DATA HIDING VIA PHASE MANIPULATION OF AUDIO SIGNALS

Xiaoxiao Dong, Mark F. Bocko, Zeljko Ignjatovic

University of Rochester, Rochester, NY, 14627 USA

ABSTRACT

Data hiding in media, including images, video, and audio, as well as in data files is currently of great interest both commercially, mainly for the protection of copyrighted digital media, and to the government and law enforcement in the context of information systems security and covert communications. In this paper we present a technique for inserting and recovering "hidden" data in audio files. In this technique the phase of chosen components of the host audio signal is manipulated in a way that may be detected by a receiver with the proper "key". Without the key, the hidden data is undetectable, both aurally and via blind digital signal processing attacks. The method described is both aurally transparent and robust and can be applied to both analog and digital audio signals, the latter including uncompressed as well as compressed audio file formats such as MP3.

1. INTRODUCTION

Data hiding is a form of steganography that embeds data into digital media for the purpose of identification, annotation, and copyright. Among different host signals, audio data hiding is especially challenging, because of the complexity of the human auditory system (HAS), [1]. The HAS operates over a wide dynamic range, while its differential range is fairly small. Moreover, the HAS is unable to perceive absolute monaural phase, except in certain contrived situations. The limitations of the HAS present opportunities for lossy compression. Thus, in order to hide data in audio files and to have the data survive lossy compression, one must focus on the features of the audio file that survive typical lossy compression methods. In this paper we describe a phase coding method that has been applied to uncompressed audio and shown to survive MPEG I layer III (MP3) compression. Furthermore the technique may be applied directly to MP3 audio files.

2. DATA HIDING BY MANIPULATION OF COVER FILE PHASE INFORMATION

In recent years, a number of audio data hiding methods have been proposed and demonstrated, including low-bit coding, spread spectrum coding, phase coding, echo data hiding, etc, [2,3,4,6,7]. Among the proposed techniques, phase coding, when it can be used, gives the best signal-to-perceived noise ratio [5].

Phase discontinuity of an audio signal is perceptible when the phase relation between each frequency component of the signal is dramatically changed. Thus, inaudible phase coding can only be achieved by keeping the modification of the phase sufficiently small and slowly varying.

In this paper we describe a method for hiding data in audio files that employs the manipulation of the phase of selected spectral components of the host audio file. This method has been demonstrated in uncompressed audio files and also has been demonstrated to survive MP3 compression and decompression. In the following we describe the method in more detail.

Naturally occurring audio signals such as music or voice contain a fundamental frequency and a spectrum of overtones with well-defined relative phases. When the phases of the overtones are modulated to create a composite waveform different from the original, the difference will not be easily detected. Thus, the manipulation of the phases of the harmonics in an overtone spectrum of voice or music may be exploited as a channel for the transmission of hidden data.

In Figure 1 below we display the magnitude and phase spectrogram of a few seconds of speech. In addition to the usual display of the magnitude of the spectral density (in the top plot) the phase of the spectrum is also displayed (in the lower plot). The phase of the spectrum is apparently random. This was verified by computing the autocorrelation in frequency of each spectral "slice"; it was found to be highly peaked at zero delay indicating no correlation.

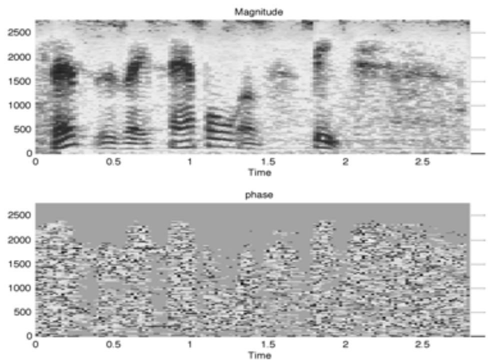

Figure 1: Spectrogram of a short excerpt of speech (male voice speaking "This is a sample of speech."). The top figure shows the magnitude of the spectrum as a function time. The bands of horizontal lines represent the overtone spectrum of the pitched portions of the signal. The lower figure displays the phase of the spectrum, which appears to be random by inspection, randomness of phase is verified by autocorrelation analysis of the phase spectrogram.

