Information Hiding in Audio Steganography using LSB Matching Revisited

R Shanthakumari¹, E M Roopa Devi², R Rajadevi³, B Bharaneeshwar⁴

¹Department of Information Technology, Kongu Engineering College, Tamilnadu
shanabiraki@gmail.com, roopadevi@kongu.ac.in, rajdevi@kongu.ac.in, bharanibala77@gmail.com

Abstract. Audio file steganography can be used as an effective and efficient method to hide messages. LSB based approach is the simplest steganographic technique among the various types for hiding secret information in audio formats. The approach is decided independent of the relationship between audio samples and the size of the secret messages. Efficiency of the output message is reduced when low data rate is used for hiding the message in lower power audio samples. The performance of data hiding in high power of audio is discussed in this paper. To identify the higher frequency, range from a given sample carrier and to hide the messages in the identified range, the LSBMR algorithm is used. The carrier wave is pre-processed in order to reduce the possibilities of intrusion detection. Experimentation of the adaptive steganography is performed on the files that are in .wav format. PSNR and MSE parameters are used for performance analysis. The threshold value of the audio carrier is depending upon the length of the message is used. The content of the secret information and threshold value of the carrier audio decides the number of samples for evaluation.

Index Terms—Data Embedding, Data Extraction, Embedding Unit, Steganography, Security, Segmentation.

1. Introduction
Audio Steganography participates in a cover (host) audio in an obscure manner in the screening activities of data. Applying the code similar (or, in most cases, the same as) to that used during the hiding stage private information of stego or of data-incorporated audio signals is recounted. Data integration is carried on audio content and multimedia data for conveying secret combat messages through a harmless audio cover. Other applications include the transmission of confidential information with security and information, like banking transactions and biometric data. Offered a misclassified channels and with help of safe audio signals, with no extra bandwidth is used in stego transmission, where encryption is widely used for the safe transmission of data. In addition, each encrypted messages suspects safe communication on unsecured channels. A chosen steganography method must meet the essential measures of stego imperceptibility, precise recuperation of information indeed in antagonistic circumstances and a solid key-base input and recuperation of data in arrange to realize the preferences of an clueless have carrying undercover information through a non-classified channel. Other specifications, such as high payloads and information retrieval, depend on applications without the original cover signal (obvious reconstruction). The stego should need a huge volume of data that can be identified with key and without the help of original host. Audio steganography is usually used for the psychoacoustic
concealing of the human sound-related since the imperceptibility in covert communication is paramount in the discernible discernment by the pirates of the data [1]-[3]. Cryptography [4-6] and Steganography [7-9] are the two techniques used for information security. Few essential qualities should be strengthened while designing a digital data hiding system, namely, Imperceptibility, Embedding capacity, Robustness and Undetectability. Imperceptibility means the veiling of primary message in covered object while the embedding capacity is the quantitative factor focused on the consignment of the primary message in the system. And robustness is pivoted on the degree of strenuousness to pull out the information from the covered one whereas the undetectability provides the possibility of venturing through various perspectives.

In the modern era, unauthorized access of information is expanding every day. Due to this, current requirement is secure information and this can be done using audio steganography technique. There are many audio steganography algorithms developed already. Using the insecure channel for transferring the data is a highly risk work for a vendor to carry out this. Most of the business vendors need two levels of security for achieve high level of security. Digital Images, particularly used files for steganography since they are the most widely used files available over the internet and web. Hiding confidential message in digital sound is more complicated when compare to hiding the content in image. For successfully embedding the secret message, LSBMR methods for concealing message in digital audio have been proposed. This paper consists of the following sections. Section 2 addresses the literature review. Section 3 contains the details of concealing the information and extracting techniques used in the implemented work. Section 4 shows the experimental outcome and discussion of the implemented system. And finally, Section 5 confers conclusion of the article.

