ISSN 2786-7447
EUROPEAN JOURNAL OF THEORETICAL AND APPLIED SCIENCES
OPEN ACCESS
PEER REVIEWED

Information Security for Audio Steganography Using a Phase Coding Method

Mohamed H. Sayed
Joaan Bin Jassim Academy for Defence Studies, Doha, Qatar
Talaat M. Wahbi
Department of computer Science, Sudan University of Science and Technology, Khartoum, Sudan

Suggested Citation
Sayed, M.H. & Wahbi, T.M. (2024). Information Security for Audio Steganography Using a Phase Coding Method. *European Journal of Theoretical and Applied Sciences, 2*(1), 634-647.
DOI: 10.59324/ejtas.2024.2(1).55

Abstract:

The art and science of steganography are dedicated to concealing the presence of a secret message from a third party, such that only the sender and recipient are aware of its content. Various types of media can be used to conceal these communications. When information is hidden in an audio signal, this is referred to as audio steganography. In this paper, two distinct steganography techniques are combined with a multi-level steganography approach: the initial message is embedded in an audio cover at the initial stage, employing, a modified LSB technique, additionally, the second message is embedded in the output from the first level, using a phase coding approach at the second level. A stego audio file is the second level's output containing two audio covers with secret messages. The message is split in multiple ways, with varying proportions between the two levels, in order to investigate how the message's size affects the two procedures used here as well as the levels. The PSNR, MSE, and histogram metrics are used to compare the original and stego audio, in order to assess the effectiveness of the suggested approach. The optimum outcome is achieved when the message is divided in the ratio (1:1). The worst outcome is achieved when the message is divided in the ratio (3:1)

Keywords: *Audio Steganography, LSB, Multi-level, MSE, Phase coding, PSNR.*

Introduction

In steganography, only the sender and recipient are aware of the presence of a secret communication, which is concealed from a third party (Mhatre et al., 2018). The Greek words stegano, which means "covered," and graphia, which means "writing" or "drawing," are the source of the term. The primary objective of steganography is to conceal the existence of a secret message (Mahajan, 2014), and it can be applied to a wide range of data types, including text, images, audio, and video. In the steganographic process, the original object is referred to as the cover or carrier object. The hidden message may be in several formats, such as text, images, audio, and video (Singh et al., 2015), and these are referred as message objects. The output file created when the steganographic approach is applied is referred to as a stego object (Milosav et al., 2023), and the process of hiding information using an audio signal is known as audio steganography. The binary sequence of an audio file is slightly altered to incorporate the hidden message (Singh et al., 2015). It is more difficult to embed secret data in digital audio than in additional media, like digital photos (Ahmed et al., 2020), as when information is inserted into an audio file through modified signals, the change must be undetectable to the auditory system of humans. The techniques used to embed data in sound files make use of the properties of the human auditory system (HAS) (Mcloughlin, 2016), which is able to identify both additive random noise and disturbances in a sound file. However, there are certain "loopholes" that can be exploited (for example, loud noises mask quiet sounds) (Mahajan, 2014). WAV, AU, and even MP3 audio files can contain embedded messages sanks to audio steganography techniques.

There are two main techniques for data hiding in audio, the first one is spatial domain techniques which consist mainly from LSB coding techniques (Shanthakumari et al., 2021), parity coding technique (Banik & Bandyopadhyay, 2018), and echo hiding technique(Xie & Wu, 2010). The second one is transform domain techniques which includes spread spectrum technique as in (Al-Najjar, 2008), phase coding technique (Alsabhany et al., 2019), discrete wavelet transform technique (Chen et al., 2021), tone incertion technique (Yousif et al., 2017) and amplitude coding technique (Bharti et al., 2019).

Al Najjar (Al-Najjar, 2008) first proposed multi-level steganography (MLS) for picture steganography. MLS is a novel approach to concealing information in communication networks that builds on the features of the upper level method to produce a new method (known as a lower-level method). The foundation of MLS is the combination of two or more steganographic techniques in which one technique, known as the upper level, serves as a carrier for another, known as the lower level (Al-Najjar, 2008).

The most basic and widely used steganography technique is the least significant bit (LSB) substitution method. The ideal approach to LSB substitution is to in order to avoid materially changing the original cover, incorporate every bit of the hidden message into the least significant portion of the cover audio (Mahajan, 2014). A significant amount of data can be encoded using LSB coding. However, in some LSB coding implementations, two message bits are substituted for the two least significant bits of the sample (Mohamad & Yasin, 2018), although this expands the amount of data that can be encoded, it also adds to the noise that ends up in the audio file. (To retrieve a secret message from an LSB-encoded cover file, the recipient needs to have access to the sample indices that were used during the embedding process (sheelu, 2013).

The phase coding technique is one of the primary methods used in audio steganography, which is based on the idea that noise is more noticeable to the human ear than the phase components of sound. It functions by replacing the first audio segment's phase with a data-representative reference phase. The relative phase between segments is then maintained by adjusting the phase of succeeding segments (Li & Kim, 2011). Phase coding has been described as effective in many review articles (Alsabhany et al., 2019), and is one of the most successful coding techniques in terms of the signal-to-perceived noise ratio. With this method, an inaudible encoding in terms of signal-to-perceived noise ratio is achieved by encoding the message bits as phase shifts in the phase spectrum of a digital signal (Al-Othmani et al., 2012). Further in this paper, the related works will be presented in third section. The proposed algorithm of audio steganography using phase coding technique will be explained in fourth section. Analysis and experimental results are depicted in fifth section. The discussion will be presented in sixth section. Finally, seventh section draws the conclusions.

Related Works

Kaur and Verma used a method for audio steganography in (Kaur & Verma, 2014) that made use of the LSB coding, parity coding, and phase coding techniques in multi-level steganography. A three-layered audio multi-level steganography method was reviewed. With a single cover file, three secret messages rather than one could be sent. Three variants of the audio steganography technique were studied, and a PSNR graph was used to compare the results for the stego audio. There are three levels to every permutation. Multi-level audio steganography can be divided into three levels, known as layers 1, 2, and 3.

In (Kumar & Banik, 2012), Kumar and Banik used the LSB modification and phase encoding techniques, which are very primitive forms of steganography. These two techniques were revisited to give an overview of how steganography works with audio file. Phase coding works by first dividing the original audio stream or cover file into blocks, then embedding the entire message data sequence into the first block's phase spectrum. The output stego file is correct and audible, and no discrepancy is found compared to the input carrier file.

