Received: January 6, 2022. Revised: February 28, 2022.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Minimizing Sample Space to Optimize Quality of Stego-Audio

Mohammad Muzayyin Amrulloh[1] **Tohari Ahmad**[1]*

[1]*Department of Informatics, Institut Teknologi Sepuluh Nopember, Indonesia*
* Corresponding author's Email: tohari@if.its.ac.id

Abstract: The rapid progress of information and communication technology forces all application aspects to be developed securely, considering that data are often transmitted over public networks. If the security is not well applied, the data can be easily compromised during transmission. So, a data hiding scheme is essential to protect confidential data. Some methods have been introduced; however, the data size and the resulted data quality are still challenging. This research works on these problems by employing audio as the cover to carry confidential data and focuses on analyzing the audio sample space accommodating those data. Here, we propose a method to optimize stego-audio quality by reducing the sample space capacity to produce better audio quality. The experimental result shows that the stego-audio quality evaluated using Peak Signal-to-Noise Ratio (PSNR) goes up by 31.3 dB without additional steps such as smoothing or payload distribution, which is simpler than the previous schemes.

Keywords: Information security, Network security, Network infrastructure, Information hiding, Data hiding.

1. Introduction

Information technology has quickly developed and implemented widely in various fields such as finance, education, and health. Its application has made it easier for users to carry out their activities. Nevertheless, it is possible that confidential data to be the target of an attacker, making them vulnerable during transmission. Therefore, a secure mechanism is essential in protecting confidential data or information privacy and security. Consequently, we must secure data by using security techniques and making it unreadable [1]; one approach is developing and implementing a data hiding or steganographic method [2].

Data hiding is an ancient art and science to embed a secret message (payload) into the media (cover) [3], whose concept is depicted in Fig. 1 [4]. It requires a medium to carry the data like image, text, audio, and video [5], as illustrated in Fig. 2, to generate a stego file containing those confidential data. This stego and the original carrier must have a high similarity level so that the public cannot identify and analyze the existence of any confidential message or data, and not raise suspicions [6, 7].

Despite its advantages, the existing data hiding methods still have challenging problems to solve, mainly are the quality of the stego data after embedding the specified secret data size into the cover file. Those two issues are usually known as the quality aspect represented by peak signal-to-noise ratio (PSNR) and the capacity aspect represented by

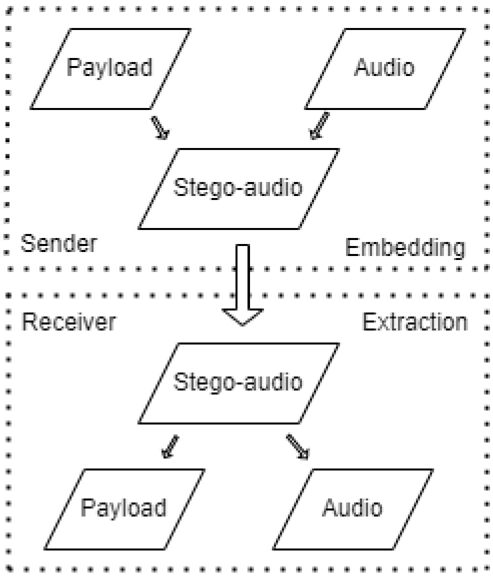

Figure. 1 Data hiding scheme

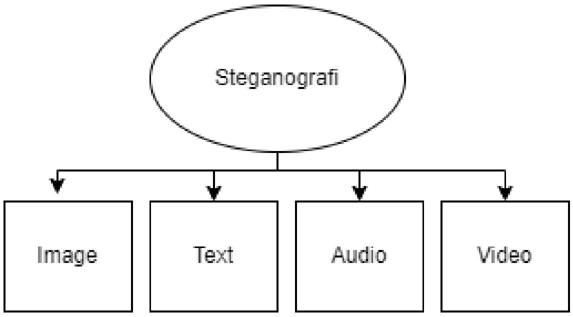

Figure 2 Types of carriers

bits. So, the research intends to obtain as high as a possible similarity between the stego dan the cover files after embedding more private data (payload). Furthermore, another problem that may arise is whether the original cover can be extracted entirely from the stego file. Although it is not compulsory, this characteristic may be required in some applications.

This research takes an audio file as the cover. Previously, Benhfid [8] proposed a reversible data hiding method based on interpolation. Besides improving the quality of stego-audio, that method can extract the payload fully. Another study [9] proposed a multiple smoothing method to get better stego-audio quality. Similarly, other research [4] proposes distributing payload evenly, which can improve the quality of stego-audio without having to do smoothing several times.

Based on these studies, we find that it is still possible to increase the method's performance by focusing on finding an appropriate audio embedding space. Therefore, we propose a method to improve the quality of stego-audio produced by several previous studies considering the results of PSNR calculations by reducing the payload embedding space. It is intended to obtain a relatively high PSNR value, which means that the stego-audio file is not much different from the corresponding original audio samples.

The structure of this paper is as follows. Related works are in section 2, which describes several previous studies related to this proposed method. section 3 explains the exact scheme, followed by the experimental results in section 4. Finally, the conclusion is in section 5.

