Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.079951

ARTICLE

Secret Sharing-Based Reversible Data Hiding for Enhanced Audio Data Security across Multiple Genres

Mohammad Muzayyin Amrulloh[1,2], Tohari Ahmad[1,*] and Royyana Muslim Ijtihadie[1]

[1]Department of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
[2]Department of Electrical and Informatics Engineering, Universitas Negeri Malang, Malang, Indonesia

*Corresponding Author: Tohari Ahmad. Email: tohari@its.ac.id

Received: 31 January 2026; Accepted: 20 April 2026; Published: 15 June 2026

ABSTRACT: The rapid development of digital technology has facilitated data exchange and communication, while simultaneously increasing security threats such as data theft and manipulation. As personal data is highly confidential, effective protection mechanisms are required in the digital era. Audio steganography hides secret messages (payload) within audio signals; however, many existing approaches rely on a single stego-audio output, which can lead to information leakage during storage or transmission if the file is intercepted. This vulnerability allows an attacker to more easily reconstruct the steganographic scheme from a single output. To address this limitation, this study proposes a secret-sharing-based audio steganography method in which the payload is divided into multiple parts prior to embedding, thereby providing an additional protection mechanism. Nevertheless, the use of secret sharing may degrade stego-audio quality during embedding, as indicated by lower Peak Signal-to-Noise Ratio (PSNR) values. To mitigate this issue, a linear interpolation technique is incorporated to optimize the quality of the stego-audio. The proposed method focuses on three main aspects: improving protection by embedding messages in multiple parts, maintaining embedding capacity without introducing additional stego outputs, and improving audio quality through interpolation-based optimization. Experimental results show that the proposed approach improves stego-audio quality by approximately 6.95% in PSNR compared with several previous studies, while maintaining relatively high PSNR under the evaluated experimental conditions. Statistical evaluation using Normalized Correlation (NC) and entropy measurements indicates consistent payload reconstruction with limited statistical variation after embedding. Overall, the method contributes to an audio steganography scheme with a balanced trade-off between security, capacity, and audio quality.

KEYWORDS: Audio steganography; cybersecurity; ICT infrastructure; information security; reversible data hiding; secret sharing

1 Introduction

Data have become crucial in the era of increasingly massive technological developments [1–3]. Various data types are sent and received via electronic media, including the Internet, without regard to the time. This makes the data vulnerable to theft or alteration by certain parties without proper safeguards to ensure their privacy or confidentiality [1]. Therefore, a reliable technique is required to address this issue.

In steganography, confidential payloads are embedded into digital carriers (text, images, audio, and video) with the primary objective of preserving perceptual transparency while remaining statistically inconspicuous [4–7]. Unlike cryptography, which transforms data into an unreadable form, steganography conceals the very existence of the communication, making it less likely to raise suspicion [8,9]. This

characteristic offers an advantage over watermarking, in which the embedded mark is often intended to remain perceptible for ownership identification [10–12]. In addition, steganography is applicable across various media types and can preserve the original quality of the carrier when appropriate embedding techniques are used [13,14]. In many cases, both the embedded message and the carrier media can be recovered with limited distortion, which is generally more difficult in watermarking-based approaches. Due to its concealment capability and relatively low detectability, steganography is widely considered a suitable approach for covert information transmission without attracting attention [15,16].

This study uses audio as the cover medium because it has been shown to provide higher embedding capacity than text or images, thereby enabling more effective steganographic insertion [8,17]. The general framework of audio-based steganography is illustrated in Fig. 1. In general, steganography techniques can be categorized into two models: reversible and irreversible data hiding [12,18]. This work adopts a reversible data-hiding model, which enables the embedded payload and the cover media to be restored to their original states after extraction [12,19]. In contrast, irreversible data hiding can recover the hidden message but does not fully preserve the original cover signal [20,21].

