International Journal of Computing and Corporate Research
IJCCR India

A Robust Audio Steganographic Technique based on Phase Shifting and Psycho – acoustic Persistence of Human Hearing Ability

Prof. Samir Kumar Bandyopadhyay, Senior Fellow of IEEE
Department of Computer Science and Engineering
University of Calcutta, Kolkata-700009, India

Tuhin Utsab Paul
Department of Computer Science and Engineering
University of Calcutta, Kolkata-700009, India

Avishek Raychoudhury
Department of Computer Science and Engineering
University of Calcutta, Kolkata-700009, India

Abstract

In this paper, a new technique for hiding the data as audio stream has been proposed. This method uses the psycho – acoustic theory of persistence of hearing in human to hide one audio stream behind another audio stream by using phase shift and signal insertion. The method proposed in the paper hides one target audio file behind one cover audio file thereby increasing the data hiding capacity greatly and without doing distortion to the cover audio file. For the cause of security only the final stego-audio file is sent over the network. In this paper we use

WAVE audio files as the carrier file, and allow messages (as WAVE audio file) of (practically) huge length to be hidden within the sound data. The sound file itself looks identical and sounds the same to the human ear, so the existence of the message is very difficult to detect.

Keywords: Audio, Hiding, Phase shift, Persistence of hearing, Security.

1. INTRODUCTION

Steganography is the study of techniques for hiding the existence of a secondary message in the presence of a primary message. Steganography, coming from the Greek words stegos (in Greek, στεγοσ) and it means roof or covered and graphia (in Greek, γραπηια) which means writing, is the art and science of hiding the fact that communication is taking place. The primary message is referred to as the carrier signal or carrier message; the secondary message is referred to as the payload signal or payload message. Generally, in steganography the following operations are performed:

1) Write a non-secret cover message.

2) Produce a stego-message by concealing a secret embedded message on the cover-message by using a stego-key.

3) Send the stego-message over the insecure channel to the receiver.

4) At the other end, on receiving the stego-message, the intended receiver extracts the secret embedded message from the stego-message by using a pre agreed stego-key.

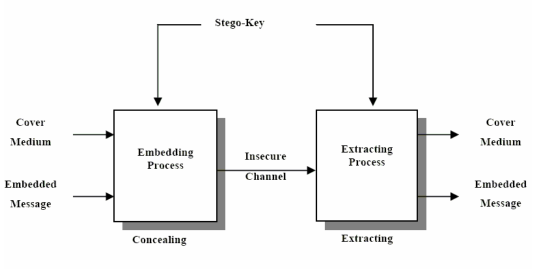

Fig. 1: The General Steganography System

Steganography is a method that establishes a covered information channel in point-to-point connections, whereas watermarking does not necessarily hide the fact of secret transmission of information from third persons. Besides preservation of the carrier signal quality, watermarking generally has the additional requirement of robustness against manipulations intended to remove the embedded information from the marked carrier object. This makes watermarking appropriate for applications where the knowledge of a hidden message leads to a potential danger of manipulation [1].

However, even knowledge of an existing hidden message should not be sufficient for the removal of the message without knowledge of additional parameters such as secret keys [2]. Obviously, the most significant applications of data hiding are covert communication.

Steganographic algorithms can be characterized by a number of defining properties. Three of them, which are most important for audio steganographic algorithms, are defined below.

Transparency evaluates the audible distortion due to signal modifications like message embedding or attacking. In most of the applications, the steganography algorithm has to insert additional data without affecting the perceptual quality of the audio host signal. The fidelity of the steganography algorithm is usually defined as a perceptual similarity between the original and stego audio sequence. However, the quality of the stego audio is usually degraded, either intentionally by an adversary or unintentionally in the transmission process, before a person perceives it. In that case, it is more adequate to define the fidelity of a steganography algorithm as a perceptual similarity between the stego audio and the original host audio at the point at which they are presented to a consumer.

