A Variant of LSB Steganography for Hiding Images in Audio

Kriti Saroha
SOIT
CDAC
U.P., INDIA

Pradeep Kumar Singh
Department of Computer Science
KIET
U.P., INDIA

ABSTRACT
Information hiding is the technology to embed the secret information into a cover data in a way that keeps the secret information invisible. This paper presents a new steganographic method for embedding an image in an Audio file. Emphasis will be on the proposed scheme of image hiding in audio and its comparison with simple Least Significant Bit insertion method of data hiding in audio.

General Terms
Audio Steganography.

Keywords
Information hiding, Steganography, LSB.

1. INTRODUCTION
The increased popularity of digital media has raised serious concerns over its security related issues. Security attacks in the form of eavesdropping, masquerading and tampering and in many other forms is very common nowadays. Data hiding is one of the emerging techniques that aim to provide for security by hiding secret information into the multimedia contents by altering some nonessential components in the host or cover file. Information hiding, Steganography, and Watermarking are three closely related fields that have a great deal of overlap and share many technical approaches. Information hiding (or data hiding) is a general term encompassing a wide range of problems beyond that of embedding messages in content. The term hiding here can refer to either making the information imperceptible (as in watermarking) or keeping the existence of the information secret. Steganography, coming from Greek word stegos, meaning covered and graphia, which means writing, is the art and science of hiding the information within information. Steganographic systems, can be divided into two categories, One in which the very existence of the message is kept secret, and nonsteganographic systems, in which the existence of the message need not be a secret. Watermarking approach is protection against removal, and is used for document marking, for embedding information about the author or for embedding a serial number; in other words, copyrights information. In this case, the goal is protection against removal; the watermark might or might not be made visible to the user using a watermarking reader tool. There are two different techniques for document marking: watermarking and fingerprinting. Watermarking is the process of embedding marks in digital documents (sounds, images, binaries, etc.) exactly like the watermarks used for example, in marking a banknote. Fingerprinting is the process of embedding a serial number into every copy of an object. This serial number can be used to detect the break of a license agreement. In both the cases, the information is supposed to be invisible, but it should be very difficult to remove it. The difference between the two processes is that in the former process the objects are all marked the same way, but in the latter process every copy has a different serial number embedded.

Modern Steganography is generally associated with embedding of secret information into the digital media like image, audio, video, and text rather than physical objects. This paper is an attempt to find a method that uses an audio file as a cover media to hide an image without making noticeable changes to the file structure and contents of the audio file. The proposed scheme is based on Least Significant Bit insertion method as it has been already proved that modification of LSB creates a minimal change in the audio file format [9].

2. AUDIO STEGANOGRAPHY
Like the document images, the sound files may be modified in such a way that they contain hidden information, like copyright information; these modifications must be done in such a way that it should be impossible for a pirate to remove it, at least not without destroying the original signal. The methods that embeds data in sound files use the properties of the Human Auditory System (HAS). The HAS perceives the additive random noise and also the perturbations in a sound file can also be detected. But there are some “holes” that we can exploit. While the HAS have a large dynamic range, it has a fairly small differential range. As a result, loud sounds tend to mask out quiet sounds. And there are also some distortions that are so common that the HAS ignores them. The digital sound is obtained from the analog sound by converting it to digital domain. This process implies two sub processes: *Sampling* and *Quantization*. Sampling is the process in which the analogue values are only captured at regular time intervals. Quantization converts each input value into one of the discrete values. The most popular file formats for sounds are the Windows Audio-Visual (WAV) and the Audio Interchange File Format (AIFF). There are also compression algorithms such as the International Standards Organization Motion Pictures Expert Group-Audio (ISO MPEG-AUDIO). The most popular format for representing samples of high- quality digital audio is a 16-bit linear quantization e.g., Windows Audio-Visual (WAV) and Audio Interchange File Format (AIFF) [8, 9]. Another popular format for lower quality audio is the logarithmically scaled 8-bit m-law. These quantization methods introduce some signal distortion, somewhat more evident in the case of 8-bit m-law. Popular temporal sampling rates for audio include 8 kHz (kilohertz), 9.6 kHz, 10 kHz, 12 kHz, 16 kHz, 22.05 kHz, and

44.1 kHz. Sampling rate impacts data hiding in that it puts an upper bound on the usable portion of the frequency spectrum (if a signal is sampled at ~8 kHz, it is not desirable to introduce modifications that have frequency components above ~4 kHz). For most data-hiding techniques developed, usable data space increases at least linearly with increased sampling rate [9].

