Audio Watermarking Forensics: Detecting malicious re-embedding

Sascha Zmudzinski[1], Martin Steinebach[1], Stefan Katzenbeisser[2] and Ulrich Rührmair[3]

[1]Fraunhofer Institute for Secure Information Technology SIT,
Rheinstr. 75, 64295 Darmstadt, Germany

[2]Technische Universität Darmstadt, Computer Science Department,
Hochschulstr. 10, 64289 Darmstadt, Germany

[3]Technische Universität München, Department of Computer Science,
Boltzmannstr. 3, 85748 Garching bei München, Germany

ABSTRACT

Digital watermarking has become a widely used security technology in the domain of Digital Rights Management and copyright protection as well as in other applications. In this work, we investigate a particular attack strategy: Embedding a new message in media content that already carries a watermark.

The possibility for such an attack strategy results from the absence of truly asymmetric watermarking schemes, especially if the watermark is to be detected in public. In public detection scenarios, every detector needs the same key the embedder used to watermark the cover. With knowledge of the embedding algorithm, everybody who is able to detect the message can also maliciously embed a new message with the same key over the old one. This scenario is relevant in the case that an attacker intends to counterfeit a copyright notice, transaction ID or to change an embedded authentication code.

This work presents experimental results on mechanisms for identifying such multiple embeddings in a spread-spectrum patchwork audio watermarking approach. We demonstrate that under certain circumstances multiple embedding can be detected.

Keywords: Audio watermarking, watermarking security, multiple embedding, asymmetric watermarking, patchwork watermarking, spread-spectrum

1. MOTIVATION AND INTRODUCTION

While robustness[7] and transparency are well developed and analyzed, the security of digital watermarking is still discussed in little detail. Primary research issues on watermarking security are strategies for asymmetric watermarking[4] and resistance to a number of specialized attacks like collusion attacks,[10] copy attacks and inversion attacks.[1] Either specially designed messages or fingerprints or specific secure protocols utilizing time stamps and content descriptors have been introduced to counter these attacks.

As long as no truly asymmetric methods for embedding and detecting digital watermarks have been introduced, watermarking shares one weakness of all symmetric methods: One cannot distinguish the individuals using the secret key for embedding and detection when the key is legitimately shared or illegitimately leaked to unauthorized users. As a consequence *"...once the key or keys are compromised, the Darknet will propagate them efficiently, and the scheme collapses"* (*Biddle et al.*[3]).

The knowledge about the detection algorithm tells the attacker how the information is hidden within the cover, allowing to reverse engineer an embedding algorithm. The secret key tells the attacker exactly where the information has been hidden. Therefore every party able to detect a watermark also has a powerful tool at hand to attack the embedded information, either by removing or counterfeiting it.

The only difference between the attacker and the original embedding party is the state of the cover: Each embedding process changes the cover. The attacker can guess to a certain degree how the original embedder modified the given cover. Because truly robust watermarking schemes are non-reversible and usually include a

pseudo-random selection of modified feature sets, he has no *exact* idea about the unmarked state of the cover. This can help to identify malicious re-embedding when examining an embedded watermark message at a forensic level of detail. In this work, we demonstrate that under certain circumstances such multiple embedding can be detected by watermarking-forensics.

2. PATCHWORK WATERMARKING

2.1. Single embedding

As an example we look at a typical patchwork embedder as described by *Bender et al.*:[2] It pseudo-randomly selects two subsets $A = \{a_i\}$ and $B = \{b_i\}$ of a selected feature set, and then modifies them in a way that these modifications can later be detected by statistical means.

Patchwork embedding is typically based on the assumption that the mean values

$$\bar{a} = \frac{1}{n}\sum a_i \quad \text{and} \quad \bar{b} = \frac{1}{n}\sum b_i \quad (i = 1...n)$$

of the elements in A and B are approximately equal, if the $\{a_i\}$ and $\{b_i\}$ are (pseudo-) randomly selected from the feature set. That is, the difference $\bar{a} - \bar{b}$ can be seen as a random variable with expectation zero. As can be seen from the literature on patchwork embedding, appropriate examples for such feature sets can be gray values in a digital image[6] or DCT coefficients in audio data.[11]

The embedding approach is done by adding (subtracting, resp.), a small quantity $d > 0$ to (from) all values in $\{a_i\}$ and $\{b_i\}$ obtaining watermarked subsets A' and B' as follows:

- If the message bit to be embedded is a *zero*, the $\{a_i\}$ and $\{b_i\}$ are modified such that $\{a'_i\} = \{a_i + d\}$ and $\{b'_i\} = \{b_i - d\}$, i.e. that the related mean value $\bar{a}'$ becomes significantly greater than $\bar{b}'$, i.e. $\bar{a}' > \bar{b}'$.
- If the message bit is a *one* the $\{a_i\}$ and $\{b_i\}$ are modified the opposite way, such that $\bar{a}' < \bar{b}'$, significantly.