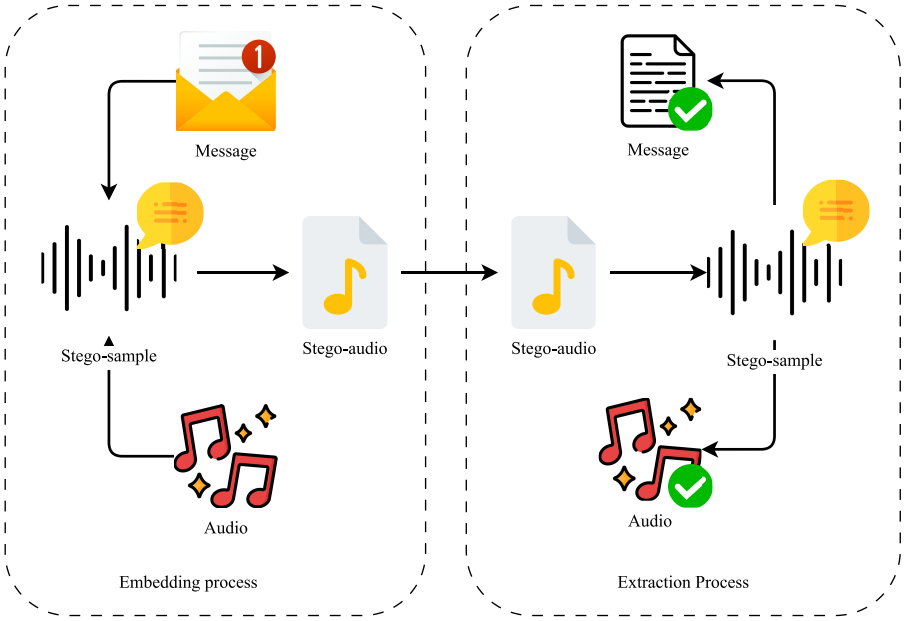

Figure 1: Basic concepts of data hiding.

With the rapid advancement of technology, steganography has increasingly faced limitations that may compromise the security of audio-based information hiding, potentially enabling attacks on stego-audio containing confidential data. Common issues include degraded stego-audio quality and unsuccessful message recovery during extraction [22,23]. Therefore, an improved method is required to address these challenges.

Several studies have attempted to improve security and quality in audio steganography. Adhiyaksa et al. [24] proposed a linear interpolation-based steganography technique to increase embedding capacity while maintaining audio quality, and demonstrated reliable payload recovery. However, this method primarily focuses on capacity and audio quality and does not explicitly address security risks associated with using a single stego-audio output. Meanwhile, Islamy et al. [25] introduced a secret-sharing-based steganography method that enhances security by dividing the payload into multiple parts and enables exact message recovery. Although more secure, this method offers limited discussion of its impact on stego-audio quality

and does not fully consider the balance among security, embedding capacity, and audio quality. Overall, existing studies tend to emphasize only one aspect, while vulnerabilities of single-stego outputs, degradation of stego-audio quality due to multi-part embedding, and the need for a method that can simultaneously balance security, capacity, and audio quality remain insufficiently addressed.

The focus of this research is to improve the quality, capacity, and security of stego-audio by using linear interpolation as the embedding location to minimize modification of the original audio samples, and by employing a secret-sharing approach that divides the hidden message into several parts. In this study, the secret shares are embedded directly during the interpolation-based insertion process rather than applied as a separate security stage. As a result, the hidden message can be reconstructed only when sufficient stego-audio shares are available, thereby reducing the risk of unauthorized access to the embedded information. The contributions and novelties of this study are as follows:

1. Employing linear interpolation to determine embedding locations, enabling message insertion while limiting modification of the original audio samples.
2. Integrating Shamir's secret sharing into the embedding process so that the payload is distributed across multiple stego-audio files, aiming to limit information exposure from a single stego object.
3. Providing an experimental evaluation that analyzes the relationship between audio quality, embedding capacity, and security-related characteristics of the proposed framework.

