PLOS ONE

Collusion-Resistant Audio Fingerprinting System in the Modulated Complex Lapped Transform Domain

Jose Juan Garcia-Hernandez[1]*, **Claudia Feregrino-Uribe**[2], **Rene Cumplido**[2]

[1] Laboratorio de Tecnologias de Informacion, CINVESTAV-IPN, Tamaulipas, Mexico, [2] Instituto Nacional de Astrofisica, Optica y Electronica, Puebla, Mexico

Abstract

Collusion-resistant fingerprinting paradigm seems to be a practical solution to the piracy problem as it allows media owners to detect any unauthorized copy and trace it back to the dishonest users. Despite the billionaire losses in the music industry, most of the collusion-resistant fingerprinting systems are devoted to digital images and very few to audio signals. In this paper, state-of-the-art collusion-resistant fingerprinting ideas are extended to audio signals and the corresponding parameters and operation conditions are proposed. Moreover, in order to carry out fingerprint detection using just a fraction of the pirate audio clip, block-based embedding and its corresponding detector is proposed. Extensive simulations show the robustness of the proposed system against average collusion attack. Moreover, by using an efficient Fast Fourier Transform core and standard computer machines it is shown that the proposed system is suitable for real-world scenarios.

Citation: Garcia-Hernandez JJ, Feregrino-Uribe C, Cumplido R (2013) Collusion-Resistant Audio Fingerprinting System in the Modulated Complex Lapped Transform Domain. PLoS ONE 8(6): e65985. doi:10.1371/journal.pone.0065985

Editor: Randen Lee Patterson, UC Davis School of Medicine, United States of America

Received March 20, 2013; **Accepted** April 29, 2013; **Published** June 6, 2013

Funding: The work was supported by CONACyT (www.conacyt.gob.mx) under grant CB-2010-1-50910. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: jjuan@tamps.cinvestav.mx

Introduction

In the Information Technology era, expansion of the Internet service together with the rapid advance of high capacity storage systems facilitated the fast and perfect copy of digital content. However, at the same time the use of these technologies causes serious problems, such as unauthorized copying and distribution of digital materials, [1]. Conventional cryptography systems encrypt digital data during its transmission and permit only authorized person to decrypt the encrypted data, nevertheless, once such data are decrypted they are totally vulnerable to illegal copying and distribution. One possible solution to this problem is the fingerprinting paradigm, where, a unique signature (which identifies to the legal user) known as a digital fingerprint is hidden using a watermarking technique into the content previously to distribution. Watermarking has several applications such as: ownership proof [2], secret communications [3], bio-security [4,5], etc. Digital fingerprinting, which is also a watermarking application, has the capacity of identifying illegal users by extracting the fingerprint of a suspicious copy. A typical attack in fingerprinting systems is the collusion attack, where a group of users combine their copies in order to remove the original fingerprint. If a sufficient number of copies are combined, the noise produced by the collusion attack can disable/confuse the fingerprint detector and prevent the content owner from identifying the illegal users. Although several linear and nonlinear operations can be utilized for a collusion attack, it has been shown that the worst one is the linear averaging [6]. Therefore, it is necessary to design collusion-resistant fingerprints that can identify the greatest number of colluders involved in a pirate copy.

Collusion-resistant fingerprint codes have been proposed as a solution to the collusion attack [6–9]. Theoretical results for collusion-resistant fingerprint codes have shown interesting properties against collusion attacks, however, in practical sceneries their performance needs further research as these can be sensible to other kinds of attacks, [9].

On the other hand, Spread Spectrum (SS) modulation is a watermarking technique that has shown to be remarkably robust to several attacks, collusion included, [10–13]; therefore, it has been frequently utilized in fingerprinting systems [14–16]. The main drawback of fingerprinting schemes based in spread spectrum modulation is their high computational complexity as the number of correlations performed is proportional to the number of possible users. A users grouping approach was proposed in [17]. That idea is based on the consideration of colluders being more likely to have similar geographical area and interests with each other. Users are grouped according to common conditions between them. When a suspicious copy is identified, the first search is about the group IDs and then for user' IDs. The computational complexity is reduced, due to the colluders search is carried out in a tree fashion, [17].

In [18], the use of PN-modulated Discrete Cosine Transform (DCT) basis as fingerprints for digital images is proposed. The DCT operation can be represented as a multiplication between the input vector and one matrix conformed by the DCT basis. That multiplication is equivalent to correlations between the input vector and each column of the DCT matrix. Therefore, a fast DCT algorithm reduces the computational complexity of correlations needed in the IDs detection to the logarithmic scale. The fingerprint is formed by the sum of two PN-modulated DCT basis, one for the group ID and the other for the user ID. In the detection stage, firstly the groups to which colluders belong are detected, and then colluders are detected for each of them. In [19] the interference due to colluder fingerprints is removed and performance of the system in [18] is improved drastically.

The music piracy produces large monetary losses around the world [20,21], therefore, a tool that helps to mitigate the music piracy is mandatory. However, most of the reported collusion-resistant fingerprinting schemes are devoted to digital images [7,8,16,18,19,22,23] and only very few are validated with audio signals [24]. This paper is about collusion-resistant audio fingerprinting. A collusion-resistant audio fingerprinting system based on some of the ideas developed for digital images in [18,19] is proposed. Instead of using the full signal as [18,19] a block-based fingerprint embedding strategy is followed and the corresponding detector is derived. In this paper, the Modulated Complex Lapped Transform (MCLT) domain is utilized as fingerprint channel due to no block-artifact property in audio watermarking systems [11,25–28].

Related Work

Work reported in [24] claims to be able to detect 80 colluders in a pirate audio clip. However, that system seems to be not suitable for real world scenarios. One weakness is about construction of component vector which is carried out using two audio channels in the Fourier domain. Due to a trigonometric function (inverse tangent) is involved in this stage, a simple attack like sign inversion in one audio channel prevents the correct ID detection as multiplication by -1 is equivalent to a phase shift by π radians. Moreover, if an audio channel is scaled (volume gain) the relation between both channels will be different to the original and the detection will fail. Detector performance after lossy compression, such as MP3 coding or Advanced Audio Coding (AAC) is not reported. It is important to mention that sign inversion, volume gain and lossy compression are real world scenarios. Neither viability of the system nor the number of users is reported. To the best of our knowledge, it is the only work about collusion-attack resistant fingerprinting in audio signals. Although there are several works about audio fingerprinting in the literature, almost all of them do not consider the collusion-attack [29–33].

On the other hand, most of the works about collusion-attack resistant fingerprinting systems are devoted to digital images mainly based in Spread Spectrum techniques. The main drawback of fingerprinting schemes based in SS techniques is their high computational complexity as it is discussed in the Introduction. In order to achieve lower computational complexity than SS-fingerprinting schemes for digital images, in [18] it is proposed to utilize PN-modulated orthogonal sequences. These orthogonal sequences can be obtained from DCT or DFT basis. In the DCT case, each user is related to a DCT matrix column which is defined in equation (26). Therefore, the SS sequence for the ith user becomes:

$$\mathbf{w_i} = \beta \cdot \mathbf{pn(s)} \otimes \mathbf{DCT(i)} \tag{1}$$

where β is a robustness factor, $\mathbf{pn(s)}$ is a PN sequence generated using an initial value $\mathbf{s}$, $\mathbf{s}$ is a secret key, $\mathbf{DCT(i)}$ is the ith DCT matrix column and $\otimes$ is the element-wise multiplication. The sequence $\mathbf{w_i}$ is embedded into the frequency components of a digital medium, in this paper audio signals. As an example, Figure 1 shows the SS sequence, $\mathbf{w_{1890}}$, for the user 1890 of 2048 and $\beta = 1$.

Unlike other watermarking applications, in the fingerprinting paradigm, detection is usually carried out in a non-blind fashion [34], i.e. the original signal is available to the detector. Under that condition, after subtracting the original sequence from the pirate copy the sequence $\tilde{\mathbf{w}}_\mathbf{i}$ is obtained. In order to carry out the detection the sequence $\tilde{\mathbf{d}}$ is obtained by applying the Inverse DCT to $\tilde{\mathbf{w}}_\mathbf{i}$ which is demodulated by the PN sequence $\mathbf{pn(s)}$ as follows:

$$\tilde{\mathbf{d}} = \mathrm{InverseDCT}(\mathbf{pn(s)} \otimes \tilde{\mathbf{w}}_\mathbf{i}) \tag{2}$$

where InverseDCT(.) denotes a fast inverse discrete cosine transform algorithm as described in the Materials and Methods section. Figure 2 shows the corresponding $\tilde{\mathbf{d}}$ for detection of the user 1890 out of 2048, as exemplified above.

From Figure 2, it is possible to observe that a threshold is necessary in order to determine the user under a statistical point of view. If $\tilde{\mathbf{d}}$ is supposed to be $N(0,\sigma^2)$ except for a fingerprinted component $\tilde{d}_k$, it is possible to calculate a threshold T according to the probability of false detection P_{fa} [18] as follows:

$$P_{fa} \leq \frac{1}{2}\mathrm{erfc}(\frac{T}{\sqrt{2\sigma^2}}) \tag{3}$$

where erfc($\cdot$) is the complementary error function defined as:

$$\mathrm{erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^\infty \exp(-u^2)\mathrm{d}u \tag{4}$$

Therefore, the threshold is given by the expression (5),

$$T = \sqrt{2\sigma^2}\mathrm{erfc}^{-1}(2P_{fa}) \tag{5}$$

where erfc^{-1}(.) stands for the inverse complementary error function.

Grouping a set of users has been proposed in the literature as a solution to high computational costs [8,16]. The assumption behind this proposal is that users who have a similar background and region are more likely to collude each other. In [18] the idea of introducing dependency between two SS sequences by exploiting the property of quasi-orthogonality of PN sequences is proposed. Thus, the fingerprint is integrated by two spread spectrum sequences related to a group ID $\mathbf{w_{i_g}}$ and an user ID $\mathbf{w_{i_u}}$ as follows:

$$\mathbf{w_{i_g}} = \beta_g \cdot \mathbf{pn(s)} \otimes \mathbf{DCT(i_g)} \tag{6}$$

where β_g is the robustness factor for groups, $\mathbf{pn(s)}$ is a PN sequence generated with the secret key $\mathbf{s}$, $\mathbf{DCT(i_g)}$ is the ith basis vector that identified to the ith group and

$$\mathbf{w_{i_u}} = \beta_u \cdot \mathbf{pn(i_g)} \otimes \mathbf{DCT(i_u)} \tag{7}$$

where β_u is the robustness factor for users, $\mathbf{pn(i_g)}$ is a PN sequence corresponding to the ith group, and $\mathbf{DCT(i_u)}$ is the ith basis vector that identified the ith user.