The modification is controlled by an appropriate perceptual model to keep the embedding process as transparent as possible.

2.2. Multiple embedding

Now we consider the security attack of changing the embedded bit by modifying the $\{a'_i\}$ and $\{b'_i\}$ once again. For example, we assume that message bit shall be flipped from *one* to *zero* by the attacker, which means that the watermarked features in the subsets show the property $\bar{a}' < \bar{b}'$. Now, for the feature subsets $A'' = \{a''_i\}$ and $B'' = \{b''_i\}$ obtained after second embedding, we must enforce their mean value $\bar{a}''$ to become greater than $\bar{b}''$. But as the content had been watermarked exactly the opposite way, simply adding d to all elements in A and subtracting d from all elements in B would end up in $\bar{a}'' \approx \bar{b}''$, resulting in no or a very weak watermark to be detected. That means an attacker would need a much higher degree of modification, say $\pm 2d$ for embedding such that the mean values

$$\bar{a}'' = \frac{1}{n}\sum\{a_i - d + 2d\} \quad \text{and} \quad \bar{b}'' = \frac{1}{n}\sum\{b_i + d - 2d\}$$

will meet the criterion $\bar{a}'' > \bar{b}''$.

For the second embedding, the perceptual model will not provide the same results like for the first embedding, as the characteristics of the cover have been changed during the first embedding. As a gross simplification, one could regard the first embedding process as removing the balance from a statistically equal set of features, which can be rather easy, while re-embedding requires to completely change the statistical state of the feature set.

Based on this approach one can design a forensic detector for re-embedding, testing if the embedded watermark is either weaker than to be expected given the masking characteristics controlling the perceptual model or if it had been embedded stronger than to be expected from the algorithm. Our goal is to perform this operation without requiring access to the original cover, while the advantages of utilizing the original cover will also be discussed.

2.3. Spread-spectrum patchwork watermarking and forensics

In our work, we focus on a modified version of a spread spectrum patchwork audio watermarking algorithm as published in earlier works.[8,9] Here, the audio stream is at first divided into audio frames of 2048 consecutive samples. Then, the frame data is transformed from the temporal domain to the spectral domain using the FFT transform. As the feature set to be modified, we consider two subsets A and B of absolute values of FFT coefficients calculated from an audio frame, especially the difference d of the two related mean values $\bar{a}$ and $\bar{b}$. The selection of the subsets from the spectrum is done pseudo-randomly and is dependent on the secret watermark key. Embedding a single message bit is done by increasing all coefficients in one of the subsets and decreasing all coefficients in the other subset. The FFT phase remains unchanged. To embed a message consisting of more than one bit, the message bits are embedded in sequence in consecutive audio frames.

Each watermark message is preceded by a synchronization pattern, which is a common synchronization approach in digital watermarking. This *sync* pattern is a predefined fixed message bit sequence that is identical for all messages to be embedded. The sync serves as a start code indicating the presence of a watermark and can be searched for by the detector as the sync bit sequence is known to both embedder and detector. If the sync pattern is found on the detector side, the feature d of consecutive audio frames can be retrieved, evaluated and the message bits can be retrieved.

In addition, the detector not only retrieves the individual value of the sync and watermark bits *one* or *zero*. The absolute value of d represents also a quantity – we call it *detection score*) – that indicates how significant the message bit could be retrieved. Furthermore, the detection score of the sync can be used for calibrating the thresholds for the detection scores, as we will see.

3. EXPERIMENTAL EVALUATION

3.1. Test setup

We embedded binary messages (64 Bit) into the pseudo-randomly chosen FFT coefficients in the range of 1 to 5 kHz as described before. The embedding strength of each modified FFT coefficient is controlled by a psychoacoustic model to avoid audible distortions. In our experiments, for the few modified FFT coefficients the algorithm was allowed to exceed the masking threshold by 3 dB, which is hardly audible for an average human listener. Our test set contains 110 audio files, approximately 8.5 hours PCM audio data (44.1 kHz mono) of different *music* genre.