2. LITERATURE REVIEW

To secure the message least significant part of the signal is used. The LSB algorithm is one of the popular audio steganography technique. This is a very handy way. In the final stego signal, the LSB bit modifications will not be speculated. A fairly large level of protection is not achieved as an easy technique. In the current LSB method, changes in pixels improve security. If the confidential data are not matched with the LSB cover, either the +1/-1 is arbitrarily added or detached from the covered signal [10], the LSB matching (also called +1, or -1 embedding) is the lesser adjustment of the LSB. Clearly, it is much more difficult to detect the matching LSB to avoid asymmetry in LSB.

In [11], Mielikainen had suggested that an improvement in LSB compliance can alter the cover image less while holding the payload. When the embedding scale is 1, the above scheme reduces the required number of corrections per pixel from 0.5 to 0.375 and also provides a great deal of resistance to steganalysis than LSB. The coding stage deals with the problem of noise-inducing audio steganography techniques. The coding of phases depends on the information that the phases element of sound is not as tangible as a noise to the human ear. The method covers the text bits as phase changes in the digital signal phase spectrum, which results in inaudible encoding for signal-to-perceived noise ratio, rather than providing the offer of anxieties. Phase code demerits are a minimum data transmission rate, since only the first signal segment encodes the secret message. Signal segment length should be increased. By increasing signal segment, it changes the relationships in each frequency part of the segment which makes easier for the message encoding. A part of this, only a few amount of data, like a technique as watermark, requires hiding is taken into account in phase coding. The fundamental spread-spectrum (SS) method, from the audio steganography aspect, aims to transmit secret messages as much as possible it make use of the audio signal - frequency spectrum. Above system make use of LSB coding to spread the information as a bit arbitrarily across the whole audio file. Although the SS method, unlike LSB coding, extend secret information over the audio file frequency spectrum, using an autonomous system of the unique signal. In the end, a bandwidth is used for transportation on the final signal [12].
To embed the data extended use the LSB from the fourth LSB position to the sixth LSB position with the least disturbances, Nedeljko and Tapio had introduced a technique. This technique is designed with a more precious way to embed the data and also the opponent cannot accurately identify the bit position wherever data is embedded. Considering the demerits of this technology, the development of the algorithm is very extra complex [13]. Wavelet domain [14] and Fourier domain [15] are used to conceal content in audio file. Samir approach [16] is the psychoacoustic resolution theory and the shifting of phases. The hearing resolution is made on the information that a pair of sounds consecutively hit our ears by less than a tenth of a second difference, so the difference between the sounds is unnoticeable.

Shanthakumari and Malliga [17] have presented used a new system for inserting the secret content with the LSBMR algorithm in the edge section of an image cover. The approach proposed by Ahmad and Mohammad to be the temporal domain [18]. This domain increases computer load and slows deployment. A multi-pixel differentiation technique using HL code had been proposed by Shanthakumari and Malliga [19] to build all blocks pixels into an embedded unit to conceal the secret bits by openly forecast the higher and lower blocks. The LSBI (Least Substantial Bit Inversion) algorithm was proposed by Shanthakumari et al. [20]. In this algorithm the secret information protection integrated with the gray code standard is sufficiently protected, which is in line with the protection layer formation in the image cover.

Beixing Deng proposed [21] a scheme to concurrently execute the concealing procedure while performing audio compressing of MP3. Modification of certain quantized spectrum values of third audio file to hide the confidential information into the audio is done and providing six-time larger capacity compared with the mp3 stego are implemented in this technology. Higher invisibility between original sound file and sound file with confident message is an important character of this system. The empirical results of this method revealed that the hiding capacity of characters is more at inside audios and extracted perfectly with confidential information that is undetectable to human ears. [22] This approach is used to embed message, and extract secret message into an usual image file using the International Data Encryption algorithm and the Minor Significant Bit Grouping algorithm. [23]. To perform the encryption for the hidden and encrypted information embedded into the covers of object by the LSB Inversion algorithm process, this approach uses the Cryptography elliptical curve algorithm.

