StirMark Benchmark: Audio Watermarking Attacks Based on Lossy Compression

Jana Dittmann, Andreas Lang and Martin Steinebach[1]

Fraunhofer Institute IPSI, Darmstadt, Germany

ABSTRACT

StirMark Benchmark is a well-known evaluation tool for watermarking robustness. Additional attacks are added to it continuously. To enable application based evaluation, in our paper we address attacks against audio watermarks based on lossy audio compression algorithms to be included in the test environment. We discuss the effect of different lossy compression algorithms like MPEG-2 audio Layer 3, Ogg or VQF on a selection of audio test data. Our focus is on changes regarding the basic characteristics of the audio data like spectrum or average power and on removal of embedded watermarks. Furthermore we compare results of different watermarking algorithms and show that lossy compression is still a challenge for most of them. There are two strategies for adding evaluation of robustness against lossy compression to StirMark Benchmark: (a) use of existing free compression algorithms (b) implementation of a generic lossy compression simulation. We discuss how such a model can be implemented based on the results of our tests. This method is less complex, as no real psycho acoustic model has to be applied. This model can be used for audio watermarking evaluation of numerous application fields. As an example, we describe its importance for e-commerce applications with watermarking security.

Keywords: Watermarking, Lossy Compression, StirMark, Audio, Attack

1. BACKGROUND AND MOTIVATION

The growing number of attacks against watermarking systems [1],[2], [3] has shown that far more research is required to improve the quality of existing watermarking methods. With StirMark Bechmark (SMBM), introduced in [4], we want to provide a well-defined benchmark for watermarking robustness and security. Researchers and watermarking software manufacturers just need to provide a table of results, which gives a good and reliable summary of the performances of the proposed scheme. So end users can check whether their basic requirements are satisfied. Researchers can compare different algorithms and see how a method can be improved or whether a newly added feature actually improves the reliability of the whole method. As far as the industry is concerned, risks can be properly associated with the use of a particular solution by knowing which level of reliability each contender can achieve.
While most research activities concentrate on still image watermarking attacks, we identified a need of audio watermarking evaluation and described first directions and results in [5]. There we analyzed watermarking robustness against various signal transformation attacks and also recognized the need of further tests regarding lossy compression algorithms. These algorithms are commonly used for audio distribution via the Internet and are therefore watermarking robustness against them is of interest for all parties providing watermarked content at web shops, online news centers or similar application areas.

As a first step towards a widely accepted way to evaluate watermarking schemes we started to implement an automated benchmark server. Users can send a binary library of their scheme to the server which in turns runs a series of tests on this library and keeps the results in a database accessible to the scheme owner or all interested parties through the Web.

1.1 Digital Watermarking

Digital watermarking is a technology for copyright protection and protection against unauthorized access and modification of multimedia material. Robust digital watermarking can be used to claim copyright protection by embedding authors or

[1] Correspondence: Email: martin.steinebach@ipsi.fraushofer.de

Steinebach, Lang, Dittmann; StirMark Benchmark: Audio watermarking attacks based on lossy compression", Photonics West 2002, 19 - 25 January 2002, Electronic Imaging 2002: Science and Technology; Multimedia Processing and Applications, Security and Watermarking of Multimedia Contents IV, San Jose, CA, USA, Proceedings of SPIE, Edward J. Delp III, Ping Wah Wong (Eds.), Vol. 4675, Bellingham, Washington, USA, S. 79 - 90, ISBN 0-8194-4415-4, 2002

producers information. Basically the principle is to hide a watermark W in a given data C (cover) by modifying some of its characteristics. Numerous watermarking algorithms have been described. For an overview regarding different applications and characteristic parameters see [6]. References [7] to [11] provide a selection of audio watermarking algorithms.

1.2 Lossy Compression

Audio data is often transferred over networks, e.g. in internet radio or live music streams. Lossy compression is the key technology to enable audio transmission over networks, as they reduce the amount of required bit rates. Without this, the required bit rates would be too high to be acceptable for most applications. Lossy compression reduces the size from factor 1:10 to 1:14 without significant loss of perceived quality. There are many different lossy compression algorithms. Well know algorithms are mp3, Ogg, VQF and WMA. They all share some basic principles (figure 1):

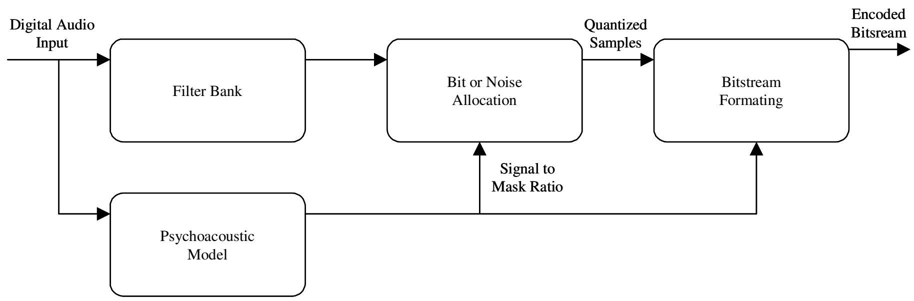

Figure 1: Encoder Block Diagramm

- Filter and analyse digital audio data. There are low and high pass filters.
- Division into frequency bands. Filter banks provide frequency and time separation and quantisation.
- A psycho acoustic model calculates the noticeable noise level for each band. It masks signals in frequency and time. By the time it distinct the pre and post noticeable signal. The output is a signal to mask ratio for each band.
- Bit or noise allocation chooses how many bits are applied for each sub band to encode the audio information.
- A bit stream formatter encodes the data and often uses further compression like e.g. Huffman codes.