3.2. Single embedding

3.2.1. Detection score

At first, for the sync detection we analyzed the behavior of the detection score of the cover data, i.e. if the audio content is in fact not watermarked. As we can see from Figure 1 the histogram of the absolute value d for the null hypothesis "*H0: no watermark present*" is concentrated to the far left, i.e. near zero. For the watermarked content, where a sync pattern is actually present, the histogram of retrieved detection scores is significantly biased to the right and the two histograms are clearly separated (see Figure 2).

Now, we consider the related error rates which can be seen in Figure 3. Closer analysis shows that the intersection of the false rejection $\alpha(d)$ and false acceptance $\beta(d)$ of the null hypothesis "*no sync*" can be found at $d \approx 0.8$. This value is expressed on an *arbitrary* scale (i.e. not normalized) given by the internals of the software implementation of the embedding algorithm[8] e.g. the chosen FFT normalization.

Then, in the first embedding step, as an example we embedded a message m_1 (in hex representation) where

$$m_1 = \texttt{0x 00 00 00 00 FF FF FF FF} \;= \texttt{0000 0000 ... 1111 1111} \quad . \tag{1}$$

Again, we plot the error probabilities for falsely accepting and rejecting message bits (see Figure 4). It should be noted that the results for the sync and the message bits are approximately similar. The slight difference between sync bits and message bits (see Table 1) can be explained by different technical settings of the two embedding modes.

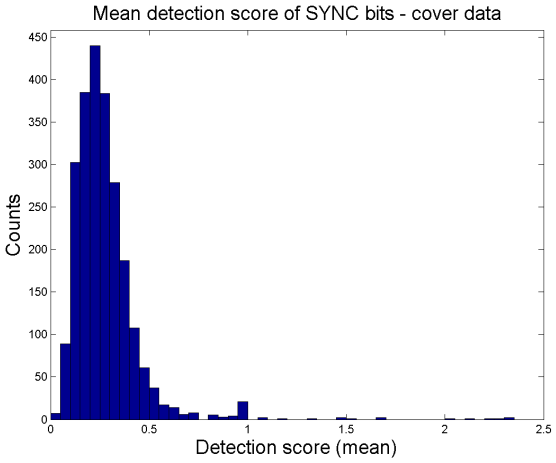


Figure 1. SYNC BITS: Histogram of absolute value of detector score for un-watermarked content

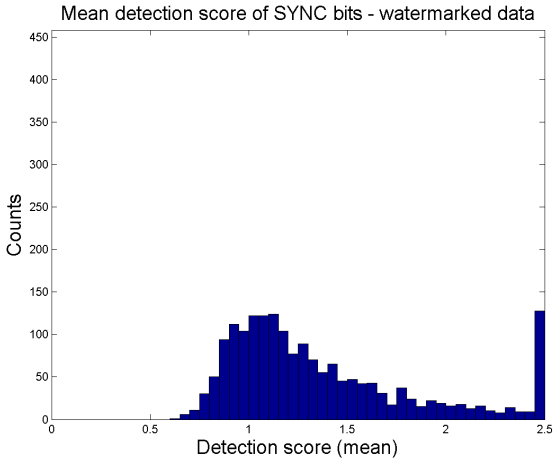


Figure 2. SYNC BITS: Histogram of absolute value of detector score for watermarked content

For the message bits, both error functions are again clearly separated, and their intersection can be found at $d \approx 0.7$. In the following, this value defines a decision threshold to distinguish un-watermarked content from correctly watermarked or maliciously re-embedded content.

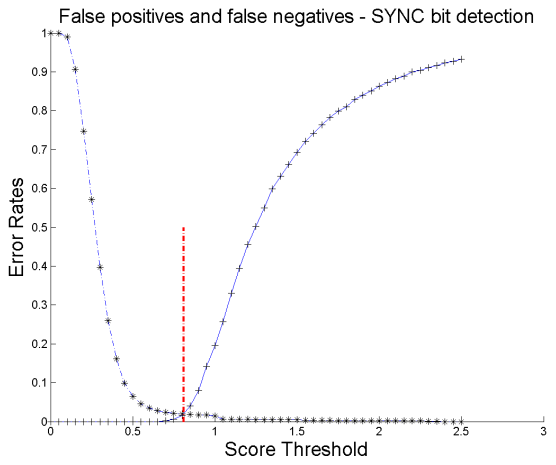


Figure 3. SYNC BITS: Error probabilities (crosses: false rejection of watermarked bits; asterisks: false acceptance of null hypothesis)

