Audio Steganalysis Based on Lossless Data-Compression Techniques

Fatiha Djebbar and Beghdad Ayad

[1] UAE university, UAE
fdjebbar@uaeu.ac.ae
[2] Canadian University in Dubai, UAE
beghdadayad@gmail.com

Abstract. In this paper, we introduce a new blind steganalysis method that can reliably detect modifications in audio signals due to steganography. Lossless data-compression ratios are computed from the testing signals and their reference versions and used as features for the classifier design. Additionally, we propose to extract additional features from different energy parts of each tested audio signal to retrieve more informative data and enhance the classifier capability. Support Vector Machine (SVM) is employed to discriminate between the cover- and the stego-audio signals. Experimental results show that our method performs very well and achieves very good detection rates of stego-audio signals produced by S-tools4, Steghide and Hide4PGP.

Keywords: audio steganalysis, active speech level, lossless data-compression.

1 Introduction

In contrast to steganography, which is the science of hiding a message in an innocuous multimedia cover file, steganalysis is the science of detecting the presence of hidden messages. Audio steganalysis techniques have been actively investigated in the last decade. This interest is attributed to the growing number of steganography algorithms and the threats they represent. In practice, the work of a steganalyst is based on finding any unnatural modification resulting from the embedding process that might exist in the suspected audio file. To date, there is no known steganographic system that hides data in a perfect secure and undetectable way. All embedding algorithms leave a fingerprint in the stego-audio unless a very small embedding capacity is achieved. Although some research works had managed to reliably detect the presence of hidden data in audio-signals, so far all of them rely on the change of the intrinsic properties (features) of the audio signals to distinguish between stego- and cover-audio signals. Once the features are extracted, most of the steganlysis methods apply a learning process to differentiate between the cover- and the stego-audio signals. The learning process is done by training a machine learning such as a support vector machine (SVM) [1] on a dataset fed with statistical properties (features) extracted from the cover and stego-audio signals. The right choice of these features reinforce the discriminatory power between cover- and stego-audio signals.

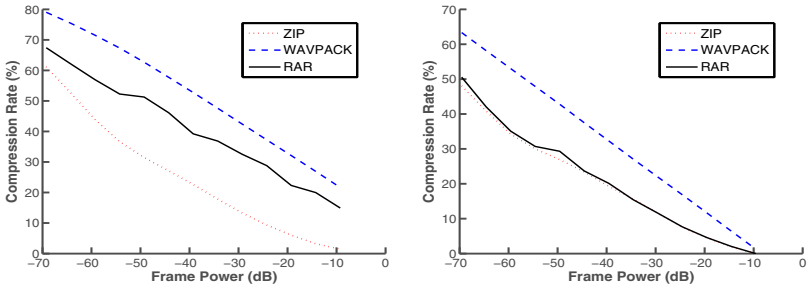
T.W. Chim and T.H. Yuen (Eds.): ICICS 2012, LNCS 7618, pp. 1–9, 2012.