The number of filters subbands or the time or frequency resolution can be different for compression algorithms. Another difference is the psycho acoustic model. Differential algorithm mask differential signals. Therefore it is possible that two lossy compression algorithms mask different signals.

In this paper we discus the changes between an original and a compressed file. We will show the importance of this manipulation of a sound file to know what going on and what happened to a possible watermark. In section 2 we describe our test scenarios, one for robustness evaluation and one for analysis of changes to the audio data. The test results are provided in section 3. We summarize the collected data and identify significant challenges for our SMBM environment. Based on this, we discuss possible automation approaches in section 4. In section 5 a example scenario is provided. Different lossy compression applications in a online music store are shown. In our conclusion we summarize our results and give some future research topics for SMBM.

2. TEST SZENARIO

In this section we describe the different test scenarios for lossy compression. We identify suitable audio test files regarding the addressed test environment and we discus our choice of lossy compression algorithms. For each of these algorithms, we provide a brief description and summarize their differences and similarities.

To made this tests we had followed situation. We took many different audio dada files with different type of content. So we had classic, rock/pop, spoken text with male and female voice, different types of noise and different types of tones and

variations of tones. The original files are all in 44,1kHz sample rate and mono. If an algorithm needed a stereo file the file was converted before.
We use different compression algorithms for our evaluation. One of the well known algorithm is the mp3 encoder from Fraunhofer. This encoder is the reference of all mp3 encoders and was the first encoder which used a psycho acoustic model to compress audio data. The LAME encoder is a open source encoder and to find under http://www.mp3dev.org/mp3/ . This encoder use the GPSYCHO model - an open source psycho acoustic and noise shaping model. An other open source encoder is the Blade encoder (http://bladeenc.mp3.no). This encoder is under the LGPL license. The third free open source encoder under GPL is the Ogg Vorbis encoder (http://www.xiph.org/ogg/vorbis/index.html). The Ogg algorithm uses different mathematical principles to compress the audio data. It claims to produce a better sound quality at the same bit rate then mp3. Another lossy compression encoder is VQF from Yamaha (http://www.vqf.com). This format can compress files approximately 25-35% more than mp3 files and claims to produce a better quality then mp3 at the same bit rate. On the other hand, VQF requires more CPU time then as a mp3 player. The last applied lossy compression encoder is the WMA encoder by Microsoft. This algorithm is said to have a lower sound quality than mp3 at high bit rates but to offer better quality at low bit rates.

This is the list of all used algorithms and their version numbers:

- Fraunhofer mp3 v3.1
- LAME mp3 v3.87
- Blade mp3 v0.91
- Vorbis Ogg v0.4
- Yamaha VQF v2.60
- Microsoft WMA v1.2.4

Test 1: Robustness of watermarks against lossy compression

In this test scenario we evaluate the robustness of a watermark against lossy compression.
This test is divided into two subtests I and II. In the first part we analyse robustness of different watermarking algorithms. The goal is to identify watermarking performance in typical Internet application scenarios.
We selected mp3 as the lossy compression applied here. Currently this is the compression used most often for numerous applications. For mp3 encoding we used the Lame encoder because of its good scripting capabilities. Five different watermarking algorithms were available to us. Four were commercial evaluation versions, the last one was our own prototypic implementation. Due to non-disclosure agreements, we will identify the algorithms only by the letters A to E.

For subtest I we choose eight different sound files as typical examples for audio data with commercial value to be compressed. The file format was 44,1 KHz sample rate, 16 bit and stereo. They consist of classical music, rock pop, different types of noise and spoken text. They represent real world examples of audio files which are sold or transmitted.
A watermark was embedded in each original file. The marked file was encoded with a bit rate from 32 kbit/s to 320 kbit/s by using the Lame encoder, producing 15 compressed versions of each audio file. After compression, we tried to retrieve the embedded watermarks. The lowest applied bit rate is 32 kbit/s. Usually an audio with less that this bit rate has too poorly quality to make it necessary to protect it with a watermark. So our test range is 32kbit/s to 320kbit/s.

Subtest II concentrated on high robustness watermarking algorithms and very critical audio data. As critical audio data the SQAM files (http://sound.media.mit.edu/mpeg4/audio/sqam/) were used. They are part of the audio quality evaluation package provided by the EBU and have also been applied at mp3 quality evaluation. We selected two algorithms, A of subtest I and E, our own prototype, to compare their robustness against lossy compression. A 32 bit message was embedded. We counted the number of times the watermark had been detected successfully. Our goal was to identify the performance of high robustness watermarks regarding lossy compression and the correlation of payload and robustness.

Test 2: Changes in audio characteristics

To build an attack model of different lossy audio compression algorithms for SMBM, we need to identify how these algorithms effect the audio data. Therefore in our first test, we compress a selection of 19 audio test files with all audio compression algorithms listed above to analyze the changes in the audio files after de-compression. As for some compression algorithms no tool for changing the data back to uncompressed PCM is available, we used the internal PCM recording capabilities of our soundcard to get a wav dump of the format players output.