In (Banik & Bandyopadhyay, 2015), Banik, and Bandyopadhyay developed a two-layered approach. In the first layer, the cover file (C) was embedded with the first secret message (S1). The stego file was denoted as C1, and formed the cover file for the next level, where the secret message was denoted as S2. The final stego file was denoted as C12, which contained both messages, S1 and S2. The two levels of steganography were identified as layers 1 and 2; at layer 1, the LSB modification technique was used, whereas in layer 2, the parity encoding technique was used.

Proposed Method

Introduction

In this paper, a layering approach is used to develop a hiding process based on two layers, using audio files as covers. We discuss the problem of hiding two secret messages in a single audio file without creating any noticeable noise in the stego object. Several steganographic methods are used in each layer, rather than a single method.

In the first layer, an enhanced LSB technique is designed to insert secret message 1 into an audio file (cover object). We treat the output of the first layer as an intermediate object that can be reused as a cover object for the second layer. (This intermediate object has nullified). In the second layer, secret message 2 is inserted into the intermediate object using a phase coding technique. Figure 1 illustrates the proposed method. Our approach uses two sites: the sender site, which handles with the embedding process at two levels, and the receiver site, which handles the extraction processes at these two levels in the reverse order.

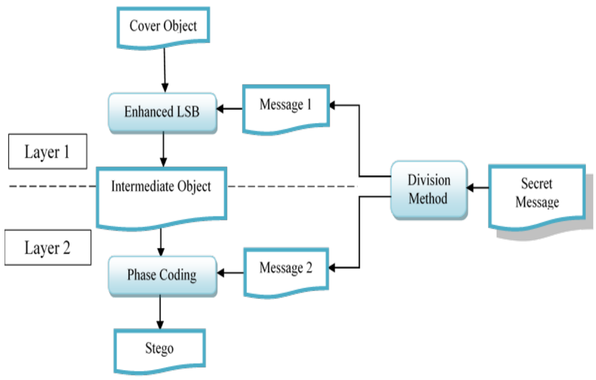


Figure 1. Proposed Multi-Level Steganography Method

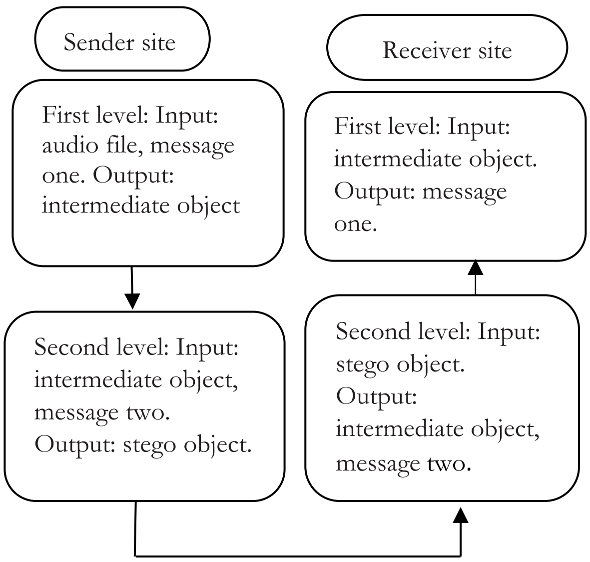


Figure 2. Sender and Receiver Sites

First Level

In this level, secret message 1 is embedded using the proposed LSB method. At the sender site, the message is hidden in the cover audio file, and the intermediate object is returned as output, whereas at the receiver site, it is extracted from the intermediate object. Figure 2 shows the input and output for this level at the sender and receiver sites.

Proposed Enhanced LSB Method

In this section, we introduce the proposed LSB method, which is employed to incorporate cover audio file's secret message 1. At the sender site, when we want to hide a message, it is first converted into 8-bit form, and the results are stored as a matrix.

Using a random key generation function, a matrix of integer key numbers is generated. These keys are used to choose the audio sample in which message bits can be hided. The number of bits that should be embedded into each step must be less than four (to avoid noise becoming noticeable).

The number of bits that are embedded in each step is determined using (1).

$$\text{Number of bits in each step} = (\text{key} \bmod 4) \quad (1)$$

For example, given a sequence of bits forming a message (10101100110010101) and a sequence of key numbers (2, 6, 3, 7, 1, 4, 3, 2), Figure 3 shows the steps of the proposed enhanced LSB method.

First Layer Algorithm at Sender Site (Encoding)

The algorithm applied at the first layer involves the following steps:

(i) Read the text (i.e. the message) to be embedded.

(ii) Convert the text into 8 bits, and store the result as a matrix.

(iii) Convert a WAV audio file (cover file) to 16 bits, and store the result as a matrix.

(iv) Using a random key generation function, generate integer key numbers and store them in a matrix.

(v) Use these key numbers to select the audio samples that will be used to hide the binary message, and to determine the number of bits that should be embedded at each step using the enhanced LSB method.

(vi) Repeat these steps until the whole message is embedded in the selected audio samples.

1	0	1	0	0	0	1	1
1	0	0	0	1	0	**1**	**0**
1	1	0	0	1	0	1	0
1	1	0	0	1	1	0	1
1	0	1	0	1	0	1	1
1	1	0	1	0	1	0	1
1	1	0	0	1	0	1	0
1	1	0	1	0	1	**1**	**0**
1	0	1	0	0	0	1	1
1	0	0	0	1	0	1	0
1	1	0	0	1	**1**	**1**	**0**
1	1	0	0	1	1	0	1
1	0	1	0	1	0	1	1
1	1	0	1	0	1	0	1
1	1	0	0	1	0	1	0
1	1	0	1	0	1	0	1
1	0	1	0	0	0	1	1
1	0	0	0	1	**0**	**1**	**1**
1	1	0	0	1	0	1	**0**
1	1	0	0	1	1	0	1
1	0	1	0	1	0	1	1
1	1	0	1	0	1	0	1
1	1	0	0	1	0	1	**0**
1	1	0	1	0	1	0	1
1	0	1	0	0	0	1	1
1	0	0	0	1	**1**	**0**	**1**
1	1	0	0	1	0	1	0
1	1	0	0	1	1	**0**	**1**
1	0	1	0	1	0	1	1
1	1	0	1	0	1	0	1
1	1	0	0	1	0	1	0
1	1	0	1	0	1	0	1

Figure 3. Steps in the Proposed Enhanced LSB Method

The output of this level is the intermediate object. Figure 4 illustrates the steps applied at the sender site.