The paper comprises five sections. The remainder of this paper is as follows. Section 2 describes previous studies relevant to this topic, especially in the last few years. Section 3 contains the method proposed in this study in detail. Next, Section 4 presents the experimental results and analyzes the method. Finally, the paper summarizes the study's main findings in Section 5.

2 Related Works

Steganography is an actively evolving research field, with major efforts focused on improving embedding capacity while preserving the perceptual quality of the cover media. One research direction that has received increasing attention is the use of linear interpolation to expand the embedding space and enable higher-capacity data hiding with minimal distortion.

Amrulloh and Ahmad [26] proposed an interpolation-based audio steganography technique that increases the available embedding space by inserting new samples between original audio points. By generating additional embedding positions, the method can increase hiding capacity while preserving important temporal and spectral characteristics and maintaining audio quality. The authors also reported accurate message extraction, which supports the reliability of the embedding process.

Similarly, Samudra and Ahmad [27] investigated linear interpolation for audio steganography and showed that interpolation-based embedding can yield higher signal quality compared to conventional approaches without interpolation. Their work highlights how interpolation can help reduce distortions that often appear in standard embedding procedures, particularly for audio signals with wide dynamic ranges. As a result, interpolation provides a flexible strategy for applications that require low perceptual impact.

Interpolation has also been explored in image steganography. Arthy et al. [28] and Chetan et al. [29] enhanced the Least Significant Bit (LSB) method by incorporating interpolation to increase embedding capacity while maintaining visual quality. While conventional LSB is widely used due to its simplicity, it is commonly associated with limited capacity and higher vulnerability to steganalysis. By generating new pixel points as embedding candidates, interpolation-based LSB approaches can enlarge the hiding space and reduce visible artifacts such as noise and distortion. These studies also emphasize reversibility, enabling

exact reconstruction of the original content after the extraction process, which is important for applications requiring strong data integrity.

In addition to steganography, secret sharing offers a complementary security mechanism by dividing a secret into multiple shares that must be combined to recover the original information. Li [1] introduced an image-based secret-sharing approach in which the secret is distributed across several images, and reconstruction is only possible when a minimum number of shares are collected. This design is intended to strengthen protection against unauthorized access and supports distributed security scenarios.

Debnath et al. [30] extended secret sharing to video media, leveraging the large capacity and dynamic characteristics of video carriers. Their work demonstrates the feasibility of hiding and distributing larger amounts of secret data while increasing the difficulty of detection. Furthermore, the authors highlighted the potential for real-time secret sharing in secure multimedia communication and highly sensitive storage applications.

In the audio domain, Firdaus et al. [31] proposed an audio-based secret-sharing scheme that segments secret audio information into multiple shares distributed among different parties. The secret can only be reconstructed when the required number of shares is available, providing resilience even when some shares are lost or compromised [32]. However, preserving reconstruction quality remains a key challenge, especially when the carrier signal must remain perceptually acceptable.

With continuing technological advances, researchers have also proposed hybrid steganography frameworks that combine interpolation with other signal-processing techniques to achieve improved robustness. For example, integrating interpolation with wavelet transform [33] or discrete cosine transform (DCT) [34] can improve robustness against noise and compression while maintaining acceptable media quality. Such methods provide more adaptive embedding strategies under diverse transmission conditions.

To support secure share management in distributed environments, Chouhan and Arora [35] investigated blockchain-based mechanisms for tracking and distributing shares. The immutability and auditability properties of blockchain can reduce the risk of manipulation and improve transparency, which is especially useful for decentralized contexts such as the Internet of Things (IoT) and cloud systems. In addition, Wang et al. [36] proposed a secret-sharing algorithm that applies data compression prior to share generation, reducing communication overhead while maintaining security. This approach is well suited for bandwidth-limited devices and networks.