We chose the following characteristics to analyze:

- **Frequency**: The most prominent frequency at the center of the file. This reflects the use of filters, as removing low or high bands will change this value. This is less detailed then a spectrum, but more easy to compare.
- **Minimal sample value / Maximal sample value:** Minimum/Maximum Sample Value: The maximum and minimum sample values in the range.
- **Peak amplitude:** The maximum sample value given in decibel form.
- **Minimum RMS power, Maximum RMS power, Average RMS power Total RMS power:** RMS is the Root Means Square and the values minimum, maximum and average RMS power provide the information about the minimum,maximum and average RMS given in decibel form. The same is with the value of total RMS power.

These characteristics do not provide a detailed insight in the workings of the compression algorithms, but enable a comparison between them. One important point is to identify the differences between the algorithms. Given the large number of test files we also determine if changes depend on the audio material. We use Cool Edit 2000 by Syntrillium to gather the information.

Lossy audio compression algorithms most often offer different bit rates of the resulting compressed audio data. Two options were available for this test: We could try to use the same characteristics for all algorithms or use different bit rates. As our goal is not a comparison between the algorithms but a overall model of the compression effects, we chose the second option. E.g. WMA is used with a very low bit rate as it is known to provide best results at low bit rates (see test results at http://www.fortunecity.com/tinpan/miles/528/audiocomp1.htm) compared to other algorithms. But we also use similar compression rates, e.g. for mp3 and ogg to identify differences between compression algorithms. We want to simulate a common user who has the choice regarding compression algorithms and will try to use the best one for his application, e.g. mp3 or ogg for high quality compression and wma for preview functions.

3. TEST RESULTS

In this section, we present the results of our tests. We discuss differences between the effect of the lossy compression algorithms on the different test files and on the embedded watermarks. We use selected examples of results for the tested scenarios.
Due to limited space we cannot provide all test results here. Please find more results including audio examples at http://ms-smb.darmstadt.gmd.de/paper.

3.1 Test 1 results: Watermark robustness

Here we present the results of our tests regarding the robustness of audio watermarking algorithms against lossy compression. Table 1 shows a example of our results of subtest I: The letters A to D represent the watermarking algorithms. A *yes* means that the watermark was retrieved completely and a *no* means that no watermark was detected. If the retrieval had to fall back to more time-consuming methods or parts of the watermarks were broken, it is called *weak*.

bit rate	A	B	C	D
32	No	No	No	No
40	No	No	No	No
48	No	No	No	No
56	No	No	No	No
64	No	No	No	No
80	No	No	No	No
96	No	Yes	No	No

112	Weak	Yes	No	Yes
128	Weak	Yes	Yes	Yes
160	Weak	Yes	Yes	Yes
192	Weak	Yes	Yes	Yes
224	Weak	Yes	Yes	Yes
256	Weak	Yes	Yes	Yes
320	Weak	Yes	Yes	Yes

Table 1: Watermark algorithms and robustness against mp3

No watermarks could be retrieved after compression at a bit rate of 80 kbps or lower. Algorithm A is only partially robust against compression with a bit rate higher than 112 kbit/s. The algorithms B, C and D are robust against typically applied compression rates. Figure 1 compares the performance of 4 watermarking algorithms (called A to D). The Y axis provides the number of detected watermarks after compression. We used 9 different sound files, in some of which multiple detections were possible.