Capacity of an information hiding scheme refers to the amount of information that a data hiding scheme can successfully embed without introducing perceptual distortion in the marked media. In the case of audio, it evaluates the amount of possible embedding information into the audio signal. The embedding capacity is the all included embedding capacity (not the payload) and can be measured in percent (%), bits per second or frame and bits per megabyte or kilo byte audio signal. In the other words, the bit rate of the message is the number of the embedded bits within a unit of time and is usually given in bits per second (bps). Some audio steganography applications, such as copy control, require the insertion of a serial number or author ID, with the average bit rate of up to 0.5 bps. For a broadcast monitoring watermark, the bit rate is higher, caused by the necessity of the embedding of an ID signature of a commercial within the first second at the start of the broadcast clip, with an average bit rate up to 15 bps.

Somehow, we need to rely on the internet to transfer confidential information. But, as we know, there are no existing technologies that can guarantee we are not attacked by information thieves. Many existing encryption algorithms are proved to be fragile indeed. Even with protection by modern encryption technologies, some people still can break the ciphers by the way of brute force attack with dictionary or rainbow tables.

In some envisioned applications, e.g. hiding speech in audio or compressed audio stream in audio, algorithms have to be able to embed message with the bit rate that is a significant fraction of the host audio bit rate, up to 150 kbps [3]. Robustness measures the ability of embedded data or watermark to withstand against intentional and unintentional attacks. Unintentional attacks generally include common data manipulations such as lossy compression, digital-to-analog conversion, re-sampling, re-quantization, etc. whereas intentional attacks cover a broad range of media degradations which include addition white and coloured noise, rescaling, rotation (for image and video steganography schemes), resizing, cropping, random chopping, and filtering attacks [2]. Also, the robustness of the algorithm is defined as an ability of the data detector to extract the embedded message after common signal processing manipulations [1].

We might want to transfer information discreetly for avoiding those attacks. Cover channel can solve the problem. The benefit of the cover channel is not only simply encrypting information into a carrier media file but also confusing attackers while the user is transferring sensitive information. It will make the encrypted information not directly exposed to attackers. Because online media is so popular, this plan is achievable. Here, we are trying to improve security by using some wave media as a cover channel. For making the cover channel robust, we introduced the Phase shift and signal insertion in the audio file based on psycho – acoustic theory of human persistence of hearing.

2. RELATED WORKS

.

In a computer-based audio steganography system, secret messages are embedded in digital sound. The secret message is embedded by slightly altering the binary sequence of a sound file. Existing audio steganography software can embed messages in WAV, AU, and even MP3 sound files. Embedding secret messages in digital sound is usually a more difficult process than embedding messages in other media, such as digital images. In order to conceal secret messages successfully, a variety of methods for embedding information in digital audio have been introduced. These methods range from rather simple algorithms that insert information in the form of signal noise to more powerful methods that exploit sophisticated signal processing techniques to hide information. The list of methods that are commonly used for audio steganography are listed and discussed below:

- LSB coding
- Parity coding
- Phase coding
- Spread spectrum
- Echo hiding

Using the least-significant bit is possible, as modifications will usually not create audible changes to the sounds. Another method involves taking advantage of human limitations. It is possible to encode messages using frequencies that are inaudible to the human ear. Using any frequencies above 20.000 Hz, messages can be hidden inside sound files and will not be detected by human checks.

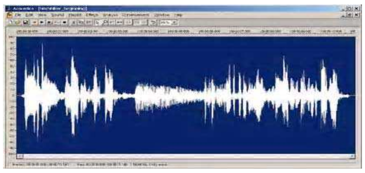

Figure 2: The signal level comparisons between a WAV carrier file before (left) and after (right) the LSB coding is done

Parity coding Instead of breaking a signal down into individual samples, the parity coding method breaks a signal down into separate regions of samples and encodes each bit from the secret message in a sample region's parity bit.

Phase coding addresses the disadvantages of the noise inducing methods of audio steganography. Phase coding relies on the fact that the phase components of sound are not as perceptible to the human ear as noise is. Rather than introducing perturbations, the technique encodes the message bits as phase shifts in the phase spectrum of a digital signal, achieving an inaudible encoding in terms of signal-to-perceived noise ratio.