Alaalsmat Al-Attili [24] had intended a method that engages the space between frames of the mp3 file for embedding the secret information. CBR type must be the MP3, is one of the deficiency of this method. But yet, offers steganography capabilities and complexity and has proposed that hiding method is robust against noise. The system is especially secure during passive attacks as the information is scrambled utilizing the RSA algorithm before information is inserted. In [25] WeiqiLuo used this LSBMR algorithm for hiding data in edges. LSB Matching Revisited (LSBMR) is the algorithm used for image, which uses a couple of pixels as an embedding unit, where one bit of mystery message carried by the LSB of the starting pixel and another bit of mystery is carried by the average of first and second pixel values (odd-even combination). This way, the proposed method uses this LSBMR algorithm to hide data in high power audios. Even though the way of successful usage of the information inserting in sound is simple, a result of few studies mentions that some inconveniences namely, choosing the region for data hiding and imperceptible payload had been crucial factors that were considered to get further improvement. But there would be sufficient improvement in the choosing the region for data hiding, payload and the proposed scheme of audio with LSB Matching Revisited algorithm to minimize the distortion between cover and stego audios. This proposed scheme can resolve all the issues. Our proposed method thereby uses this LSBMR algorithm to hide data in high power audio.
3. PROPOSED METHOD

Normally assume that the LSB of natural covers is unimportant and random enough in most Steganographic strategies, and thus regions taken for data hiding are selected employing a Pseudo-Random Number Generator. Such a hypothesis is not ever true, particularly for audio with several low power sample values. Usually, the sample values positioned at the high power audio offers major confused factual features and are profoundly subordinate on the audio file. Besides, it is major difficult to recognize discrepancies at higher power of audio than in low power audio samples. In this paper, an audio with the LSBMR algorithm is an efficient adaptive scheme is analysed. The proposed Data Embedding and Data Extraction flow diagrams are depicted in figure 1 and figure 2.

![Data Embedding Diagram](image1)

**Figure 1. Data Embedding**

On the data embedding stage, first cover audio is segmented into clips of samples and the samples are selected randomly which depends on the length of audio samples. First initialize the parameters, that are used for subsequent sample selection and data pre-processing. As per the shared key K, the segment of each individual clips are phase shifted and are combined. Later, estimate the

![Data Extraction Diagram](image2)

**Figure 2. Data Extraction**
quantity of the selected samples and if the samples are large enough for hiding, the given text message using LSBMR, then data hiding is done on the chosen samples. Else, alteration the parameters, and after that, the process of capacity estimation is repeated and sample selection until the message content can be inserted totally. Here, the mystery message is covered up in the interior of the audio along with the parameters which are different for different audio files and text messages. We require these parameters as side data to ensure the legitimacy of information extraction. High power of audio signal, represented by means of high sample value is taken as parameters for sample selection and LSBMR is used as the data hiding algorithm. In the cover audio, a sample pair (ai, bi) is an embedding unit. The unit is modified as (a'i, b'i) in the stego audio which satisfies, \( \text{LSB}(a'i) = \text{mi}, \text{LSB}((a'i/2) + b'i) = \text{mi} + 1 \) by LSBMR. Here the function of \( \text{LSB}(f) \) denotes the LSB of the sample values is done after embedding the message. The two secret bits are \( \text{mi} \) and \( \text{mi} + 1 \). The resulting audio is divided into non-overlapping \( \text{Cz} \) clips after data hiding. The clips are shifted by a random number of degrees, based on \( K \). This process is very similar to preprocess other than rearrangements being the opposite. Now the phase shifted and segmented clips are combined to generate the stego audio.

To get the stego audio a little post-processing is done in data processing. This method extracts the data, clip size and threshold from the stego audio as parameters. Based on the side data, it at that point does the pre-processing to identify the samples used for data hiding according to the corresponding extraction algorithm. In data extraction, first, it creates a traveling order by PRNG with the help of a shared key. At that point, for each of the inserting unit, two bits can be extracted. The first secret bit can be obtained by the LSB of the first sample, and by calculating the relationship between the two samples the second bit can be obtained. The proposed audio steganography mechanism consists of two stages. The first stage perform embeds the messages in the cover audio whereas the afterward stage extracts the covered up data from the stego audio.