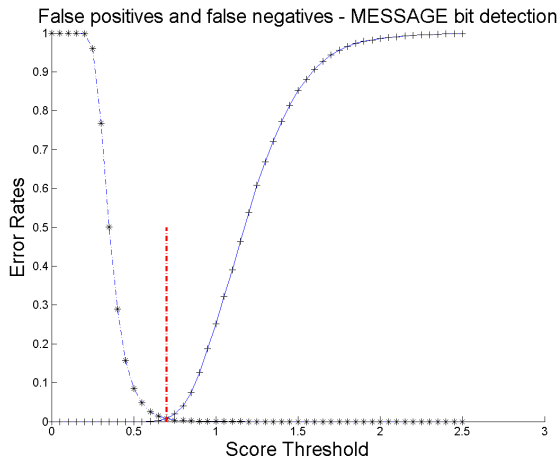


Figure 4. MESSAGE BITS: Error probabilities (crosses: false rejection of watermarked bits; asterisks: false acceptance of null hypothesis)

	lower 25%	median	upper 25%
cover (no message)	0.12	0.27	0.49
m_1 sync bits	1.02	1.22	1.59
m_1 message bits	0.75	1.07	1.48

Table 1. Median value and upper and lower 25%-quartiles for first embedding

These previous results did not consider at which bit position in the 64 bit message those detection scores were obtained. Thus, we also analyzed if the detection score is approximately equally distributed among the 64

message bits. For simplicity reasons, Figures 5 and 6 show the mean detection score for each retrieved bit. It can be seen that the decision threshold identified in Section 2 clearly separates watermarked from un-watermarked content.

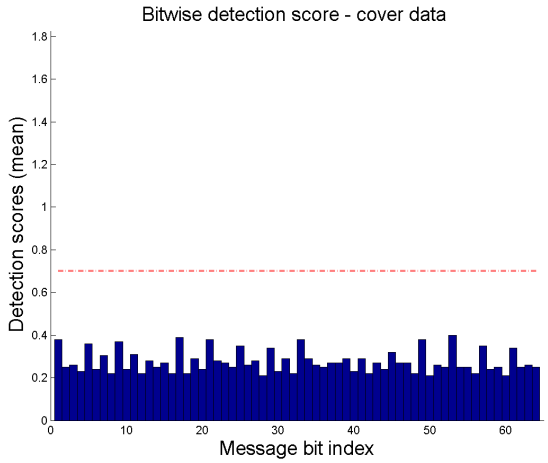


Figure 5. Detection score for single bits - un-watermarked cover (dashed line: decision threshold)

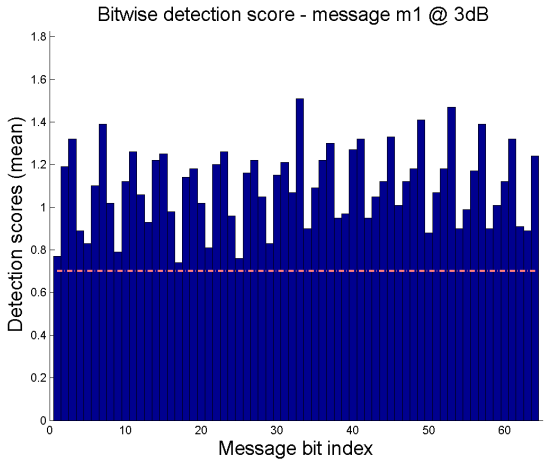
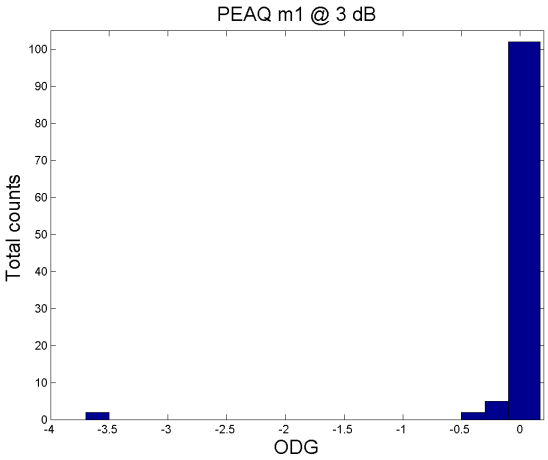


Figure 6. Detection score for single bits - watermark message m_1 (dashed line: decision threshold)

3.2.2. Objective sound quality

Finally, we analyzed which level of quality degradation was caused by the watermark embedding. Audio quality loss was estimated in terms of *objective difference grades* (ODG) using the *Opera Objective Perceptual Analyzer* by *Opticom* using the PEAQ[5] model. As can be seen from Figure 7 the distribution of ODG values is clearly



Explanation:

ODG	sensation of sound difference
0	imperceptible
1	perceptible, not annoying
2	slightly annoying
3	annoying
4	very annoying

Figure 7. Objective sound quality assessment for single embedding (histogram among all files of average ODG), mean = -0.03; definition of *objective difference grades (ODG)*

centered around zero. Closer analysis of the outliers (see left plot in Figure 7, near ODG=-3.5) showed that they were cause by incorrect delay compensation during the *Opera* analysis. That is, single watermark embedding does not introduce audible or even annoying artifacts here.