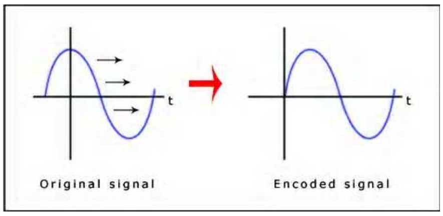

Figure 3: The signals before and after Phase coding procedure

In the context of audio steganography, the basic spread spectrum (SS) method attempts to spread secret information across the audio signal's frequency spectrum as much as possible. This is analogous to a system using an implementation of the LSB coding that randomly spreads the message bits over the entire sound file. However, unlike LSB coding, the SS method spreads the secret message over the sound file's frequency spectrum, using a code that is independent of the actual signal. As a result, the final signal occupies a bandwidth in excess of what is actually required for transmission.

In echo hiding, information is embedded in a sound file by introducing an echo into the discrete signal. Like the spread spectrum method, it too provides advantages in that it allows for a high data transmission rate and provides superior robustness when compared to the noise inducing methods. If only one echo was produced from the original signal, only one bit of information could be encoded. Therefore, the original signal is broken down into blocks before the encoding process begins. Once the encoding process is completed, the blocks are concatenated back together to create the final signal.

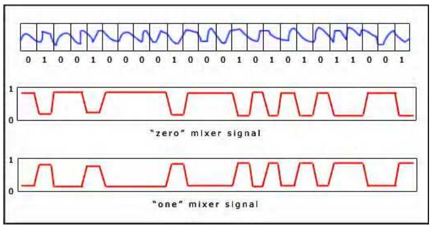

Figure 4: An example of echo hiding

Also, a message can be encoded using musical tones with a substitution scheme. For example, a Fis-tone will represent a 0 and a C tone represents a 1. A normal musical piece can now be composed around the secret message or an existing piece can be selected together with an encoding scheme that will represent a message.

3. PROPOSED WORKS

3.1 Psycho – acoustic theory of persistence of hearing

Memory is crucial to the existence of human beings and animals. Without it we would not be able to piece together the past with the present and would be living without any reference as to what, how, or why we are alive. Psychology defines memory as the brain's ability to encode, store, retain, and retrieve thoughts and sensory experiences. Memories are stored throughout the entire brain in an intricate matrix or web.

Iconic memory allows us to perceive visual information but does not give us the ability to recall it. It simply serves as a stepping stone between our eyes and our brain's memory.

Much like iconic memory, echoic memory is a form of sensory memory. It allows us to store auditory information for as long as four seconds so that our brains may translate and encode what it is we are hearing. Echoic memory is used when trying to determine one's location relative to a noise. When our brain is subconsciously determining the amount of time that passes between the recognition of a noise in one ear to the next, the data is stored in echoic memory. If two sounds hit our ear drum in a succession of less than a tenth of a second, our echoic memory cannot distinctly perceive the difference between the two sounds. This effect is called Persistence of hearing [4-8].

International Journal of Computer Science and Security (IJCSS is seeking research papers, technical reports, dissertation, letter etc for these interdisciplinary areas. The goal of the IJCSS is to publish the most recent results in the development of information technology. These instructions are for authors of submitting the research papers to the International Journal of Computer Science and Security (IJCSS).

3.2 Phase Shifting

Phase shift is any change that occurs in the phase of one quantity, or in the phase difference between two or more quantities. θ is sometimes referred to as a phase-shift, because it represents a "shift" from zero phase. But a change in θ is also referred to as a phase shift. For infinitely long sinusoids, a change in $\theta$is the same as a shift in time, such as a time-delay. If $x(t)$is delayed (time-shifted) by $\frac{1}{4}$ of its cycle, it becomes:

$$\begin{aligned} x(t - \tfrac{1}{4}T) &= A \cdot \cos(2\pi f(t - \tfrac{1}{4}T) + \theta) \\ &= A \cdot \cos(2\pi f t - \tfrac{\pi}{2} + \theta), \end{aligned}$$

whose "phase" is now $\theta - \frac{\pi}{2}$. It has been shifted by $\frac{\pi}{2}$radians.

