Detection and Classification of Double Compressed MP3 Audio Tracks

Tiziano Bianchi
Dept. of Electronics and Telecommunications
Politecnico di Torino (Italy)
tiziano.bianchi@polito.it

Alessia De Rosa
National Inter-University Consortium for Telecommunications
University of Florence (Italy)
alessia.derosa@unifi.it

Marco Fontani
Dept. of Information Engineering and Mathematical Sciences
University of Siena (Italy)
marco.fontani@unisi.it

Giovanni Rocciolo
Dept of Information Engineering
University of Florence (Italy)
g.rocciolo@gmail.com

Alessandro Piva
Dept of Information Engineering
University of Florence (Italy)
alessandro.piva@unifi.it

ABSTRACT

In this paper, a method to detect the presence of double compression in a MP3 audio file is proposed. By exploiting the effect of double compression in the statistical properties of quantized MDCT coefficients, a single measure is derived to decide if a MP3 file is single compressed or it has been double compressed and also to devise the bit-rate of the first compression. Experimental results confirm the performance of the detector, mainly when the bit-rate of the second compression is higher than the bit-rate of the first one.

Categories and Subject Descriptors

D.4.6 [**Software**]: Security and Protection—*Authentication*

Keywords

Audio forensics, double compression, audio tampering

1. INTRODUCTION

The presence of artifacts due to a double compression in the statistics of transformed coefficients has received a lot of attention in the image forensics field: Popescu et al. in [9] observed that consecutive quantizations introduce periodic artifacts into the histogram of DCT coefficients; these periodic artifacts are visible in the Fourier domain as strong peaks in medium and high frequencies. Their seminal work has been the basis of several works dealing with double JPEG compression, as an example we cite [4, 2, 1].

On the contrary, there are few works in the current literature dealing with MP3 audio files and with double audio compression. In [14] to defeat Fake-Quality MP3 files (i.e. MP3 files recompressed at a higher bit-rate) authors observed that there are many more quantized MDCT coefficients with small values in a single compressed MP3 file than that in the fake-quality MP3, no matter which bit rate the fake quality MP3 is transcoded from. In particular, the detector just measures the number of MDCT coefficients assuming ± 1 values and comparing this value with a given threshold: if it is lower than the threshold, it is decided that the file is a fake-quality one, otherwise that it is a single compressed one. The same authors in [15] proposed to detect double MP3 compression through the use of support vector machine classifiers with feature vectors formed by the distributions of the first digits of the quantized MDCT coefficients; in particular, a global method is proposed, where the statistics on the first digits of all quantized MDCT coefficients is taken, and the computed probability distributions of nine digits are used as features (9 dimensions) for training of a SVM. A so called band distribution method is also proposed, where a procedure of band division is added before computing the statistics on the first digits; this modification allows to increase the performance. In [5, 10] to detect double MP3 compression, some statistical features on the MDCT are extracted and a support vector machine is applied to the extracted features for classification. In particular, a set of statistical features of zero MDCT coefficients and non-zero MDCT coefficients from the whole frequency as well as individual scale bands are adopted. In [12, 13] a forgery detection method for MP3 audio files is proposed. Based on the observation that forgeries break the original frame segmentation, frame offsets are used to locate forgeries automatically, allowing to detect most common forgeries, such as deletion, insertion, substitution, and splicing. However, experimental results are carried out in audio files that after the manipulation have not been reencoded in MP3 format. Finally, in [8] the inverse decoder problem is considered. In this scenario, only the uncompressed samples are known to the analyst and the goal is to recover the parameters of a possible previous compression.

In this work, we propose an approach exploiting the effect of double compression in the statistical properties of quantized MDCT coefficients in a MP3 compressed audio file. The method relies on a single measure derived from the statistics of MDCT coefficients, allowing us to apply a simple threshold detector to decide if a given MP3 file is single compressed or it has been double compressed. Moreover, the proposed method is able to derive the bit-rate of the first compression by means of a Nearest Neighbour classifier, as it will be described in the following.