3.2.3. Robustness to lossy compression

To allow a comparison of our results with common tests of watermarking robustness against lossy compression, we analyzed the detection score for mp3 compression after the first embedding. To this end, the watermarked content was encoded to mp3 format at different bit rates. As can be seen from Figures 8 and 9, such lossy

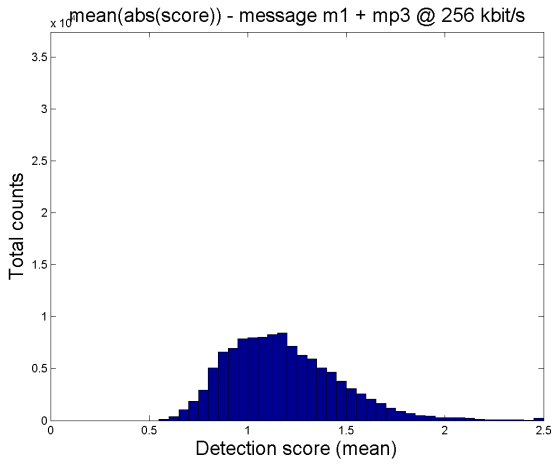


Figure 8. Absolute value of detector score after mp3 encoding at 256 kbit/s

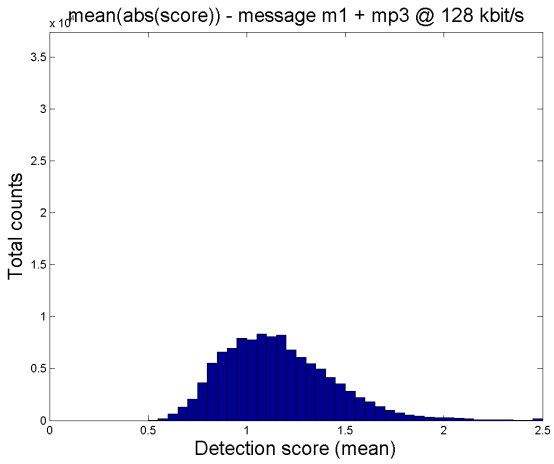


Figure 9. Absolute value of detector score after mp3 encoding at 128 kbit/s

	lower 25%	median	upper 25%
mp3 @ 256 kBit/s	0.75	1.07	1.48
mp3 @ 128 kBit/s	0.73	1.05	1.45

Table 2. Median value and upper and lower 25%-quartiles after mp3 re-encoding

compression does not significantly change the distribution of detection scores. The same can be seen comparing the results for the quartiles and medians in Table 2 with Table 1. In addition, Table 3 shows that moderate mp3 encoding does not significantly reduce the number of correctly detected messages. That is, the embedded watermark is significantly robust to moderate mp3 compression. We will also show in a later section that lossy compression is transparent to identifying malicious re-embedding.

	correct messages
message m_1	1130
m_1 + mp3 @ 256 kBit/s	1136
m_1 + mp3 @ 128 kBit/s	1047

Table 3. Number of correctly detected watermark messages (single embedding)

3.3. Multiple embedding

3.3.1. Detection score

After the first embedding, we embedded a second watermark with message m_2, where

$$m_2 = \texttt{0x FF FF FF FF 00 00 00 00} \;\; = 11111111...00000000 \quad .$$

Message m_2 was chosen to be the *bit-wise inverse* of m_1, this experiment can be regarded as a *training stage* for forensic distinction between single and double embedding.

Here, the embedding is done on the previously watermarked content. This simulates multiple embedding when the cover data is not available. In contrast to our previous work,[8] the same secret key was used for the second embedding. As all other technical parameters were set identical to the first embedding, message m_2 is embedded at the same positions in the time-frequency domain as m_1.

Then, as outlines in the introduction in Section 2.2, we tried different levels of embedding strength, e.g. 3 to 9 dB above the masking threshold. In Figure 10 one can see that the distribution of detection scores is rather similar to un-watermarked content. Here, the median absolute detection score of 0.23 (see Table 4) is approximately similar to the result from unmarked audio content (0.27). Thus, at least an embedding strength of 6 dB needs to be applied for obtaining successful detection results.