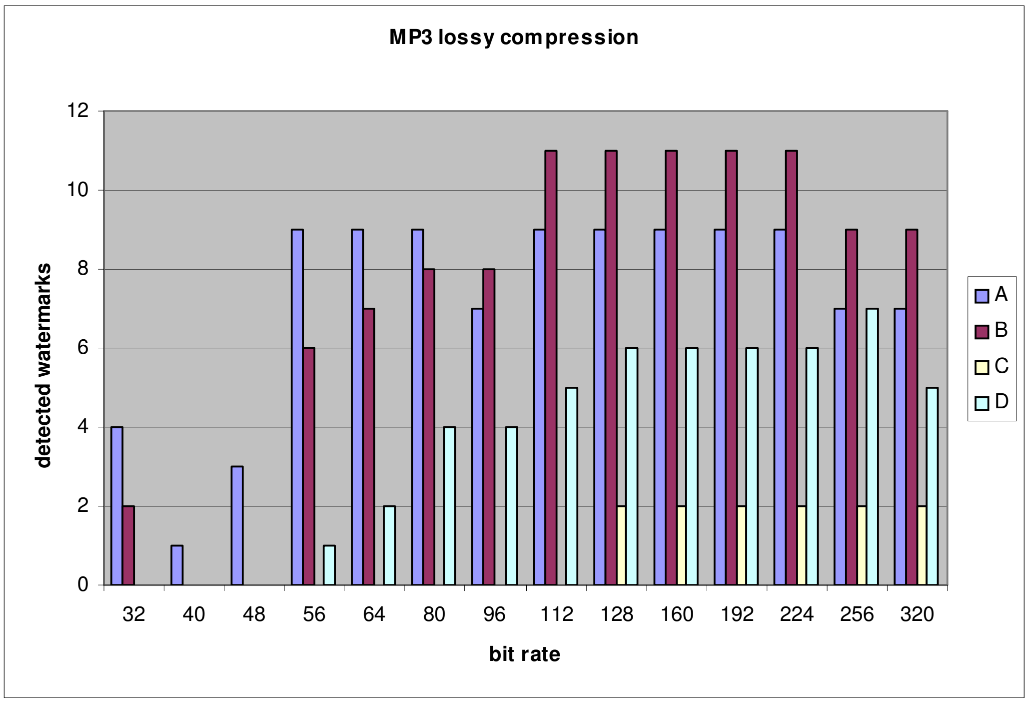

Figure 1: mp3 compression results capacity comparison

Some watermarking algorithms have a better robustness against mp3 compression as others. For example algorithm A has a good robustness in many bit rates. Algorithm B has a higher capacity at most bit rates but is less robust against lower bit rates. Although mp3 is very popular there are many watermarking algorithms not robust against being compressed by it. Therefore these watermarking algorithms not suited for many applications. For SMBM a lossy compression scenario will be integrated able to simulate different bit rates and compression schemes.

Table 2 summarizes our results of subtest II. It was not possible for algorithm B to embed a watermark in audio file “horn” and algorithm A could embed the watermark only once. In comparison, both algorithm could embed a watermark in the example “Quarte”. Algorithm A detected the watermark twice, algorithm B once. Both files were encoded by the Lame encoder. If a watermark was embedded by E, then it resistant against mp3 compression. A is only resistant against mp3 compression at bit rates higher or equal 112kbit/s.

Horn	A	E
32	0	0
40	0	0
48	0	0
56	0	0
64	0	0
80	0	0
96	0	0
112	1	0
128	1	0
160	1	0
192	1	0
224	1	0
256	1	0
320	1	0

Quarte	A	E
32	0	1
40	0	1
48	0	1
56	0	1
64	0	1
80	0	1
96	0	1
112	2	1
128	2	1
160	2	1
192	2	1
224	2	1
256	2	1
320	2	1

Table 2: Mp3 test results. Left: critcal sound file Right : typical sound file

As already stated above, more test results are available at http://ms-smb.darmstadt.gmd.de/paper.
This result presents us that some watermarking algorithms can embed or can not embed a watermark in an audio file depends on the content of it. But this watermark is very robust against mp3 encoding.

Figure 2 compares the detection rates of A and E applied to the SAQM test files at different compression rates.

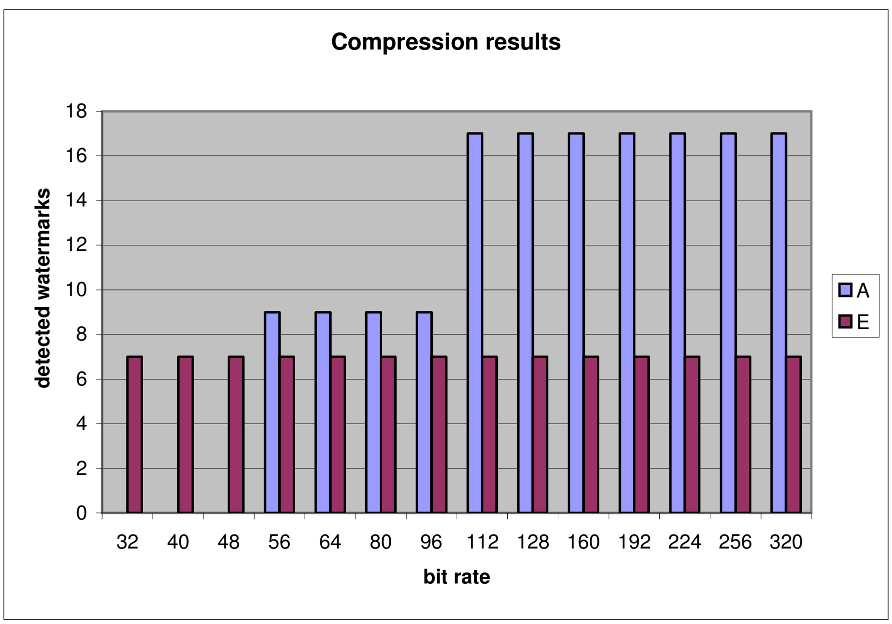

Figure 2: Capacity and mp3 lossy compression with different bit rate

A has a higher capacity then E but its robustness is worse then E at bit rates below 56 kbit/s. At 96 kbit/s first data loss occurs. Algorithm E can retrieve the correct watermark at every bit rate. Here the capacity is not affected .

This result shows the correspondence of robustness and capacity. A offers a high capacity, E a high robustness. It is important to evaluate different bit rates with different algorithms. As there are many different application and requirements, not only typically applied bit rates should be evaluated, but a wide range of possible compression rations are of interest for our SMBM model. Our model for lossy compression attacks therefore must be able to simulate high quality compression at high bit rates as well as low bit rate compression, which is e.g. used for previews.

3.2 Test 2 results: Changes of characteristics

With test 2 we want to characterize the effects of the compression algorithms on a selection of audio files. Due to the large number of resulting test samples (19 files x 6 algorithms x 9 features) we will first give some representative examples and then discuss the overall results.

High quality compression

Here we compare the mp3 blade codec and ogg, both a 128 kbps / mono. This is a unusual high bit rate for audio compression, reducing file size only to about 20 % of the original.