2. Related works

This proposed research takes audio as the medium since it likely provides more embedding capacity than an image that can be explored further [10]. Moreover, not much research has explored it. The basis of reference used in this study can be explained as follows.

Based on its capability of recovering the original medium, data hiding can be grouped into two categories: reversible and irreversible [11]. The reversible data hiding can recover both the private data and the medium completely. On the contrary, irreversible data hiding focuses on successfully retrieving only the secret data [12].

Several data hiding methods are often taken, such as the interpolation method on audio samples [4]. Besides restoring the entire payload, this method can also obtain the audio cover properly. Therefore, it is convenient in the extraction process. Some studies, like [8], have increased the sample space capacity to maximize the embedded payload. However, it has a weakness: the stego-image quality and imperceptibility tend to drop; so, it is necessary to design additional methods to maintain the quality. For example, implementing the smoothing method [9] or the payload distribution [13].

In the previous study, Al-Ahmad [9] proposed a method of hiding data implementing several smoothing steps. This study can generate a good quality stego-audio. It is because its smoothing stage can reduce the difference between the cover audio and the resulting stego-audio well [4]. However, the drawback of the smoothing method is that it tends to use fewer samples for insertion, forcing the embedding process to take a large payload capacity on audio samples. Meanwhile, Pei et al. [13] proposed a method of hiding data using an even distribution of payload to obtain better stego-audio quality without multiple smoothing stages. The drawback of the evenly distributed method is that it does not consider the available sample space. Thus, forcing each sample to accommodate the number of distributed payloads.

3. Proposed method

Similar to other schemes, the application consists of 2 processes: the embedding and the extraction.

3.1 Embedding

This first process is to embed the payload into the audio sample of the cover, as provided in Fig. 3. The detailed steps are as follows.

1. First, the audio file of type .wav is normalized by adding all samples with the highest range of 16-bit audio, which is 32768, making all sample values positive. Therefore, their range has become from 0 to 32768.
2. In this step, these samples are interpolated by determining midpoints between the

Received: January 6, 2022. Revised: February 28, 2022.

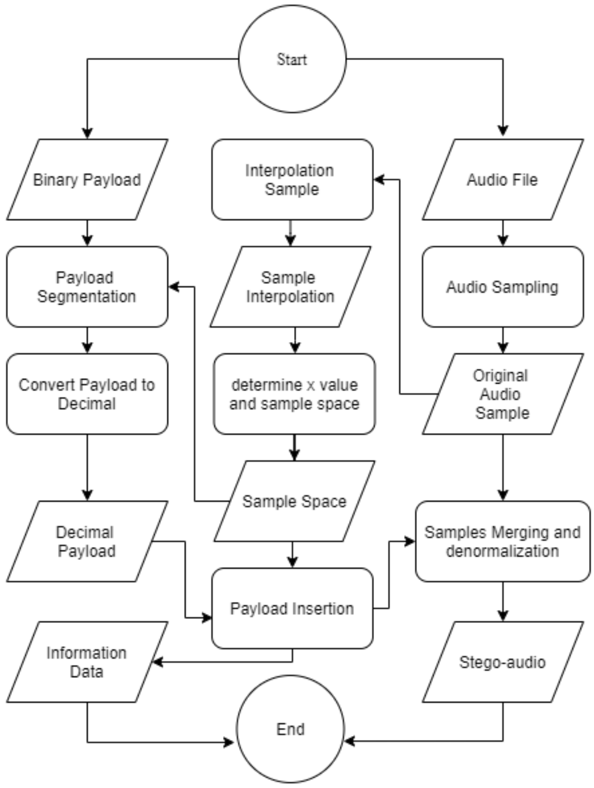

Figure 3 The embedding process

normalized audio samples. This is carried out by implementing Eq. (1).

$$S'_n = \left\lfloor \frac{(S_n + S_{(n+1)})}{2} \right\rfloor \quad (1)$$

The notation S'_n is the result of interpolation at n index, whereas S_n is the sample value on the n index.

3. Next, it needs to determine the constant value of the distance between the normalized sample and the interpolated sample using Eq. (2).

$$C = \frac{audio\ time}{audio\ length \times 2} \quad (2)$$

The notation C indicates the distance between the interpolated sample and the normalized sample in the audio signal, which is used as a constant value to calculate the sample space. The $audio\_time$ notation determines the audio file's duration in milliseconds, while the $audio\_length$ is the length of audio samples.

4. In this step, we determine sample space to find the number of bits that can be embedded in each interpolated sample. Before finding the sample space, we must calculate the distance between the original audio samples using the Euclidean method shown in Eq. (3), whose results are used to calculate sample space in each audio sample in Eq. (4).

$$d = \sqrt{(x_n - x_{n+1})^2 + (y_n - y_{n+1})^2} \quad (3)$$

$$N_n = \lfloor log_2\,(C \times d_n) \rfloor \quad (4)$$

Description
- d = Distance between the original audio samples
- x_n = The x-value of the n-th audio sample based on the x-axis of the audio signal.
- y_n = The y-value of the n-th audio sample based on the y-axis of the audio signal
- N_n = The number of how many bits can be embedded in the n-th index.