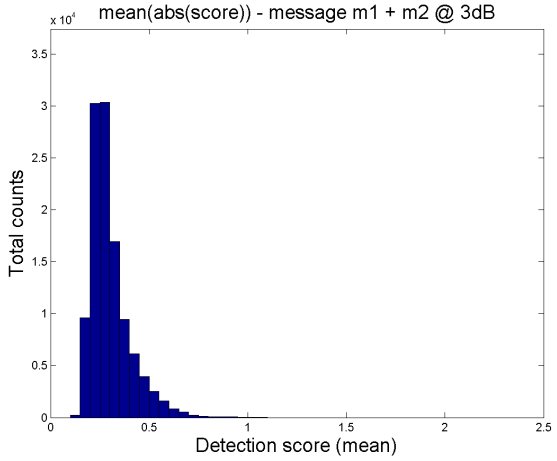


Figure 10. Absolute value of detector score after re-embedding with embedding strength 3 dB

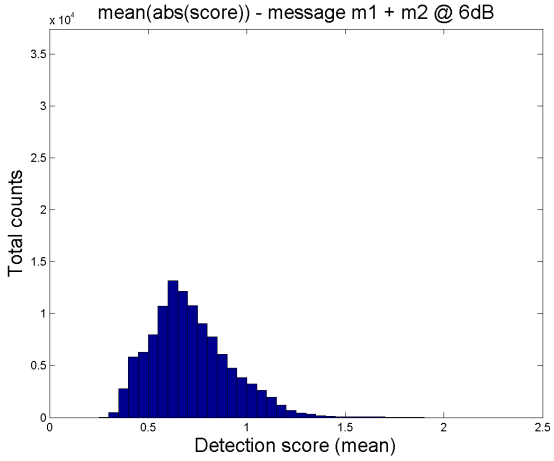


Figure 11. Absolute value of detector score after re-embedding with embedding strength 6 dB

	lower 25%	median	upper 25%
m_1 @ 3 dB + m_2 @ 3 dB	0.10	0.23	0.42
m_1 @ 3 dB + m_2 @ 6 dB	0.41	0.65	0.91

Table 4. Median value and upper and lower 25%-quartiles after re-embedding

Even if higher embedding strength of 6 dB was chosen, the median detection score is significantly lower (0.65) than for the single embedding (1.07), see Tables 4 and 2. This result is in accordance with the theoretical embedding model as the first embedding stage violates the basic assumption for the patchwork watermarking scheme (see Section 2) for any later embedding.

3.3.2. Objective sound quality

For the evaluation of the sound quality, the message

$$m_2' = \texttt{0x AA AA AA AA AA AA AA AA} \;\; = \texttt{1010 1010 ... 1010 1010}$$

was chosen. Here, only 50% of the message bits of m_1 need to be changed by the embedding. This is more realistic as 50% of the message need to be changed in average when counterfeiting the first message. Again, almost *no* correct watermark message can be detected and retrieved if the second message is embedded "over" the first message with embedding strength 3 dB. Table Table 5 shows that at least an embedding strength of 6 dB is required.

But although embedding a counterfeited message is possible, an attacker has to accept a *significant loss of sound quality*, as can be seen from Figure 12. Compared to Figure 7 one can see that the ODG histograms are significantly shifted to the left. For a significant number of audio files the sound quality degradation becomes *perceptible* and even *annoying*.

	correct messages
$m_1 + m_2$ @ 3 dB	6
$m_1 + m_2$ @ 6 dB	1627
$m_1 + m_2$ @ 9 dB	1969

Table 5. Number of correctly detected watermark messages (double embedding)