Authors in [2] presented a universal steganalysis algorithm for high quality recorded speech. In this work, a statistical model is constructed to capture the irregularities between the cover and the stego signals' spectrograms. The use of audio quality measures for audio steganalysis was proposed by [3]. The authors selected a set of nineteen perceptual and non-perceptual domain audio quality measures (i.e, Signal-to-noise ration, Log likelihood) to distinguish between the stego-signal and its de-noised version (used as an estimate to the cover-signal). ANOVA test [4] and SFS (Sequential Floating Search) [5] were used to select the most appropriate measures to better detect the presence of hidden messages. To improve the latter technique, [6] proposed a content independent distortion measures as features for the classifier design. Instead of creating a reference signal via a de-noised version of the stego-signal, they proposed to use a single reference signal that is common to all signals to be tested. The author has also minimized the set of quality measure used in [3] to only five. In [7], the features are extracted from the histograms of both statistical moments and frequency domain of the tested audio signal. The same methodology was applied by [8]. However, only higher order statistical moments of histogram and frequency histogram for both signal and its wavelet sub-bands are extracted and used as features to train an SVM classifier. The features (mean, variance, skewness, and kurtosis) wavelet coefficients are supposed to provide information about the frequency distribution of the audio signal and information about the difference between the wavelet coefficients and their linear predictive values. The same principle in selecting the features was followed by [9]. However, the signal reference used is a self-generated signal via linear predictive coding. In [10], the authors used the mean and the standard deviation as features captured from high frequencies of first, second, third and fourth order derivatives of the audio signal spectrum. A reference signal is generated by randomly modifying the least significant bits of the stego signal. The latter method was further improved in [11] by extending the features developed in [10] to include mel-cepstrum coefficients (widely used in speech recognition) [12] extracted from the second derivative and also from wavelet spectrums of the audio signal. This method is also an improvement of the work presented in [13], where mel-cepstrum coefficient were exploited for the first time in audio steganalysis and used as features to train the classifier. More recently, the same authors [14] proposed to use stream data mining for high complexity audio signals steganalysis. Their approach is based on extracting the second order derivative based Markov transition probabilities and high frequency spectrum statistics as features of the audio streams. A steganalysis method based on features extracted from the co-occurrence matrix of audio signals is presented in [15]. The statistic features are calculated from the amplitude components of audio signals. Preprocessing of principal component analysis (PCA) is performed on statistic features trained with SVM classifier. In the same perspective, [16] proposed to use Hausdorff distance [17]. Wavelet de-noising is applied on the stego-signal to create a cover-signal estimate. The Hausdorff distance measure is computed at various wavelet decomposition levels from which the statistical moments are generated. In [18], the steganalysis method is based on negative resonance phenomenon in audio signal created due to data embedding. The proposed method uses features such as mean, variance, skewness and kurtosis derived from the stego and its linear predicted value. In general, the features used in previous work are extracted from the entire tested audio-signals, a

process that could dilute the hiding error effect on the stego-audio signal. To select our features, we exploit the disparities in lossless data-compression ratios between audio files with different information quantities (i.e., cover- and the stego-signals). We compute the compression rates of distinct parts of each audio signal to collect additional features to collect more informative data allowing to enhance the classifier capability. Thus, each tested signal is split into four energy level parts: noise, low, medium and high using active speech level (ASL) which is defined in ITU-T Recommendation P.56 [19]. The rational to not utilize only the entire signal for the compression process is that different energy level parts (power classes) in the audio signal could be impacted differently during the hiding process. Extra features collected from this energy parts provide more informative data allowing to enhance the classifier capability. Since the original signal will not be available during the testing stage, we need to create a reference signal for the received audio-files in order to compute the features vector. By randomly modifying the 1st LSB layer of the tested signal, a signal version is created and is used as the reference signal. We show the efficacy of our proposed algorithm on a large database of audio signals and on different steganographic algorithms such as Steghide, Stool and Hide4PGP [20,21,22]. This paper is organized as follows: the impact of lossless data-compression and ASL on cover and stego signals are presented in Section 2 and Section 3. Section 4 discusses the preprocessing steps to generate our features. In Section 5.2, classification results by SVM and evaluation study are revealed. Finally, we conclude this chapter with a summary of our work in Section 6.

2 Lossless Data-Compression and Signal Energy

Lossless data compression involves the compression of any type of files in a way that they can be latter recovered bit-wise identical to the original. It is based on removing redundant or "unnecessary" bits of data to reduce the file to its smallest version. When data is hidden in a cover file, the quantity of information of the file changes and so does the compression ratios. However, since the audio signal content and its energy vary as time progresses, performing lossless data-compression on distinct energy parts of the signal allows to capture all subtle changes in the audio-signal. For the compression operation, we selected three utilities (zip, rar and wavpack). These tools give distinct