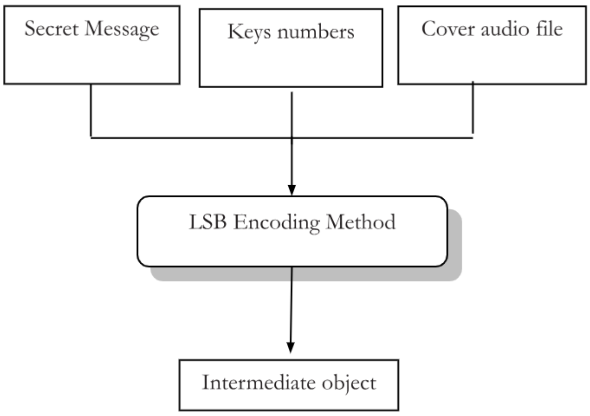


Figure 4. First Layer Encoding Method Applied at the Sender Site

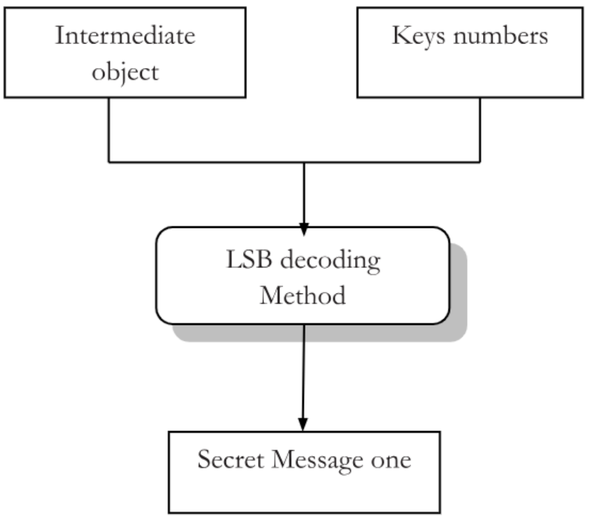


Figure 5. At the Recipient Location, the First-Level Decoding Technique is Used

First Layer Algorithm at Receiver Site (Decoding)

Decoding takes place in the following steps:

(i) Read the intermediate object (the output of level 2 after decoding) and store it as a matrix.

(ii) Extract the control data to restore the random key numbers.

(iii) Convert the audio matrix to binary. Use the key numbers to select the bits that represent message 1, and store these bits in a matrix.

(iv) Convert the matrix of data bits created in the previous step to a string.

The output of this level will be secret message one. Figure 5 illustrates the method applied at the receiver site.

Second Level

At this level, the phase coding approach is applied. The main aim at the sender site is to hide secret message 2 in the intermediate object (the output of level 1), and at the receiver site, the aim is to extract message 2 from the stego object (the output of level 2 at the sender site).

Second Layer Algorithm at Sender Site (Encoding)

(i) Break the sound sequence s[i], (0 < i < i–1), into a series of N short segments, whose lengths are equal to the size of the message to be encoded [i], where (0 < n < N–1).

(ii) Apply a K-point discrete Fourier transform (DFT) to the n-th segment, s[i], where (K = i/N), and create a matrix with phase ϕn(ωk) and magnitude An(ωk) for (0 < k < K–1).

(iii) Store the phase difference between each adjacent segment for (0 < n <N–1).

$$\Delta\varphi n+1(\omega) = \varphi n+1(\omega k) - \varphi n(\omega k)$$

(iv) A binary set of data is represented as φdata= π/2 or –π/2 representing zero or one:

$$\varphi 0' = \varphi data'$$

(v) Re-create the phase matrices for n > 0 using the phase difference as follows:

$$\begin{bmatrix} (\phi_1'(\omega_k) = \phi_0'(\omega_k) + \Delta\phi_1(\omega_k)) \\ \ldots \\ (\phi_n'(\omega_k) = \phi_{n-1}'(\omega_k) + \Delta\phi_n(\omega_k)) \\ \ldots \\ (\phi_N'(\omega_k) = \phi_{N-1}'(\omega_k) + \Delta\phi_N(\omega_k)) \end{bmatrix}$$

(vi) Use the modified phase matrix φ'n(ωk)and the original magnitude matrix An(ωk) to reconstruct the sound signal by applying the inverse DFT (Banik & Bandyopadhyay, 2018). Figure 6 illustrates the method used at the sender site.

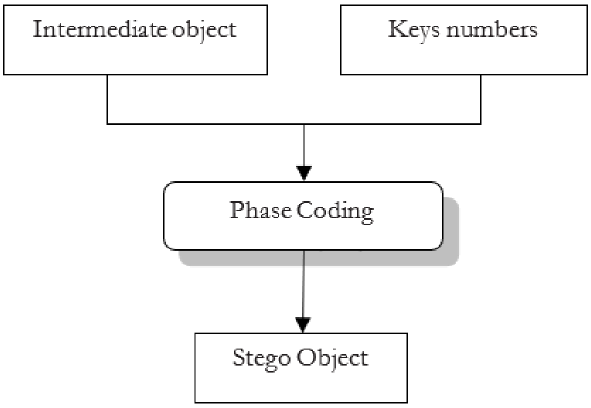


Figure 6. Second Layer Encoding Method at the Sender Site

Second Layer Algorithm at Receiver Site (Decoding)

Prior to decoding, the sequence is synchronized (Banik & Bandyopadhyay, 2018).

(i) The receiver needs to be aware of the segment's length, the DFT points, and the data interval.

(ii) The value of the underlying phase of the first segment is detected as a zero or one, which represents the coded binary string.

(iii) The absolute phases of the subsequent segments are adjusted correspondingly since φ0' = (ωk) is altered. Nonetheless, every subsequent frame's relative phase difference is maintained. The ear is most sensitive to this relative phase difference (Banik & Bandyopadhyay, 2018). Figure 7 illustrates the method used at the receiver site.