In figure 3 we show the frequency changes of both compression schemes. They are given in percentages, calculated (original feature/changed feature)*100). While most example files have not been effected strongly by the compression, a few files stand out: E.g. the test files 17 and 18, consisting of single frequency signals have been strongly changed by ogg, producing a very high detected frequency. There are increases and decreases of detected frequencies. An interesting fact is that when the frequency has been changed only by a small amount (examples 7, 15, 16, 19), this is the case both for ogg and mp3. It seems the algorithms work similar on some material, maybe the one easy to compress, but react differently on critical audio files.

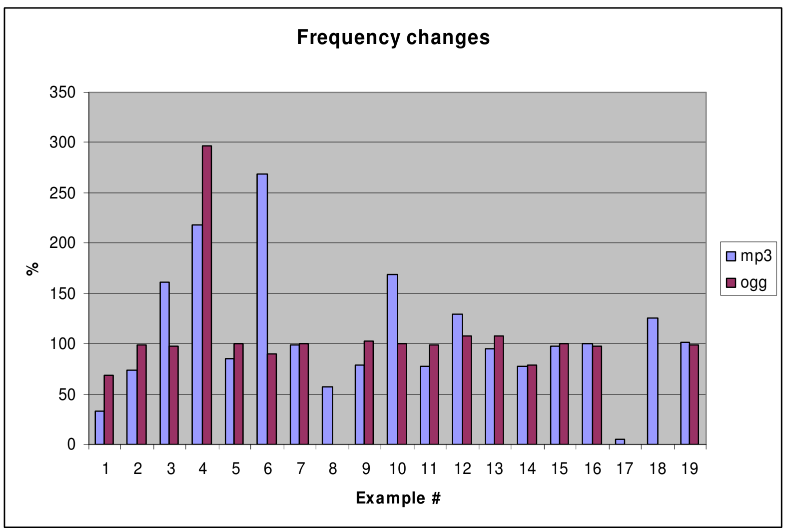

Figure 3: High quality compression test (Frequency changes)

The peak volume changes (figure 4) have only be changed slightly in most cases. Most peak volumes seem to be unaffected by compression algorithms as they are not masked. The only exception is mp3 at example 10 and 12 where the peak of the mp3 is changed about 50 % compared to the original.

Peaks mask other parts of the audio file. This indicates a dynamic volume compression does not take place in high quality lossy compression algorithms and is therefore unnecessary for our model. We can verify this be looking at the average RMS (figure 5), which is also not subject to significant changes most of the times.

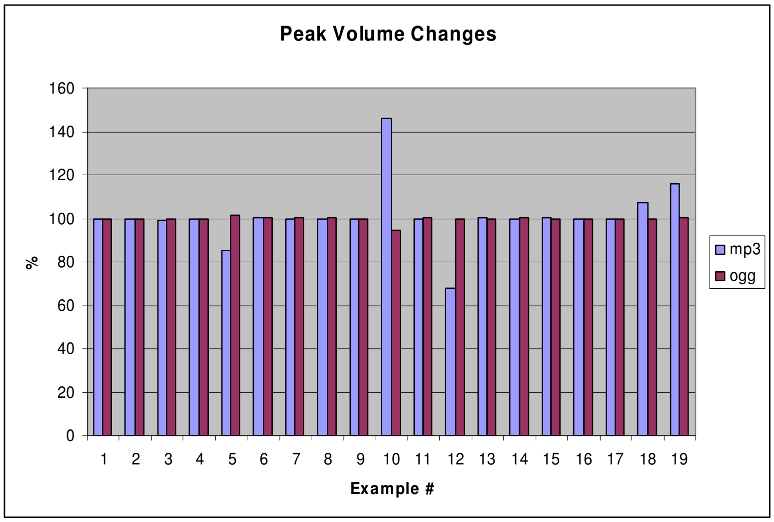

Figure 4: High quality compression test (Peak volume changes)

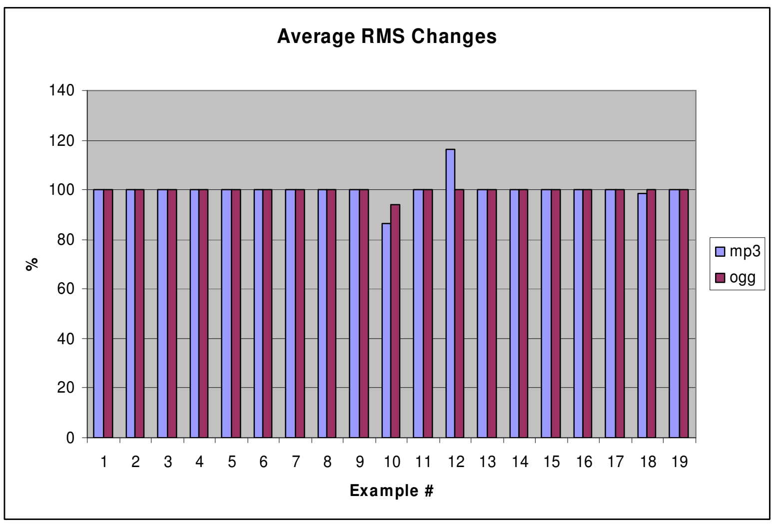

Figure 5: : High quality compression test (Average RMS changes)

Low quality compression

For our low quality compression comparison, we chose Yamaha VQF at 48K/bits and Microsoft WMA at 32 kbit/s for a 44.100Hz and mono source file. Given the low bit rates and therefore the necessity to remove more information of the original audio file, the impact on the characteristics is much stronger. We used 21 examples here, two more then in the high quality tests; all test files of the high quality test are included.