Examination of various music and speech spectrograms indicates an apparent randomness of phase, which is not surprising since the analysis frequencies of the spectral analysis are not phase coherent with the frequencies present in the signal. The fact that the phases are random presents an opportunity to replace the random phase in the original sound file with any pseudo-random sequence in which one may embed hidden data. In such an approach the embedded data is encoded in the larger features of the cover file, which enhances the robustness of the method. To extract the embedded data one must have a "key" to distinguish the phase modulation encoding from the inherent phase randomness of the audio signal.

2.1. The Phase Encoding Scheme

2.1.1. Relative Phase Encoding

A first method of phase encoding is indicated in Figure 2. In the illustrated method, during each time frame one selects a pair (or more) of frequency components of the spectrum and re-assigns their relative phases. The choice of spectral components and the selected phase shift can be chosen according to a pseudo-random sequence known only to the sender and receiver. To decode, one must compute the phase of the spectrum and correlate it with the known pseudo-random carrier sequence.

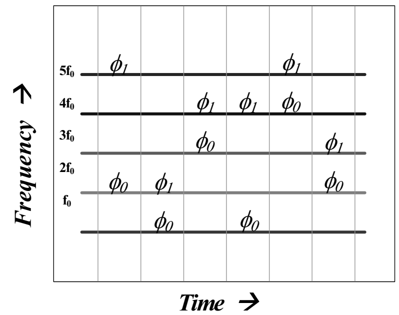

Figure 2: A phase encoding scheme is indicated in which information is inserted as the relative phase of a pair of partials in the sound spectrum. In each time frame a new pair of partials may be chosen according to a pseudo-random sequence known only to the sender and receiver. The relative phase, between the two chosen spectral components is then modified according to a pseudo-random sequence onto which the hidden message is encoded.

2.1.2. Quantization Index Modulation Phase Encoding

In an alternative phase encoding scheme we employ a variable set of phase quantization steps as illustrated below in Figure 3.

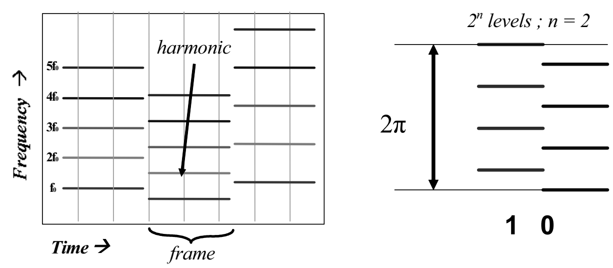

Figure 3: In an alternative phase encoding method the following steps are employed. One first computes the spectrum of a frame of audio data, then selects an apparent fundamental tone and its series of overtones as shown in the left figure; it is convenient to select the strongest frequency component in the spectrum. Then, one or more of the overtones in the selected series is "phase quantized" according to one of two quantization scales, as shown on the right. The choice of quantization levels indicates a "1" or "0" datum. The phase-quantized spectrum is then inverse transformed to convert back to the time domain.

In the phase quantization index modulation (QIM) [8] method illustrated in Figure 3, the following steps are employed.

Step1:

Segment the time representation of the audio signal $S[i]$, ($0 \le i \le I-1$) into series of frames of L points $S_n[i]$ where ($0 \le i \le L-1$).

Step2:

Compute the spectrum of each frame of audio data and calculate the phase of each frequency component within the frame, $\Phi_n(\omega_i)$ $(0 \le i \le L-1)$. An idealization of a typical spectrum with a fundamental and accompanying overtone series is shown in Figure 3.

Step3:

Quantize the phases of one or more of the overtones in the selected frame according to one of two quantization scales, as shown on the right of Figure 3.

$$\Delta\Phi = \pi / 2^{n}$$

If '1' is to be embedded,

$$\Phi_n(\omega_i) = \Delta\Phi \times \text{round}(\Phi_n(\omega_i) / \Delta\Phi)$$

If '0' is to be embedded,

$$\Phi_n(\omega_i) = \Delta\Phi \times \text{round}(\Phi_n(\omega_i) / \Delta\Phi \text{ -0.5})+ \Delta\Phi /2$$

The number of quantization levels 'n' is variable; The greater the number of levels the less audible the effect of phase quantization. However, when a greater number of quantization levels are employed the probability of data recovery error increases.

