on Signals, Systems, and Computers, Pacific Grove, California (2009)
13. L1 magic: `http://users.ece.gatech.edu/~justin/l1magic/`
14. Gemmeke, J.F., Virtanen, T., Hurmalainen, A.: Examplar Based Sparse Representations for Noise Robust Automatic Speech Recognition. IEEE Trans. Audio, Speech and Language Processing 19(9), 2067–2080 (2011)
15. Sprechman, P., Sapiro, G.: Dictionary Learning and Sparse Coding for Unsupervised Clustering. In: ICASSP 2010, pp. 2042–2045 (2010)
16. Wright, J., Yi, M., Mairal, J., Sapiro, G., Huang, T., Yan, S.: Sparse Representation for Computer Vision and Pattern Recognition. Proc. of IEEE 98(6), 1031–1044 (2010)
17. Sheikh, M., Baraniuk, R.: Blind Error-Free Detection of Transform-Domain Watermarks. In: IEEE Int. Conf. on Image Processing (ICIP), San Antonio, Texas, vol. 5, pp. V-453–V-456 (September 2007)

18. Tagliasacchi, M., Valenzise, G., Tubaro, S.: Hash-Based Identification of Sparse Image Tampering. IEEE Transactions on Image Processing 18(11), 2491–2504 (2009)
19. Valenzise, G., Prandi, G., Tagliasacchi, M., Sarti, A.: Identification of Sparse Audio Tampering using Distributed Source Coding and Compressive Sensing Techniques. Eurasip Journal on Image and Video Processing 2009, 1–13 (2009)
20. Fakhr, M.W.: Robust Watermarking using Compressed Sensing Framework with Application to MP3. International Journal of Multimedia and its Applications, IJMA 4(6), 27–43 (2012)
21. Fakhr, M.W.: Sparse Watermark Embedding and Recovery using Compressed Sensing Framework for Audio Signals. In: International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, Sanya, China, pp. 535–539 (2012)