2. THE PROPOSED METHOD

The core idea of the algorithm is to measure the similarity between the histogram of quantized MDCT coefficients of the MP3 file under analysis, that has possibly undergone a double compression, and the histogram of the coefficients computed on a single compressed version of the same file, that is of the single compressed MDCT coefficients. Intuitively, if the distance between the two distributions is low, this will indicate that the file under analysis has not been MP3 encoded twice, viceversa the file will be considered as double compressed.

Obtaining a reliable estimate of the distribution of the single quantized MDCT coefficients from the corresponding quantized or double quantized coefficients appears to be a difficult task. However, it has already been observed in the image forensic literature [6] that the DCT coefficients obtained by applying a slight shift to the grid used for computing the block DCT usually do not exhibit quantization artifacts. Hence, in a similar way the distribution of the single compressed MDCT coefficients can be approximated by considering the simulated single compressed file, achieved by removing a given number of PCM samples of the decompressed audio file and recompressing the remaining samples to the same compression quality of the file under analysis.

To demonstrate that the idea is effective, an uncompressed audio track, 4 sec long, has been MP3 compressed at several bit-rates in $[64, 96, 128, 192]$ kbit/s, and then recompressed to 160 kbit/s, in such a way to obtain 4 double compressed versions, 3 with increasing bit-rate, and one with decreasing bit-rate. Then, the uncompressed file is also single compressed to 160 kbit/s. To each of these compressed files, the previous procedure has been applied to obtain a simulated single compressed file, by removing the first 10 PCM samples to the decompressed file and recompressing the remaining samples to 160 kbit/s. Then, the histograms of MDCT coefficients for the input original files and the corresponding simulated single compressed files have been compared. In Figure 1, the histograms related to the MP3 file double compressed at 64 kbit/s and then 160 kbit/s, and to the simulated MP3 file single compressed at 160 kbit/s are shown. It is evident that the first histogram exhibits the characteristic pattern of a distribution of coefficients that have undergone a double compression, whereas in the second these artifacts have been removed. Similar results are obtained when the first compression was done with higher bit-rates, but still lower than 160 kbit/s, even if the effect of the double quantization becomes smaller. If the same procedure is applied to a single compressed MP3 file, it will happen the histograms of the input file and the corresponding simulated one are very similar, as it is shown in Figure 2. A similar situation shows up when a double compression has been applied, but with the first bit-rate (i.e. 192 kbit/s) higher than the second one (i.e. 160 kbit/s), see Figure 3.

The processing blocks composing the proposed algorithm are illustrated in Figure 4: the MP3 file is decompressed obtaining a sequence of PCM samples. The *Pattern breaker* just removes a given number of PCM samples starting from the beginning of the PCM sequence. This operation allows to maintain the original characteristics of the signal, while removing quantization artifacts. The *Filterbank + MDCT* block takes the PCM samples and applies filtering and the MDCT transform, achieving a set of unquantized MDCT coefficients. The *Parameter extraction* allows to extract from the original MP3 bitstream the quantization parameters, i.e. the quantization pattern and the original quantization values. The quantization pattern is needed by the *Re-quantizer* to simulate a distribution of MDCT coefficients that have undergone only a single compression. In addition, the *Re-quantizer* smooths the sequence of simulated coefficients through a *Laplace Smoothing* [7], a technique used to

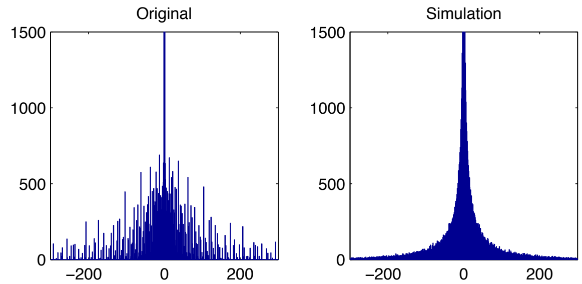


Figure 1: Histograms of MDCT coeffs of a MP3 file double compressed at 64 kbit/s and then 160 kbit/s (left), and of the corresponding simulated single compressed file at 160 kbit/s (right).