Step4:

Inverse transform the phase-quantized spectrum to convert back to the time representation of the signal by applying an L point IFFT.

Recovery of the embedded data requires the receiver to compute the spectrum of the signal and to know which spectral component has been phase quantized.

2.2. Decoding Error Rates versus SNR

We applied the method described above to the 2nd strongest harmonics of a 23 seconds long guitar solo. A Gaussian noise was introduced prior to decoding. The decoding error rate at 3 different quantization levels with increasing signal to noise ratio (SNR) is shown in Figure 4.

Applying the method described here to 512 points frames of 44,100 samples/sec audio one may encode 86 bits per second per chosen spectral line. This is slightly over 5 kbits/minute. We have also employed the method on up to 4 harmonics of the overtone spectrum with satisfactory results, raising the data capacity to approximately 20 kbits/minute.

2.3. Phase Discontinuities Reducing

An artifact of the phase manipulation method is a small discontinuity at the frame boundaries caused by reassignment of the phase of one of the spectral components.

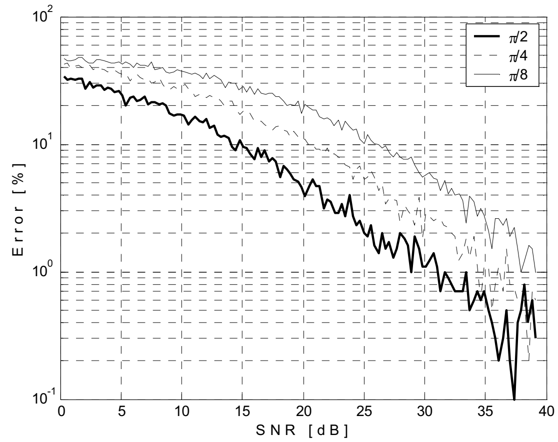

Figure 4: The decoding error rate vs. SNR. The 2nd strongest harmonic of the music file was phase quantized and embedded with 1 kbits of binary data then followed with the decoding process in the presence of Gaussian noise. The above was done at 3 quantization levels of n=1,2 and 3 respectively.

Depending upon the magnitude of the discontinuity a broad spectral component may appear in the background of the host file spectrum. In order to reduce the magnitude of the discontinuity three techniques have been employed. In the first, rather than reassigning the phase of a single spectral component we do so for a band of frequencies in the neighborhood of the spectral component of interest. We typically use a band of frequencies of width equal to a few percent of the signal bandwidth. A second method is to employ an error diffusion technique using a sigma delta modulator. Although both of these methods proved to be acceptable a third method proved to be the simplest and most effective.

The third method for reducing the phase discontinuities at the frame boundaries is simply to force the phase shifts to go to zero at the frame boundaries. In our implementation we employed a raised cosine function $(1+\cos)^n$ with n = 10. At the frame boundaries the phase of the chosen harmonic is not shifted and in the central region of the frame the phase is shifted by an amount equal to the difference of the original phase of the chosen harmonic and the nearest phase quantization step. the audible artifacts are eliminated in this method.

3. LOSSY COMPRESSION AND ROBUSTNESS OF EMBEDDED DATA

MP3 is a common form of lossy audio compression that employs human auditory system features, frequency and temporal masking, to compress audio by a factor of approximately 1:10, [9].