Time is sometimes used (instead of angle) to express position within the cycle of an oscillation.

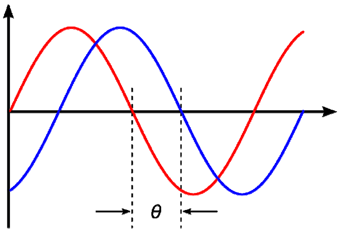

Figure 5: Illustration of phase shift. The horizontal axis represents an angle (phase) that is increasing with time.

3.3 WAV file format

Audio file format is a container format for storing audio data and metadata on a computer system. There are three major groups of audio file format nowadays: [i] Uncompressed audio formats, [e.g., WAV] [ii] Formats with lossless compression, [e.g., WMA] and [iii] Formats with lossy compression. [e.g., MP3] Our design is based on uncompressed WAV audio format. WAV file format is a subset of Microsoft’s RIFF specification for the storage of multimedia files. A RIFF file begins with a file header followed by a sequence of data chunks. In fact, a WAV file is a RIFF file with a signal “WAV” chunk. The signal “WAV” chunk is subdivided into two sub-chunks: [a] “fmt” chunk specifies the data format with some audio metadata information, such as number of channels, sample frequency rate, byte rate, and [b] block alignment. “data” chunk specifies the data sample [9-10].

The WAVE form is defined as follows. Programs must expect (and ignore) any unknown chunks encountered, as with all RIFF forms. However, <fmt-ck> must always occur before <wave-data>, and both of these chunks are mandatory in a WAVE file.

```
<WAVE-form> =>
          RIFF( 'WAVE'
             <fmt-ck>                 // Format
             [<fact-ck>]              // Fact chunk
             [<cue-ck>]               // Cue points
             [<playlist-ck>]          // Playlist
             [<assoc-data-list>]            // Associated data list
             <wave-data> )            // Wave data
```

- The WAVE format chunk <fmt-ck> specifies the format of the <wave data>
- The WAVE format chunk <fmt-ck> specifies the format of the <wave data>. The “fact”

chunk is required if the waveform data is contained in a “wavl” LIST chunk and for all compressed audio formats. The "fact" chunk will be expanded to include any other information required by future WAVE formats.

- The <cue-ck> cue-points chunk identifies a series of positions in the waveform data stream.
- The <playlist-ck> playlist chunk specifies a play order for a series of cue points.
- The <assoc-data-list> associated data list provides the ability to attach information like labels to sections of the waveform data stream.
- The <wave-data> contains the waveform data.

Here, we are concerned with the <wave-data> portion of the file. It is defined as follows:

```
<wave-data> => { <data-ck> | <data-list> }
<data-ck> => data( <wave-data> )
<wave-list> => LIST( 'wavl' { data-ck> |            // Wave samples
<silence-ck> }... )                                  // Silence
<silence-ck> => slnt( <dwSamples:DWORD> )                    // Count of silent samples
```

3.4 Size constraint between target and cover medium

Let, the sampling rate and the total duration of the Cover and Target Audio file is S1, T1 and S2, T2 respectively.

Now, we are splitting the Target Audio file after every Θ seconds where 0<Θ< $\frac{1}{10}$ .So, each piece will contain (S2*Θ) samples.

So, total number of pieces is calculated as: $\frac{S2*T2}{S2*\Theta}$ = T2/Θ

Now, incase of Cover Audio file, sampling rate is S1.So, for embedding (S2*Θ) samples of the Target Audio file inside the Cover Audio file time required(i.e Phase Shift) is ((Θ*S2)/S1) second. For using only integer value, we are using the ‘floor’ function.