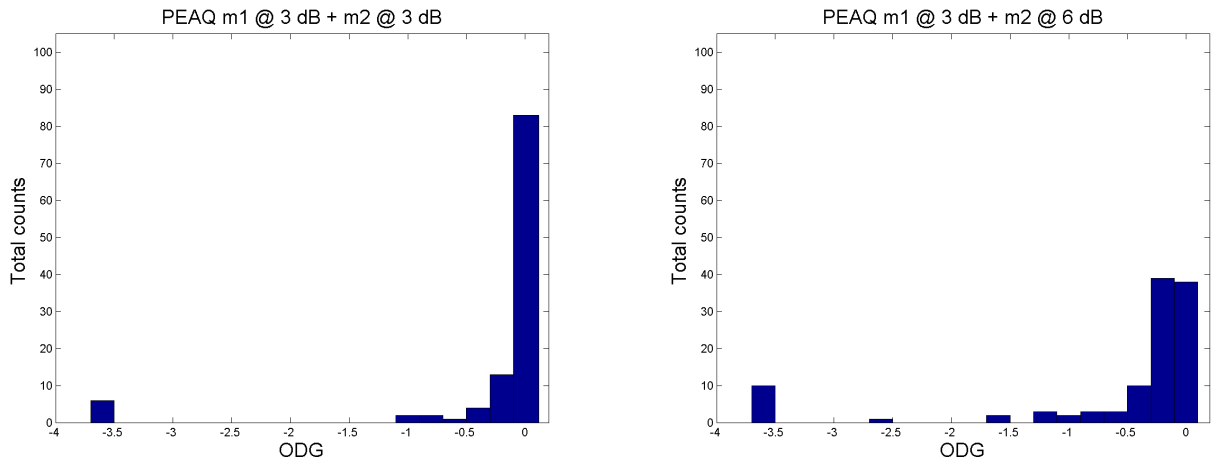


Figure 12. Objective sound quality after multiple embedding; left: mean ODG re-embedding at 3 dB (mean = -0.27), right: re-embedding at 6 dB (mean = -0.57)

3.3.3. Multiple embedding versus lossy compression

In addition to the previous simulations, we analyzed the sensitivity of our identification of multiple embedding to lossy compression. Therefore, the watermarked audio content was subject to lossy compression before *and* after the counterfeited message m_2' was embedded. We used mp3 re-encoding at different bit rates. Figure 13 shows the median of the absolute value of detection score for each of the 64 message bits.

The detection score of the attacked audio shows an interesting alternating sequence of high and low detection scores. That can be understood from our selection of the first and second message embedded:

$$\begin{aligned} m_1 &= \texttt{0000 0000 ... 1111 1111} \\ m_2' &= \texttt{1010 1010 ... 1010 1010} \end{aligned}$$

Here, the absolute value of the score is decreased when two different message bits partially "annihilate" each other (for example in the first bit index) while it will be increased when identical values are embedded (like in the second bit index, for example).

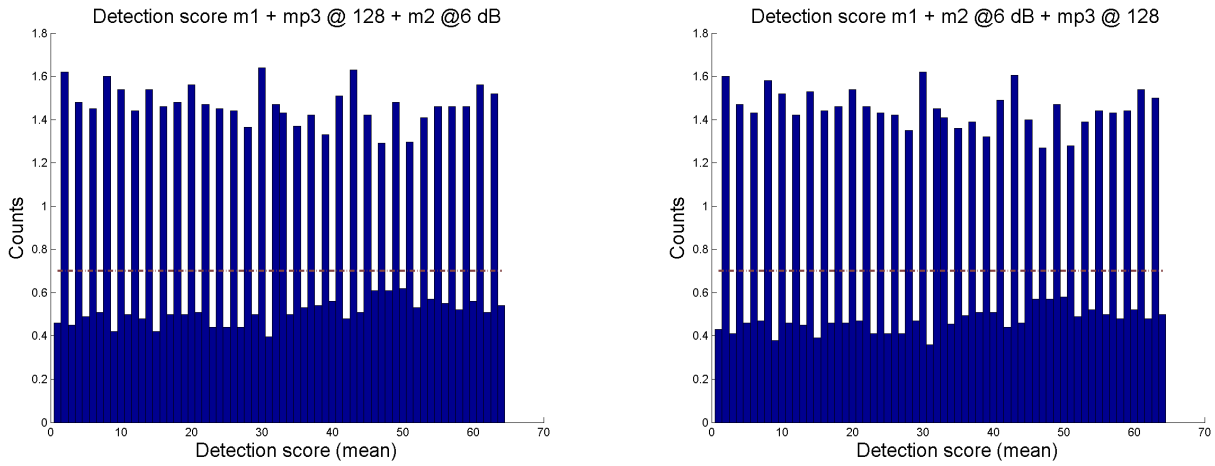


Figure 13. Median of detection score for each message bits - left: mp3 encoding before second embedding; right: mp3 encoding before second embedding - dashed horiz. line: decision threshold as obtained above

Obviously, the score of half of the bits in Figure 13 is significantly smaller than the decision threshold obtained in the previous sections. This behavior can be expected from the results in the previous sections. That means that lossy encoding is transparent also for the detection of multiple embedding. In addition, that means that an attacker can not use lossy compression to disguise malicious re-embedding.