(a) noise part **(b)** audio signal

Fig. 1. Compression rates (1- compression ratio) for noisy part (a) and entire signal (b)

compression ratios when applied on the same audio file, which result in augmenting the detection rates of stego-audio signals as more informative data are collected. The Figures (1a) and (1b) illustrate the varying effect of lossless data-compression performed with zip, rar and wavpack utilities, on noise-only as well as an entire speech signals. The figures also describe the relation between the compression rate and the energy associated to the audio signal. More precisely, the higher are the energies the lower are compression rates of the audio signals (noise or signal). This result shows that using lossless data compression, stego and cover audio signals are more distinguishable at their lower energy parts.

3 Active Speech Level

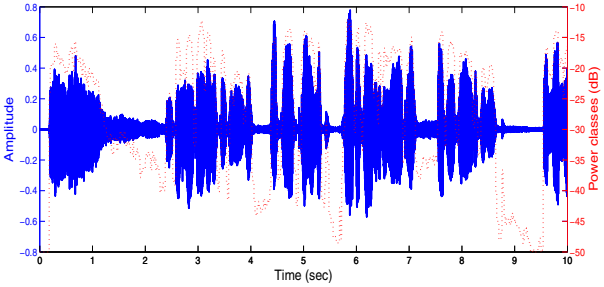
ASL determines a speech activity factor (Sp_l) representing the fraction of time where the signal is considered to be active speech and the corresponding active level for the speech part of the signal [19]. The speech activity algorithm computes the speech energy value at each sample time (frame) and is computed using the library tool voicebox available at `http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html` . To determine which frames belong to high, medium, low and noisy (pause) power classes of active speech, Sp_l (dB) is compared with a discrete set of thresholds. The thresholds set are chosen based on experimental considerations and they are specific to normalized audio files of our datasets which statistics are shown in Table 1. An example of speech-signal division process to parts with different power classes using ASL and the thresholds set is illustrated in Figure (2a). Final division result is shown in Figure (2b).

The impact of lossless data-compression on different power classes of audio signals is illustrated in Figures (3a) and (3b). These figures show that stego-audio signals are less compressible than the original audio signals (Figure 3b). In addition, the figures also indicate that the compression rates are more less important in high energy audio

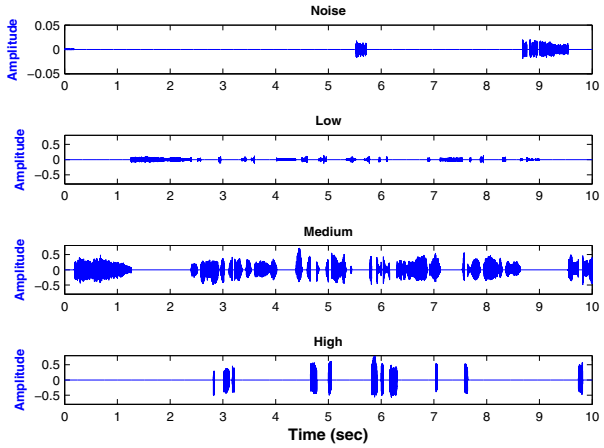
Table 1. Statistics about the composition of the datasets in terms of different audio signal parts and thresholds set used to categorize the frames as noisy, low, medium or high

Power classes	Audio (%) 2700 sec	Speech (%) 1550 sec	Music (%) 1150 sec
Noise	14.76	24.03	2.26
Low	15.54	22	6.84
Medium	50.03	39.19	64.64
High	19.67	14.78	26.26