3.1 Data Embedding

The Algorithm for embedding message in cover audio is given below: -

3.2 Data Embedding

Step 1: Parameter Initialization

The cover audio is first divided into non-overlapping clips of samples \( \text{Cz} \) with the Clip size being randomly selected in a range between 1 to length of audio samples.

Step 2: Preprocess stage which includes segmentation, phase shift and combine clips.

Step 2.1: Segmentation

As determined by the samples \( \text{Cz} \), the audio file is segmented into clips. For example, when the clip size is segmented as 4, 7 or 100, the resultant clips are shown in figure 3(a) to figure 3(d).
Step 2.2: Phase shift

After segmentation, (Eg: The clip size with segment 6 was taken) in a pseudorandom order as per the shared key K, each of the individual small clips is phase shifted. (look into figure 4(a) and figure 4(b)). The advantage gained via the random phase shift is that it blocked from getting the exact implanting units outward from phase shift key, and hence level of security is expanded. Also, the operation of brute force attack gets complicated when the phase shifts further.

Figure 4(a). Audio clip before applying phase shift.

Figure 4(b). Audio clip after applying phase shift $\{90^\circ, 120^\circ, 180^\circ, 280^\circ, 230^\circ, 360^\circ\}$.

Step 2.3: Combine clips

The segmented clips are now combined and figure 5 shows the resultant audio file after performing preprocess.

Figure 5. Audio clip after performing preprocess.
Step 3: Sample Selection

Two secret bits can be inserted into every embedding section according to LSBMR method. Let us consider a message (m), the sample selection can be determined by the threshold T as mentioned below. Let ES (T) be the set of sample pairs with absolute values equal to or greater than parameter T

\[ ES (T) = \{(ai, bi) \mid |ai - bi| \geq T, \text{for all } (ai, bi) \in V\} \]

Step 4: Capacity Estimation

The threshold estimation is calculated by

\[ T = \max \{2 \times |ES| \geq |E(m)|\} \]

Where T \( \in \{1\text{ to } -1\} \), E (m) is the message size. The embedding section is denoted by ES. For example: When threshold 0.7532 is applied, embedding units as shown in figure 6. are considered for the samples in audio clips satisfying ‘T’ alone.

Figure 6. Threshold is applied for audio samples.

Step 5: Data Hiding

Data hiding is performed on the set ES (T) = \{(ai, bi) \mid |ai - bi| \geq T, \text{for all } (ai, bi) \in V\}. The above-implanting units are processed in a pseudorandom order that is decided by a mystery key K for each unit, (ai, bi). These sample pairs are converted to binary. Information covering is performed according to the taking after four cases.

Case 1: LSB (ai) = mi & LSB (f (ai, bi)) = mi+1 then

\[ (ai', bi') = (ai, bi) \]

Case 2: LSB (ai) =mi & LSB (f (ai, bi)) ≠ mi+1 then

\[ (ai', bi') = (ai, bi + r), \text{ where } r = \pm 0.001. \]

Case 3: LSB (ai) ≠ mi & LSB (f (ai-1, bi)) = mi+1 then

\[ (ai', bi') = (ai-1, bi+1) \]

Case 4: LSB (ai) ≠ mi & LSB (f (ai-1, bi)) ≠ mi+1 then

\[ (ai',bi+1') = (ai+1, bi+1) \]

Where mi and mi+1 denote two secret bits to be embedded, the function ‘f’ is defined as f (a, b) = (a/2) +b. r is a random value in \{-0.001, +0.001\} and (ai’, bi’) denotes the pixel pair after data hiding. After the over alterations ai’ and bi’ may be less than the threshold T. In such cases, we need to readjust them as (ai”, bi”) by ai” = ai’+A1 and bi” = bi’+A2. A1 and A2 possess the value of either 0.001 or 0.002. Lastly, LSB (ai”) = mi & LSB (f (ai/2’, bi’)) = mi+1

Step 6: Post Process

After data hiding, the non-overlapping Cz clips were divided from the resulting audio, which were then shifted by random number of degrees depending on K. Post process is identical to
preprocess except that the rearrangements made are reverse. Now, the stego audio is generated after the phase shifted and segmented clips are combined and is then sent to the receiver. This can be shown in figure 7(a) -7(b) respectively.