Overall, existing studies indicate a trade-off between embedding capacity, perceptual quality, and security in steganography and secret-sharing approaches. Linear interpolation–based steganography generally increases embedding capacity and maintains audio quality by expanding the embedding space. However, when relying on a single stego output, such methods remain vulnerable to information leakage. In contrast, secret-sharing techniques enhance security by dividing the payload into multiple parts, but this often results in reduced stego-media quality and increased system complexity [37]. Moreover, many studies treat interpolation and secret sharing as separate solutions, leaving joint optimization of security, capacity, and media quality unaddressed. Therefore, an integrated audio steganography approach is needed to preserve audio quality while enhancing security without sacrificing embedding capacity.

3 Proposed Method

The application design in this study consists of two main processes: embedding and extraction. The details of each process are described in the following sections. The embedding process inserts a payload into an audio medium, while the extraction process performs the reverse operation by retrieving the embedded payload from the stego-audio. The extracted payload is expected to be identical to the original payload prior

to embedding. The embedding stage requires an input audio file in *.wav format and a payload file in *.txt format. Both inputs are processed by the system to generate multiple stego-audio files as output.

However, the extraction process only requires a sufficient number of stego-audio files in *.wav format and does not depend on any additional auxiliary files. This differs from the method in [24], which utilizes extra files to store embedding-related information. An illustration of the proposed approach is shown in Fig. 1.

3.1 Embedding Process

The process of embedding a message into an audio medium combines two main approaches: linear interpolation and the secret-sharing method. Linear interpolation inserts new data points between the original audio samples by calculating the average or drawing a straight line between two adjacent points. This technique enables modifications to the audio signal while limiting structural changes, which is associated with maintained objective audio quality metrics and may reduce statistical detectability.

The secret-sharing method divides the payload into several smaller parts that can be distributed or stored separately. This adds an extra layer of protection because the original message can only be revealed when all parts are recombined.

The overall scheme and each process step are shown in Fig. 2, which illustrates how the message data are processed, embedded, and retrieved from the audio medium. For clarity and reproducibility, the detailed embedding procedure is summarized in Algorithm 1.

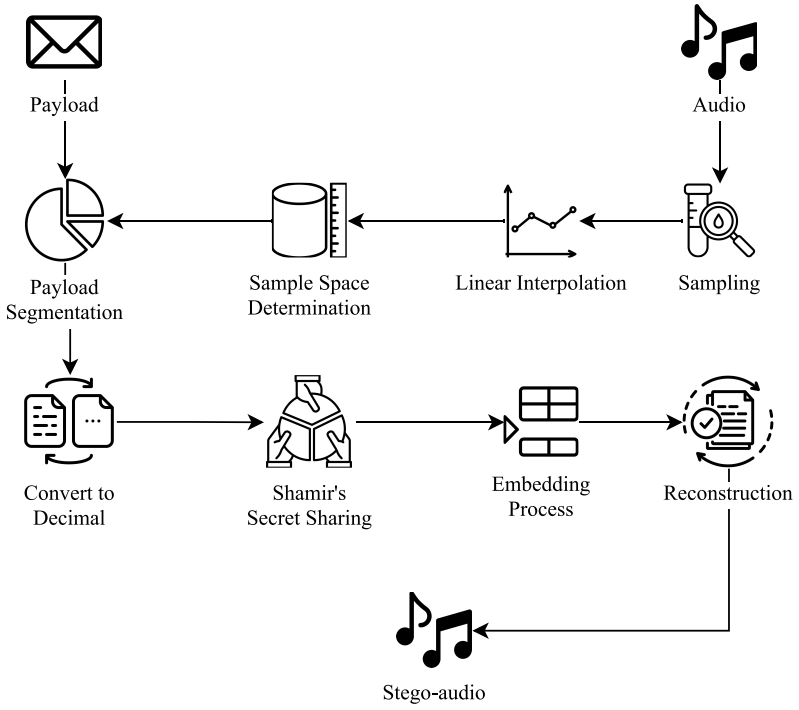

Figure 2: Embedding process.