4. CONCLUSION AND FUTURE WORK

In this work, we discuss a particular security attack on digital watermarking, namely embedding a second watermark message using the same key. This issue is of relevance whenever a *public* detection and retrieval of a watermark message is required or, at least, can not be avoided. This is the case in copyright watermarking applications where the detector algorithm and the correct key needs to be available to a large set of distributed users or devices. Here, when watermark keys leak from a protected environment maliciously or accidentally to the public or to the *Darknet*[3] the watermarking scheme can be compromised.

Therefore, we introduce a forensic approach in the context of spread-spectrum / patchwork watermark detection for audio data. We analyze the suspicious watermarked media by statistical means without having access to the un-marked cover. Especially the comparison of the detection score from the potentially attacked message bits with the score of the related un-attacked sync pattern seems promising for the detection of re-embedding. As result from this first experimental evaluation, there is strong evidence that an adaptation to existing watermarking detection algorithms allows to evaluate if the audio content has previously been subject to watermarking with the same symmetric key. Attackers can only circumvent such enhanced detection if they accept a significant loss of sound quality.

Even when an attacker has the key and the embedding algorithm at hand, he can not be sure to create a copy of the cover marked with his message that can stand against a detailed forensic analysis without introducing audible artifacts. This again is in some cases an acceptable alternative to asymmetric embedding and is based on the non-reversible nature of the utilized watermarking algorithm.

As our approach and the related patchwork mechanisms involved are independent from the media type, our contribution is expected to be applicable to image and video watermarking, as well. In future work, we will address for example the psychoacoustic properties. It can be expected that the first, second, third etc. stage of embedding will gradually modify the psychoacoustic properties of the audio content significantly. Those modifications might provide forensic approaches for detection of multiple embedding.

ACKNOWLEDGMENTS

This work was supported by the *CASED Center for Advanced Security Research Darmstadt*, Germany, (*http://www.cased.de/en.html*), and the Physical Cryptography Project at the Technische Universität München, Germany (*http://www.pcp.in.tum.de*).

REFERENCES

1. André Adelsbach, Stefan Katzenbeisser, and Helmut Veith. Watermarking schemes provably secure against copy and ambiguity attacks. In *DRM '03: Proceedings of the 3rd ACM workshop on Digital rights management*, pages 111–119, New York, NY, USA, 2003. ACM.
2. W. Bender, D. Gruhl, N. Morimoto, and A. Lu. Techniques for data hiding. *IBM Systems Journal, MIT Media Lab*, 35(3,4):313–336, 1996.
3. Peter Biddle, Paul England, Marcus Peinado, and Bryan Willman. The darknet and the future of content protection. In *Security and Privacy in Digital Rights Management, ACM CCS-9 Workshop, DRM 2002, Washington, DC, USA, November 18, 2002*, pages 344–365, 2003.
4. Gal Hachez and Jean-Jacques Quisquater. Which directions for asymmetric watermarking? In *Proceedings of the XI European Signal Processing Conference (EUSIPCO 2002*, pages 283–286, 2002.
5. ITU International Telecommunications Union. Method for objective measurements of perceived audio quality. *ITU-R Recommendation BS.1387*, May 1998.
6. B. Purna Kumari and V. P. Subramanyam Rallabandi. Modified patchwork-based watermarking scheme for satellite imagery. *Signal Processing*, 88(4):891–904, 2008.
7. Petitcolas, Fabien A. P., Anderson , Ross J., and Kuhn, Markus G. Attacks on copyright marking systems. In David Aucsmith, editor, *Lecture Notes in Computer Science Second Workshop on Information Hiding, Portland, Oregon, USA, April 14–17, 1998, ISBN 3540653864*, volume 1525, pages 218–238, September 1998.
8. Martin Steinebach. *Digitale Wasserzeichen fuer Audiodaten*. Shaker Verlag Aachen, 2003.
9. Martin Steinebach and Sascha Zmudzinski. Evaluation of robustness and transparency of multiple audio watermark embedding. In III Delp, Edward J., Ping Wah Wong, Jana Dittmann, and Nasir D. Memon, editors, *Proceeding of SPIE Int., Symposium on Electronic Imaging, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X, San Jose, USA*, volume 6819, 2008.
10. Y. Wang and P. Moulin. Capacity and optimal collusion attack channels for Gaussian fingerprinting games. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 6505 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, February 2007.
11. In-Kwon Yeo and Hyoung Joong Kim. Modified patchwork algorithm: A novel audio watermarking scheme. In *Proceedings of the International Conference on Information Technology: Coding and computing (ITTC '01), 02–04 April 2001, Las Vegas, USA*. IEEE, April 2001.