3. TYPESET TECHNIQUES OF DATA HIDING IN AUDIO

3.1 Least Significant Bit Encoding

Least significant bit (LSB) coding is the simplest way to embed information in a digital audio file. By substituting the least significant bit of each sampling point with a binary message, LSB coding allows for a large amount of data to be encoded.

In LSB coding, the ideal data transmission rate is 1 kbps per 1 KHz. In some implementations of LSB coding, however, the two least significant bits of a sample are replaced with two message bits. This increases the amount of data that can be encoded but also increases the amount of resulting noise in the audio file as well. A novel method which increases the limit up to four bits was given by Nedeljko Cvejic, Tapio Seppben, and Media Team Oulu at Information Processing Laboratory, University of Oulu, Finland [4].

To extract a secret message from an LSB encoded sound file, the receiver needs access to the sequence of sample indices used in the embedding process. Normally, the length of the secret message to be encoded is smaller than the total number of samples in a sound file. One must decide then on how to choose the subset of samples that will contain the secret message and communicate that decision to the receiver. One trivial technique is to start at the beginning of the sound file and perform LSB coding until the message has been completely embedded, leaving the remaining samples unchanged. This creates a security problem, however in that the first part of the sound file will have different statistical properties than the second part of the sound file that was not modified. One solution to this problem is to pad the secret message with random bits so that the length of the message is equal to the total number of samples.

A more sophisticated approach is to use a pseudorandom number generator to spread the message over the sound file in a random manner. One popular approach is to use the random interval method, in which a secret key possessed by the sender is used as a seed in a pseudorandom number generator to create a random sequence of sample indices. The receiver also has access to the secret key and knowledge of the pseudorandom number generator, allowing the random sequence of sample indices to be reconstructed. Checks must be put in place, however, to prevent the pseudorandom number generator from generating the same sample index twice. If this happened, a collision would occur where a sample already modified with part of the message is modified again. The problem of collisions can be overcome by keeping track of all the samples that have already been used. Another approach is to calculate the subset of samples via a pseudorandom permutation of the entire set through the use of a secure hash function. This technique insures that the same index is never generated more than once.

There are two main disadvantages associated with the use of methods like LSB coding. The human ear is very sensitive and can often detect even the slightest bit of noise introduced into a sound file. Second disadvantage however, is that this is not robust. If a sound file embedded with a secret message using LSB coding was resampled, the embedded information would be lost. Robustness can be improved somewhat by using a redundancy technique while encoding the secret message. However, redundancy techniques reduce data transmission rate significantly.

3.2 Phase Coding

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. Phase coding is explained in the following procedure [7]:

- The original sound signal is broken up into smaller segments whose lengths equal the size of the message to be encoded.
- A Discrete Fourier Transform (DFT) is applied to each segment to create a matrix of the phases and Fourier transform magnitudes.
- Phase differences between adjacent segments are calculated.
- Phase shifts between consecutive segments are easily detected. In other words, the absolute phases of the segments can be changed but the relative phase differences between adjacent segments must be preserved. Therefore the secret message is only inserted in the phase vector of the first signal segment as follows:

$$phase\_new = \begin{cases} \pi/2 & if \quad message \quad bit = 0 \\ -\pi/2 & if \quad message \quad bit = 1 \end{cases}$$

- A new phase matrix is created using the new phase of the first segment and the original phase differences.
- Using the new phase matrix and original magnitude matrix, the sound signal is reconstructed by applying the inverse DFT and then concatenating the sound segments back together.

To extract the secret message from the sound file, the receiver must know the segment length. The receiver can then use the DFT to get the phases and extract the information. One disadvantage associated with phase coding is a low data transmission rate due to the fact that the secret message is encoded in the first signal segment only. This might be addressed by increasing the length of the signal segment. However, this would change phase relations between each frequency component of the segment more drastically, making the encoding easier to detect. As a result, the phase coding method is used when only a small amount of data, such as a watermark, needs to be concealed.

3.3 Spread Spectrum

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 [7].

The SS method has the potential to perform better in some areas than LSB coding, parity coding, and phase coding techniques in that it offers a moderate data transmission rate while also maintaining a high level of robustness against removal techniques. However, the SS method shares a disadvantage with LSB and parity coding in that it can introduce noise into a sound file.