If the average value of the sample space is 0, the C notation will be updated until the average value of the sample space is more than 0. It is to ensure all payloads can be embedded.

5. After the sample space has been obtained, the payload is segmented according to the available sample space. At this stage, the payload is embedded into the interpolated audio sample. The uses method can be seen in Eq. (5).

$$S'_n = S_n - P_n \quad (5)$$

The notation S'_n represents the newly generated sample value embedded by the payload at the n-th index, while P is the segmented payload at the n-th index.

6. In this step, we merge the embedding sample with the interpolated audio sample to create a stego-audio file using Eq. (6) [4].

$$S_i = \begin{cases} S_{i/2}, & if\ i\ is\ even\ number \\ S'_{t,(i-1)/2}, & if\ i\ is\ odd\ number \end{cases} \quad (6)$$

7. Finally, the stego file is created by denormalizing its audio samples, which are then restored in the range between -32768 and 32768, as in the original values. The stego audio format is still in .wav.

 DOI: 10.22266/ijies2022.0630.17

3.2 Extraction

The extraction process retrieves the payload data from the stego-audio file. This extraction scheme is given in Fig. 4, while its corresponding steps can be explained as follows.

1. First, we normalize stego-audio files of type .wav using the same method as the first step of the embedding.
2. Then, we split the stego-audio samples into even and odd sample indexes. The odd index determines the sample containing the payload, whereas the even index is the initial audio sample. It is done by implementing Eq. (7) and Eq. (8) [4].

$$S_{n/2} = SSA_n \; for \; n \; is \; an \; even \; number \quad (7)$$

$$S''_{(n-1)/2} = SSA_n \; for \; n \; is \; an \; odd \; number \quad (8)$$

   The notation S_n shows the original audio sample at the n index. The Sn'' is the n-th sample that has been embedded with a payload, and SSA_n depicts the stego-audio sample at n-th index [4].
3. After the stego-audio samples are split into two parts, we interpolate the original audio samples by determining the midpoint of the original audio sample using the same formula as in step 2 of the embedding.
4. At this stage, we find the constant value of the distance between the normalized audio sample and the interpolated audio sample using the same algorithms implemented in step 3 of the previous embedding process.
5. The audio sample space is obtained to find the number of payload bits embedded in each interpolated sample using the same formula in step 4 of the embedding stage.
6. This step retrieves the payload from the stego-audio sample by calculating the difference between the interpolated stego-audio sample and the new interpolated sample. The method can be seen in Eq. (9).

$$Pd_n = Si_n - Sp_n \quad (9)$$

   Description
   Pd_n = Decimal payload at n-th index
   Si_n = Interpolated sample at n-th index
   Sp_n = Sample audio with payload at the n-th index

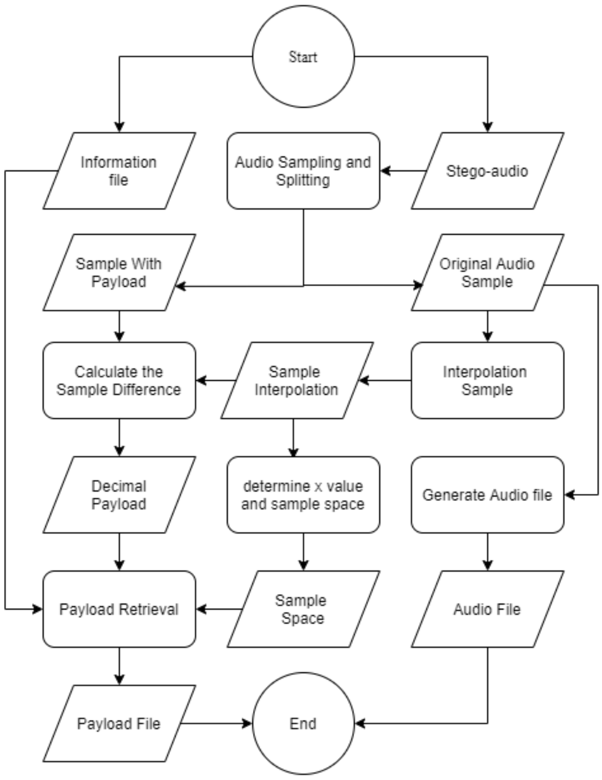

Figure 4 The extraction process

7. Then, the decimal payload is converted back into a binary payload.
8. The original audio file is reconstructed by denormalizing the audio sample using the same formula is in the last step of the embedding process.

4. Experiment result

The dataset used in this experiment consists of two types: audio and payload files. As previously described, the audio files are in .wav format because it does not experience compression, so no information is lost even though they are embedded by other data [14].

The audio cover files are taken from instrument recognition in musical audio signals (IRMAS) [15], comprising 15 audio files with 16-bit depth, mono channels, and a bit rate of 44100 Hz. This data set consists of 5 instruments, each of which is combined with three genres. The period of those audio cover files is 3 seconds, whose detail is provided in Table 1. The payload files are 12 txt data types with different sizes, which are 1, 10, 20, 30, 40, 50, 60, 70, 80, 90, and 100 Kb. This payload is obtained from one of the text generator pages, namely www.id.lipsum.com [16].