Figure 6 shows the frequency changes. VQF produces all the strong peaks, with a maximum at example 14 with 1500 % difference to the original. WMA seems to be more capable of reproducing the true frequencies at low bit rates. The extreme changes in some examples will make it necessary to include options of strong frequency manipulations in a model for low bit rate compression.

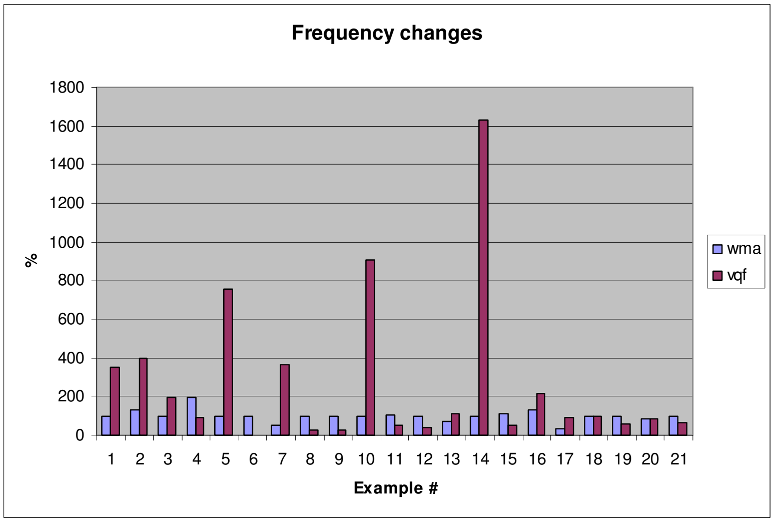

Figure 6: Low quality compression (Frequency Changes)

The peak volume changes (see figure 7) are not much stronger then with high quality compression with the exception of example 1. Here the original is very loud, the peak is at about –1 dB, while after the compression the peak is at about –8 dB. Therefore for loud originals and with low bit rate compression a dynamics function must be present in a SMBM model. The average RMS power (figure 8) is also subject to stronger changes then with high quality compression, but compared to frequency and peak volume these changes are rather small.

The low quality compression test show that in spite of the results from high quality compression, a volume change model, like controlled dynamic compression is necessary to simulate the effects of lossy compression on audio material if one plans to cover the whole possible compression range.

The choice of different compression algorithms could also be the reason for the changed characteristics. But as both algorithms are said to perform well at low bit rates and SMBM plans to offer a complete model, this does not change the consequence given above.

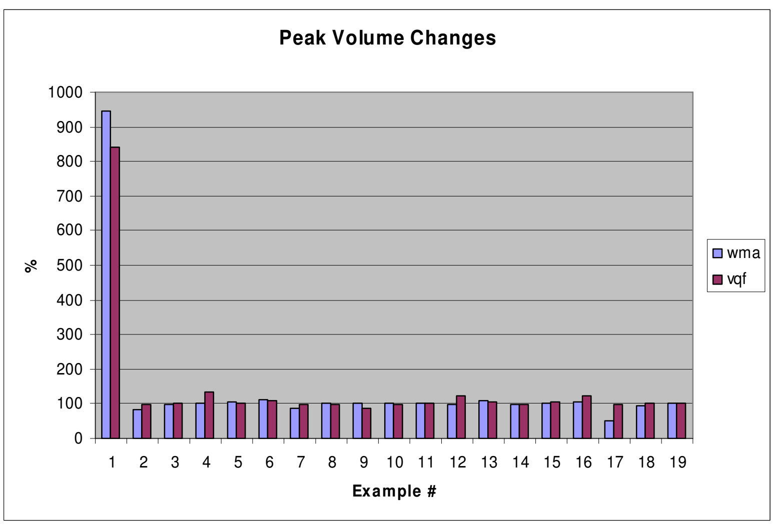

Figure 7: Low quality compression (Peak Volume Changes)

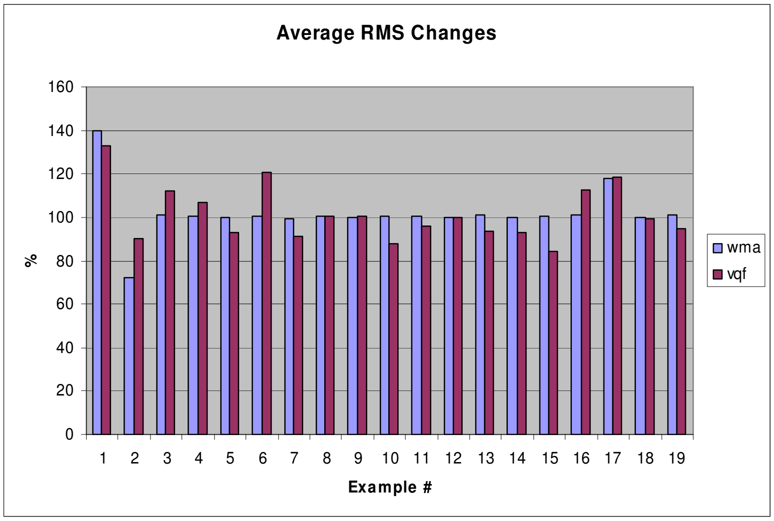

Figure 8: Low quality compression (Average RMS Changes)