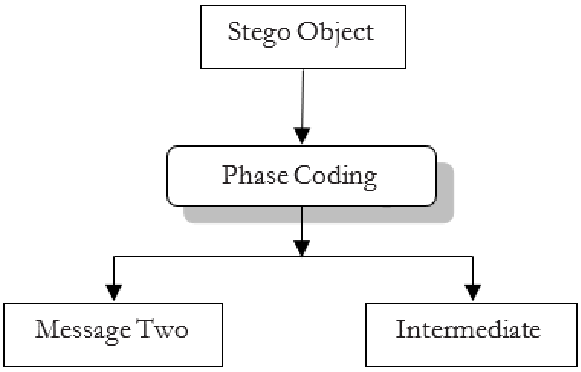


Figure 7. Second Layer Decoding Method at the Receiver Site

Results

Five experiments were conducted with an audio file of size of 188 kB, with a length of 4 s and a frequency of 2 kHz. In each experiment, the size of the secret message was varied to get the peak values of the signal-to-noise ratio (PSNR) and mean squared error (MSE), which were used to measure the quality of the stego audio in each experiment.

The square of the error between the cover audio signal and the stego audio signal is known as the mean squared error (MSE), as indicated in (2). The MSE can be used to calculate the degree of audio signal distortion (Ahmed et al., 2020).

$$MSE = \frac{\sum[f(i,j)-F(i,j)]^2}{N^2} \quad (2)$$

Where i and j are the number of rows and columns, and N is the number of samples in the input audio files. f(i, j) represents the cover audio signal, and F(i, j) represents the stego audio signal (Ahmed et al., 2020).

Peak signal-to-noise ratio, or PSNR, compares the cover and stego audio to determine the quality of an audio signal (Ahmed et al., 2020), as shown in (3). The PSNR is typically expressed in terms of the logarithmic decibel scale because many signals have a very wide dynamic range, which is defined as the ratio between the largest and smallest possible values of a changeable quantity:

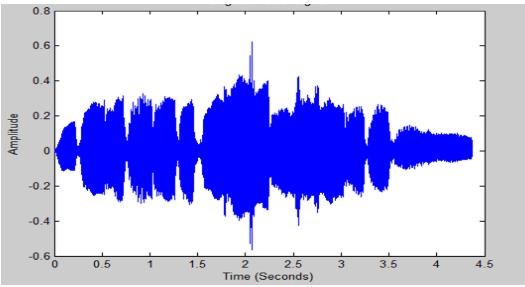
$$PSNR = 20log10\left(\frac{255}{MSE}\right) \quad (3)$$

Results of the First Experiment

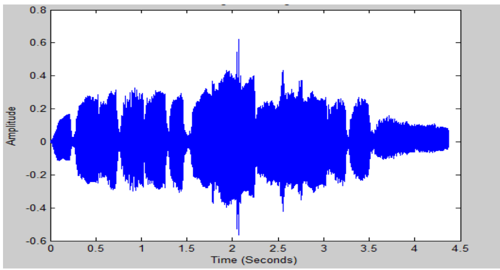
The results of the first experiment, where the secret message was divided in the ratio (1:1), are shown in Table 1. It can be seen that the best PSNR was obtained for a message size of 4 kB, and the worst for a message size of 256 bytes. Figure 8 (a) shows the original audio file, while (b) and (c) show the stego audio histograms for message sizes of 4 kB (best PSNR) and 256 bytes (worst PSNR).

Table 1. The flute's PSNR and MSE Measurements .wav Tone in the Initial Experiment, Dividing the Secret Message in a 1:1 Ratio

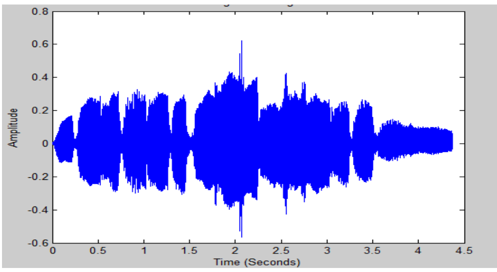
Audio size	Message size	SMS1	SMS2	PSNR	MSE	Notes
4 s	256 bytes	128 bytes	128 bytes	98.2816	0.000010	Worst
4 s	512 bytes	256	256	102.9424	0.000003	-
4 s	1 kB	512	512	108.8217	0.000001	-
4 s	2 kB	1 kB	1 kB	118.5449	0.000000	-
4 s	3 kB	6144	6144	122.3658	0.000000	
4 s	4 kB	2 kB	2 kB	128.7861	0.000000	Best
4 s	5 kB	2560	2.5 kB	126.0543	0.000000	
4 s	8 kB	-	-	-	-	Large message



(a)



(b)



(c)

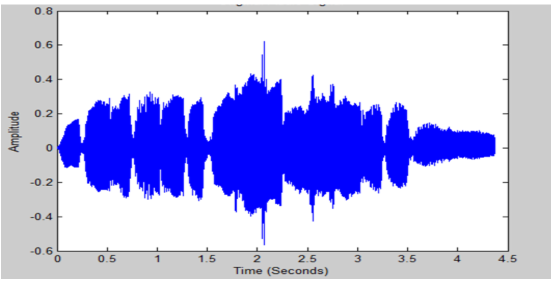
Figure 8. Flute.wav Audio File: (a) Original Audio File, and Results from the First Experiment, for Hidden Messages of (b) 256 bytes and (c) 4 KB

Results of the Second Experiment

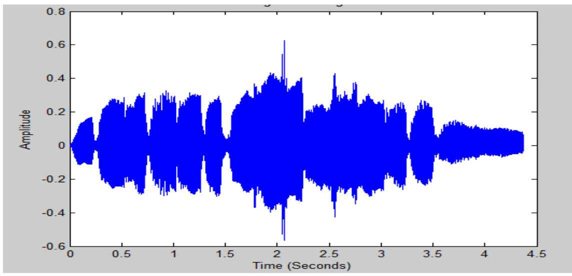
Table 2 shows the results of the second experiment, where the secret message was divided in the ratio (1:2). The best PSNR was obtained for a message size of 4 kB, and the worst for a message of size 256 bytes. Figure 9 (a) shows the original audio file, while (b) and (c) show the stego audio histograms for message sizes of 4 kB (best PSNR) and 256 bytes (worst PSNR).