Algorithm 1: Proposed audio steganography embedding

Require: Cover audio C, payload M, total shares n, threshold k
Ensure: Stego-audio shares S
Convert M into binary sequence B
$I \leftarrow \text{LinearInterpolation}(C)$
$Cap \leftarrow \text{DetermineSampleSpace}(I)$
$Seg \leftarrow \text{SegmentPayload}(B, Cap)$
for each segment $s \in Seg$ **do**
    $Share \leftarrow \text{ShamirSecretSharing}(s, n, k)$
    Embed $Share$ into I
end for
$S \leftarrow \text{CombineWithOriginal}(C, I)$
return S

3.1.1 Audio Sampling

The first step in this process was to sample and normalize all the original audio samples. Normalization was performed by adding each audio sample to the maximum bit depth value of 32,768 so that all samples were in the range of 0 to 32,767. In this way, negative values in the cover audio samples can be removed, making the calculation process easier and more accurate than before. An illustration of this process is presented in Fig. 3.

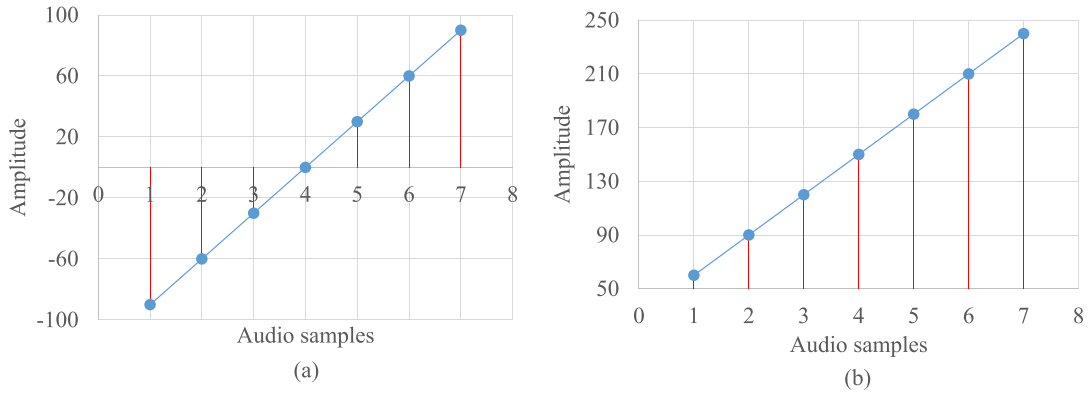

Figure 3: Illustration of audio sampling. (**a**) Before normalization; (**b**) After normalization.

3.1.2 Linear Interpolation

The next step is to insert a new sample between the two original audio samples at a new insertion location, so that the number of samples doubles. Later, the embedding process does not affect the original audio sample, so the message is easier to retrieve during extraction. At this stage, linear interpolation was used, as defined in Eq. (1), where S'_n is the new sample generated at index n and S_n is the original sample at index n. An illustration of the linear interpolation is shown in Fig. 4.

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

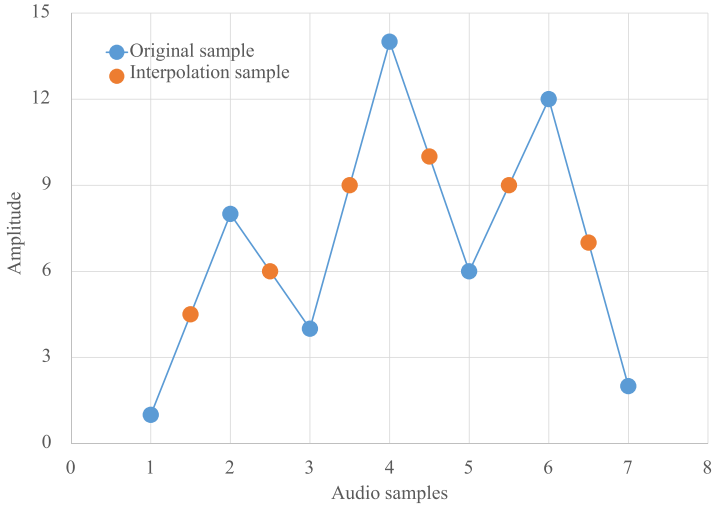

Figure 4: Illustration of linear interpolation.