Table 1. Table captions should be placed above the table

Graphics	Top	In-between	Bottom
Tables	End	Last	First
Figures	Good	Similar	Very well

3.4 Echo Hiding

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.

To hide the data successfully, three parameters of the echo are varied: amplitude, decay rate, and offset (delay time) from the original signal. All three parameters are set below the human hearing threshold so that the echo is not easily resolved. In addition, offset is varied to represent the binary message to be encoded. 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 [7].

To extract the secret message from the stego-signal, the receiver must be able to break up the signal into the same block sequence used during the encoding process. Then the autocorrelation function of the signal's cepstrum (the cepstrum is the forward fourier transform of the signal's frequency spectrum) can be used to decode the message because it reveals a spike at each echo time offset, allowing the message to be reconstructed.

4. PROPOSED SCHEME

The proposed scheme uses an image file as the secret data to be hidden in the audio file taken as cover object because the size of the image is generally quite small compared to the size of the audio file in which it must be hidden.

Captions should be Times New Roman 9-point bold. They should be numbered (e.g., “Table 1” or “Figure 2”), please note that the word for Table and Figure are spelled out. Figure’s captions should be centered beneath the image or picture, and Table captions should be centered above the table body. In order to hide secret information in the form of gray scale image file, we use cover file i.e. 16 bit CD quality wave audio file at 44.1 kHz which have best quality of sound characteristics. The proposed method is based on the basic design principle for hiding secret data in audio. According to the proposed method, least significant bits up to three LSB positions are used as a stego-key to encode the image bits in stego-object. Thus, the image data can be embedded according to the following embedding algorithm.

4.1 Embedding Algorithm

1. Input the gray scale image that is to be embedded and convert it into binary form.

2. Read the cover audio file. Leave the header of the audio file untouched.

3. Start with the first audio data sample and first bit of the image.

4. Do the following:

   a. Compare the image bit to be embedded with the audio sample’s 1st MSB to 7th MSB position till the first match is found.

   b. If any MSB (from position 1 to 7) of audio sample matches with the image bit, replace the three LSBs of the audio sample with the binary equivalent of the MSB position (where match is found) else,

   c. Insert all 0s into the three LSBs of the audio sample (indicating that this sample does not contain an image bit embedded into it) and,

   d. Move to the next audio sample.

5. Repeat steps 4a - 4d till the match for the image bit is found in some MSB (from position 1 to 7) of an audio sample.

6. Move to the next image bit.

7. Repeat steps 4 - 6 till all the image bits are successfully hidden into the audio file.

8. Write the wave audio file with stego-key that contains number of samples used to embed secret image in the audio file and number of rows/columns in the image.

9. Output wave audio file is the stego-object.

4.2 Extracting Algorithm

The data extraction at the receiver’s end follows the same logic as the embedding algorithm.

1. Read the stego-object i.e. Cover audio after encoding.

2. Extract the stego-key from the stego-object that contains information about the number of samples used to embed the image in the audio and number of rows/columns in the image.

3. Do the following:

a. Select the audio samples one by one and extract the image bits from the MSB position of the audio samples with respect to the decimal equivalent of the value at three LSB positions.

b. If the value at three LSB positions is all 0s, move to next sample.

4. Repeat steps 3a – 3b till all the samples used to embed the image bits are used.

5. Store the image bits retrieved from the audio file into an array.

6. Divide the array into number of rows and columns and create the secret image.

7. Display the secret image.

5. SYSTEM EVALUATION

In order to evaluate the sound quality after embedding the secret image into the audio files, two types of test were carried out.

5.1 Mean Opinion Score

Subjective quality evaluation for the image hiding in audio has been done by listening tests involving twenty persons. Ten of them had basic or medium music education or have been active musicians. A total number of the twenty audio pieces were used as test signals. Clips were 44.1 KHz sampled audio files, represented by 16 bits per sample. Duration of each sample was 20-30 seconds. The audio files are categorized as sounds of music songs, instrumental and echo sounds. The entire tests have been carried out at each category of sound files. Initially, in the first part, the listeners were repeatedly presented with the audio clips with hidden image and audio clips without any hidden image, in a random order. Listeners were then asked to differentiate between the audio files with and without hidden data.