Power classes	Threshold (dB)
noise	-45
Low	-35
Medium	-25
High	-15

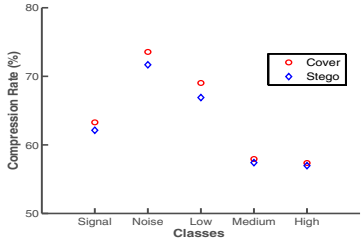


(a) Speech signal

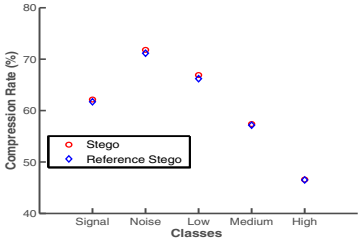


(b) Power audio parts classes

Fig. 2. Speech audio-signal division based on thresholds set. The blue curve in (2a) is the temporal representation of a speech signal (left y-axis) while the red curve represents the energy in (dB) per speech signal frame (right y-axis). The energy is computed by ASL and classified to four power classes (2b) using the thresholds set presented in Table 1.



(a) Cover and its reference audio signal



(b) Stego and its reference audio signal

Fig. 3. Compression rates for cover- (a) and stego-signal (b) with their respective reference versions

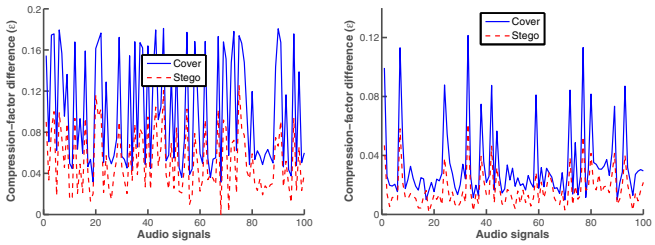
signal parts (medium and high). This confirms our statement made in the previous section where cover and stego signals are better discriminated in the lower energetic parts of the audio signals.

4 Features Extraction

The features extraction step starts by creating the features vector representing the compression factors difference (ϵ) between received (tested) audio-signals and their self-generated reference versions. Features extraction are done as follows:

1. The speech signal is split into M frames of 10 ms and N samples each, $s_t(m,n)$, $1 \leq m \leq M$ and $1 \leq n \leq N$.
2. For each frame, compute Sp_l using voicebox tool.
3. Classify the frame as high, medium, low or noisy by comparing its Sp_l to the values shown in Table 1.
4. Reassemble the frames of the same category into one part as shown in Figure 2b. At the end of the process, each audio file will be divided into four audio signal parts: noisy, low, medium and high energy.
5. Compute the compression ratio η_i for each part of the audio-signal as well as the entire signal using lossless data compression utilities (zip, rar and wavpack). For each audio signal, 15 compression ratios are computed (η_i, i=1...15) .
6. Calculate the compression factors difference (ϵ_i, i=1...15) between similar categories of tested s_t and its reference audio-signal s_r sush as: $\epsilon_i = \sqrt{|\eta_{ti} - \eta_{ri}|}$. The square root, a non linear amplification, is used to augment feeble η_i values and therefore to signify their impact in the classification process. The Figures (4a) and (4b) show the relative difference between ϵ values extracted from noisy parts as well as entire signals.

The features vector of each audio signal contains 15 coefficients: $Features = \epsilon_1, \epsilon_2, ..., \epsilon_{15}$



(a) noise audio signal part **(b)** entire audio signal

Fig. 4. Variation of the compression factors difference (ϵ) in noisy parts (4a) and in entire signals (4b) of 100 audio-files. The compression is performed by zip utility and the stego files are generated with Hide4PGP.