Then, the fingerprint assigned to the jth user of the ith group is conformed by:

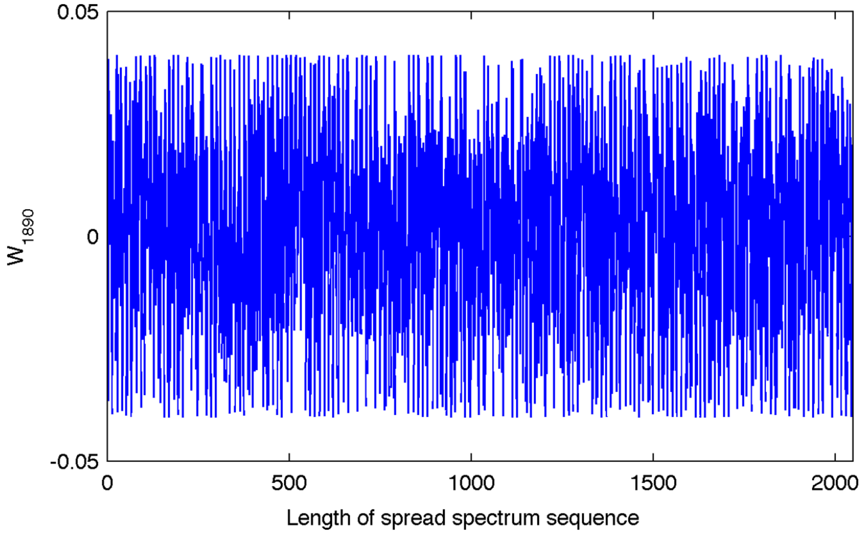

Figure 1. Spread spectrum sequence for the user 1890 of 2048.
doi:10.1371/journal.pone.0065985.g001

$$\mathbf{w_{i,j}} = \mathbf{w_{j_u}} + \mathbf{w_{i_g}} \tag{8}$$

The energy of the fingerprint is represented by

$$\beta^2 = \beta_g^2 + \beta_u^2. \tag{9}$$

From equation (8) it is easy to see that a couple of detectors is required, one for the spread spectrum sequence related to group

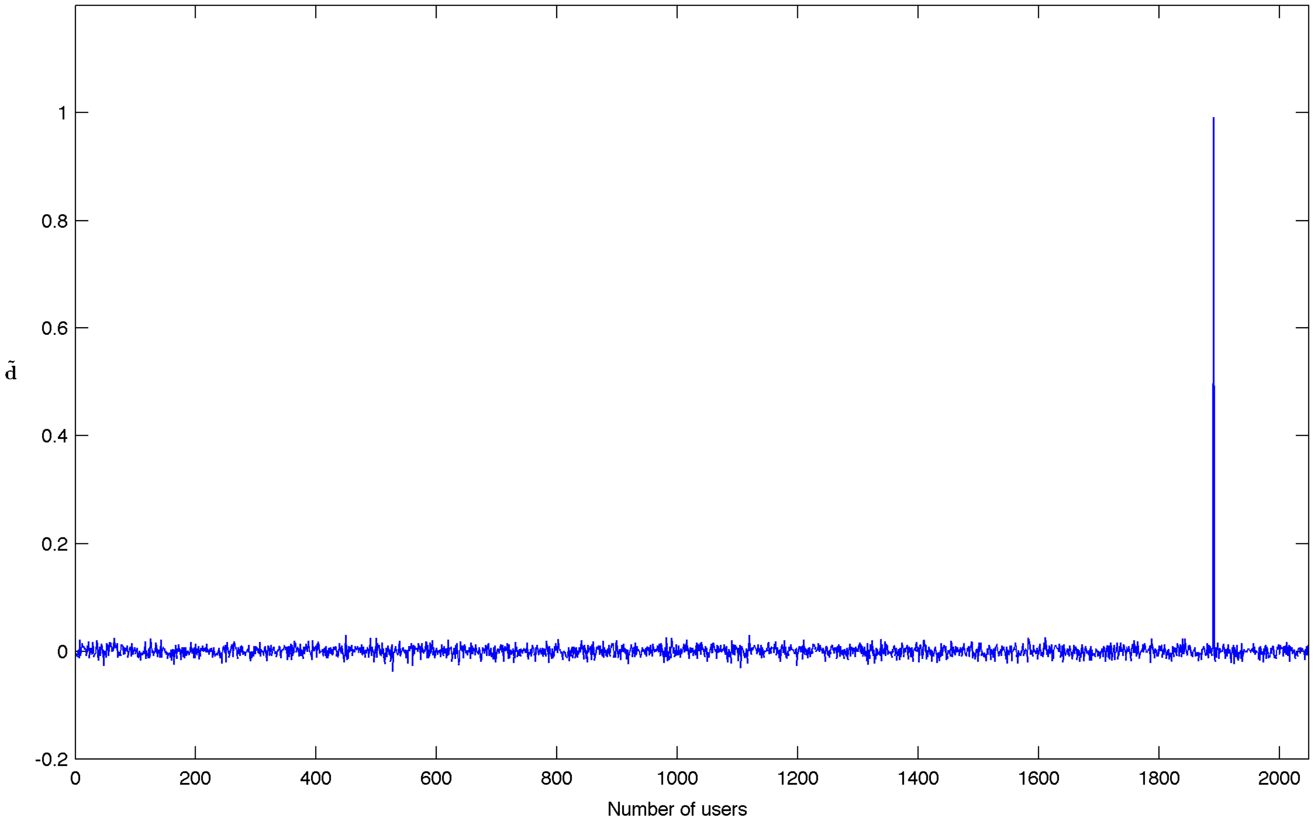

Figure 2. $\tilde{\mathrm{d}}$ for detection of the user 1890 of 2048.
doi:10.1371/journal.pone.0065985.g002

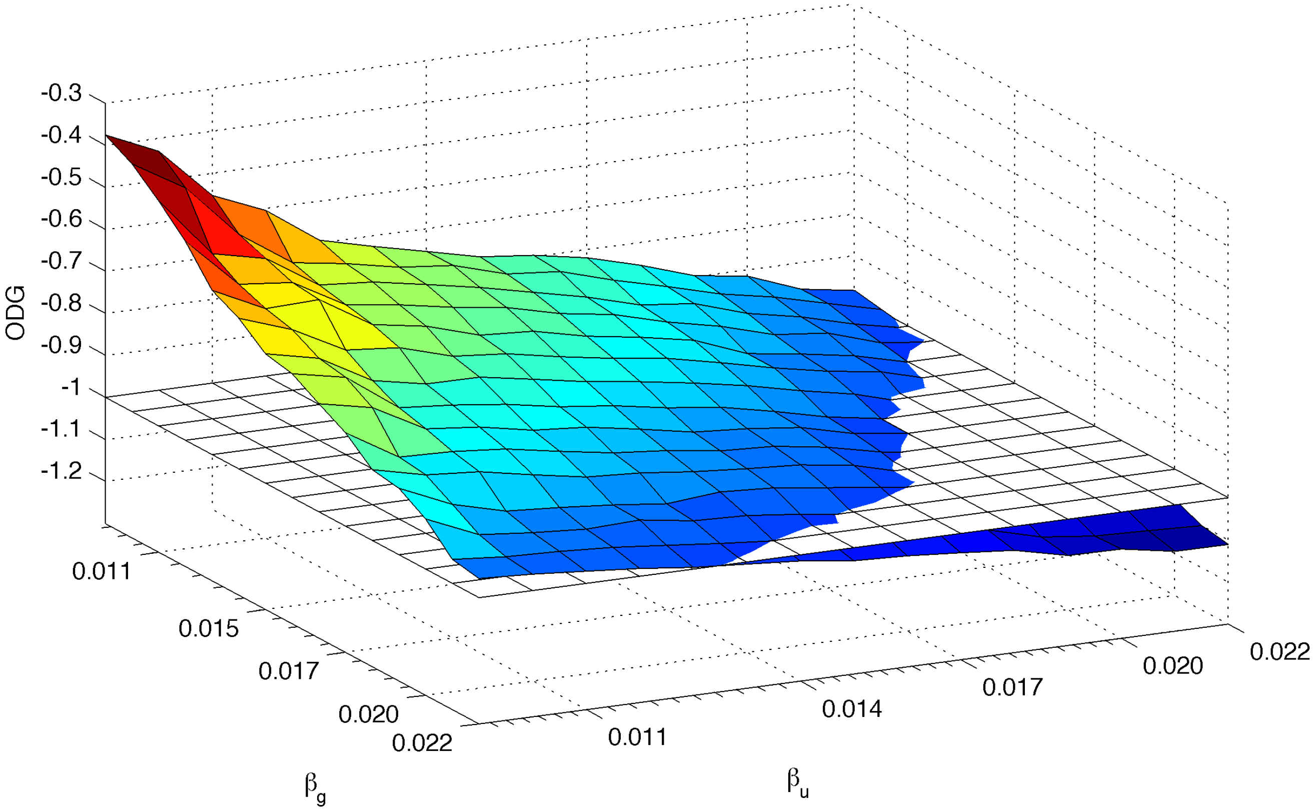

Figure 3. ODG region for fingerprinted audio clips.
doi:10.1371/journal.pone.0065985.g003

ID $\mathbf{w_{i_g}}$ and other for the user ID $\mathbf{w_{j_u}}$. These detectors are derived from equation (2) as follows:

For group ID detection:

$$\widetilde{\mathbf{d}}_{\mathbf{g}} = \text{InverseDCT}(\mathbf{pn}(\mathbf{s}) \otimes \widetilde{\mathbf{w}}_{\mathbf{i,j}}) \tag{10}$$

and for user ID detection:

$$\widetilde{\mathbf{d}}_{\mathbf{u}} = \text{InverseDCT}(\mathbf{pn}(\mathbf{i_g}) \otimes \widetilde{\mathbf{w}}_{\mathbf{i,j}}) \tag{11}$$

with thresholds, T_g and T_u, derived according to equation (5) as follows:

$$T_g = \sqrt{2\sigma_g^2}\text{erfc}^{-1}(2P_{fag}) \tag{12}$$

$$T_u = \sqrt{2\sigma_u^2}\text{erfc}^{-1}(2P_{fau}) \tag{13}$$

where P_{fa_g} and P_{fa_u} are given false positive probabilities for the group and user ID detection procedures respectively. σ_g^2 and σ_u^2 are the variance of the group and user ID detection sequences respectively.