Table 2. The Flute's PSNR and MSE Measurements. In the Second Experiment, a Wave Tone was Used, and the Secret Message was Divided in a Ratio of 1:2

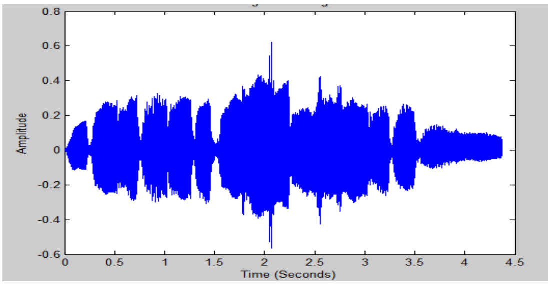
Audio size	Message size	SMS1	SMS2	PSNR	MSE	Notes
4 s	256 bytes	85 bytes	171 bytes	98.2816	0.000010	Worst
4 s	512 bytes	170	342	104.6864	0.000002	-
4 s	1 kB	341	683	112.1520	0.000000	-
4 s	2 kB	682	1366	117.5200	0.000000	-
4 s	3 kB	1 kB	2 kB	122.7473	0.000000	
4 s	4 kB	1365	2731	128.7481	0.000000	Best
4 s	5 kB	1706	3414	126.1244	0.000000	
4 s	8 kB			-	-	Large message



(a)



(b)



(c)

Figure 9. Flute.wav Audio file: (a) Original Audio File, and Results from the Second Experiment, for Hidden Messages of (b) 256 bytes and (c) 4 KB

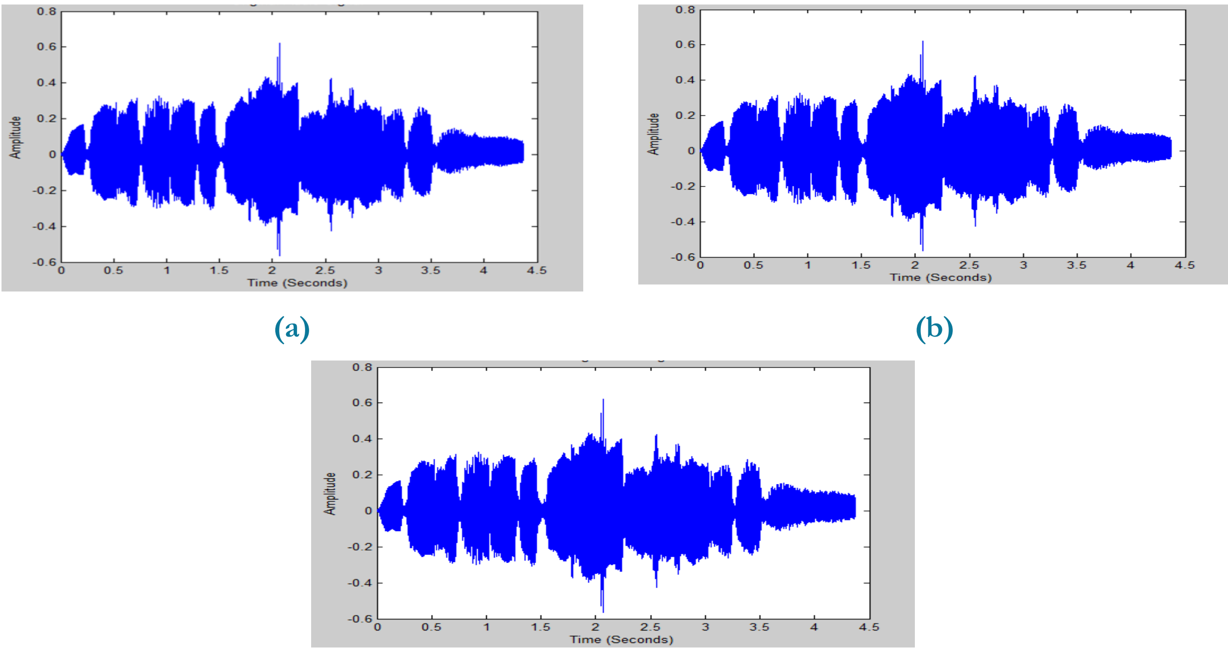
Results of the Third Experiment

The results for the third experiment, where the secret message was divided in the ratio (2:1), are shown in Table 3, the best PSNR was obtained for a message size of 4 kB, and the worst for a message size of 256 bytes. Figure 10 (a) shows the original audio file, while (b) and (c) show the stego audio histograms for message sizes of 4 kB (best PSNR) and 256 bytes (worst PSNR).

Table 3. The Flute's PSNR and MSE Measurements. In the Third Experiment, the Secret Message was Divided in a Ratio of 2:1 with a Wave Tone

Audio size	Message size	SMS1	SMS2	PSNR	MSE	Notes
4 s	256 bytes	170 bytes	86 bytes	95.9499	0.000017	Worst
4 s	512 bytes	340	172	101.9284	0.000004	-
4 s	1 kB	682	342	108.4033	0.000001	-
4 s	2 kB	1364	684	116.2635	0.000000	-

4 s	3 kB	2 kB	1 kB	118.4480	0.000000	-
4 s	4 kB	2730	1366	124.1015	0.000000	Best
4 s	5 kB	3412	1708	123.3313	0.000000	-
4 s	8 kB	-	-	-	-	Large message



(a) (b)

(c)