3.1.3 Sample Space Determination

After generating interpolated samples, the embedding capacity of each sample is determined to avoid distortion in low-amplitude regions. Because interpolated samples S'_n exhibit varying amplitudes, samples that cannot safely carry embedded bits are excluded from the embedding process.

Eq. (2) defines the sample space SS_n for each interpolated sample S'_n. The logarithmic term relates the amplitude of S'_n to the available embedding levels, the square root limits excessive capacity growth, and the floor function ensures that SS_n is an integer. This formulation balances embedding capacity and audio quality.

$$SS_n = \left\lfloor \sqrt{\log_2 S'_n} \right\rfloor \tag{2}$$

3.1.4 Payload Segmentation

After getting the number of bits that can be accommodated, the payload will be cut according to the results of the bit capacity that can be accommodated in the previous stage. Then, each payload cut will be converted to decimal. Table 1 shows the illustration at this stage.

Table 1: Illustration of payload segmentation.

Sample	1	2	3	4	5	6
Sample Space	2	1	3	4	1	0
Binary Payload	01	1	101	1010	0	–
Decimal Payload	1	1	5	9	0	–

3.1.5 Shamir's Secret Sharing

In this stage, we implemented Shamir's Secret Sharing (SSS) to divide the payload into multiple shares. SSS relies on Lagrange polynomial interpolation. Let a secret S be distributed among n participants such that any subset of at least k participants can reconstruct it. The secret-sharing procedure is illustrated in Fig. 5 and is formulated in Eqs. (3) and (4).

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_{k-1} x^{k-1} \mod p \tag{3}$$

The notation a_0 (or S) represents the secret to be shared, while $a_1, a_2, \ldots, a_{k-1}$ are randomly generated coefficients. The value of p is a prime number greater than S.

$$y_i = f(x_i) \mod p, \quad \text{for } i = 1, 2, \ldots, n \tag{4}$$

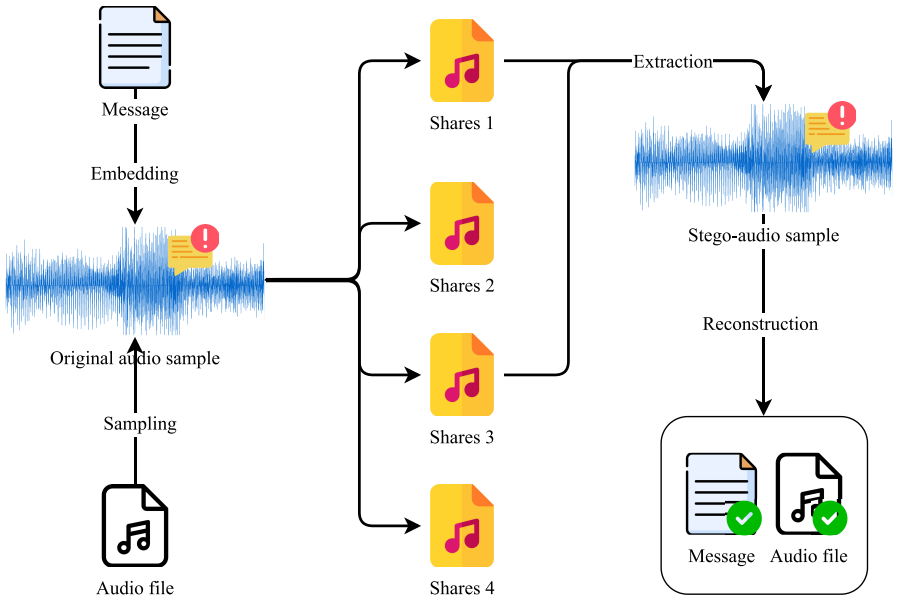

Figure 5: Illustration of the embedding and extraction process using a secret sharing–based audio steganography scheme with four generated shares.