![Audio File after Phase Shift](image1)

![Plot of Audio file](image2)

**Figure 7 (a).** After Data Hiding the Resulting Audio  
**Figure 7(b).** Stego Audio.

### 3.3 Data Extraction:

Using the decoding procedure received from the sender, the same sequences of embedding process are followed in the reverse direction in order to extract the private content from the stego audio. The following is the detailed data retrieving algorithm:

**Step 1: Parameter Extraction**

To extract the data, clip size $C_z$ and the threshold are first extracted from the stego audio which is same as data embedding in Step 1.

**Step 2: Preprocess**

The stego audio is segmented into $C_z$ clips and are shifted by arbitrary degrees based on the hidden key $K$, at that point, the resultant clips are united.

**Step 3: Sample Identification**

All the covered up bits are extricated completely, according to a pseudorandom grouping based on the mystery key. The progress the embedding units which value are equal to or higher than the threshold.

**Step 4: Data Extraction**

For each qualified implanting unit, say, $(a'i, b'i)$, where $|a'i - b'i| \geq t$, the two mystery bits $m_i$ and $m_{i+1}$ are extricated as: $m_i = \text{LSB} (a'i)$ & $m_{i+1} = \text{LSB} ((a'i/2) + b'i)$

**Step 5: Original Message Extraction**

Convert the resultant bits into ASCII values which is then converted to characters. At last, original message is obtained.

### 4. Results and Discussions

MATLAB 7.0 is used for the proposed work. The intention of the work is to examine the execution of the audio steganographic algorithm based on LSBMR algorithm. We used this algorithm for audio samples though LSBMR algorithm is applied for image pixels generally. On various .wav audio file, our proposed method has been executed and the performance of the plan has been calculated and examined on the basis of two parameters namely, Mean Squared Error (MSE) and Peak Signal to Noise Ratio (PSNR) where both are computed as follows
- **Mean Squared Error (MSE):**

  MSE dealt with audio signal distortion. It is defined as squared error between stego audio signal and the original audio signal is given in Equation

  \[ MSE = \frac{1}{N} \sum_{n=0}^{N}(x(n) - y(n))^2 \quad (1) \]

  \[ \sum_{n=0}^{N}(x(n) - y(n))^2 \sum_{n=0}^{N}(x(n) - y(n))^2 \]

  Here \( y(n) \) represents stego audio file and \( x(n) \) represents cover audio file.

- **Peak Signal to Noise Ratio (PSNR):**

  PSNR dealt with audio signal quality. PSNR compares the original audio signal with stego signal. The formula for PSNR is given in Equation 2.

  \[ PSNR = 10 \log_{10}\frac{\sum_{n=0}^{N}x(n)^2}{\sum_{n=0}^{N}(x(n) - y(n))^2} \quad (2) \]

  These measures are calculated for ‘road.wav’ audio file by varying threshold and phases. Here the results are shown for the ‘road.wav’ audio file which was segmented into 6 clips and applied with phases of \{45°, 90°, 120°, 180°, 270° and 360°\} and the corresponding results of MSE and PSNR are shown in Table 1.

### Table 1. MSE and PSNR values.

| Threshold | Embedded Data (in bits) | MSE          | PSNR         |
|-----------|-------------------------|--------------|--------------|
| 0.5920    | 30                      | 1.33483e-13  | 95.754       |
| 0.5810    | 50                      | 5.04106e-13  | 93.497       |
| 0.5632    | 336                     | 2.13805e-12  | 90.348       |
| 0.5192    | 550                     | 5.11926e-12  | 89.567       |
| 0.5291    | 1092                    | 1.0444e-11   | 88.358       |
| 0.5000    | 2184                    | 2.03394e-11  | 87.222       |
| 0.4930    | 4914                    | 4.59119e-11  | 85.845       |
| 0.4930    | 7630                    | 7.2314e-11   | 80.749       |
| 0.4875    | 11998                   | 1.11838e-10  | 84.333       |
| 0.4500    | 17458                   | 1.6435e-10   | 83.682       |
| 0.3999    | 28658                   | 2.71998e-10  | 82.827       |