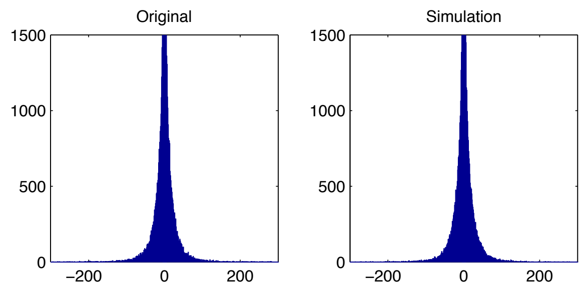


Figure 2: Histograms of MDCT coeffs of a MP3 file single compressed at 160 kb/s (left), and of the simulated single compressed file at 160 kbit/s (right).

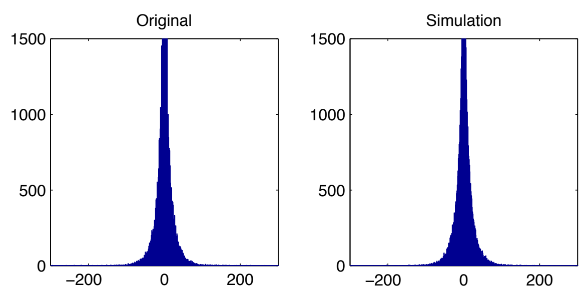


Figure 3: Histograms of MDCT coeffs of a MP3 file double compressed at 192 kb/s and then 160 kb/s (left), and of the simulated single compressed file at 160 kbit/s (right).

smooth categorical data (in particular, the smoothing parameter α equal to 1 was adopted). This operation aims at filling possible empty bins present in the data histogram, and thus avoiding numerical errors in the following computations. The original quantized values and the simulated single quantized values are then compared through the *Histograms distances* block.

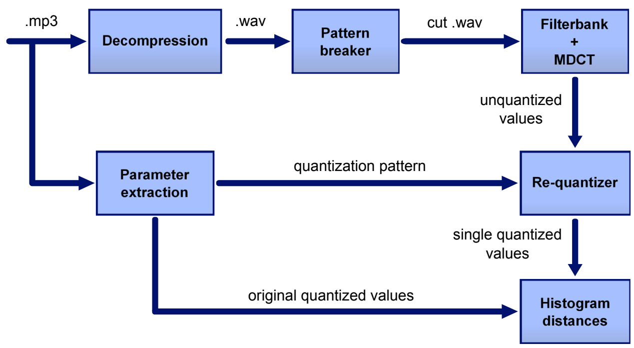


Figure 4: Scheme of the proposed method.

Indicating with X and Y respectively the observed and the simulated distributions of MDCT coefficients, their histograms are built, and a similarity measure is computed. Among the possible measures that can be used to compute the distance between two histograms, we adopted the the Chi-square distance $D_\chi(X,Y)$ [11], defined as:

$$D_\chi(X,Y) = \sum_{i=1}^{N} \frac{(X_i - Y_i)^2}{2(X_i + Y_i)} \tag{1}$$

where N is the number of bins of the histograms.

The computed distance measure represents the feature the proposed method uses to detect if a MP3 file has been single or double compressed (i.e. when the first bit-rate is lower than the second one).

Furthermore, such a proposed feature (as it will be shown in the next Section) is able to provide additional information about the compression suffered by the analyzed MP3 file, in particular concerning the difference between the second and the first bit-rate ($\Delta = BR2 - BR1$): when such a value is positive the proposed algorithm is able to classify double compressed MP3 audio tracks with respect to the first bit-rate. In fact, the values assumed by the feature range quite differently according to the second bit-rate and the Δ factor, thus allowing to cluster double compressed MP3 audio tracks by applying a Nearest Neighbour classifier.