The experiment is done to evaluate the capability of the proposed method based on the following scenarios.

Table 1 Audio dataset

instrumen	name	data	genre
Cello (cel)	data 1	[cel][jaz_blu] 0011__2	Country-Folk (cou_fol)
	data 2	[cel][cla] 0007_1	Classical (cla)
	data 3	[cel][pop_ roc]0060_2	Pop-Rock (pop-rock)
Acoustic guitar (gac)	data 4	[gac][jaz_ blu]0549_2	Country-Folk (cou_fol)
	data 5	[gac][cla] 0530_1	Classical (cla)
	data 6	[gac][pop_ roc]0560_3	Pop-Rock (pop-rock)
Piano (pia)	data 7	[pia][jaz_blu] 1348_1	Country-Folk (cou_fol)
	data 8	[pia][cla] 1291_1	Classical (cla)
	data 9	[pia][pop_r oc]1306_1	Pop-Rock (pop-rock)
Saxo-phone (sac)	data 10	[sax][jaz_blu] 1605_2	Country-Folk (cou_fol)
	data 11	[sax][cla] 1598_1	Classical (cla)
	data 12	[sax][pop_ro c]1588_1	Pop-Rock (pop-rock)
Human Singing Voice (voi)	data 13	[voi][jaz_blu] 2358_1	Country-Folk (cou_fol)
	data 14	[voi][nod][co u_fol]2442_1	Classical (cla)
	data 15	[pop_roc] 2547_3	Pop-Rock (pop-rock)

1. Measuring the quality of stego-audio according to the peak signal to noise ratio (PSNR) and its respective mean squared error (MSE).
2. Obtaining the average sample space for each audio.
3. Comparing the average of PSNR results with other research.
4. Measuring the capacity of each audio file.
5. Determining the suitable genre with the proposed method based on the PSNR value for each audio genre.

In the first scenario, the quality of the stego-audio is compared to that of the initial audio file using the PSNR value after obtaining the MSE, as given in Eq. (10) and Eq. (11) [4].

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(S_i - S_i')^2 \qquad (10)$$

Mean Squared Error (MSE) calculates the error value between the stego-audio and the original audio samples. The notation N is the number of samples in audio, S_i represents a sample of the initial audio, while S_i' is a sample of stego-audio.

$$PSNR = 10 \times log_{10}\left(\frac{(2^b - 1)^2}{MSE}\right) \qquad (11)$$

Peak signal to noise ratio (PSNR) measures the similarity between the initial audio and stego-audio. The symbol 2^b is the highest value of the bit-depth, where b is 16 bits. After calculating the PSNR, we need to compare the method in this study with related methods in previous research, namely the distribution method [4] and the smoothing method [17].

In scenario 2, we calculate the average sample space for the proposed method in each audio file to prove that the proposed method has a relatively small sample space but has an even size. After that, we compared the PSNR results of the proposed method with several schemes in the previous studies to prove that this proposed method can improve the quality of stego-audio in the previous studies in scenario 3. In the following scenario, we measure the capacity of each audio to determine how many bits it can hold. Finally, we determine the appropriate audio genre according to the proposed method based on the PSNR value for each audio genre in scenario 5.

4.1 Measuring the quality of stego-audio

Based on the experiment, it is found that, on average, the highest and lowest PSNR values are obtained from 1 Kb and 100 Kb of payload, which are 114.39 dB and 92.29 dB, respectively. Specifically, the best quality is when 1 Kb data are embedded into audio 7, while the worst is 100 Kb data hidden to audio 3, which are 118.90 dB and 81.40 dB, respectively. It is because the more payload embedded into the cover, the more noises in the stego data. Therefore, the smaller payload, the more similar the stego to the cover.

The results are compared with that of other research using the payload distribution method [4, 18] and smoothing methods [17], whose results are given in Fig. 5. It is depicted that [4] and [18] produce the highest average PSNR than the proposed method,