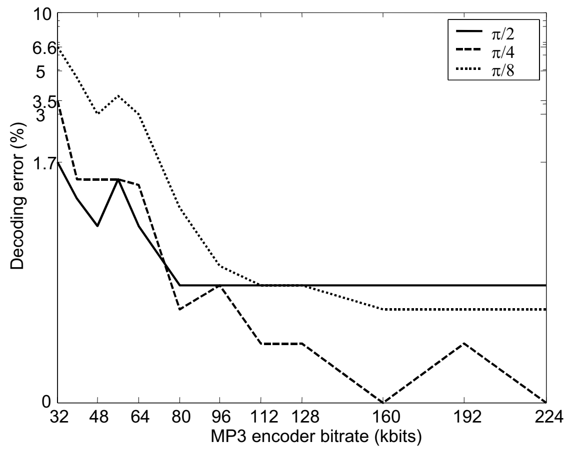

Figure 5: The decoding error rate was illustrated as the MP3 encoder bitrate ranging from 32 kbits to 224 kbits. 3 quantization levels were used separately. The frame length is 576 points. The sampling frequency is 44,100.

The robustness of the data hiding techniques described above has been assessed by applying QIM to uncompressed (.wav) audio file followed by conversion to MP3 format and then back to .wav format. The spectrograms of the final wav files were indistinguishable from the originals and the audio quality was typical of MP3 compressed audio. In the example of decoding the embedded binary data from the wav file that was converted back from mp3, the mp3 file was created from the original wav file that was embedded with 1 kbits data in the 2nd harmonic. In figure 5, the decoding error rate was illustrated as the MP3 encoder bitrate ranging from 32 kbits to 224 kbits. 3 quantization levels were applied separately. The frame length was 576 points. The sampling frequency was 44,100.

It was found that the data recovery error rate could be reduced to near zero by employing an amplitude threshold in the selection of the segments of audio data to be encoded. A weak form of error correction could be employed to guard against such infrequent errors. One also may implement the techniques described above directly in compressed audio files, which would eliminate recovery errors. Efforts to do so are ongoing.

4. CONCLUSION

Two new phase coding methods for hiding data in audio files have been proposed. The methods allow up to 20 kbits of data per minute to be embedded in uncompressed or compressed audio files. Additionally, we described three effective techniques for reducing the phase discontinuity, and the associated aural artifacts, caused by the phase coding methods.

5. ACKNOWLEDGEMENT

This work was supported by the Air Force Research Laboratory/IFEC under grant number F30602-02-1-0129.

6. REFERENCES

[1] L. W. Chang, and I. S. Moskowitz, "Critical Analysis of Security in Voice Hiding Techniques", in Information and Communications Security – First International Conference, Beijing, China, Nov. 1997, vol. 1334 of Lecture Notes in Computer Science, Springer Verlag, Berlin, Germany, pp. 203–216.

[2] S. Katzenbeisser, and F.A.P. Petitcolas, "Information Hiding: Techniques for Steganography and Digital Watermarking", Artech House, Norwood, MA, 2000.

[3] F.A.P. Petitcolas, R.J. Anderson, and M.G. Kuhn, "Information Hiding – A Survey", Proceedings of the IEEE, 87(7), 1999, pp. 1062-1078.

[4] R.J. Anderson, and F.A.P. Petitcolas, "On the Limits of Steganography", IEEE Journal of Selected Areas in Communications, 16(4), 1998, pp. 474-481.

[5] W. Bender, D. Gruhl, N.Morimoto, A Lu, "Techniques for data hiding", IBM System Journal, Vol 35, NOS 3&4, 1996.

[6] M. D. Swanson, B. Zhu, A. H. Tewfik, L. Boney , "Robust audio watermarking using perceptual masking", Signal Processing, 66(3), 1998, pp. 337–355.

[7] D. Gruhl, W. Bender, and A. Lu, "Echo hiding, in Information Hiding," First International Workshop, R. J. Anderson, Ed., vol. 1174 of Lecture Notes in Computer Science, Isaac Newton Institute, Cambridge, England, May 1996. Springer-Verlag, Berlin, Germany, pp. 295–315.

[8] B. Chen and G. W. Wornell, "Digital watermarking and information embedding using dither modulation," in Proc. of IEEE Workshop on Multimedia Signal Processing (MMSP-98), Redondo Beach, CA, USA, Dec. 1998, pp. 273–278.

[9] ISO/IEC 11 172-3, "Information technology-Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s-Part 3: Audio," First edition, Aug.1993.