5 Evaluation Measures

5.1 Datasets

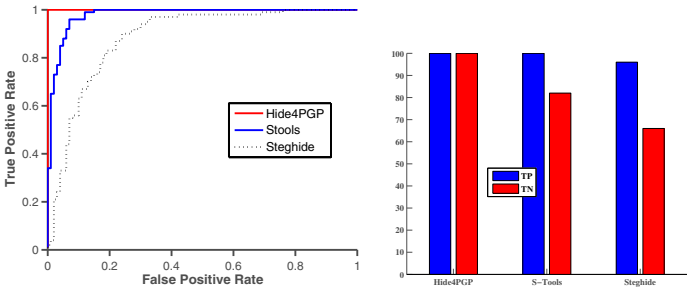
For each tested steganography algorithm, two datasets are produced: training and testing (Tr and Ts). Each dataset contains 270 stego and cover WAV audio signals of 10 s length. All signals are sampled at 44.1 kHz and quantized at 16-bits. Each training and testing dataset contains 135 positive (stego) and 135 negative (cover) audio samples. We used on-line audio files from different types such as speech signals in different languages (i.e, English, Chinese, Japanese, French, and Arabic), and music (classic, jazz, rock, blues). All stego-audio signals are generated by hiding data from different types: text, image, audio signals, video and executable files. The datasets Tr and Ts consist of a matrix of $\{\epsilon_i, l_i\}$,where ϵ_i refers to 15 compression-factors difference, and $l_i \in \{\pm 1\}$. The values +1 and -1 correspond to "Stego-audio" and "non Stego-audio" classes respectively. The performance of the proposed steganalysis algorithm is measured by the ability of the system to recognize and distinguish between stego and cover-audio signals. Next, we present a performance analysis of our steganalysis algorithm.

5.2 Results

In this section, we investigate the detection rate of our steganographic algorithm based on classification results of the SVM classifier used in conjunction with the Radial Basis Function (RBF) kernel [23]. In this study, we used SVMs library tool available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. The detection rates of our algorithm are reported in Tab.2, more details are reported in the Figures 5b and 5a. The performance of the proposed steganalysis method is measured by how well the system can recognize and distinguish between the stego and the cover-audio signals. In order to analyze the evaluation measures, we firstly define the following:

- *TP*: stego-audio signal classified as stego-audio signal
- *TN*: cover-audio signal classified as cover-audio signal
- *FN*: stego-audio signal classified as cover-audio signal
- *FP*: cover-audio signal classified as stego-audio signal

In the subsequent formula, *all* represents the number of all positive and negative audio signals. The value of the above information is used to calculate the $Accuracy(AC) = \frac{TP+TN}{all}$. Following the preparation of the training and testing features vectors for for each studied steganographic tool, we use SVM classifier in conjunction with the RBF kernel for the classification process. The results of the performance evaluation study are reported in Table 2 where the accuracy of each tool is measured by the AC value. The true positive rate and false negative rates are reported in Figure 5a while in Figure 5b true-positive versus true-negative rate of the proposed steganalysis algorithm are presented. Higher AC values and ROC correspond to more accurate steganalysis detection performance. The results show that the features extracted by our method are very informative for the classification process. In addition, we only used 15 features for the classification which results shorten the computation time needed for the classification process. Most importantly, the proposed method offers very high accuracy in regards to stego-audio files detection. Stego files generated by Hide4PGP and Stools are 100% detected (Figure (5b)) versus 96% in Steghide.



(a) ROC curves

(b) True positive versus true negative rates

Fig. 5. Lossless compression-based audio-steganalysis results on tested steganographic methods [20,21,22]

Table 2. Overall lossless data compression-based audio steganalysis

Hiding methods	AC
Hide4PGP	1
S-Tools	0.91
Steghide	0.81

6 Conclusion

In this paper, we proposed a simple to implement yet effective new blind audio steganalysis method. This method is based on lossless data compression techniques. To improve the detection rates of our method, more informative features are extracted from distinct energy parts of the audio signals. The proposed method have shown better accuracy rates when compared with existing landmark methods. Finally, the success of the proposed steganalysis method in detecting steganographic audio signals encouraged us to plan future investigations such as minimizing the features vector and further extending our proposed method to other steganographic applications which involve hiding small amount of data in the audio signals.