Here, we have also calculated the mean opinion score of audios where different LSB positions of audio file i.e. simply 1st or 2nd or 3rd or 4th LSB positions were used for hiding data into it and for audios where all bits upto 4th LSB position were used for hiding data bits into it.

To calculate the mean opinion score, a 5-point impairment scale is used by the individual after listening the music file and final mean of all the scores is M.O.S. Mean Opinion Score for all the three categories of sound, for both the cases are as shown in Table1.The 5-point scale is defined in the following manner.

5: Imperceptible

4: Perceptible but not annoying

3: Slightly annoying

2: Annoying

1: Very Annoying

Table 1 shows that for music and instrumental audios, the proposed scheme score is close to upto 3LSB position score, while for echo files, it is very close to upto 2LSB position.

This can be analyzed that for echo files, there is more imperceptibility while perceptibility can be found in instrument and music songs about noise. According to the MOS table we have analyzed that our proposed scheme is better than the 3LSB and 4LSB position, where all bits upto 3rd or 4th LSB positions were used to hide the image bits. So according to the MOS score we can conclude that chances of identifying the audio with embedded image having echo sounds are very much less with respect to music or instrumental sounds.

Table1. Mean Opinion Score for stego-object containing echo, instrumental and music sounds. (ps-proposed scheme)

Case-1:MOS at 1,2,3,4 LSB Position				Case-2 : MOS upto all 4 LSB Positions		
Method	Music	Instru-ment	Echo	Musi c	Instru -ment	Echo
PS	4.5	4.2	4.7	4.5	4.2	4.7
4LSB	4.4	4.0	4.6	4.2	3.6	4.3
3LSB	4.6	4.4	4.8	4.5	4.2	4.5
2LSB	4.8	4.6	4.9	4.7	4.4	4.7
1LSB	4.9	4.8	5.0	4.9	4.8	5.0

5.2 Objective Quality Evaluation

Objective quality evaluation for the image hiding in audio has been done by SNR formula 6.02n dB, n is the number of quantized bits in audio i.e. 16, in this case. Signal to Noise ratio is calculated from the original cover file and stego-object using SNR formula for all the three categories of sounds in both cases i.e. when data bits are inserted at 1st or 2nd or 3rd or 4th LSB position and when data bits are inserted in all bits up to 4th LSB position with proposed method (PS). SNR graphs for upto 4th LSB position and proposed scheme are as shown in Figure 1, 2, and 3.

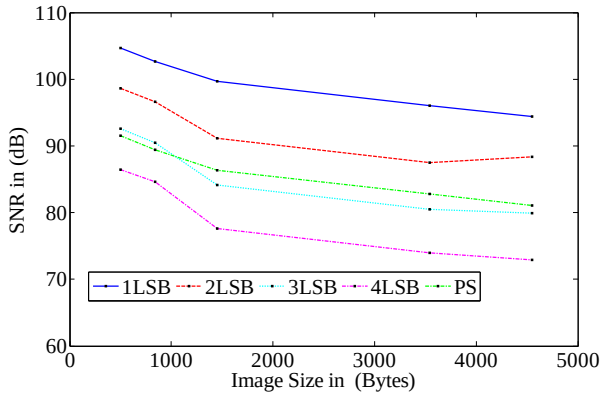

Figure1. Signal to noise ratio for different size images in music sound file upto 4th LSB position.

Figures 1, 2, and 3 shows that signal to noise ratio for all three categories of sound file i.e. (echo, instrument, music) with proposed scheme is higher than with respect to 3LSB and 4LSB positions. Higher SNR shows that signal power is higher and there is less noise in cover file i.e. stego-object as compared with 3,4,LSB positions for hiding image in audio. So, for either the cases it has been proved that our proposed scheme gives better result as compared to when 3 or 4LSB positions are used for hiding data. According to our proposed scheme, the size of the cover audio file remains the same, which is the basic principle of steganoraphy.

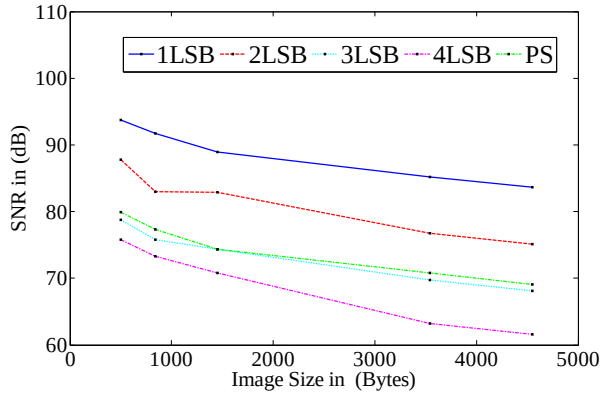

Figure2. Signal to noise ratio for different size images in instruments sound file upto 4th LSB position.