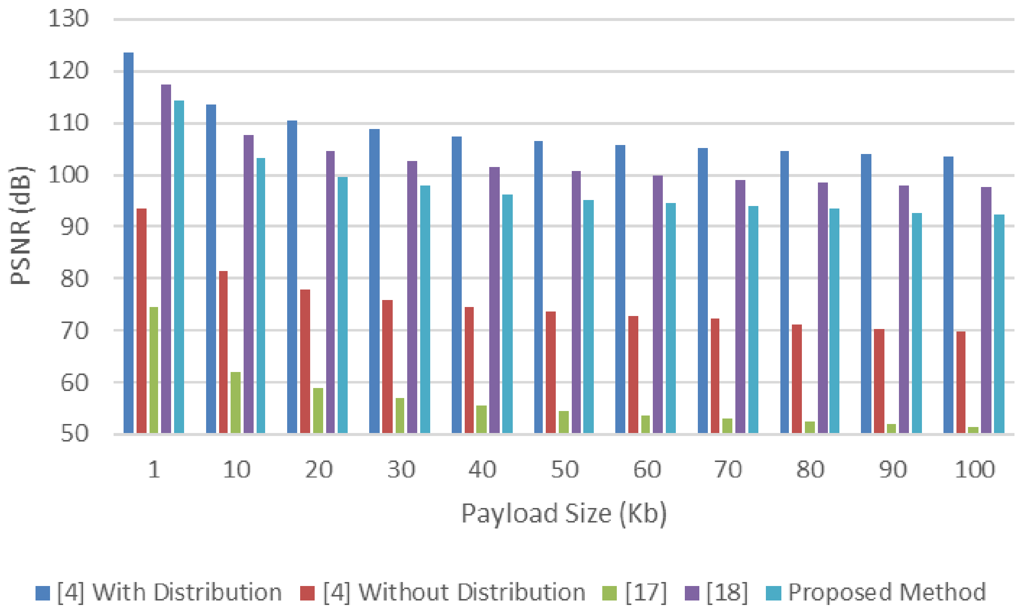

Figure. 5 The average of PSNR values taken from [4] with distribution, [4] without distribution, [17], [18], and the proposed method

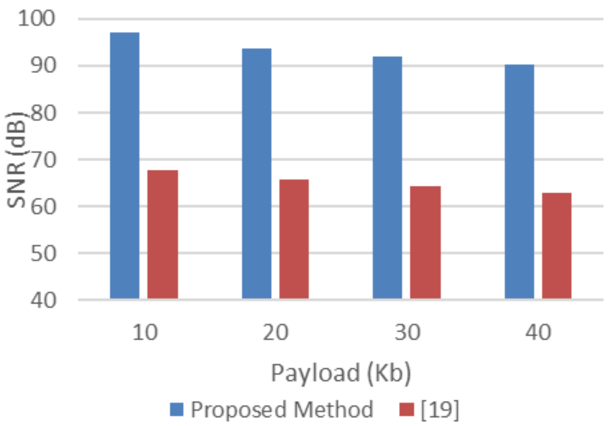

Figure. 6 The average SNR value taken from the proposed method and [19]

which are respectively 108.52 dB and 102.5 dB on all payload sizes; while the proposed method is 97.56 dB. It is because [4] and [18] implement a payload distribution step that allows the embedding process to employ more samples to get better PSNR results.

Next, the comparison is made between the proposed method and [4] without using the payload distribution and [17] with the smoothing approach. The results show that the proposed method has the highest average PSNR, 97.5 dB, while [4] with no payload distribution produces 75.7 dB, and that of [17] with the smoothing approach is 56.8 dB. In [4] without using the payload distribution, it is found that there is a decrease in the quality of stego-audio based on PSNR value, which is about 30%. It shows that the payload distribution approach has a significant impact in improving the quality of stego-audio based on the PSNR value.

We also compare the results with those of [19] that implement modification lagrange interpolation (MLI) and reduce difference expansion (RDE) methods to increase the storage capacity and combine it with smoothing method to improve the quality of stego-audio based on the resulting signal to noise ratio (SNR) value. The SNR value is positively correlated with the PSNR value. Therefore, when the SNR value is getting bigger, the quality of the stego-audio is also getting better. The comparison results can be seen in Fig. 6.

Based on each payload size, it can be concluded that the proposed method has a better SNR value than [19] that implements the smoothing method. This is indicated by an increase in the SNR value of around 30% in the proposed method without applying the smoothing method. In general, this proposed study can generate better stego-audio quality than [4, 17, 18, 19] without using additional steps such as smoothing or payload distribution on all samples. This factor can also be the strength of this proposed research because it is simpler than the previous one. At the implementation level, complexity is considered as one of the essential factors.

4.2 Analyzing the effect of sample space on the quality of each audio file

In this scenario, we monitor the average number of bits embedded in each sample. This experiment aims to prove that the proposed method can produce a relatively small sample space without distributing the payload. As provided in Fig. 7, the results show that the proposed method can reduce the number of required sample spaces. Consequently, the proposed method can produce a relatively small MSE value, which improves the quality.

As presented in Fig. 7, the proposed method produces an average sample space that ranges from 1 to 2 bits in each sample. It means that the payload embedding process requires a relatively small sample space, which does not vary much. If the available sample space gets smaller, the payload embedding is also smaller. It can be inferred that the average value of the sample space influences the quality of the stego-audio.

However, in the case of audio 3 with specifications of the pop-rock genre with cello instruments, its average sample space value is significantly different from other audio. It is because this audio type has a different distance between samples. Meanwhile, this research method states that if the difference between samples gets bigger, the stego-audio quality will decrease. This characteristic is shown in Fig. 8, which compares the distance between samples haracteristic is shown in for each audio.

The result shows the highest difference between audio samples in audio 3 (pop-rock genre with cello instruments) with 3169. In contrast, the lowest difference in audio samples is in audio 2 (a classical genre with cello instruments) with 28. Based on these results, it is found that when the difference between samples is bigger, the sample space is bigger, too. Furthermore, when the sample space gets bigger, the quality of the stego-audio falls, as depicted in Fig. 9.