The outline of the paper is as follows: First, experimental results and discussion are offered. In the Materials and Methods section, we recall the Modulated Complex Lapped Transform and Discrete Cosine Transform and their fast algorithms used in this work. In the Fingerprinting System section steps are described comprising audio fingerprinting method by DCT modulation in the MCLT domain. Finally, conclusions are offered.

Results and Discussion

The proposed audio fingerprinting system is evaluated under averaging collusion attacks. Through abundant experiments; the operation parameters are determinate too. For experimentations, CD-quality audio files are utilized from a set of 1000 popular music recordings. The probability of false detection is set to 10^{-6} for both group (P_{fa_g}) and user ID detection (P_{fa_u}) procedures, as this is a typical value in audio spread spectrum-based watermarking systems [11].

Fingerprint Robustness Determination

In order to determinate the adequate β_g and β_u values in equations (6) and (7); an audio transparency metric is utilized, the Objective Difference Grade (ODG) [35]. An ODG value between 0 and -1 is considered a good perceptual transparency [35]. In the experiment, several audio clips are fingerprinted with different combinations of β_g and β_u values and the ODG metric for each combination is obtained. The limit for practical β_g and β_u values is determinate for ODG ≥ -1 as the bigger the fingerprint energy β the lower the ODG value. In the spread spectrum watermarking, it is well known that the bigger the watermark energy the bigger the watermark robustness [2]. Therefore, it is interesting to investigate the biggest fingerprint energy values that maintain a good

Table 1. ODG values for combinations of β_g and β_u values.

β_g/β_u	**0.0008**	**0.0009**	**0.0010**	**0.0011**	**0.0012**	**0.0013**	**0.0014**	**0.0015**	**0.0016**	**0.0017**	**0.0018**	**0.0019**	**0.0020**	**0.0021**	**0.0022**
0.0008	−0.38	−0.41	−0.47	−0.53	−0.61	−0.64	−0.69	−0.71	−0.74	−0.78	−0.83	−0.84	−0.88	−0.94	−0.95
0.0009	−0.43	−0.49	−0.61	−0.65	−0.67	−0.69	−0.72	−0.73	−0.77	−0.85	−0.86	−0.90	−0.94	−0.95	−0.96
0.0010	−0.55	−0.61	−0.64	−0.66	−0.67	−0.75	−0.79	−0.82	−0.84	−0.86	−0.87	−0.93	−0.95	−0.96	−0.97
0.0011	−0.61	−0.70	−0.72	−0.73	−0.74	−0.76	−0.82	−0.83	−0.85	−0.87	−0.89	−0.94	−0.96	−0.97	−0.98
0.0012	−0.69	−0.72	−0.73	−0.75	−0.76	−0.80	−0.82	−0.85	−0.87	−0.89	−0.92	−0.95	−0.95	−0.98	−1.00
0.0013	−0.73	−0.74	−0.75	−0.77	−0.80	−0.81	−0.84	−0.87	−0.88	−0.91	−0.92	−0.96	−0.97	−0.99	−1.01
0.0014	−0.76	−0.75	−0.77	−0.79	−0.81	−0.84	−0.86	−0.88	−0.91	−0.92	−0.94	−0.95	−0.97	−1.00	−1.01
0.0015	−0.78	−0.79	−0.80	−0.82	−0.84	−0.86	−0.88	−0.91	−0.93	−0.94	−0.95	−0.96	−0.99	−1.02	−1.03
0.0016	−0.80	−0.80	−0.83	−0.84	−0.86	−0.87	−0.90	−0.92	−0.94	−0.95	−0.96	−0.98	−1.01	−1.01	−1.04
0.0017	−0.82	−0.84	−0.87	−0.87	−0.88	−0.90	−0.92	−0.95	−0.95	−0.97	−0.97	−0.99	−1.02	−1.03	−1.05
0.0018	−0.86	−0.87	−0.88	−0.90	−0.93	−0.93	−0.95	−0.98	−0.98	−0.99	−1.01	−1.01	−1.04	−1.04	−1.06
0.0019	−0.90	−0.91	−0.92	−0.94	−0.95	−0.95	−0.97	−1.00	−1.00	−1.01	−1.02	−1.02	−1.04	−1.08	−1.09
0.0020	−0.92	−0.93	−0.94	−0.95	−0.97	−0.97	−1.01	−1.02	−1.04	−1.05	−1.06	−1.05	−1.06	−1.09	−1.09
0.0021	−0.96	−0.96	−0.98	−0.99	−1.00	−1.01	−1.02	−1.03	−1.05	−1.06	−1.07	−1.06	−1.07	−1.10	−1.11
0.0022	−0.98	−0.99	−1.00	−1.00	−1.02	−1.03	−1.05	−1.05	−1.07	−1.06	−1.07	−1.06	−1.07	−1.10	−1.11

doi:10.1371/journal.pone.0065985.t001

perceptual transparency. Figure 3 shows the ODG region for an average of 10 sets of 225 fingerprinted audio clips.

In order to provide a reference for practical β_g and β_u values, Table 1 shows the corresponding ODG values for combinations of β_g and β_u values.

Figure 4 shows the collusion-attack robustness for the combination with the higher acceptable β_g and the combination with the higher acceptable β_u, with colluders from the same group and block length, $M = 2048$. It is interesting to note that the detection performance is better when the robustness factor for users is greater than the robustness factor for groups, i.e. $\beta_u > \beta_g$, moreover, according to [17], users in a group are more likely to collude with each other, therefore, the number of group IDs involved in a pirate copy must be smaller than the number of colluder IDs. As a consequence, the energy of the user ID PN-sequence must be higher than the group ID PN-sequence, i.e. $\beta_u > \beta_g$. Under that asseveration, combinations $(\beta_g = 0.0012, \beta_u = 0.0022)$, $(\beta_g = 0.0014, \beta_u = 0.0021)$, $(\beta_g = 0.0015, \beta_u = 0.0020)$ and $(\beta_g = 0.0017, \beta_u = 0.0019)$ seem to be good candidates for fingerprint embedding.

Block Length Influence

From Figure 4 it is possible to observe that in the best case, $(\beta_g = 0.0012, \beta_u = 0.0022)$, the number of detected colluders appears low for practical applications. In order to improve the performance of the proposed system, the influence of the block length, M, is investigated. Collusion-attack robustness is studied for different block lengths and Figure 5 shows the results for such study.

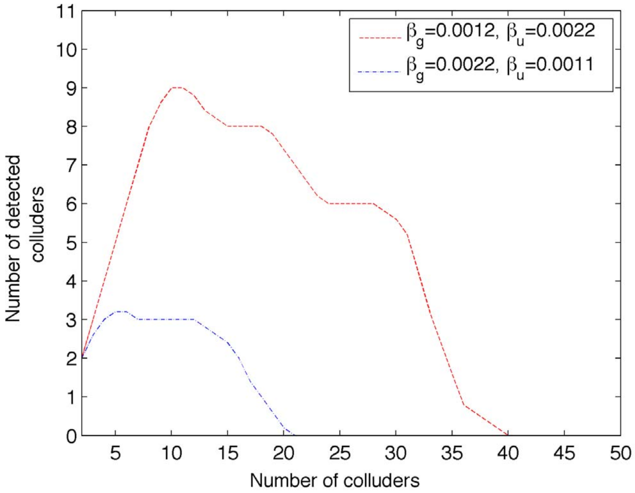

Figure 4. Collusion-attack robustness for $(\beta_g = 0.0012, \beta_u = 0.0022)$ and $(\beta_g = 0.0022, \beta_u = 0.0011)$.
doi:10.1371/journal.pone.0065985.g004

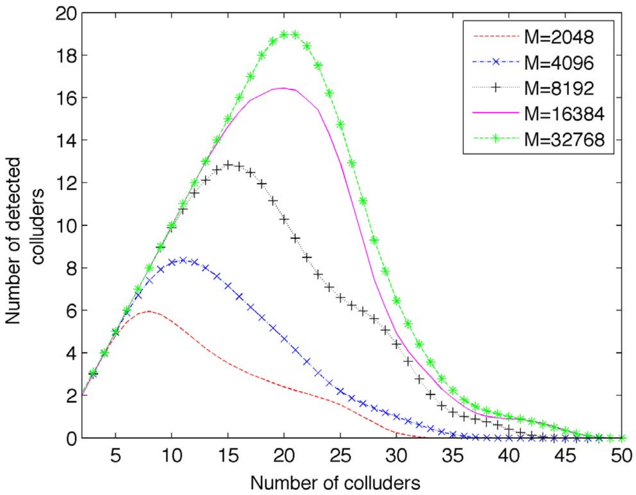

Figure 5. Collusion-attack robustness for several block length, M.
doi:10.1371/journal.pone.0065985.g005

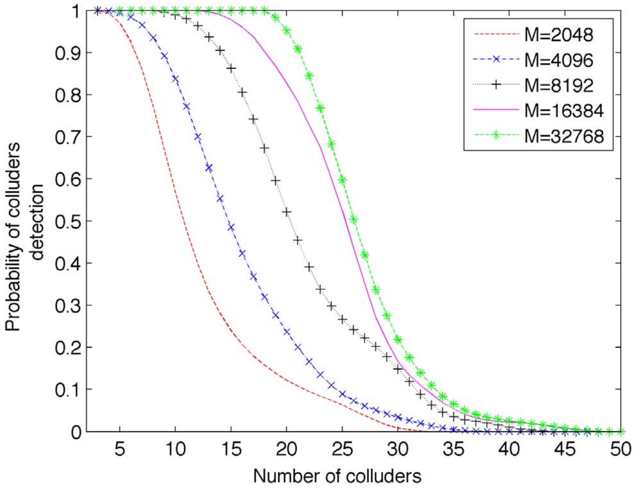

Figure 6. Detection rate of colluders in function of block length, M.
doi:10.1371/journal.pone.0065985.g006

The higher block length the higher collusion-attack robustness. However, for $M=32768$ the increase in performance is not significant in comparison with $M=16384$ as it can be seen in Figure 5. Moreover, the computing resources for computing FFT in the MCLT and DCT transforms can be critical for some platforms when the number of points is larger [36]. Therefore, $M=16384$, seems to be the best option as it is possible to detect more colluders users from the totality of them. Figure 6 shows the detection rate of colluders in function of block length, M. This confirms what mentioned from results in Figure 5.