3. EXPERIMENTAL RESULTS

To validate the ideas proposed in the previous section, an audio dataset has been built, trying to represent as much as possible heterogeneous sources. To this aim, the database includes uncompressed audio files belonging to four different categories: *Music*: royalty free music audio tracks, with 5 different musical styles [3], *Speech*: music audio files containing dialogues, *Outdoor*: audio files relative to recording outdoors, and *Commercial*: file containing dialogues combined with music, as often happens in advertising. Each category collects about 17 minutes of audio divided into 250 segments 4 s long, for a total of 1.000 uncompressed audio files. Each file has been compressed, in dual mono, with bit-rate *BR1*

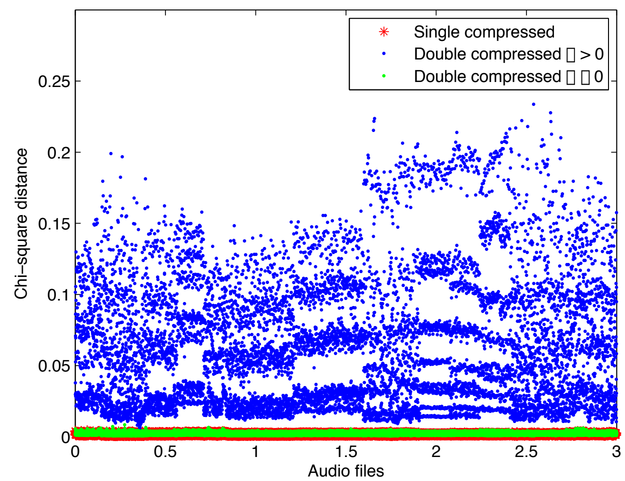


Figure 5: Chi-square distances computed for the 30.000 audio segments composing the dataset.

chosen in $[64, 96, 128, 160, 192]$ kbit/s, obtaining 5.000 single compressed MP3 files. Finally, these files have been compressed again using as *BR2* one of the bit-rate values (also the same value as the first one was considered) achieving 25.000 MP3 double compressed files. Among these, 10.000 files have a difference $\Delta = BR2 - BR1$ between the second and the first bit-rate which is positive, taking value in $[32, 64, 96, 128]$ kbit/s; 10.000 files have a negative difference Δ taking value in $[-128, -96, -64, -32]$ kbit/s and 5.000 files have $\Delta = 0$. The overall dataset is composed by 30.000 MP3 files, including 5.000 single compressed files.

3.1 Double Compression Detection

As a first experiment, we computed the values assumed by the proposed feature for all the 30.000 files belonging to the test dataset. In Figure 5 the Chi-square distances are visualized: single compressed files and double compressed files with negative or null Δ show a distance D near to zero, whereas the other files have D rather higher than zero. By comparing D with a threshold τ is possible to discriminate these two kinds of files: double compressed files with $\Delta > 0$ and the other ones.

By adopting a variable threshold τ, we then computed a Receiver Operating Characteristic (ROC) curve, representing the capacity of the detector to separate single compressed from double compressed MP3 files (including $\Delta >$ or ≤ 0). The trend of the obtained ROC curve is shown in Figure 6 (left): it reflects the bimodal distribution of distances of double compressed files (blue and green colored in Figure 5) and highlights that the detector is able to distinguish only one of the two components (the one with $\Delta > 0$). If we separate the previous ROC in one relating to files double compressed with positive Δ, and one to files double compressed with negative or zero Δ, as is shown in Figure 6 (right), when double compressed file with positive Δ are considered, we obtain a perfect classifier, while when the cases with negative or zero Δ are analyzed, we are next to the random classifier.