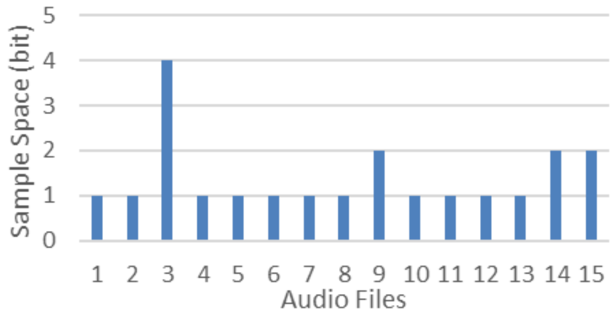

Figure. 7 Comparison of sample space

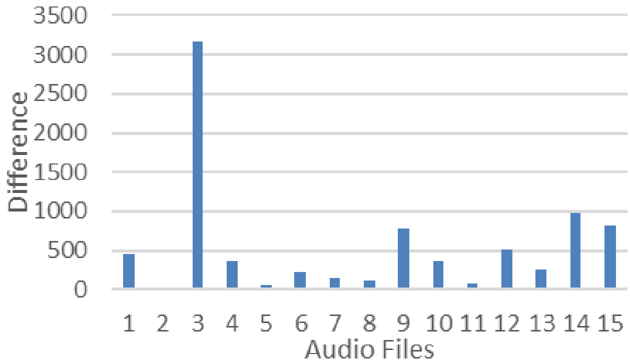

Figure. 8 Comparison of the distance between samples

4.3 Comparing the average of PSNR values

Next, the average of PSNR values for each stego-audio of the proposed method is compared with other methods [20, 21, 22], whose results are provided in Fig. 9. Those studies are implemented using the same dataset to make it as fair as possible. The figure presents that the proposed method is better than the methods in [20],[21], and [22], where it has the highest PSNR average, 97.6 dB for all audio. In comparison, the lowest PSNR is in [22], whose average PSNR value is 43.6 dB.

4.4 Measuring the capacity of each audio file

In this section, we calculate the capacity of sample space for each audio to compare the correlation between capacity and quality in stego-audio files. The results can be found in Fig. 10, showing the results of the bit capacity that can be accommodated in each audio. Here, the third audio can hold the maximum number of payload bits. At the same time, the second audio can take the least payload.

4.5 The comparison result of PSNR on each audio genre

This section evaluates the type of audio according to the proposed method based on the PSNR audio results for each genre, given in Fig. 11. Based on the comparison results, it is found that the highest average PSNR is in the classical audio genre, with an average PSNR of 99.4 dB. On the contrary, the lowest average PSNR results are in the pop-rock genre, with an average PSNR result of 95.2 dB. These results indicate that this proposed study is suitable for audio with the classical genre and less suitable for audio with the pop-rock genre.

Next, the audio capacity is evaluated to determine which audio genre is the most appropriate, considering the sample space capacity required in the embedding process. The evaluated sample space capacity is shown in Fig. 11.

Previously, Fig. 10 has explained that if the capacity of the audio file is getting smaller, the quality of stego-audio is getting better. Here, Fig. 12 shows the average capacity value for each audio genre that the lowest capacity is in the country folk

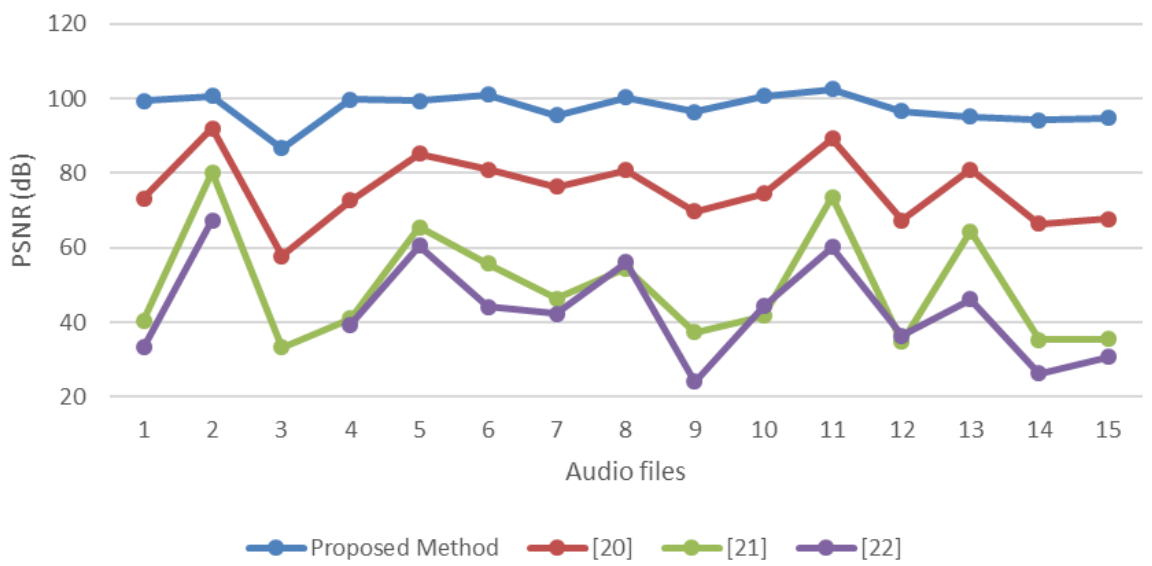

Figure. 9 The average PSNR value taken from the proposed method, [20], [21], and [22]