Implementation Issues

In this subsection, the viability of the proposed system is addressed. It can be interesting to measure the computing time of fingerprint embedding and detection as a function of the block sizes as several block sizes have been studied. Table 2 shows time requirements for several block sizes in real-time terms. It is interesting to point out that the computing complexity increases very slightly when the block size increases in two-power factor.

According to Table 2, an 80 min. music album can be fingerprinted in about 2.37 min. (80/33.68) which could be attractive for on-line music distribution services.

Let N_{g_c} be the number of groups of colluders involved in a pirate copy, it is necessary $N_{g_c}+1$ detection process operations in order to find all of the colluders as one detection process is utilized for group IDs detection and 1 detection process for each detected group in order to identify colluder user IDs. As an example, if an 80 min. music album is pirated by 40 users from 5 groups, the colluders detection in the whole album requires about 15.3 min. $((5+1)*(80/31.36))$ which seems to be a non-prohibitive amount of time for commercial applications. Moreover, if the number of colluders is higher but the number of groups is the same, the computational complexity will be maintained about 15.3 min. as it only depends of number of detected groups.

Audio Clip Requirements for IDs Detection

Due to the nature of the fingerprint insertion process, it is possible to assume that it is not necessary the whole audio clip in the detection process. The IDs detection is carried out by a counter of events that exceed thresholds, therefore, if there are enough events the system achieves its maximum detection capacity. This is expected to happen after a certain number of events and after the behavior of the detector goes stable. In order to validate that claim, the next experiment was carried out: a set of audio clips were fingerprinted with different IDs, and a pirate copy was generated for 2 to 50 colluders; for 1 to 55 seconds of the audio clip ID detection is executed and detected colluders are counted. This experimentation was carried out with 100 different pirate audio clips with $M=16384$ and their results are averaged. Figure 7 shows the detector behavior in function of pirate audio clip duration and number of colluders.

It is interesting to observe that the curve remains without notable changes from 26 seconds to 55 seconds. In other experimentation, using several 30 seconds pirate audio clips, the detector capacity is the same as compared with detection using the whole pirate audio clips, which corroborates the behavior shown in Figure 7. On the other hand, according to Figure 8, for a probability of colluders detection equal to 1, the detector behavior is practically the same for durations longer that 2 seconds.

Lossy Compression Attack

In order to validate the proposed system in a practical scenario, robustness to collusion attack after lossy compression is explored. Advanced Audio Coding (AAC) is used for experimentation as it has shown better performance in perceptual transparency and compression rates terms as compared with MPEG-1 and MPEG-2 Audio Layer 3 [37]. The block length utilized in the experiment is $M=16384$, and the number of audio clips involved is 225. Figure 9 shows the detector performance under collusion attack after AAC compression for several bitrates.

It is possible to see from Figure 9 that the lower AAC bitrate the lower performance. Figure 10 shows the detection probability of colluded attacked audio clip after AAC compression for several bitrates. The detector performance reduces about 12% after high quality lossy compression; which is competitive for real work environments.

Gain and Inverse Attack

It is well known that SS watermarking is strong against gain attack, however, in order to corroborate that claim an experiment

Table 2. Time requirements for several block sizes.

Block Size	Fingerprint embedding (real-time)	Fingerprint detection (real-time)
2048	38.04x	33.24x
4096	36.23x	32.50x
8192	35.65x	32.39x
16384	33.68x	31.36x

doi:10.1371/journal.pone.0065985.t002

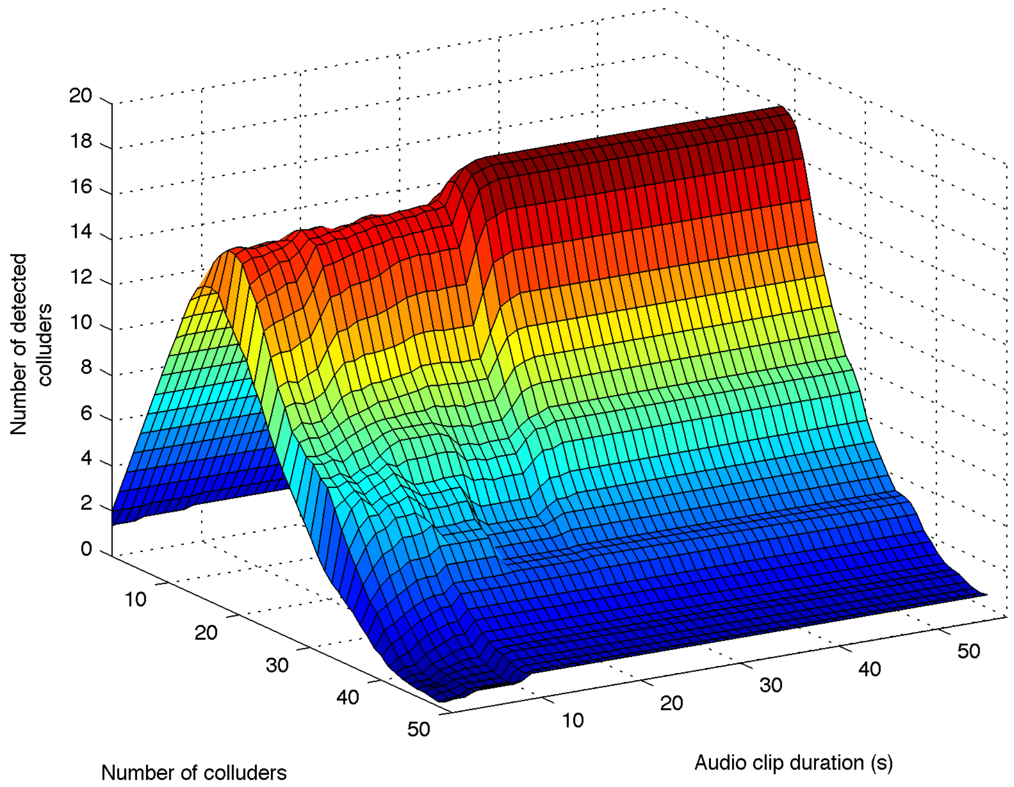

Figure 7. Colluders detection in function of pirate audio clip duration.
doi:10.1371/journal.pone.0065985.g007

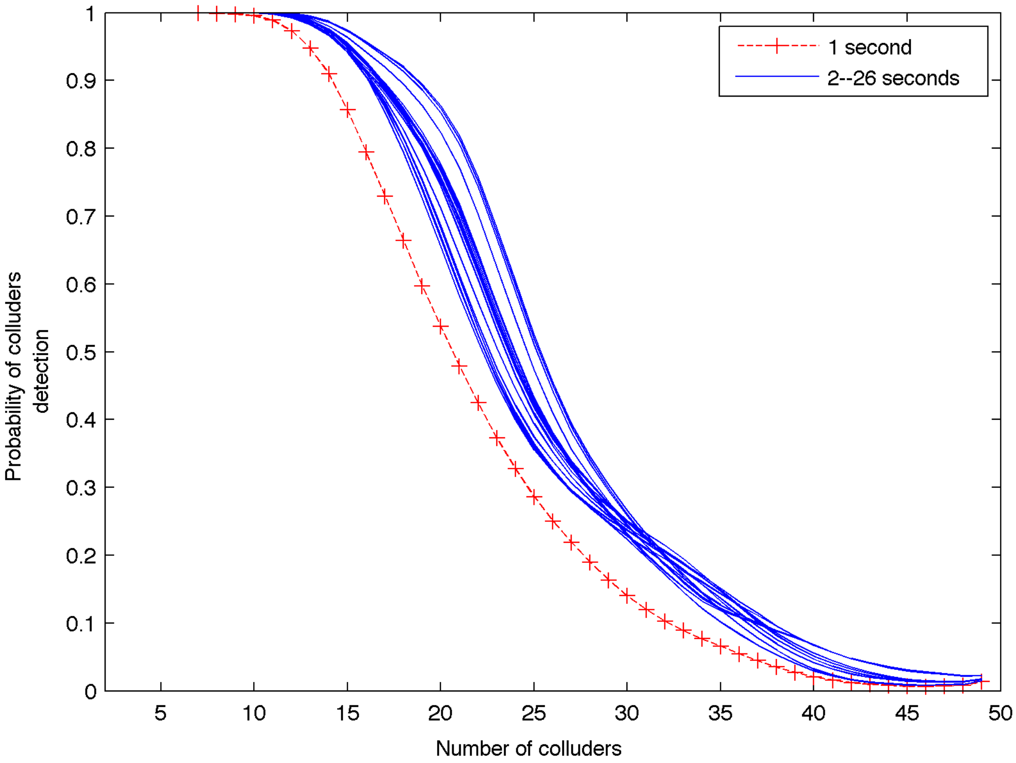

Figure 8. Probability of colluders detection for several pirate audio clip durations.
doi:10.1371/journal.pone.0065985.g008

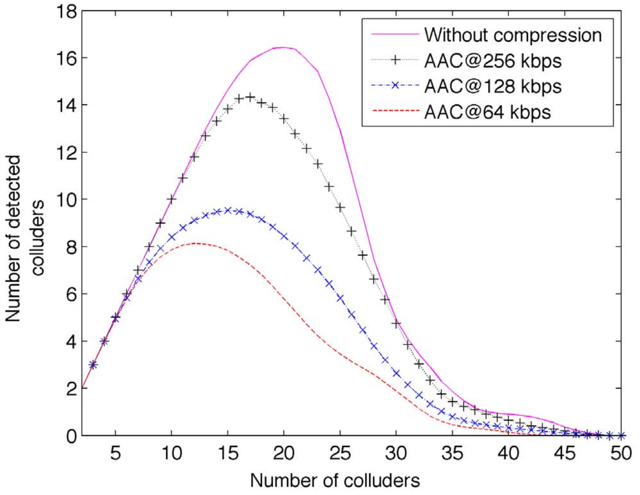

Figure 9. Collusion-attack robustness for several AAC bitrates.
doi:10.1371/journal.pone.0065985.g009

was carried out. After several pirate audio clips were uniformly scaled in the range [0.5,1.5] in 0.1 steps, the detector performance was the same for all of the scaled values.