As anticipated in Section 2, the distance representing the proposed feature assumes a large range of values, as it can

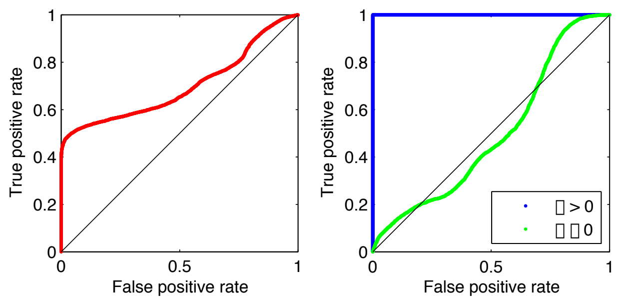


Figure 6: ROC curve obtained by varying the threshold τ of the classifier (left). Separation of cases with positive Δ and negative or zero Δ (right)

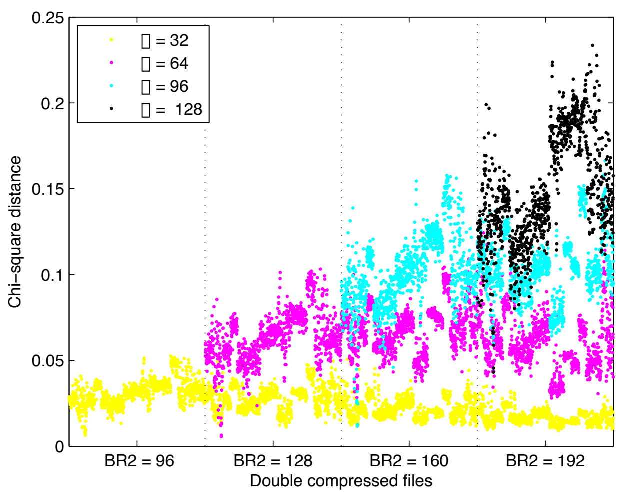


Figure 7: Chi-square distances computed for double compressed files with positive Δ, grouped according to both the values assumed by Δ and BR2.

be clearly observed in Figure 5. In order to highlight the relationship between the values taken by the feature and the compression parameters (i.e. BR2 and Δ), we examined in more detail the 10.000 double compressed files with positive Δ, plotting their Chi-square distances in Figure 7. Such values were plotted according the different Δ: in particular there are 4.000 files with $\Delta = 32$ (yellow), 3.000 files with $\Delta = 64$ (violet), 2.000 files with $\Delta = 96$ (sky-blue), and finally 1.000 files with $\Delta = 128$ (black), and grouped for different BR2: $[96, 128, 160, 192]$. Double compressed audio tracks with same Δ factor obtain similar values of Chi-square distance (see plotting with same color) and to increasing Δ values correspond increasing Chi-square distances between the observed and simulated distribution. On the other hand, given a value of Δ, for different bit-rate of the second compression, different values of distance are obtained, since at a lower bit-rate (e.g. values for BR2=96 vs. values for BR2=192) corresponds a greater number of traces left in the content and thus a larger distance from the simulated single compression.

The dependency of the results on the bit-rate of the second compression (that is a parameter observable from the bitstream), suggests the possibility of improving the perfor-

BR1 vs BR2	64	96	128	160	192
64	-	99.9	99.9	99.9	99.9
96	49.9	-	99.9	99.9	99.9
128	49.9	49.9	-	99.9	99.9
160	69.5	47.9	49.1	-	96.7
192	56.1	66.4	57.8	67.4	-

Table 1: Detection accuracy of the proposed method for different bit-rates.

BR1 vs BR2	64	96	128	160	192
64	-	70.9	96.7	98.6	99.4
96	53.4	-	93.2	99.3	99.4
128	64.9	63.7	-	93.0	95.5
160	64.1	67.6	65.6	-	82.5
192	59.0	64.9	70.0	75.0	-

Table 2: Detection accuracy of LSQ method for different bit-rates.

BR1 vs BR2	64	96	128	160	192
64	-	99.9	99.9	99.9	99.3
96	96.2	-	99.7	99.7	98.8
128	80.2	99.0	-	99.2	98.1
160	84.5	94.1	96.3	-	98.0
192	67.6	88.5	89.9	90.3	-

Table 3: Detection accuracy of YSH method for different bit-rates.

mance of the double compression detection by taking into account a specific threshold τ for each specific BR2. With this observation in mind, we performed a set of experiments in order to compare the detection accuracy of the proposed detector with respect to the detection accuracy of the method proposed in [5, 10] by Liu, Sung, Qiao (LSQ method) and in [15] by Yang, Shi, Huang (YSH method). Corresponding results are shown in Tables 1, 2 and 3 respectively. The proposed method achieves nearly optimal performances for all combinations such that $\Delta > 0$. Also the other methods generally achieve good performances for $\Delta > 0$, however there are some bit-rate combinations for which they are slightly inferior. Conversely, for $\Delta < 0$ the proposed method is not able to reliably detect double MP3 compression, whereas the other methods have better performances, especially the YSH method that has good performances also in this scenario.