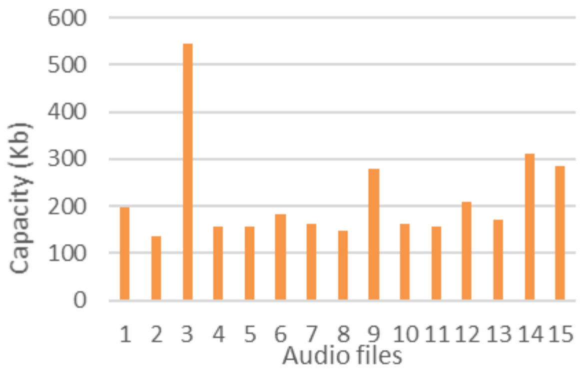

Figure. 10 Comparison of capacity for each audio file

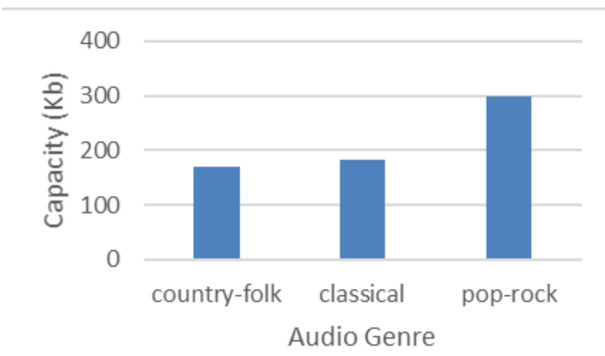

Figure. 12 Comparison of capacity for each audio genre

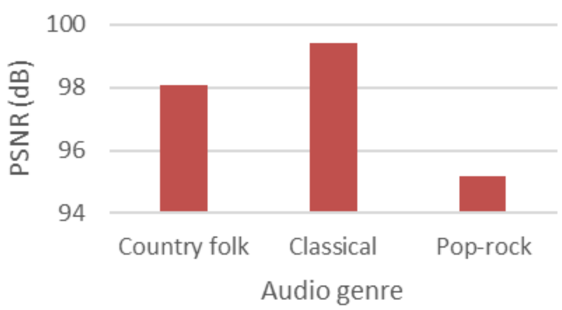

Figure. 11 Comparison of capacity for each audio

audio genre with 170556, while the highest value is the pop-rock genre with 300126. These results indicate that the most suitable audio genre to use is country-folk. On the contrary, pop-rock may not be applicable.

5. Conclusion

This research finds that the payload size influences the quality of the stego-audio, indicated by lowering the quality for a bigger payload. Without using additional steps, like payload distribution and smoothing, the proposed method reaches better results than others.

Furthermore, the size of the sample space affects the quality of the stego-audio; that smaller sample space results in better stego-audio quality. This proposed method produces a relatively small sample space, which does not differ much for each audio. If the payload embedding on the sample is relatively small, then the original sample does not change much from the audio stego sample. This results in improved stego-audio quality. It is also found that this method is more suitable to use in the classical audio genre on all instruments. The highest average PSNR indicates this in the classical audio genre of 99.4 dB.

This research can be extended to obtain a better quality in the future. For example, the most optimum size of embedding space should be considered and taken for holding the payload.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

Conceptualization, MMA and TA; methodology, MMA and TA; software, MMA; validation, MMA; formal analysis, MMA; investigation, MMA; resources, MMA and TA; data curation, MMA; writing—original draft preparation, MMA; writing—review and editing, TA; visualization, MMA; supervision, TA; project administration, TA; funding acquisition, TA.

References

[1] N. Kundu and A. Kaur, "A Secure Approach to Audio Steganography", *Int. J. Eng. Trends Technol.*, Vol. 44, No. 1, pp. 1–7, Feb. 2017, doi: 10.14445/22315381/IJETT-V44P201.

[2] C. C. Sobin and V. M. Manikandan, "A Secure Audio Steganography Scheme using Genetic Algorithm", In: *Proc. of the IEEE International Conference Image Information Processing*, Vol. 2019-Novem, pp. 403–407, Nov. 2019, doi: 10.1109/ICIIP47207.2019.8985689.

[3] N. Manohar and P. V. Kumar, "Data Encryption Decryption Using Steganography", In: *Proc. of Int. Conf. Intell. Comput. Control Syst. ICICCS 2020*, No. Iciccs, pp. 697–702, 2020, doi: 10.1109/ICICCS48265.2020.9120935.

[4] M. M. Amrulloh and T. Ahmad, "Utilizing fuzzy logic in developing reversible data hiding method", *Int. J. Intell. Eng. Syst.*, Vol. 13, No. 5, 2020, doi: 10.22266/ijies2020.1031.30.

[5] Lindawati and R. Siburian, "Steganography implementation on android smartphone using the LSB (least significant bit) to MP3 and WAV audio", In: *Proc. of ICWT 2017 3rd Int. Conf. Wirel. Telemat. 2017*, Vol. 2017-July, pp. 170–174, 2018, doi: 10.1109/ICWT.2017.8284161.