For uniform distribution of the amplitudes of the Target Audio file, (S2*Θ) samples are embedded after every $\frac{S1*T1}{\frac{T2}{\Theta}} = \frac{S1*T1*\Theta}{T2}$ samples of the Cover Audio file. Hence, after every $\frac{T1*\Theta}{T2}$ second in the Cover Audio file the amplitudes of the Target Audio file are embedded. One necessary condition for this technique is that the Cover Audio file must be longer than Target Audio file (T1>T2) such that, $\frac{T1*\Theta}{T2} \geq 3$, i.e T1 ≥ $\frac{3*T2}{\Theta}$

3.5 Sender Side Approach

The steganographic approach proposed here is based on the Psycho – acoustic theory of persistence of hearing which states that audio signals of duration less than one – tenth of a second cannot be perceived by human year. This property is used to hide one audio file (target) behind another (cover) audio file in an imperceptible way. The target audio file signal is fragmented in fragments of time duration less than one – tenth second. These fragments are inserted at regular interval of the cover audio file by segmenting the cover audio signal by phase shifting. Thus the updated cover signal i.e. with target signal segments inserted in original cover signal, is the final stego – audio signal which is transmitted to the receiver end. This stego – audio sound similar to the original cover signal as the inserted signal segments are less than the duration of one – tenth second and thereby imperceptible to human year.

3.6 Receiver side approach

At the receiver end, only the stego – audio file is received. The inserted signal fragments are retrieved from regular intervals (in which they were inserted) of the stego – audio file. These signal segments were concatenated to recreate the original target audio signal. Thus the final target audio signal is recreated without any data loss.

3.7 Algorithms

The formal algorithms for the sender end and receiver end are as follows:

3.7.1 Sender end

This is the main function in our algorithm. This function will be used in the sender side to encrypt the Target Audio file inside the Cover Audio file.

Input: This function will take Target Audio file and Cover Audio file as input.

Output: It will output the Stego-Audio file.

1. Read the Cover Audio file and the Target Audio file. Obtain their corresponding time-amplitude plot and sampling rate, total duration say S1, T1 and S2, T2 respectively.
2. Split the Target Audio file after every Θ seconds where 0<Θ< $\frac{1}{10}$ into (T2/Θ) different pieces containing (S2*Θ) samples in each piece.
3. After every (floor ((T1*Θ)/T2)) second shift the phase of the Cover Audio file for (floor (Θ*S2/S1)) second.
4. Update the 1st four amplitude value of the Cover Audio file with the value of T1, S2, T2, Θ in an encrypted form.
5. Embed each pieces of the Target Audio file one by one in the generated vacant space inside the Cover Audio file due to Phase Shifting. This step is continued until the entire Target Audio file is embedded within the Cover Audio file.
6. Save the Audio file with the sampling rate S1 as Stego-Audio file.
7. Send the Stego-Audio file to the receiver.

3.7.1 Receiver end

This function is used in the algorithm to decrypt the Target Audio file from the received Stego-Audio file.

Input: This function will take the Stego-Audio file as input.

Output: This function will output the Target Audio file as the Final audio file.

1. Read the received Stego-Audio file. Obtain its sampling rate say S1 and its corresponding time-amplitude plot.
2. Decrypt and obtain T1,S2, T2, Θ from the 1st four amplitude values of the Stego-Audio file respectively .
3. Take NULL signal ‘A’.

4. After (floor ((T1*Θ)/T2)) second of the Stego-Audio file, cut the next signal component having (S2*Θ) amplitude values for (floor (S2*Θ)/S1) seconds and concatenate with A.
5. Start from the next amplitude value and repeat Step 4 until A contains a signal with (S2*T2) amplitude sample values.
6. Sample the amplitude values stored in A with the sampling rate S2 and save it as the Final Audio file.
7. Return the Final Audio File.