All the results shown in Tables 1, 2 and 3, have been achieved considering audio tracks 4 s long. We evaluated the degradation of the performances of the three detectors when the duration of the audio segments is reduced from 4 s to $[2, 1, 1/2, 1/4, 1/8, 1/16]$ s. The reason behind this experiment is that analyzing very small portions of audio potentially opens the door to fine-resolution splicing localization (i.e., detect if part of an audio file has been tampered). Practically, instead of taking all the MDCT coefficients of the 4 seconds long segment, only the coefficients belonging to a subpart of the segment are retained, where the subpart is just one half, one fourth and so on. For BR2=192 kbit/s and BR2=128 kbit/s the detection accuracies (averaged with respect to BR1) have been plotted with varying audio file duration in Figure 8 for our method, the LSQ method and the YSH method. The proposed method achieves a nearly

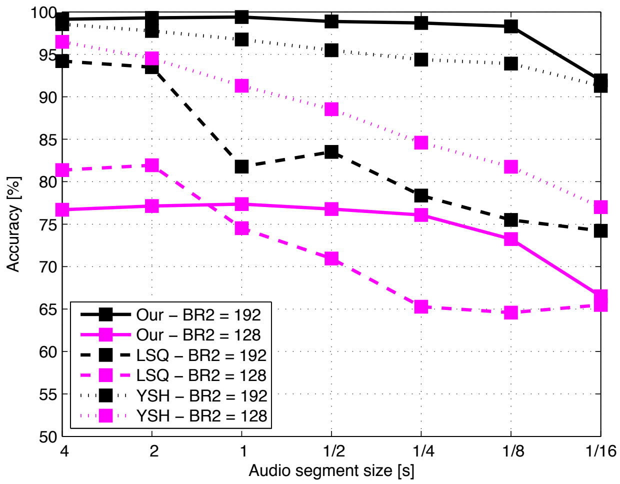


Figure 8: Detection accuracy with varying audio file duration for BR2=192 kbit/s and BR2=128 kbit/s.

constant detection performance up to 1/8 s audio segments, whereas the performance of the other methods drastically drop for audio segments under 2 s. Our method achieves very good performance in the case of high quality MP3 files. For BR2 equal to 192 kbit/s, our method achieves the best performance irrespective of the audio segment duration. For BR2 equal to 128 kbit/s, the proposed method is able to outperform the LSQ method for short audio segments, but remains inferior with respect to the YSH method.

3.2 First Compression Bit-Rate Classification

By taking into account the feature distribution highlighted in Figure 7, we considered the capability of the proposed feature to classify the double compressed file according to the first compression bit-rate, as BR1=BR2-Δ. A Nearest Neighbor classifier is adopted for each different BR2 and the corresponding classification accuracy results are shown in Table 4 for BR2=192 kbit/s and 6 for BR2=128 kbit/s. The rows of the tables represent the actual bit-rate of the first compression, and the columns the values assigned by the classifier. A comment about this experiment is in order: as shown in the previous section, the proposed method will hardly detect double compression for negative or null Δ. Similarly, the output of the classifier on double encoded files with negative or null Δ is not reliable. In particular, since a single compressed file can be considered like having undergone a (virtual) first compression at infinite quality, the classifier cannot distinguish between single encoded files and double encoded files with negative Δ.

For comparison, only the LSQ method is considered, since the YSH method was not proposed for compression classification. The corresponding results obtained on 4 s audio segments for BR2= 192, 128 kbit/s are shown in Tables 5 and 7 respectively.