[6] S. Teotia and P. Srivastava, "Enhancing Audio and Video Steganography Technique Using Hybrid Algorithm", In *Proc. of 2018 IEEE Int. Conf. Commun. Signal Process. ICCSP 2018*, pp. 1059–1063, 2018, doi: 10.1109/ICCSP.2018.8524182.

[7] A. B. W. Putra, M. T. Jufri, D. Lantara, A. Assyauqi, A. Wajiansyah, and A. Pranolo, "Steganography for Data Hiding in Digital Audio Data using Combined Least Significant Bit and 4-Wrap Length Method", In: *Proc. of 2nd East Indones. Conf. Comput. Inf. Technol. Internet Things Ind. EIConCIT 2018*, pp. 336–339, 2018, doi: 10.1109/EIConCIT.2018.8878598.

[8] A. Benhfid, E. B. Ameur, and Y. Taouil, "Reversible steganographic method based on interpolation by bivariate linear box-spline on the three directional mesh", *J. King Saud Univ. - Comput. Inf. Sci.*, Vol. 32, No. 7, pp. 850–859, 2020, doi: 10.1016/j.jksuci.2018.09.016.

[9] A. A. Ahmad, O. S. Almousa, and Q. Abuein, "Enhancing Steganography by Image Segmentation and Multi-level Deep Hiding", *Int. J. Commun. Networks Inf. Secur.*, Vol. 13, No. 1, pp. 143–150, 2021.

[10] A. H. Ali, M. Mohdrosmadimokhtar, and L. George, "A Review on Audio Steganography Techniques", *Res. J. Appl. Sci. Eng. Technol.*, Vol. 12, No. 2, pp. 154–162, Jan. 2016, doi: 10.19026/rjaset.12.2316.

[11] V. Chandrasekaran and P. Sevugan, "Applying reversible data hiding for medical images in hybrid domain using haar and modified histogram", *Int. J. Intell. Eng. Syst.*, Vol. 10, No. 4, pp. 126–134, 2017, doi: 10.22266/ijies2017.0831.14.

[12] A. K. Sahu and G. Swain, "High fidelity based reversible data hiding using modified LSB matching and pixel difference", *J. King Saud Univ. - Comput. Inf. Sci.*, No. xxxx, 2019, doi: 10.1016/j.jksuci.2019.07.004.

[13] Y. Pei, X. Luo, Y. Zhang, and L. Zhu, "Multiple Images Steganography of JPEG Images Based on Optimal Payload Distribution", *Comput. Model. Eng. Sci.*, Vol. 125, No. 1, pp. 417–436, 2020, doi: 10.32604/cmes.2020.010636.

[14] K. Sharma and K. Gupta, "Lossless data compression techniques and their performance", In: *Proc. of 2017 International Conference on Computing, Communication and Automation* (*ICCCA*), 2017, pp. 256-261, doi: 10.1109/CCAA.2017.8229810.

[15] J. J. Bosch, J. Janer, F. Fuhrmann, and P. Herrera, "A comparison of sound segregation techniques for predominant instrument recognition in musical audio signals", In: *Proc. of 13th Int. Soc. Music Inf. Retr. Conf. ISMIR 2012*, pp. 559–564, 2012.

[16] "Lorem Ipsum.", www.id.lipsum.com (accessed Oct. 24, 2021).

[17] T. Ahmad and T. P. Fiqar, "Enhancing the performance of audio data hiding method by smoothing interpolated samples", *Int. J. Innov. Comput. Inf. Control*, Vol. 14, No. 3, pp. 767–779, 2018.

[18] Y. Samudra and T. Ahmad, "Mirroring-Based Data Hiding in Audio", *Int. J. Intell. Eng. Syst.*, Vol. 14, No. 5, pp. 550–558, 2021, doi:

10.22266/ijies2021.1031.48.

[19] R. Setiawan and T. Ahmad, "Improving the performance of data hiding by designing three samples-based smoothing in audio", *Int. J. Intell. Eng. Syst.*, Vol. 13, No. 2, pp. 196–204, 2020, doi: 10.22266/ijies2020.0430.19.

[20] K. H. Jung and K. Y. Yoo, "Steganographic method based on interpolation and LSB substitution of digital images", *Multimed. Tools Appl.*, Vol. 74, No. 6, pp. 2143–2155, Mar. 2015, doi: 10.1007/s11042-013-1832-y.

[21] T. Ahmad, M. H. Amrizal, W. Wibisono, and R. M. Ijtihadie, "Hiding data in audio files: A smoothing-based approach to improve the quality of the stego audio", *Heliyon*, Vol. 6, No. 3, p. e03464, Mar. 2020, doi: 10.1016/j.heliyon.2020.e03464.

[22] A. Bobeica, I. C. Dragoi, I. Caciula, D. Coltuc, F. Albu, and F. Yang, "Capacity control for prediction error expansion based audio reversible data hiding", In: *Proc. of 2018 22nd Int. Conf. Syst. Theory, Control Comput. ICSTCC 2018 - Proc.*, pp. 810–815, 2018, doi: 10.1109/ICSTCC.2018.8540672.