Due to the linearity of the embedding domain, when the sign of the pirate audio signal is inverted, the same happens in the embedding domain. Therefore, in order to guarantee the correct detection in an inverse attack scenario, the ID in a block is counted when correlation value is bigger that thresholds T_g and T_u or lower that $-T_g$ and $-T_u$, which is a very small change to the detector. This claim was corroborated with several experiments where, in the presence of the inverse attack, the detector performance is not altered

Comparison

To the best of our knowledge, the work reported in [24] is the only one addressing the collusion-resistant fingerprinting problem with audio signals. Table 3 shows a detailed feature comparison of the proposed system against that proposed in [24]. As it was described in the Related Work section, the work reported in [24] is not robust against sign inversion attack whereas, according to the results described in the Results and Discussion section, the proposed system is robust against this type of attack. A very common real-world audio processing operation is volume gain, the proposed system is able to resist this processing while the work reported in [24] does not. Moreover, unlike [24], this paper reports results for lossy compression and system viability, which are real-world scenarios as music distribution is nowadays in compressed format and real-time. Comparing the work reported in [24] with the proposed system in terms of detection probability per number of colluders is a difficult task as it is unclear which value of false alarm, P_{fa}, is considered in that work. The aforementioned work lacks in a statistical analysis of the system performance, therefore, a deepest comparison with the proposed system can be biased.

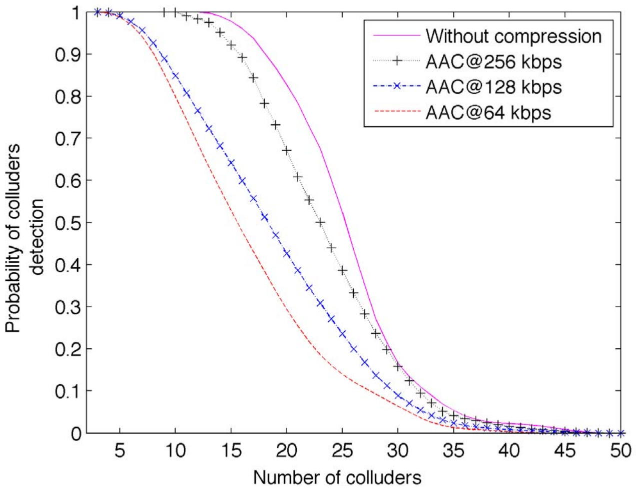

Figure 10. Detection rate of colluders in function of several AAC bitrates.
doi:10.1371/journal.pone.0065985.g010

Summary of Results

In this paper, a block-based approach for fingerprinting is considered. This consideration is due to two facts: 1) a frequency transform for a full typical audio clip is practically intractable and 2) if the fingerprint is replied each block, then, for detection is not necessary the full pirate audio signal. As a consequence of the block-based approach, the detection is carried out according to the half-normal distribution. Through experimentation, it was shown that about 1 second of CD-quality pirate audio signal is enough for probability of colluders detection equal to 1.

The optimal energy for user and group ID fingerprints in function of ODG metric is also studied. It was observed that the bigger user ID fingerprint energy, β_u, the better detection performance. This characteristic is interesting because users in a group are more likely to collude with each other [17], therefore; the number of group IDs involved in a pirate copy must be minor to the number of colluder IDs.

The impact in the fingerprint detection process of the block length was investigated through experimentation. It was observed that the higher block length the higher collusion-attack robustness. However, for a block length bigger than 2^{14} samples the performance improvement is not significative. Moreover, for a bigger block length the needed computing resources are also bigger and even intractable for some platforms.

For validation purposes, the proposed fingerprinting system was implemented in an standard modern computer using free libraries. The performance is guaranteed to be several times better that the real-time restriction. The proposed system viability is demonstrated.

Finally, the robustness of the proposed system to typical attacks in real-world scenarios, such as lossy compression, gain and inverse attacks, was shown. Then results suggest that the proposed fingerprinting system is suitable for practical applications, therefore, attractive for the music industry.

Materials and Methods

Due to the proposed fingerprinting system utilizes the DCT basis as fingerprint modulators and the insertion domain is the set of MCLT magnitudes, in this section are recalled two Fast Fourier Transform (FFT)-based fast algorithms for MCLT and DCT calculations which are utilized for the proposed fingerprinting system implementation.

Modulated Complex Lapped Transform

The Modulated Complex Lapped Transform (MCLT) is a particular kind of a 2x oversampled generalized DFT filter bank proposed in [38] whose basis are:

Table 3. Comparison between the proposed work and [24].

Algorithm	Robustness			Number of users	System viability
	Lossy compression	Inverse sign	Volume gain		
Tirkel *et al.* [24]	no	no	no	not reported	not reported
Proposed	high quality	yes	yes	M^2	yes

doi:10.1371/journal.pone.0065985.t003

$$p(n,k)=p_c(n,k)-jp_s(n,k) \tag{14}$$

$$p_c(n,k)=h(n)\sqrt{\frac{2}{M}}\cos(phase) \tag{15}$$

$$p_s(n,k)=h(n)\sqrt{\frac{2}{M}}\sin(phase) \tag{16}$$

with:

$$h(n)=-\sin[(n+\frac{1}{2})\frac{\pi}{2M}] \tag{17}$$

and

$$phase=(n+\frac{M+1}{2})(k+\frac{1}{2})\frac{\pi}{M} \tag{18}$$

where n is the time-domain index, k is the frequency-domain index, M is the sample block length and $j=\sqrt{-1}$. The MCLT coefficients of the input vector $\mathbf{x}$ are calculated as $X(k)=X_c(k)-jX_s(k)$ with:

$$\begin{aligned} X_c(k) &= \sum_{n=0}^{2M-1} x(n)p_c(n,k), \\ X_s(k) &= \sum_{n=0}^{2M-1} x(n)p_s(n,k) \end{aligned} \tag{19}$$

Fast MCLT Algorithm. In [39] it was proposed a FFT-based fast MCLT algorithm. The MCLT coefficients $X(k)$ can be obtained as follows:

$$X(k)=jV(k)+V(k+1) \tag{20}$$

where

$$\begin{aligned} V(k) &= c(k)U(k) \\ c(k) &= W_8(2k+1)W_{4M}(k) \\ U(k) &= \sqrt{\tfrac{1}{2M}}\sum_{n=0}^{2M-1} x(n)W_{2M}(kn) \end{aligned} \tag{21}$$

and $W_M(r)$ is the common notation for the complex exponential used in Fourier transforms, namely:

$$W_M(r)=\exp(\frac{-j2\pi r}{M}) \tag{22}$$

$U(k)$ is a $2M$ point FFT with orthonormal basis function of the input block $x(n)$, which means that MCLT coefficients can be computed by computing FFT of $x(n)$ to obtain $U(k)$ and carring out the operations with factors $c(k)$.

Fast Inverse MCLT Algorithm. In order to carry out the inverse MCLT, in [39] is developed the next relation:

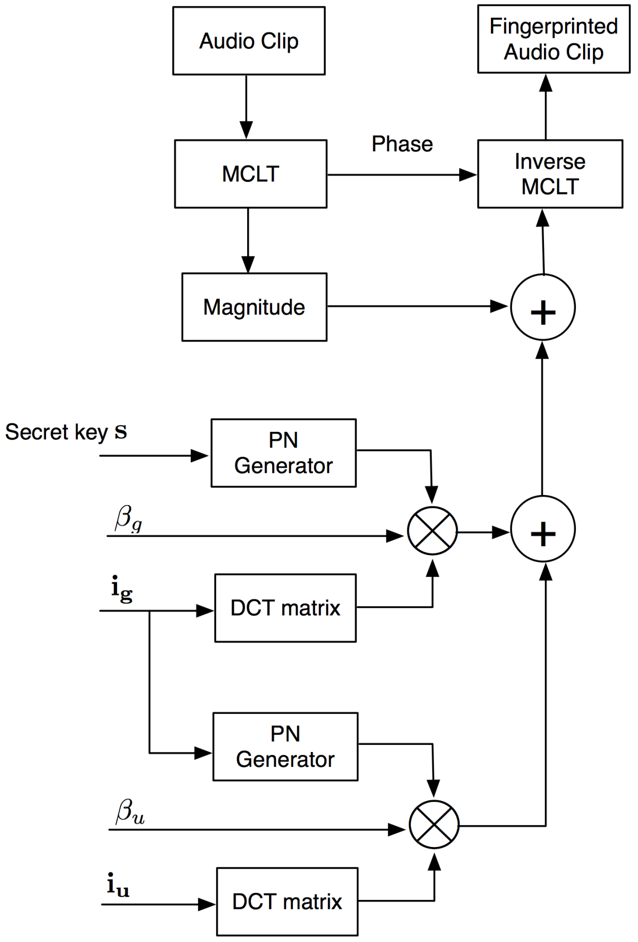

Figure 11. Fingerprint embedding system.
doi:10.1371/journal.pone.0065985.g011

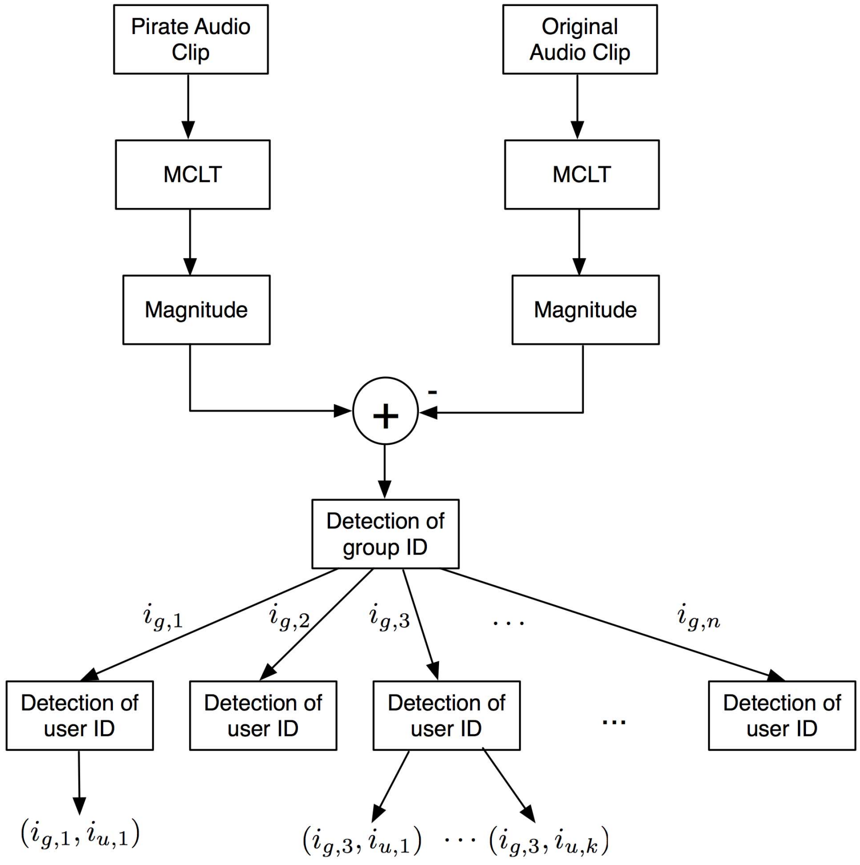

Figure 12. Colluder detection system.
doi:10.1371/journal.pone.0065985.g012

$$Y(k) = \frac{c^*(k)}{4}[X(k-1) - jX(k)] \tag{23}$$

where $X(k)$ are the MCLT coefficients, the superscript * denotes complex conjugation, and the modulation $c(k)$ is the same as that in (21). Using (23) we compute the M first FFT coefficients of $y(n)$, but it is well known that FFT coefficients must satisfy the conjugate symmetry property:

$$Y(2M-k) = Y^*(k) \tag{24}$$

Finally, we know that $Y(0)$ and $Y(M)$ must be real-valued, and after some manipulations:

$$\begin{aligned} Y(0) &= \tfrac{1}{\sqrt{8}}[\Re\{X(0)\} + \Im\{X(0)\}] \\ Y(M) &= -\tfrac{1}{\sqrt{8}}[\Re\{X(M-1)\} + \Im\{X(M-1)\}] \end{aligned} \tag{25}$$

with $\Re$ and $\Im$ taking the real and imaginary parts, respectively.

Discrete Cosine Transform

The Discrete Cosine Transform (DCT) is a linear and invertible function in the Real Numbers set, originally derived from Chebyshev polynomials [40]. The DCT basis are orthogonal and defined as follows:

$$a(n,k) = c(k)\sqrt{\frac{2}{M}}\cos[(n+\frac{1}{2})\frac{k\pi}{M}] \tag{26}$$

where

$$c(k) = \begin{cases} 1/\sqrt{2} & \text{if } k=0 \\ 1 & \text{otherwise} \end{cases} \tag{27}$$

Fast DCT and Inverse-DCT Algorithms. It is known that the Fourier transform of a real-even function $f(-x) = f(x)$ is real-even, and i times the Fourier transform of a real-odd function $f(-x) = -f(x)$ is real-odd, thus for these symmetry conditions it is not necessary to use complex inputs/output. Therefore, it is possible to compute the DCT or the Discrete Sine Transform (DST) by utilizing an FFT algorithm.

Let be the input vector $x(n = 0..M-1)$ even around $n = -0.5$ and even around $n = M-0.5$, it is possible to show that DFT(x) is the non-normalized DCT of x, $Y_{nonO}(k)$ described as follows:

$$Y_{nonO}(k) = 2\sum_{n=0}^{M-1} x(n)\cos[(n+\frac{1}{2})\frac{k\pi}{M}] \tag{28}$$

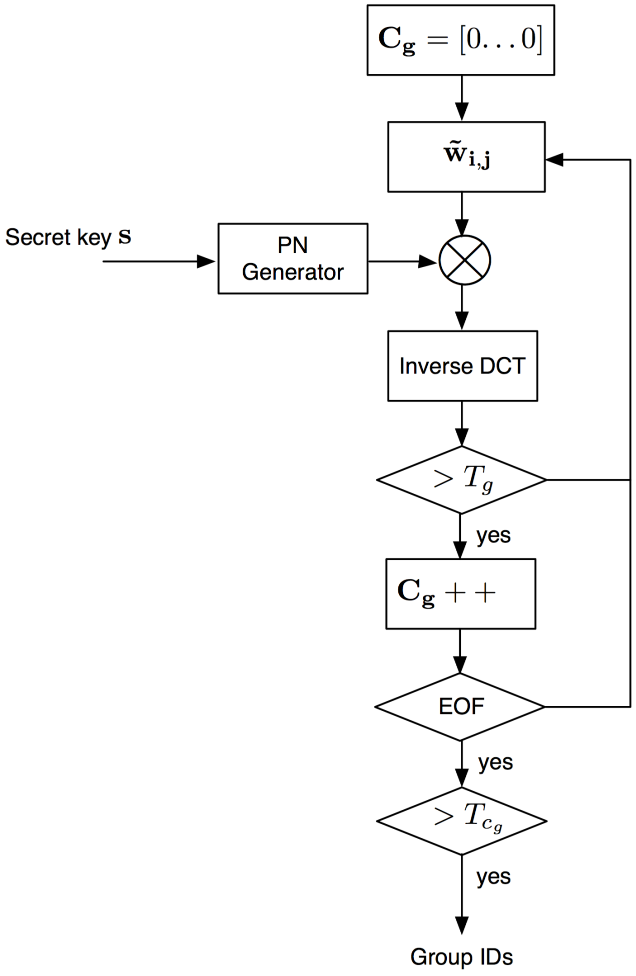

Figure 13. Group IDs detection system.
doi:10.1371/journal.pone.0065985.g013

with basis:

$$b(n,k) = 2\cos[(n + \frac{1}{2})\frac{k\pi}{M}] \tag{29}$$

The basis set described in equation (29) is non-orthogonal, therefore, it is necessary to normalize equation (28) in order to get the orthogonal transform as follows:

$$Y_O(k) = \begin{cases} (1/\sqrt{4M})\,Y_{nonO}(k) & \text{if } k=0 \\ (1/\sqrt{2M})\,Y_{nonO}(k) & \text{otherwise} \end{cases} \tag{30}$$

On the other hand, let be the input vector $Y(k=0..M-1)$ even around $k=0$ and odd around $k=M$, it is possible to show that DFT(Y) is the non-normalized Inverse DCT of Y, $x_{nonO}(n)$ described as follows:

$$x_{nonO}(n) = Y(0) + 2\sum_{k=1}^{M-1} Y(k)\cos[(n + \frac{1}{2})\frac{k\pi}{M}] \tag{31}$$

As in equation (28), it is necessary a normalization procedure in order to get the orthogonal transform. The normalization is carried out as follows:

$$x_O(k) = \frac{1}{\sqrt{2M}} x_{nonO}(n) \tag{32}$$

In the literature, fast algorithms for the DFT have been extensively reported and very efficient software libraries exist [41]. In this work, these libraries are utilized as a module of the DCT and MCLT computing, reducing the effort required for efficient implementation to a butterfly stage implementation for MCLT and a normalization stage implementation for DCT.

The Fingerprinting System

The frequency domain for embedding is the Modulated Lapped Complex Transform (MCLT). In order to bring perceptual transparency, the fingerprint is embedded into MCLT magnitudes while keeping phases without changes.

Fingerprint Embedding. Instead of [18,19], in this paper the fingerprint is replicated several times along the audio signal in a block-processing fashion as typical CD-quality music clips are conformed by about 8 million of samples and the embedding/detecting process can become intractable if an orthogonal transform is applied to the whole audio clip. Moreover, by splitting the audio signals in blocks for fingerprinting it is possible to detect colluders with a fraction of the whole audio clip which is demonstrated in the Results section. Each samples-block is 50% overlapped as the MCLT is a lapped transform. Due to MCLT is a 2x oversampled DFT filter bank, $2M$ audio samples are required in order to compute M MCLT coefficients. Figure 11 shows a block diagram of the embedding system.

The fingerprint embedding process is carried out as follows: firstly host audio signal is divided into frames of $2M$ samples per frame. Next, each frame is transformed using the MCLT. Subsequently both magnitude and phase of MCLT are computed. The fingerprint is then added to the MCLT magnitudes while keeping phase without change. The additive technique is utilized for embedding as follows:

$$\hat{\mathrm{X}} = \mathrm{X} + \mathrm{w}_{\mathrm{i,j}}, \tag{33}$$

where $\hat{X}$ is the fingerprinted MCLT magnitude, X is the original MCLT magnitude and $\boldsymbol{w_{i,j}}$ is the fingerprint assigned to the jth user of the ith group. Finally, inverse MCLT is applied to both processed magnitude and original phase to get the audio signal with hidden fingerprint. The fingerprint is conformed according to equation (8), the secret key s provides the system security in a symmetric-key fashion. The $\mathbf{i_u}$ and $\mathbf{i_g}$ variables represent the authorized user and its group respectively. The PN-Generators produce pseudo-noise with an uniform distribution.

Fingerprint Detection. Figure 12 shows the colluders detection system. In fingerprinting systems is a common assumption to get access to the original media. That consideration is taken into account for the proposed system.

Detection procedure is carried out in a block fashion as the fingerprint is embedded in the same way. In this paper, a detection strategy using several MCLT magnitude blocks is proposed.