4. TEST RESULT

Sender end :

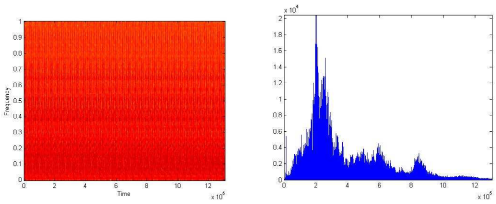

Figure 6: Frequency vs Time plot of the cover WAV file and histogram

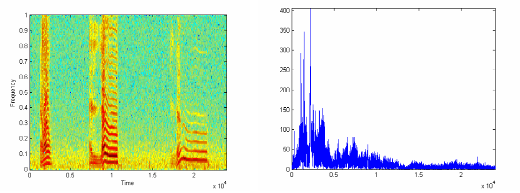

Figure 7: Frequency vs Time plot of the target WAV file and histogram

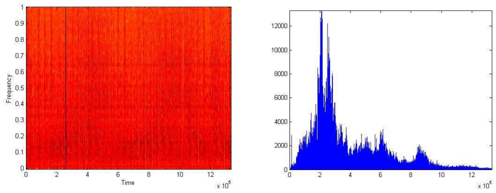

Figure 8: Frequency vs Time plot of the stego WAV file and histogram

Reciever End :

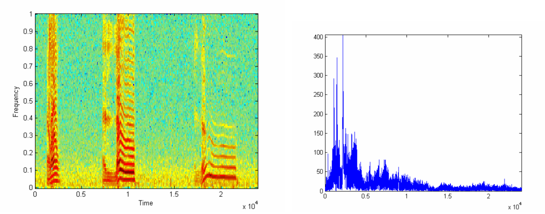

Figure 9: Frequency vs Time plot of the decrypted WAV file and histogram

5. Computation Complexity:

In the Sender end, splitting of the Target Audio file, phase shifting the Cover audio file and then the embedding procedure is performed sequentially. So, for total number of samples m, n for Cover and Target Audio file, the time complexity is O(max(m,n)).

In the Receiver end, the decryption of the Target Audio file is also performed sequentially. So, for total number of samples m, n for Cover and Target Audio file, the time complexity is O(max(m,n)).

Conclusion

A new approach is proposed to resolve two problems of substitution technique of audio steganography.

The proposed algorithm for steganography has the following unique features :

- The property of psycho – acoustic theory of persistence of human hearing ability makes the signal insertion imperceptible to ordinary hearing thereby having no cause for suspicion.
- The results of steganolysis as shown above shows that the insertion is almost undetectable.
- The algorithm runs in linear time, so it is feasible for real time application.
- There is no data loss in transmission and reconstruction of the target audio file.
- Since the cover medium is audio file, so the robustness is largely increased.

- Data insertion capacity is large.
- Since only the stego – audio file is transmitted to the receiver end, so it provides extra security and less chance of suspicion.

REFERENCES

1. Martín Alvaro, Sapiro Guillermo and Seroussi Gadiel, “Is Image Steganography Natural?” IEEE Transactions On Image Processing, Vol. 14, No. 12, December, 2005.

2. Cvejic N. and Seppänen T. ”Increasing the capacity of LSB based audio steganography”, Proc. 5th IEEE International Workshop on Multimedia Signal Processing, St. Thomas, VI, December 2002, pp. 336-338.

3. Lee, Y. K. and Chen L. H. “High Capacity Image Steganographic Model”. IEEE Proceedings Vision, Image and Signal Processing, pp. 288-294, 2000.

4. Pal S.K., Saxena P. K. and Mutto S.K. “The Future of Audio Steganography”. Pacific Rim Workshop on Digital Steganography, Japan, 2002.

5. Westfeld A. and Pitzmann A. "Attacks on Steganographic Systems". Lecture Notes in Computer Science, vol. 1768, Springer-Verlag, Berlin, pp. 61-75, 2000.

6. Johnson, N. F. and Jajodia, S. (1998). Exploring steganography: Seeing the unseen. Computer, 31(2):26–34.

7. Niels Provos, Peter Honeyman, Hide and Seek: Introduction to Steganography (2003).

8. Robert Krenn. Steganography and steganalysis

9. Westfeld, A. (2001). F5-a steganographic algorithm: High capacity despite better steganalysis. In Proc. 4th Int’l Workshop Information Hiding, pages 289–302.

10. E. Koch, J. Rindfrey, and J. Zhao, “Copyright Protection for Multimedia Data,” Proc. Int’l Conf. Digital Media and Electronic Publishing, Leeds, UK 1994.