From the Table 1, the threshold value is taken from higher to lower, it is proved that the MSE esteem drop between the ranges 0 to 1. The lower estimation of MSE signifies lower mistake. In the proposed algorithm, the PSNR values range from 82.8 dB to 95.75dB. Further, the MSE is inversely proportional to PSNR, and an increase in MSE leads to the decrease in PSNR value and vice versa.

From the above Table 2, note that capacity estimate is high while the threshold estimate is low when 79,362 characters are embedded inside the audio. By the end, it is noticed that if the threshold value is higher to implant as it were the least set of characters. While the threshold value diminishes, the number of characters implanted is more. This is illustrated in Table 2. Thus, the embedding unit we can adjust as per the requirement of data size with the threshold. This is graphically shown in figure 8.

### Table 2. Characters Embedded Based on Threshold.

| S. No | Threshold | No. of characters that can be embedded |
|-------|-----------|--------------------------------------|
| 1     | 0.5920    | 119                                  |
| 2     | 0.5810    | 245                                  |
| 3     | 0.5632    | 366                                  |
| 4     | 0.5291    | 987                                  |
| 5     | 0.5000    | 3526                                 |
| 6     | 0.4930    | 4857                                 |
| 7     | 0.4875    | 6201                                 |
| 8     | 0.4500    | 24177                                |
| 9     | 0.4132    | 60630                                |
| 10    | 0.3999    | 79362                                |
This figure 8 shows the effect of threshold on characters. When threshold is 0.3999, maximum number of 79,362 characters is embedded and if the threshold is 0.5920, less number of 119 characters is embedded. Table 3 and 4 demonstrate the result of embedding a disparate size of secret message in the different cover audio used to assess the proposed method. The performance measures are also calculated for various .wav audios by varying threshold values and phase angles. It is shown in Table 3 and 4. Here, all the .wav audio files are segmented into 3 clips and are given phase rotations of \(\{45^0, 90^0, 120^0\}\) respectively. The resultant values of MSE and PSNR obtained after this process are given in Table 3 and 4.

**Figure 8.** Characters Embedded Based on Threshold.

### Table 3. MSE of Various Audio Files.

| Data size in characters | Sample Audio (.wav) File | Goat- bleet  | Flute  | Tenor sax | Muted Trumpet |
|-------------------------|--------------------------|-------------|--------|-----------|---------------|
|                         |                          | MSE         |        |           |               |
| 100                     | 0.002                    | 0.003       | 0.005  | 0.002     |               |
| 250                     | 0.005                    | 0.009       | 0.006  | 0.004     |               |
| 500                     | 0.009                    | 0.001       | 0.007  | 0.008     |               |
| 1000                    | NA                       | 0.002       | 0.008  | 0.015     |               |
| 1500                    | NA                       | 0.004       | 0.009  | 0.015     |               |
| 2000                    | NA                       | 0.004       | 0.026  | 0.032     |               |
| 3000                    | NA                       | 0.004       | 0.047  | 0.045     |               |
| 4000                    | NA                       | 0.010       | 0.054  | 0.063     |               |

*NA- Not applicable

### Table 4. PSNR of Various Audio Files.

| Data size in characters | Sample Audio (.wav) File | Goat- bleet  | Flute  | Tenor sax | Muted Trumpet |
|-------------------------|--------------------------|-------------|--------|-----------|---------------|
|                         |                          | PSNR (dB)   |        |           |               |
| 100                     | 100.7                    | 125.8       | 134.4  | 132.6     |               |
| 250                     | 96.8                     | 118.4       | 130.7  | 128.8     |               |
| 500                     | 94.02                    | 115.4       | 127.4  | 125.7     |               |
| 1000                    | NA                       | 113.6       | 124.8  | 122.7     |               |
| 1500                    | NA                       | 112.4       | 122.1  | 119.8     |               |
| 2000                    | NA                       | 110.1       | 121.8  | 118.2     |               |
| 3000                    | NA                       | 110.6       | 119.3  | 116.6     |               |
| 4000                    | NA                       | 109.3       | 118.7  | 112.8     |               |