Group ID Detection. Figure 13 shows the group IDs detection system. For each available MCLT coefficients block, group detection is carried out according to the threshold, T_g,

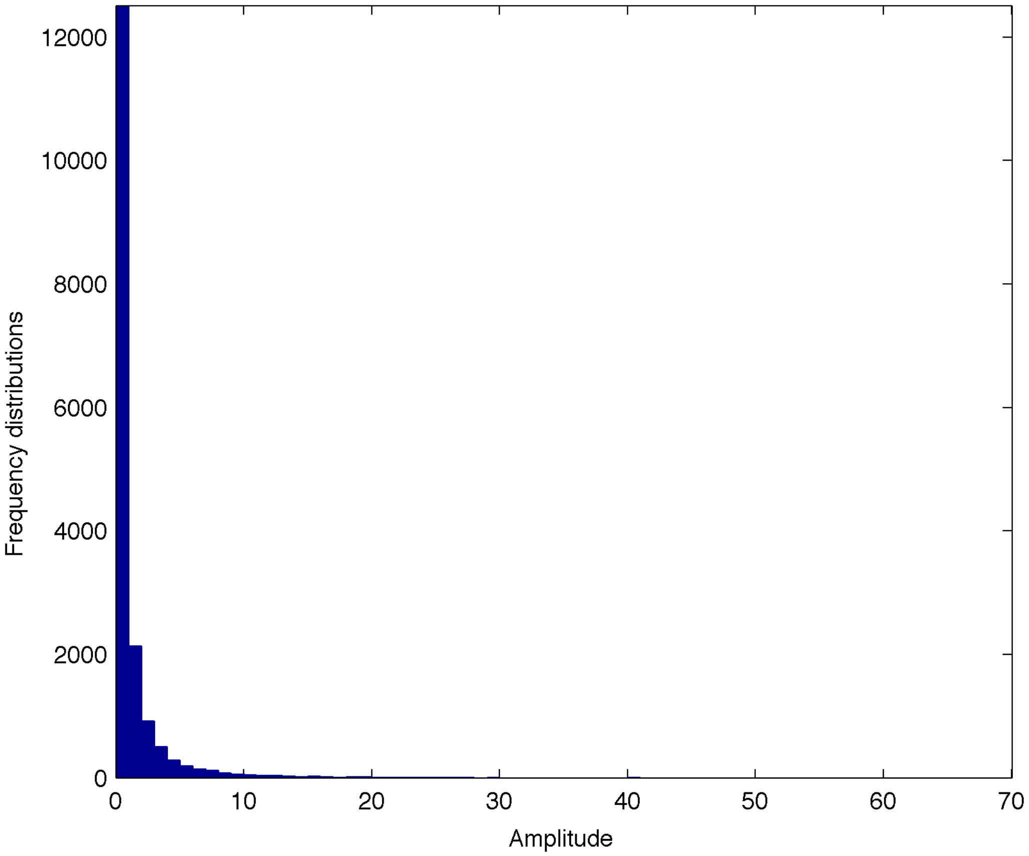

Figure 14. Distribution of the counter vector $\mathrm{C_g}$.
doi:10.1371/journal.pone.0065985.g014

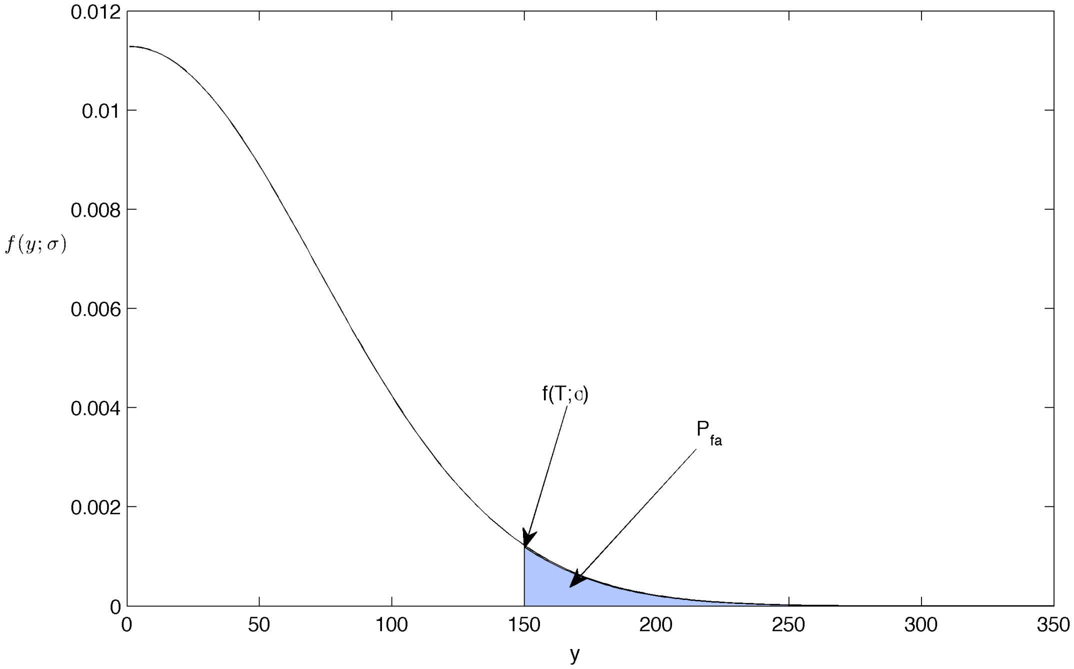

Figure 15. Half-normal distribution.
doi:10.1371/journal.pone.0065985.g015

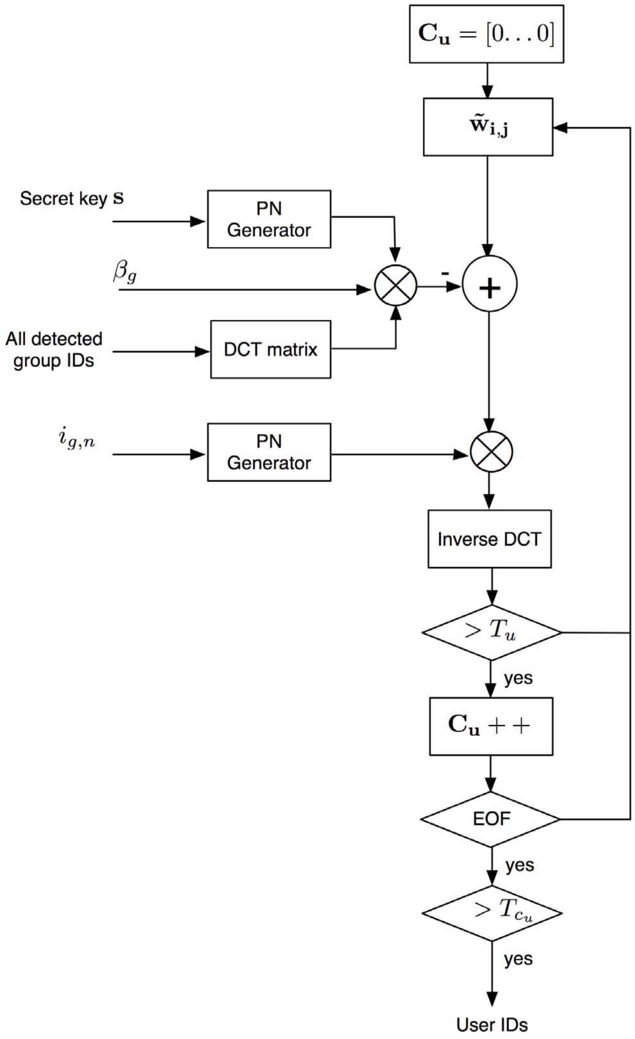

Figure 16. User IDs detection system.
doi:10.1371/journal.pone.0065985.g016

described in equation (12). For the whole pirate audio clip, there is a counter vector $\mathbf{C_g}$ that registers the number of times that each component of $\tilde{\mathbf{d}}_{\mathbf{g}}$ exceeds T_g.

Instead of the group ID detection in each block where the threshold is computed assuming a Gaussian distribution, the threshold, T_{c_g}, for detection in the counter vector $\mathbf{C_g}$ must consider other distribution as the lower limit of that distribution will always be zero. In order to know the statistical behavior of the counter vector $\mathbf{C_g}$, 120 different fingerprinted audio clips are utilized. Figure 14 shows the distribution of the counter vector $\mathbf{C_g}$.

As can be seen from Figure 14, $\mathbf{C_g}$ can be modeled by the Half-normal distribution, which is defined as follows:

$$f(y;\sigma) = \sqrt{\frac{2}{\sigma^2\pi}}\exp\left(-\frac{y^2}{2\sigma^2}\right) y > 0 \qquad (34)$$

with cumulative distribution function $F(y;\sigma)$ as follows,

$$F(y;\sigma) = \int_0^y \sqrt{\frac{2}{\sigma^2\pi}}\exp\left(-\frac{x^2}{2\sigma^2}\right)\mathrm{d}x \qquad (35)$$

For a given threshold, T, the false detection probability P_{fa}, is computed by subtracting the cumulative distribution function to the unit as follows:

$$P_{fa} = 1 - \int_0^T \sqrt{\frac{2}{\sigma^2\pi}}\exp\left(-\frac{y^2}{2\sigma^2}\right)\mathrm{d}y \qquad (36)$$

Figure 15 shows the P_{fa} for a given threshold T in a Half-normal distribution.

Using a change of variable $z = y/(\sqrt{2\sigma^2})$ in equation (35) it becomes:

$$\begin{aligned} F(y;\sigma) &= \tfrac{2}{\sqrt{\pi}}\int_0^{y/(\sqrt{2\sigma^2})}\exp(-z^2)\mathrm{d}z \\ &= \mathrm{erf}\left(\frac{y}{\sqrt{2\sigma^2}}\right) \end{aligned} \qquad (37)$$

where erf(.) is the error function and is related to the complementary error function as:

$$\mathrm{erfc}(x) = 1 - \mathrm{erf}(x) \qquad (38)$$

From equations (36), (37) and (38); the false detection probability P_{fa} for a given threshold T can be rewritten as:

$$P_{fa} = \mathrm{erfc}\left(\frac{T}{\sqrt{2\sigma^2}}\right) \qquad (39)$$

Therefore, the threshold, T_{C_g} for a P_{fa} given for a group ID detection in a pirate audio clip can be computed as follows:

$$T_{Cg} = \sqrt{2\sigma^2_{C_g}}\mathrm{erfc}^{-1}(P_{fa}) \qquad (40)$$

where $\sigma^2_{C_g}$ is the variance of $\mathbf{C_g}$.