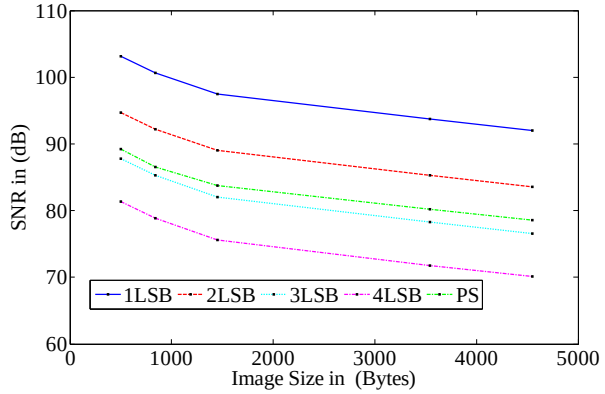

Figure3. Signal to noise ratio for different size images in echo sound file upto 4th LSB position.

6. CONCLUSION

A new method of embedding an image data into the host audio file using LSB based Audio Steganography has been successfully developed and implemented as discussed in this paper. Proposed method is found to be better than the 3LSB and 4LSB insertion methods where 3 & 4 LSB positions of audio are used to hide the image bits. Better results are analyzed by both the subjective listening testing i.e. Mean Opinion Score and Objective testing i.e. signal to noise ratio graphs. Moreover, how the performance is affected by changing different bit positions has also been reported in this work using SNR.

The main aim of this research work was embedding of image into audio as a case of audio steganography. In test cases, the image data has been successfully embedded and extracted from the audio file.

7. FUTURE SCOPE

Future research direction is to explore the possibilities of improvements in audio steganograhy system with respect to each technique of data hiding in audio. One of the areas is to enhance the storage capacity of the system. This focuses on improving the maximum capacity of the audio signal to carry hidden data into it and making it robust to steganalysis. Further, the methods can be improved by applying mixed approaches, making the system more secure towards detection by using the combination of various techniques of data hiding in audio signals.

8. REFERENCES

[1] Deshpande Neeta, Kamalapur Snehal, and Daisy Jacobs, "Implementation of LSB Steganography and Its Evaluation for Various Bits",2001.

[2] Dr. D. Mukhopadhyay, A. Mukherjee, S. Ghosh, S. Biswas, and P. Chakarborty, "An Approach for Message Hiding using Substitution Techniques and Audio Hiding in Steganography", IEEE 2005.

[3] Fabien A. P. Petitcolas, Ross J. Anderson, and Markus G. Kuhn, "Information Hiding—A Survey", Proceedings Of The IEEE, vol. 87, no. 7, July 1999.

[4] Nedeljko Cvejic, and Tapio Seppben, "Increasing The Capacity of LSB-Based Audio Steganography", IEEE 2002.

[5] Ping Wah Wong and Edward J. Delp, editors, "Security and Watermarking of Multimedia Contents", Volume 3657, Society of Photo-optical Instrumentation Engineers, 1999.

[6] Ping Wah Wong and Edward J. Delp, editors, "Security and Watermarking of Multimedia Contents II", Vol. 3971, Society of Photo-optical Instrumentation Engineers, 2000.

[7] Poulami Dutta, Debnath Bhattacharyya, and Tai-hoon Kim, "Data Hiding in Audio Signal: A Review", International Journal of Database Theory and Application, vol. 2, no. 2, June 2009.

[8] R.Anderson, and F.Petitcolas, "On the limits of the Steganography", IEEE Journal Selected Areas in Communications, vol .16, no. 4 , May 1998.

[9] W. Bender, D. Gruhl , N. Morimoto, and A. Lu, "Techniques For Data Hiding", IBM Systems Journal, vol. 35, nos 3&4, 1996.

[10] Zhou Lin-na, Liu Cui-qing, Ping Xi-jian, and Wang Yun-he, "Information Hiding Method in Digital Audio Signal Based on Perfect Codes", Information Hiding and Multimedia Signal Processing, IEEE 2006.