*NA- Not applicable
From the above table, note that the PSNR estimate is low while the MSE estimate is high when the range from 100 to 4000 characters are embedded inside audios. PSNR results are still low, as well as the MSE value is still high, meaning that the original audios are closer to the stego audios. By comparing, PSNR values against the embedding capacity in the figure 9. From figure 9, it is inferred that the PSNR value decreases as the data rate increases.

![Figure 9. PSNR of Different Audio Files.](image)

### 4.1 Threshold Vs Data Rate

Table 5 demonstrate the result of varying the threshold, embedding a disparate size of secret message in the different cover audio used to assess the proposed method. By varying threshold values, the performance measures are also calculated for various .wav audios.

| Audio (.wav) | T=0.0832 | T=0.7932 | T=0.7732 | T=0.7432 |
|--------------|----------|----------|----------|----------|
| Goat Bleet   | 300      | 600      | 1000     | 1367     |
| Flute        | 250      | 4700     | 6690     | 12375    |
| Tenorsax     | 8770     | 13850    | 18500    | 21800    |
| Muted-Trumpet| 4260     | 7050     | 9600     | 24190    |

From the above table, threshold value of different audio files in which the proposed strategy gives large capacity with superior sound quality. For “Muted-Trumpet” with the threshold $T=0.7432$, the payload obtained is 24,190 characters. Thus, we can adjust the embedding unit with threshold as per the requirement of data size. This is graphically shown in figure 10. From the below figure 10, it is inferred that if the threshold value is minimum, the number of characters implanted is more. In case the limit is higher to insert as it were the least set of characters.

![Figure 10. Threshold Vs Data Rate on various Audio File.](image)
4.2 Spectrogram

Spectrogram of ‘road.wav’ original audio and stego audio are shown in figure 11 and 12 respectively. It is observed that the stego audio and the host audio signal cannot be perceived differently. However, in practice some of the LSB of the received stego may not remain same because of applying LSBMR.

5. CONCLUSION

In this paper, audio steganography scheme based on LSB matching revisited algorithm is applied to embed concealed information into audio without inventing perceptible distortions. By predicting the threshold value, the concept of LSBMR algorithm is utilized to make selected samples as an inserting unit for covering up the mystery messages. LSBMR algorithm is applied for the samples of high power audio based on the threshold of the required samples considered for modification. The performance of audio steganography for various .wav audio files are analysed. In addition to that, the data hiding safeguard is increased and then segmentation and phase shift are performed. Algorithms are applied in the sequence of pre-process, which, then data hiding and followed by post process increases the security. The quality of audio can be calculated by Peak Signal to Noise Ratio (PSNR) and Mean Squared Error (MSE). When threshold is minimum, more characters are embedded or else if the threshold is maximum, less number of characters are embedded. The future work can be extended to other covers such as video.