User ID Detection. Figure 16 shows the user ID detection system. In similar form that group ID detection, for each available MCLT coefficients block, user detection is carried out according to the threshold, T_u, described in equation (13). For the whole pirate audio clip, there is a counter vector $\mathbf{C_u}$ that registers the number of times that each component of $\tilde{\mathbf{d}}_{\mathbf{u}}$ exceeds T_u. Counter vector $\mathbf{C_u}$ is modeled as a Half-normal distribution and the corresponding threshold T_{C_u} for a given P_{fa} is calculated according to:

$$T_{Cu} = \sqrt{2\sigma^2_{C_u}}\mathrm{erfc}^{-1}(P_{fa}) \qquad (41)$$

where $\sigma^2_{C_u}$ is the variance of $\mathbf{C_u}$. In order to improve user ID detection, the interference due to group ID is previously removed.

Number of Users. Due to fingerprints are formed by two DCT modulated PN-sequences, the number of possible IDs for each of them is equal to their respective lengths, M for both. Therefore, the maximum number of possible users of the system is M^2.

Software Implementation

Audio signals are processed as vectors of float numbers in the range $[-1,1)$. For audio file manipulations the libsndfile library [42] is used. The entire fingerprinting system was implemented in C language in a Intel Core i7 CPU and 8 GB RAM. The compiler used in this work is GCC version 4.2.1 and the operating system is Mac Os X version 10.7.5. The programs are compiled with -o3 optimization flag. In order to compute the FFT for the fast MCLT and DCT algorithms described above, the FFTW library [41] is utilized.

FFTW is an optimized library that implements most of the variants of the Discrete Fourier Transform. Moreover, FFTW is able to exploit the Message Passing Interface (MPI) and multi-threaded strategies in order to utilize the full power of modern personal computers. Due to the computer used for validation of the proposed fingerprinting system is a multi-core shared-memory computer, the instantiation of FFTW is carried out using multi-threaded calls.

Conclusions

In this paper, a collusion-resistant fingerprinting system for audio signals is proposed. Each fingerprint is formed by a PN-sequence representing a group ID and other representing one user ID, following state-of-the-art ideas for fingerprinting systems in digital images. Due to nature of audio signals, the fingerprint is replied several times along the audio clip, therefore, it is not necessary the whole audio clip in the detection process. This characteristic guarantees the performance to be several times better that the real-time restriction. The detector performance after high quality lossy compression remains competitive for real work environments. The number of users available, the low computational complexity and the high quality lossy compression robustness make the proposed algorithm attractive for a number of audio processing applications.

Author Contributions

Conceived and designed the experiments: JJG CFU RC. Performed the experiments: JJG CFU RC. Analyzed the data: JJG CFU RC. Contributed reagents/materials/analysis tools: JJG CFU RC. Wrote the paper: JJG CFU RC.

References

1. Zhang X, Liu Q, Wang H (2012) Ontologies for intellectual property rights protection. Expert Systems with Applications 39: 1388–1400.
2. Cox I, Miller M, Bloom J, Fridrich J, Kalker T (2007) Digital Watermarking and Steganography, 2nd Ed. (The Morgan Kaufmann Series in Multimedia Information and Systems). Morgan Kaufmann, 2nd edition.
3. Fridrich J (2009) Steganography in Digital Media: Principles, Algorithms, and Applications. Cambridge University Press.
4. Jupiter D, Ficht T, Samuel J, Qin Q, de Figueiredo P (2010) Dna watermarking of infectious agents: Progress and prospects. PLoS ONE 6: e1000950.
5. Liss M, Daubert D, Brunner K, Kliche K, Hammes U, et al. (2012) Embedding permanent watermarks in synthetic genes. PLoS ONE 7: e42465.
6. Kiyavash N, Moulin P, Kalker T (2009) Regular simplex fingerprints and their optimality properties. IEEE Transactions on Information Forensics and Security 4: 318–329.
7. Boneh D, Shaw J (1998) Collusion-secure fingerprinting for digital data. IEEE Transactions on Information Theory 44: 1897–1905.
8. Trappe W, Wu M, Wang J, Liu R (2003) Anti-collusion fingerprint for multimedia. IEEE Transactions on Signal Processing 51: 1069.
9. Tardos G (2003) Optimal probabilistic fingerprint codes. In: Proceedings of 35th ACM Symposium on Theory Computing. pp. 116–125.
10. Cox I, Kilian J, Leighton T, Shamoon T (1997) Secure spread spectrum watermarking for multimedia. IEEE Transactions on Image Processing 6: 1673–1687.
11. Kirovski D, Malvar H (2003) Spread spectrum watermarking of audio signals. IEEE Transactions on Signal Processing 51: 1020–1033.
12. Tsai MJ, Luo YF (2009) Service-oriented grid computing system for digital rights management (gc-drm). Expert Systems with Applications 36: 10708–10726.
13. Geetha S, Ishwarya N, Kamaraj N (2010) Audio steganalysis with hausdorff distance higher order statistics using a rule based decision tree paradigm. Expert Systems with Applications 37: 7469–7482.
14. Kirovski D, Malvar H, Yacobi Y (2004) A dual watermark-fingerprint system. IEEE Multimedia 11: 59–73.
15. Zhao HV, Wu M, Wang ZJ, Liu KJR (2005) Forensic analysis of nonlinear collution attacks for multimedia fingerprinting. IEEE Transactions on Image Processing 14: 646–61.
16. Wang ZJ, Wu M, Zhao HV, Trappe W, Liu KJR (2005) Anti-collusion forensics of multimedia fingerprinting using orthogonal modulation. IEEE Transactions on Image Processing 14: 804–821.
17. Wang ZJ, Wu M, Trappe W, Liu KJR (2004) Group-oriented fingerprinting for multimedia forensics. EURASIP Journal on Advances in Signal Processing 2004: 2153–2173.
18. Kuribayashi M (2011) Hierarchical spread spectrum fingerprinting scheme based on the cdma technique. EURASIP Journal on Information Security 2011.
19. Kuribayashi M (2012) Interference removal operation for spread spectrum fingerprinting scheme. IEEE Transactions on Information Forensics and Security 7: 403–417.
20. Siwek SE (2007) The true cost of sound recording piracy to the u. s. economy. Policy Report 188, Institute for Policy Innovation.
21. IIPA (2010) Iipa 2010 "special 301" recommendations. Available: http://www.iipa.com/rbc/2010/2010SPEC301LOSSLEVEL.pdf. Accessed 2013 March 6.
22. Kim Wg, Lee S, Seo Ys (2006) Image fingerprinting scheme for print-and-capture model. In: Zhuang Y, Yang SQ, Rui Y, He Q, editors, Advances in Multimedia Information Processing -PCM 2006, Springer Berlin/Heidelberg, volume 4261 of *Lecture Notes in Computer Science*. pp. 106–113.
23. Celik MU, Sharma G, Tekalp AM (2004) Collusion-reilient fingerprinting by random pre-warping. IEEE Signal Processing Letters 11: 831–835.
24. Tirkel A, Hall TE, Osborne CF, Meinhold N, Moreno O (2011) Collusion resistant fingerprinting of digital audio. In: Proceedings of the 4th international conference on Security of information and networks. ACM, pp. 5–12.
25. Kirovski D, Malvar H (2001) Robust covert communication over a public audio channel using spread spectrum. In: 4th International Information Hiding Workshop. pp. 354–368.
26. Zezula R, Misurec J (2006) Audio signal watermarking in mclt domain with the aid of 2d pattern. In: Proceedings of 2nd International Conference on Digital Telecommunications, ICDT '07. p. 40.
27. Garcia-Hernandez JJ, Nakano M, Perez H (2008) Data hiding in audio signals using rational dither modulation. IEICE Electron Express 5: 217–222.
28. Garcia-Hernandez JJ, Feregrino-Uribe C, Cumplido R, Reta C (2011) On the implementation of a hardware architecture for an audio data hiding system. Journal of Signal Processing Systems 64: 457–468.
29. Cano P, Batlle E, Kalker T, Haitsma J (2002) A review of algorithms for audio fingerprinting. In: IEEE Workshop on Multimedia Signal Processing. IEEE, pp. 169–173.
30. Cano P, Batlle E, Gomez E, Gomes LCT, Bonnet M (2005) Audio Fingerprinting: Concepts and Applications, Springer-Verlag, chapter 2. Studies in Computational Intelligence. pp. 233–245.
31. Jang D, Jin M, Lee J, Lee S, Lee S, et al. (2006) Automatic commercial monitoring for tv broadcasting using audio fingerprinting. In: AES, editor, 29th International Conference: Audio for Mobile and Handheld Devices. 3–2.
32. Baluja S, Coveli M (2008) Waveprint: Efficient wavelet-based audio fingerprinting. Pattern Recognition 41: 3467–3480.
33. Bellettini C, Mazzini G (2010) A framework for robust audio fingerprinting. Journal of Communications 5: 409–424.
34. Liu KJR, Trappe W, Wang ZJ, Wu M, Zhao H (2005) Multimedia Fingerprinting Forensics for Traitor Tracing, volume 4 of *EURASIP Book Series on Signal Processing and Communications*. Hindawi Publishing Corporation.
35. Thiede T, Treurniet W, Bitto R, Schmidmer C, Sporer T, et al. (2000) Peaq - the itu standard for objective measurement of perceived audio quality. AES 48: 3–29.
36. Smith SW (2002) Digital Signal Processing: A Practical Guide for Engineers and Scientists. Newnes.
37. Brandenburg K (1999) Mp3 and aac explained. In: AES, editor, Audio Engineering Society Conference: 17th International Conference: High-Quality Audio Coding. 1709, pp. 1–12.
38. Malvar HS (1999) A modulated complex lapped transform and its applications to audio processing. In: IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, volume 3, pp. 1421–1424.

39. Malvar HS (2003) Fast algorithm for the modulated complex lapped transform. IEEE Signal Processing Letters 10: 8–10.
40. Ahmed N, Natarajan T, Rao KR (1974) Discrete cosine transform. IEEE Transactions on Computers 23: 90–93.
41. Frigo M, Johnson SG (2005) The desing and implementation of fftw3. Proceedings of the IEEE 93: 216–231.
42. de Castro Lopo E (2011) Libsndfile. Available: http://www.mega-nerd.com/libsndfile/. Accessed 2013 January 25.