4. LOSSY COMPRESSION INCLUSION IN STIRMARK BENCHMARK

Many applications are known which use lossy compression. Test 1 showed the importance of robustness evaluation of watermarking algorithms against mp3 as the most popular lossy compression format. In test 2 we analyzed the differences between different lossy compression algorithms and bit rates. It is important to know the effects of lossy compression on sound files to add a lossy compression model to the StirMark Benchmark list of attacks. It may not be necessary to add the feature of a real encoder. This will reduce performance requirements. Also source codes will not always be available.

Our concept to add lossy compression to SMBM is either to simulate different algorithms and bit rates or to include open source code lossy compression as attacks. In the second case, is not necessary to add the complete encoding process. We do not need to write the header or the file to a disk. Only the core of the lossy compression algorithm is important for us.

Different lossy compression algorithms have different effects on the audio file. They change various characteristics. To imitate the effects of different lossy compression algorithms we need at least use the following components:

- High, low and band pass filter
- Amplify manipulation and compressor
- Frequency analysis and manipulation
- Dynamic manipulation
- Noise
- Feature detection

Figure 9 shows the basic concept of our simulation: A watermark WM is embedded in an audio file. The file is then analyzed with respect to different features. The lossy compression model receives the features of the audio file, like e.g. the loudness or the frequency range and the parameters provided by SMBM. These describe the lossy compression algorithm and bit rate to be simulated. Now the model controls different attack algorithms like filters, dynamic compression or equalizers to reproduce the effects of the lossy compression algorithm on the marked audio file. A random number generator can be used to modify the parameter settings in a certain range to imitate unpredictable changes. A watermark detector tries to receive the watermark from the attacked audio file and as a result provides the number of successful detections .

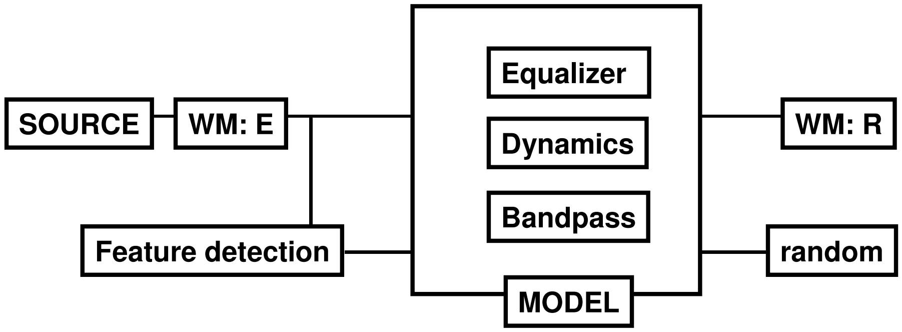

Figure 9: SMBM lossy compression simulation concept

5. APPLICATION EXAMPLE: AUDIO WEBSHOP WATERMARKING

In this section we provide an example for the necessity of watermarking robustness against lossy compression.
A web shop selling music files over the Internet is a common e-commerce concept. In a typical secenario, audio files will be provided by the copyright owners in CD-quality. The web shop stores them in high quality for CD-on-demand services and at two different compression qualities. A high bit rate version is available for online sales, a low bitrate version for previews or internet radio advertisement.
The web shop owner earns money by selling the content and is also responsible for protecting the right of the copyright owner. Therefore he wants to ensure that illegal copies auf audio files bought at this shop can be traceded or at least identified. He decides to embed copyright watermarks in the CD-quality material, as both compressed versions of the audio data are created from it. Chosing this way of protection, he has to ensure the following issues regarding the robustnes of the applied watermarking algorithm:

- All audio files have to be marked. If there are problems with some files, like in the "Horn" example of Test 1 / Subtest II, he can not be sure if all his content is protected
- The high quality compression must not remove the watermark. The resulting files are the products of his web shop and have to be protected.
- The low quality compression should not remove the watermark. It may be acceptable to lose the mark here, as there is only a reduced value of the files. But as there are many new buisness concepts regarding advertisment based on watermarking, his requirements could also include robustness against low quality compression

Another important question is whether it is possible to retrieve the watermark if the lossy compression algorithm is changed as these are constantly improving. Today mp3 is the common Internet format, but one day a new algorithm may take its place. If the watermarking algorithm applied to the CD quality audio is removed by the new compression format, all audio files have to be re-marked. If there are no stored originals available, this can be a major challenge.
The consequence for the shop owner is the necessity to evaluate the watermarking algorithms he is interested in because of their other characteristics like high transparency or low complexity regarding the performance at compression rates he plans to apply. He also has to identify his payload-requirements. Only the combination of robustness and payload will show if his requirements are met by the algorithm. SMBM will help him by offering an adjustable compression model.