REFERENCES

[1] Bender W., Gruhl D., Morimoto N. and Lu A., “Techniques for data hiding”, IBM Systems Journal, Vol. 35, Issue 3-4, (1996): 313-336.
[2] Ross J. Anderson, Fabien A.P. Petitcolas, “On the limits of steganography”, IEEE Journal of Selected Areas in Communications, 16.4 (1998): 474-481.
[3] Swanson M.D., Kobayashi M. And Tewfik A.H, “Multimedia data embedding and watermarking technologies”, Proc. IEEE, Vol. 86, (1998): 1064- 1087.
[4] BruceSchneier, Applied Cryptography Protocols, Algorithm and Source Code in C, Second edition. Wiley India edition, 2007.
[5] W. Diffie and M. E. Hellman, “Exhaustive Cryptanalysis of the NBS Data Encryption Standard”, IEEE Journal on Computer Applications, Vol. 21, No. 20, (1977): 74-84.
[6] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems”, Communications of the ACM, Vol. 21, No.2, (1978): 120–126. doi:10.1145/357980.358017
[7] Abbas Cheaidid, Joan Cordell, Kevin Curran, PaulMeKevitt, Digital image steganography: Survey and analysis of current method signal Processing, 2010, 727–752.
[8] R. Amirtharajan, R. John BoscoBalaguru, “Constructive Role of SFC & RGB Fusion versus Destructive Intrusion”, International Journal of Computer Applications, Volume 1 – No. 20. ISSN: 0975 – 8887, 34-40.
[9] Fatih Djebbar, Baghdad Ayad., Karim Abed Meraim., Habib Hamam, “Comparative study of digital audio steganography techniques”, EURASIP Journal on Audio, Speech, and Music Processing, December 2012.
[10] Sharp T., “An implementation of key-based digital signal steganography”, in Proc. 4th International Workshop on Information Hiding, Vol.2137, Springer LNCS, (2001): 13–26.
[11] Miilikainen, J., “LSB matching revisited”, IEEE Signal Processing Letters, Vol.13, No. 5, (2006): 285-287.
[12] PoulamiDutta, 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.
[13] Nedeljko C. and Tapio S.A., “Increasing robustness of LSB audio steganography by reduced distortion LSB coding”, Journal on
Universal Computer Science 11.1:2005: 56-65.

[14] N. TaraghiDelgarm, Speech Watermarking, M.Sc. Thesis, Computer Engineering Department, Sharif University of Technology, Tehran, IRAN, May 2006.

[15] L. Gang, A.N. Akansu, and M. Ramkumar, “MP3 resistant oblivious steganography”, Proceedings of 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP’01), Vol.3, pp. 1365-1368, May 2001.

[16] Samir, K.B. Tuinh, U.P. and Ra., Avibhek, “A robust audio steganographic technique based on phase shifting and psycho-acoustic persistence of human hearing ability”, Int. J. Comput. Corporate Res., 1.1(2011).

[17] Shanthakumari R and Malliga S, “Data hiding in image using tree based parity check with LSB matching revisited algorithm”, International Journal of Innovative Research in Computer and Communication Engineering 3.6 (2015): 5472–5479.

[18] Ahmad, D. and F. Mohammad, “Adaptive and Efficient Audio Data Hiding Method in Temporal Domain”, International Conference on Information and Communication Systems, 2009. doi:10.1109/ICICS.2009.5397486

[19] Shanthakumari R and Malliga S, “Information hiding in digital images using modified LSB substitution with multipixel differencing and HL code”, Asian Journal of Research in Social Sciences and Humanities 7.1 (2017): 198–207.

[20] Shanthakumari R, Malliga S and Dheepika S, “Data hiding scheme in spatial domain”, International Journal of Computer Science Engineering and Technology 4.12 (2014): 400–403.

[21] Beixing Deng, Jie Tan, Bo Yang, Xing Li, “A Novel Steganography Method Based on Modifying Quantized Spectrum Values of MPEG/Audio Layer III”, Proceedings of the 7th WSEAS International Conference on Applied Informatics and Communications, Athens, Greece, Vol.7. (2007): 325-330.

[22] Shanthakumari R and Malliga S, “Dual-layer security of image steganography based on IDEA and LSBG algorithm in the cloud environment”, 44.5 (2019):1-12.

[23] Shanthakumari, R and Malliga, S “Dual layer security of data using LSB inversion image steganography with elliptic curve cryptography encryption algorithm”, International Journal of Multimedia Tools and Applications, 79.5-6 (2020): 3975-3991.

[24] AlaaAlmat Al-Attili, OsamahAbdulgader Al- Rababah, “New technique for hiding data in audio file”, International Journal of Computer Science and Network Security, 10.7 (2010).

[25] Weiqi Luo, fangjun Huang, Jiwu Huang, “Edge Adaptive Image Steganography Based on LSB Matching Revisited”, IEEE transaction on Information forensics and security, 5.2 (2010).