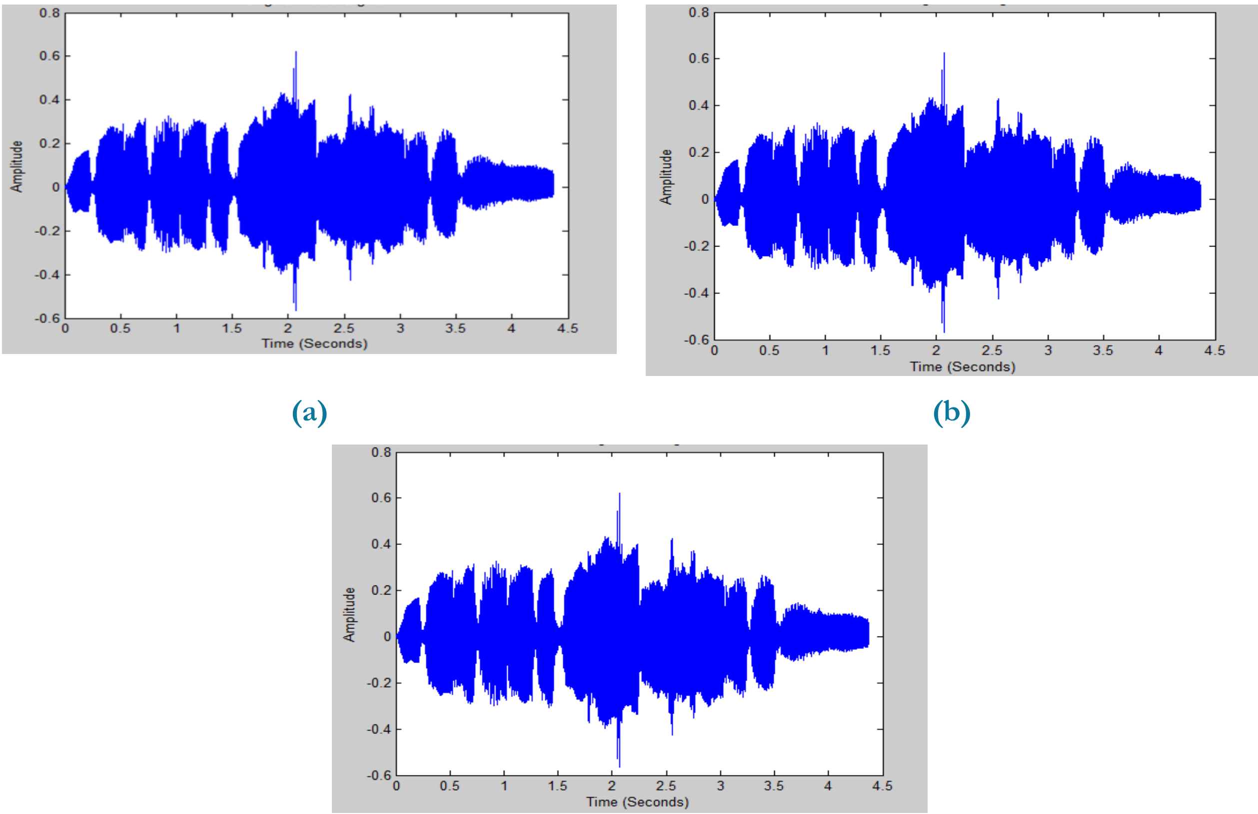
Figure 10. Flute.wav Audio File: (a) Original Audio File, and Results from the Third Experiment, for Hidden Messages of (b) 256 bytes and (c) 4 KB.

Results of the Fourth Experiment

Table 4 shows the results of the fourth experiment, where the secret message was divided in the ratio (1:3). It can be seen that the best PSNR was obtained for a message size of 5 kB, and the worst for a message size of 256 bytes. Figure 11 (a) shows the original audio file, while (b) and (c) show the stego audio histograms for message sizes of 5 kB (best PSNR) and 256 bytes (worst PSNR).

Table 4. The Flute's PSNR and MSE Measurements. In the Fourth Experiment, the Secret Message was Divided in a Ratio of 1:3, Using a Wave Tone

Audio size	Message size	SMS1	SMS2	PSNR	MSE	Notes
4 s	256 bytes	64 bytes	192 bytes	99.2838	0.000008	Worst
4 s	512 bytes	128	384	105.3374	0.000002	-
4 s	1 kB	256	768	111.2693	0.000000	-
4 s	2 kB	512	1536	119.2462	0.000000	-
4 s	3 kB	768	2304	123.2732	0.000000	
4 s	4 kB	1024	3072	126.1376	0.000000	-
4 s	5 kB	1280	3840	126.7631	0.000000	Best
4 s	8 kB			-	-	Large message



(a) **(b)** **(c)**

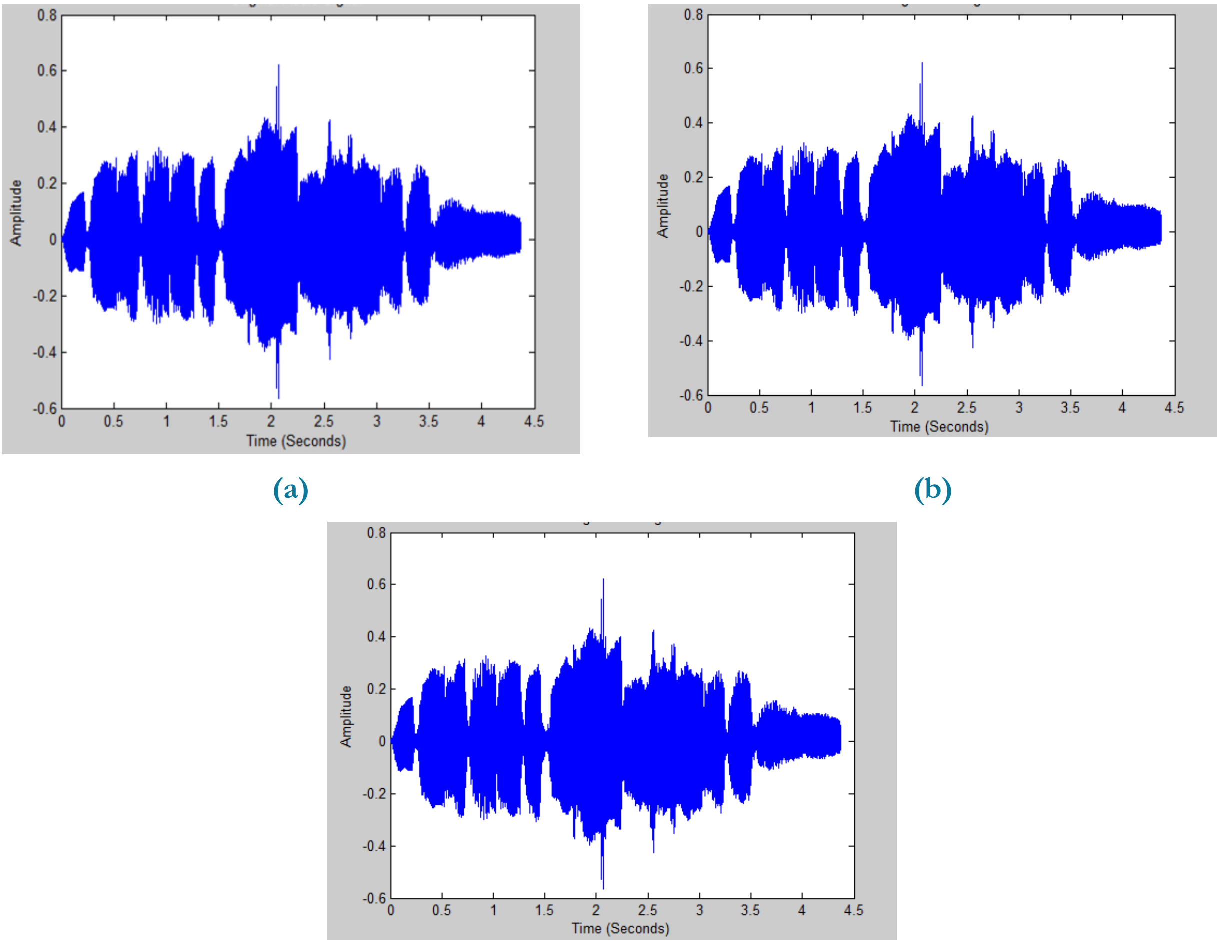
Figure 11. Flute.wav Audio File: (a) Original audio file, and results from the fourth experiment, for hidden messages of (b) 256 bytes and (c) 4 KB.