In this scenario, both methods achieves similar classification performance. It is worth noting that the proposed method achieves better classification rates for higher bit-rates, whereas the LSQ method appears slightly more accurate in the case of low bit-rates.

As in the case of detection, we experimented how the classification performances vary with respect to audio file duration. The average classification accuracy for different BR2 (i.e. [192, 160, 128, 96, 64] kbit/s) and decreased audio file duration (i.e. [4, 2, 1, 1/2, 1/4, 1/8, 1/16] s) are shown in Figure 9 for both our method (a) and LSQ method (b). In the case of higher bit-rates, it is evident that the proposed method achieves better classification performance than the LSQ method. Noticeably, the performances of the proposed method suffer only a slight degradation up to 1/8 s audio segments, whereas the performances of the LSQ method usually drop for audio segments shorter than 1-2 seconds.

Actual vs Pred.	192	160	128	96	64
192	99.9	0.1	0.0	0.0	0.0
160	0.0	98.9	1.1	0.0	0.0
128	0.0	0.4	92.8	6.6	0.2
96	0.0	0.0	13.0	69.5	17.5
64	0.0	0.0	1.2	15.1	83.7

Table 4: Classification accuracy of the proposed method for BR2 = 192.

Actual vs Pred.	192	160	128	96	64
192	93.5	6.3	0.2	0.0	0.0
160	6.3	90.9	2.7	0.0	0.0
128	0.4	10.6	83.4	4.5	1.1
96	0.0	0.0	1.0	98.2	0.7
64	0.0	0.2	0.2	0.4	99.3

Table 5: Classification accuracy of LSQ method for BR2 = 192.

Actual vs Pred.	192	160	128	96	64
192	51.4	18.8	29.7	0.0	0.0
160	1.8	87.0	11.2	0.0	0.0
128	42.0	23.3	34.5	0.2	0.0
96	0.0	0.0	0.0	97.7	2.3
64	0.0	0.0	0.0	5.2	94.8

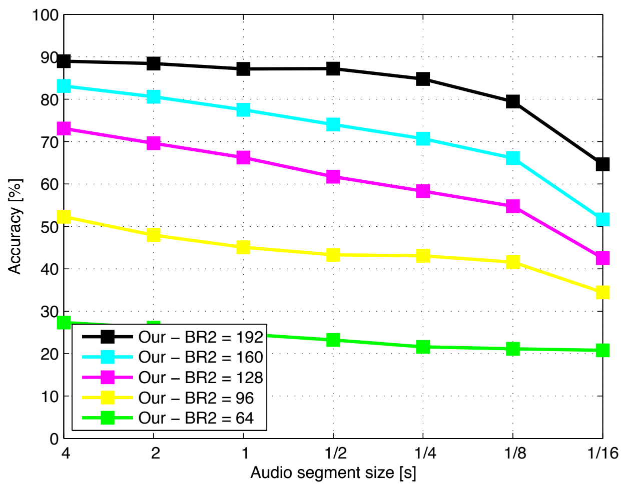
Table 6: Classification accuracy of the proposed method for BR2 = 128.

Actual vs Pred.	192	160	128	96	64
192	69.9	23.9	6.2	0.0	0.1
160	47.4	48.4	4.2	0.0	0.0
128	24.7	15.5	59.6	0.0	0.1
96	0.7	0.2	0.1	93.0	6.0
64	0.1	0.0	0.1	0.4	99.4

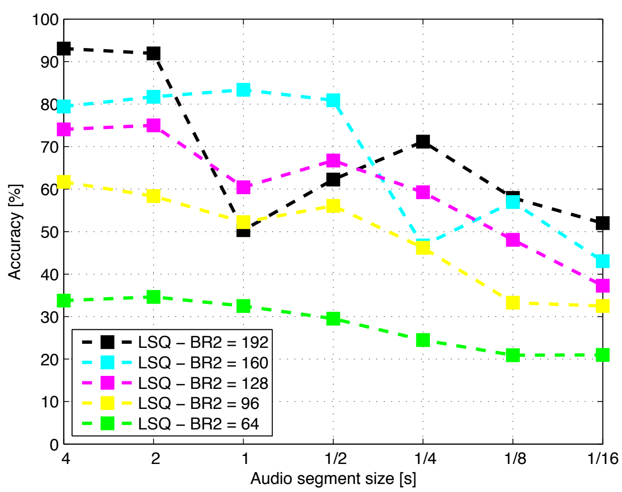
Table 7: Classification accuracy of LSQ method for BR2 = 128.