References

1. Cristianini, N., Shawe-Taylor, J.: An introduction to Support Vector Machines Cambridge University Press (2000)
2. Johnson, K.M., Lyu, S., Farid, H.: Steganalysis of Recorded Speech. In: Proceedings of the Conference on Security, Steganography and Watermarking of Multimedia (SPIE), San Jose, USA, pp. 664–672 (January 2005)
3. Ozer, H., Avcibag, I., Sankur, B., et al.: Steganalysis of Audio Based on Audio Quality Metrics. In: Proceedings of SPIE, Santa Clara, CA, USA, pp. 55–66 (June 2003)
4. Rencher, A.C.: Methods of Multivariate Data Analysis, 2nd edn. John Wiley (2002)
5. Pudil, P., Novovicova, J., Kittler, J.: Floating Search Methods in Feature Selection. Pattern Recognition Letters, 1119–1125 (November 1994)
6. Avcibas: Audio steganalysis with content independent distortion measures. IEEE Signal Process Letter 13(2), 92–95 (2006)

7. Qi, Y., Fu, J., Yuan, J.: Wavelet domain audio steganalysis based on statistical moments of histogram. Journal of System Simulation 20(7), 1912–1914 (2008)
8. Xuan, G., Shi, Y.Q., Gao, J., et al.: Steganalysis based on multiple features formed by statistical moments of wavelet characteristic functions. In: Proceeding of Information Hiding Workshop, pp. 262–277 (2005)
9. Ru, X., Zhang, H., Huang, X.: Steganalysis of Audio: Attaching the Steghide. In: Proceeding of the Fourth International Conference on Machine Learning and Cybernetics, pp. 3937–3942 (2005)
10. Liu, Q., Sung, A., Qiao, M.: Detecting information hiding in WAV audios. In: Proceeding of 19th International Conference on Pattern Recognition, pp. 1–4 (2008)
11. Liu, Q., Sung, A.H., Qiao, M.: Temporal derivative-based spectrum and mel-cepstrum audio steganalysis. IEEE Transactions on Information Forensics and Security 4(3), 359–368 (2009)
12. Kraetzer, C., Dittmann, J.: Pros and Cons of Mel-cepstrum Based Audio Steganalysis Using SVM Classification. In: Furon, T., Cayre, F., Doërr, G., Bas, P. (eds.) IH 2007. LNCS, vol. 4567, pp. 359–377. Springer, Heidelberg (2008)
13. Kraetzer, C., Dittmann, J.: Mel-cepstrum based steganalysis for voip-steganography. In: Proceedings of SPIE, Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, pp. 650505.1–650505.12 (2006)
14. Liu, Q., Sung, A.H., Qiao, M.: Novel Stream Mining for Audio Steganalysis. In: Proceedings of the 17th ACM International Conference on Multimedia, Beijing, China, pp. 95–104 (October 2009)
15. Qi, Y., Wang, Y., Yuan, J.: Audio Steganalysis Based on Co-occurrence Matrix and PCA. In: International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), vol. 1, pp. 433–436 (2009)
16. Liu, Y., Chiang, K., Corbett, C., Archibald, R., Mukherjee, B., Ghosal, D.: A Novel Audio Steganalysis based on Higher-Order Statistics of a Distortion Measure with Hausdorff Distance. LNCS, pp. 487–501 (September 2008)
17. Huttenlocher, D.P., Klanderman, G.A., Rucklidge, W.J.: Comparing Images using the Hausdorff Distance. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(9), 850–863 (1993)
18. Ru, X., Zhang, Y., Wu, F.: Audio steganalysis based on negative resonance phenomenon caused by steganographic tools. Journal of Zhejiang 7(4), 577–583 (2006)
19. ITU-T Recommendation P56, Telephone Transmission Quality: Objective Measuring Apparatus (March 1996)
20. Steghide, `http://steghide.sourceforge.net/`
21. Stools Version 4.0, `http://info.umuc.edu/its/online_lab/ifsm459/s-tools4/`
22. Hide4PGP, `http://www.heinz-repp.onlinehome.de/Hide4PGP.html`
23. Vapnik, V.: Statistical Learning Theory. Wiley, Hoboken (1998)