Results of the Fifth Experiment

The results for the fifth experiment, where the secret message was divided in the ratio (3:1), are shown in Table 5 below. It can be seen that the best PSNR was obtained for a message size of 4 kB, whereas the worst PSNR was found for a message size of 256 bytes. Figure 12 (a) shows the original audio file, while (b) and (c) show the stego audio histograms for message sizes of 4 kB (best PSNR) and 256 bytes (worst PSNR).

Table 5. The Flute's PSNR and MSE Measurements. In the Fifth Experiment, the Secret Message was Divided in the Ratio (3:1) Using a wav Tone

Audio size	Message size	SMS1	SMS2	PSNR	MSE	Notes
4 s	256 bytes	192 bytes	64 bytes	95.7244	0.000017	Worst
4 s	512 bytes	384	128	101.6952	0.000004	-
4 s	1 kB	768	256	105.8164	0.000002	-
4 s	2 kB	1536	512	113.6644	0.000000	-
4 s	3 kB	2304	768	119.5227	0.000000	
4 s	4 kB	3072	1024	123.0610	0.000000	Best
4 s	5 kB	3840	1280	122.2606	0.000000	
4 s	8 kB			-	-	Large message



(a) (b) (c)

Figure 12. Flute.wav Audio file: (a) Original Audio File, and Results from the Fifth Experiment, for Hidden Messages of (b) 256 bytes and (c) 4 KB

Discussion

We performed five experiments, and compared the values of the PSNR and MSE for each, which showed that the best values for PSNR and MSE were obtained by dividing the secret message in the ratio (1:1), i.e., in the first of our experiments. The worst values for PSNR and MSE were obtained when the secret message was divided in the ratio (3:1), i.e., in the fifth experiment.

The effectiveness of phase coding depends on the size of the segment, which increases with the message size. The best values for PSNR and MSE were obtained in the first experiment due to the use of phase coding. The effectiveness of this method depends on the message size, which in turn depends on the size of the segment. The opposite result was obtained in the fifth experiment, where the message size was smaller.

Based on the results of these experiments, we conclude that the effectiveness of the proposed method is affected by phase coding.

The security is increased by dividing the secret messages into two levels. Since the stego audio is not visibly distorted and the stego object's audio graph and original audio graph do not visibly differ from one another, a malevolent user watching a communication cannot discern that a message is present in the audio.

Figure 13 shows the results of all experiments, and Figure 14 shows the best and worst PSNR values for all experiments.