4. CONCLUSIONS

A method to detect the presence of double compression in a MP3 audio file has been presented. A statistical measure has been derived to measure the effect of double compression, that allows to decide if a MP3 file is single compressed or it has been double compressed and also to derive the bit-rate of the first compression. The algorithm is effective when the bit-rate of the second compression is higher than the bit-rate of the first one, and exhibits a good performance even when analyzing short temporal windows, opening the possi-



(a)



(b)

Figure 9: Accuracy of the classifiers with varying audio file duration: (a) proposed feature; (b) LSQ.

bility of using this feature for the localization of tampering in a MP3 file, whose study is left as future work.

5. ACKNOWLEDGMENTS

This work was partially supported by the REWIND Project funded by the Future and Emerging Technologies (FET) programme within the 7FP of the European Commission, under FET-Open grant number 268478.

6. REFERENCES

[1] T. Bianchi and A. Piva. Image forgery localization via block-grained analysis of JPEG artifacts. *Information Forensics and Security, IEEE Transactions on*, 7(3):1003 –1017, june 2012.

[2] T. Bianchi, A. D. Rosa, and A. Piva. Improved DCT coefficient analysis for forgery localization in JPEG images. In *Proc. of ICASSP 2011*, pages 2444–2447, May 2011.

[3] K. M. (incompetech.com). Artifact, danse macabre, just nasty, airport lounge, chee zee beach.

[4] Z. C. Lin, J. F. He, X. Tang, and C. K. Tang. Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis. *Pattern Recognition*, 42(11):2492–2501, Nov. 2009.

[5] Q. Liu, A. Sung, and M. Qiao. Detection of double MP3 compression. *Cognitive Computation*, 2:291–296, 2010.

[6] J. Lukáš and J. Fridrich. Estimation of primary quantization matrix in double compressed JPEG images. In *Digital Forensic Research Workshop*, 2003.

[7] C. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[8] S. Moehrs, J. Herre, and R. Geiger. Analysing decompressed audio with the inverse decoder - towards an operative algorithm. In *Audio Engineering Society Convention 112*, 4 2002.

[9] A. C. Popescu and H. Farid. Statistical tools for digital forensics. In *In 6th International Workshop on Information Hiding*, pages 128–147. Springer-Verlag, Berlin-Heidelberg, 2004.

[10] M. Qiao, A. H. Sung, and Q. Liu. Revealing real quality of double compressed MP3 audio. In *Proceedings of the international conference on Multimedia*, MM '10, pages 1011–1014, New York, NY, USA, 2010. ACM.

[11] G. Snedecor and W. Cochran. *Statistical Methods*. Number v. 276 in Statistical Methods. John Wiley & Sons, 1991.

[12] R. Yang, Z. Qu, and J. Huang. Detecting digital audio forgeries by checking frame offsets. In *ACM Multimedia and Security Workshop*, pages 21–26, 2008.

[13] R. Yang, Z. Qu, and J. Huang. Exposing MP3 audio forgeries using frame offsets. *ACM Trans. Multimedia Comput. Commun. Appl.*, 8(2S):35:1–35:20, Sept. 2012.

[14] R. Yang, Y. Q. Shi, and J. Huang. Defeating fake-quality MP3. In *ACM Multimedia and Security Workshop*, pages 117–124, 2009.

[15] R. Yang, Y. Q. Shi, and J. Huang. Detecting double compression of audio signal. In *SPIE Conference on Media Forensics and Security*, 2010.