6. CONCLUSION

In this paper, we identify the need for inclusion of lossy audio compression models into the SMBM evaluation suite for audio watermarking. The performance of currently available watermarking solutions is not satisfying all requirements a user could have in an e-commerce environment. The results of test 1 show a lack of robustness against low compression rates. It also proofs the connection of robustness and payload. In test 2 we analyze the impact of lossy compression on audio characteristics like frequency and volume peaks. Lowering bit rates have more effect on these characteristics then choosing another compression algorithm.
In conclusion to our test results, we discuss possibilities of simulating the compression effects. We can use existing compression algorithms if their source codes are available freely: They can be integrated into SMBM as attacks. This is similar to test scenario 1. For some of the algorithms the source codes are not public. Based on this information gathered in

test 2 we will implement a generic lossy compression simulation. To achieve this, we have to use different types of filters to affect the audio files in a similar way lossy compression algorithms do. For example, most algorithms use a low and a high pass filter to remove all frequencies the average listener can not perceive. This method will be less complex then the actual compression algorithms, as no real psycho acoustic model has to be applied.

Our final goal is to simulate an application environment as the ones mentioned above where lossy audio compression algorithms are applied, possibly together with other media manipulations like filtering or dynamic compression.

REFERENCES

1. Jonathan K. Su and Bernd Girod. Fundamental performance limits of power-spectrum condition-compliant watermarks. In Ping Wah Wong and Edward J. Delp, editors, proceedings of *electronic imaging, security and watermarking of multimedia contents II*, vol. 3971, pp. 314–325, San Jose, California, U.S.A., 24–26 January 2000. The Society for imaging science and technology (I.S.&T.) and the international Society for optical engineering (SPIE). ISSN 0277-786X. ISBN 0-8194-3589-9.
2. Martin Kutter. Watermark copy attack. In Ping Wah Wong and Edward J. Delp, editors, proceedings of *electronic imaging, security and watermarking of multimedia contents II*, vol. 3971, pp. 371–380, San Jose, California, U.S.A., 24–26 January 2000. The Society for imaging science and technology (I.S.&T.) and the international Society for optical engineering (SPIE). ISSN 0277-786X. ISBN 0-8194-3589-9.
3. Fabien A. P. Petitcolas, Ross J. Anderson and Markus G. Kuhn. Attacks on copyright marking systems. In David Aucsmith, editor, *second workshop on information hiding*, in vol. 1525 of *lecture notes in computer science*, pp. 218– 238, Portland, Oregon, U.S.A., 14–17 April, 1998. ISBN 3-540-65386-4.
4. Steinebach, Martin; Petitcolas, Fabien A. P.; Raynal, Frederic; Dittmann, Jana; Fontaine, Caroline; Seibel, Christian; Fates, Nazim; Croce Ferri, Lucilla (2001). StirMark Benchmark: Audio watermarking attacks. In: Int. Conference on Information Technology: Coding and Computing (ITCC 2001), April 2 - 4, Las Vegas, Nevada, IEEE Computer Society, Los Alamitos, CA, USA, pp. 49 - 54, ISBN 0-7695-1062-0, 2001.
5. Petitcolas, F. A. P.; Steinebach, Martin; Raynal, F.; Dittmann, Jana; Fontaine, C.; Fates, N. (2001). Public automated web-based evaluation service for watermarking schemes: StirMark Benchmark. In: Security and Watermarking of Multimedia Contents III, Ping Wah Wong, Edward J. Delp III, Editors, Proceedings of SPIE Vol. 4314, Bellingham WA, USA, pp. 575 - 584, ISBN 0-8194-3992-4, 2001.
6. Dittmann, Jana; Mukherjee, Anirban; Steinebach, Martin (2000). Media-independent Watermarking Classification and the need for combining digital video and audio watermarking for media authentication. In: Proceedings of the International Conference on Information Technology: Coding and Computing, 27 - 29 March, Las Vegas, Nevada, USA, pp. 62 - 67, IEEE Computer Society, ISBN 0 - 7695 - 0540 - 6, 2000.
7. Laurence Boney, Ahmed H . Tewfik , and Khaled N. Hamdy , "Digital Watermarks for Audio Signals," EUSIPCO-96, VIII European Signal Proc. Conf., Trieste, Italy, September, 1996. Notes in Computer Science, Portland, Oregon, USA, 14{17 April, 1998, p. 218-238}.
8. D. Kirovski, H. Malvar, Spread-Spectrum Audio Watermarking: Requirements, Applications, and Limitations, 2001 IEEE 4th Workshop on Multimedia Signal Processing, October 3 - 5, 2001, Cannes, France, Electronic Proceedings
9. G. Caccia, R. Lancini, M. Ludovico, F. Mapelli, S. Tubaro, Watermarking for Musical Pieces Indexing used in Automatic Cue Sheet Generation Systems, 2001 IEEE 4th Workshop on Multimedia Signal Processing, October 3 - 5, 2001, Cannes, France, Electronic Proceedings
10. J.Chou, K, Ramchandran, A. Ortega; High capacity audio data hiding for noisy channels, Int. Conference on Information Technology: Coding and Computing (ITCC 2001), April 2 - 4, Las Vegas, Nevada, ISBN 0-7695-1062-0, 2001.
11. I.K. Yeo, H.Kim, Modified Patchwork Algorithm: A novel audio Watermarking scheme, Int. Conference on Information Technology: Coding and Computing (ITCC 2001), April 2 - 4, Las Vegas, Nevada, ISBN 0-7695-1062-0, 2001.