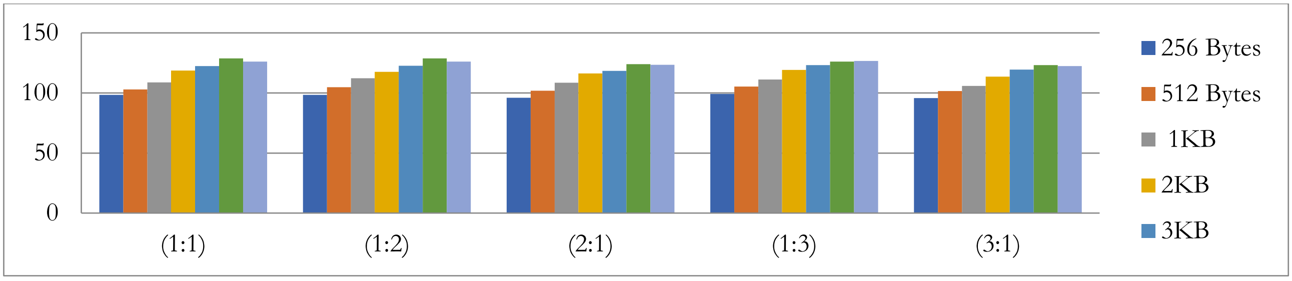


Figure 13. Results of All Experiments

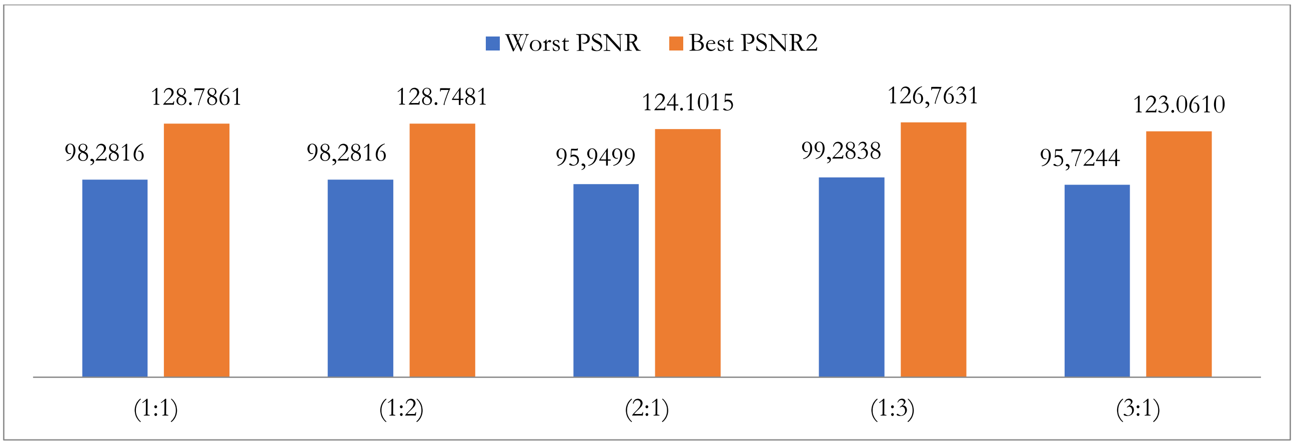


Figure 14. Best and Worst PSNR Values from All Experiments

Conclusion

Cryptography systems make the attackers suspected and give them motivations to direct efforts to crack systems. Audio can be used as a cover even in time domain or in frequency domain. In Phase Coding, Fourier transform is applied onto the audio divided into chunks, phase changes according to the secret message are applied onto the first chunk and the audio file is regained using the inverse Fourier transform.

In the techniques of audio steganography phase coding is a very effective and subtle way of encoding a message into the file without much distortions in the audio file since it only involves phase shifting.

In this paper, we tried to study the effect of dividing the message in the phase coding method on the effectiveness of the algorithm by conducting several experiments and then measuring the PSNR, MSE, and histogram metrics which used to compare the original and stego audio, in order to assess the effectiveness of the suggested approach. The comparative results showed that the effectiveness of the proposed method is affected by phase coding mechanism.

Conflict of Interests

No conflict of interest.

References

Ahmad, T., Amrizal, M. H., Wibisono, W., & Ijtihadie, R. M. (2020). Hiding data in audio files: A smoothing-based approach to improve the quality of the stego audio. *Heliyon*, *6*(3), e03464. https://doi.org/10.1016/j.heliyon.2020.e03464

Alsabhany, A.A., Ridzuan, F., & Azni, A.H. (2019). The Adaptive Multi-Level Phase Coding

Method in Audio Steganography. *IEEE Access, 7*, 291-306. https://doi.org/10.1109/ACCESS.2019.2940640

Al-Najjar A. J. (2008). The Decoy: Multi-Level Digital Multimedia Steganography Model, *12th WSEAS International Conference on Communications.* Computer Engineering Department, King Fahd University of Petroleum & Minerals, Saudi Arabia (pp. 445-450). https://doi.org/10.5555/1580987.1581065

Al-Othmani A. Z., Manaf A. A. & Zeki A. M. (2012). A survey on steganography techniques in real time audio signals and evaluation. *International Journal of Computer Science Issues*, *9*(1), 30-37. https://doi.org/10.14569/issn.2156-5570

Banik B. G. & Bandyopadhyay S. K. (2015). An image steganography method on edge detection using multiple LSB modification technique. *Journal of Basic and Applied Research International*, *9*(2), 75-80.

Banik B. G. & Bandyopadhyay S. K. (2018). Blind key-based attack resistant audio steganography using cocktail party effect, *Security and Communication Networks*, 1–21. https://doi.org/10.1155/2018/1781384

Bharti, S., Gupta, M. & Agarwal, S. (2019). A novel approach for audio steganography by processing of amplitudes and signs of secret audio separately. *Multimedia Tools and Applications, 78,* 179-201. https://doi.org/10.1007/s11042-019-7630-4

Chen S. T., Huang T. W., & Yang C. T. (2021). High-SNR steganography for digital audio signal in the wavelet domain, *Multimedia Tools and Applications,* *80*(6), 597–614, https://doi.org/10.1007/s11042-020-09980-6

kaur K. & Verma D. (2014). Multi-level steganographic algorithm for audio steganography using LSB, parity coding and phase coding technique. *International Journal of Advanced Research in Computer Science and Software Engineering*, *4*(1).

Kumar S. & Banik B. G. (2012). LSB modification and phase encoding technique of audio steganography revisited. *International Journal of Advanced Research in Computer and Communication Engineering,* 1(4). https://doi.org/10.17148/ijarcce

Li Y. and Kim J. (2011). Multiphase image segmentation using a phase-field model, *Computer & Mathematics with Application,* 62, 737-745. https://doi.org/10.1016/j.camwa.2011.05.054

Mahajan P. (2014). Steganography: A data hiding technique. *International Journal of Advanced Research in Computer Science and Software Engineering*, *4*(11), 759-763.

Mcloughlin I. V. (2016). *Speech and audio processing*, Cambridge University Press.

Mhatre D., R. Yadav, S. Jha, & A. Bhargava. (2018). The method of covering and hiding secret data by using various methods: Steganography. *International Journal of Engineering Research & Technology*, 6(12). https://doi.org/10.17577/IJERTCONV6IS12002

Milosav P., Milosavljević M., & Banjac Z. (2023). Steganographic method in selected areas of the stego-carrier in the spatial domain. *Symmetry*, *15*, 1015. https://doi.org/10.3390/sym15051015

Mohamad F. S., & Yasin N. S. (2013). Information hiding based on audio steganography using least significant bit. *International Journal of Engineering & Technology*, *7*, 536-538. https://doi.org/10.14419/IJET.V7I4.15.28363

Shanthakumari, R., Devi, E.M., Rajadevi, R., & Bharaneeshwar, B. (2021). Information Hiding in Audio Steganography using LSB Matching Revisited. *Journal of Physics: Conference Series, 1911.* https://doi.org/10.1088/1742-6596%2F1911%2F1%2F012027

Sheelu. (2013). Enhancement of data hiding capacity in audio steganography. *IOSR Journal of Computer Engineering,* *13*(3), 30-35. https://doi.org/10.9790/0661-1805025457

Singh A. K., Singh J., & H. V. Singh. (2015). Steganography in images using LSB technique. *International Journal of Latest Trends in Engineering and Technology*, *5*(1),426-430.

Xie C., Cheng Y., & Wu F. (2010). A new detection scheme for Echo Hiding. *2010 IEEE International Conference on Information Theory and Information Security*, Beijing. https://doi.org/10.1109/icitis.2010.5688776

Yousif S. A., Wahbi T. M., & Sayed M. H. (2017). Audio Steganography Using Tone Insertion Technique. *International Journal of Computer Applications Technology and Research*, *6*(6), 254-258. https://doi.org/10.7753/ijcatr0606.1004