3.1.6 Embedding

This stage is the primary step in the embedding process. This stage works by inserting the output of Shamir's secret-sharing method into an interpolation sample in each created stego-audio. The equation at this stage is given in Eq. (5). Therefore, the new sample point is always below the original interpolated sample. An illustration of this process is shown in Fig. 6. The notation S''_n indicates a new interpolation sample that already contains the payload at the n index. S'_n is the interpolation sample at the n index, and P_n is the data share at the n index.

$$S''_n = S'_n - P_n \tag{5}$$

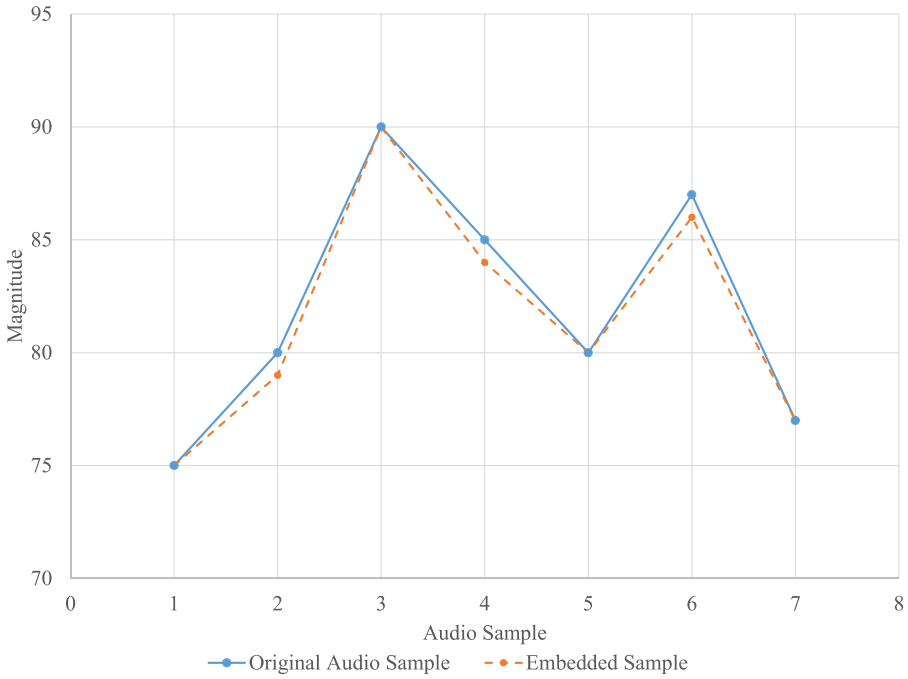

Figure 6: Illustration of the embedding process.

3.1.7 Sample Combining

Next, the original samples are combined with the interpolated samples containing the embedded payload to form the final stego signal. Eq. (6) places the interpolated samples $S'_{t,\frac{(i-1)}{2}}$ at odd indices and the original samples $S_{\frac{i}{2}}$ at even indices, preserving the temporal structure of the audio signal while integrating the embedded information.

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

3.1.8 Denormalization

Finally, we restore the audio samples to their original range, which is between −32,768 to 32,767. This process is done by adjusting the value of each audio sample by adding a value of 32,768 so that it can be reconstructed back into an audio file in *.wav format. This process illustration is the reverse of Fig. 3.

3.2 Extraction Process

The process of extracting a message from a stego-audio file uses the same methods as those used during the embedding phase, namely, linear interpolation and secret sharing. Linear interpolation identifies data points that are modified or inserted into the audio signal by comparing the patterns between the original and embedded samples. Using this technique, the system can recalculate the interpolated values used during embedding to reveal the location and content of the hidden message.

Meanwhile, the secret-sharing method allows the extracted fragments of the message from the audio to be recombined into a complete and meaningful message. Because the original message was previously

divided into several smaller parts and distributed across the audio file, the reconstruction process requires these fragments to accurately recover the hidden messages.

The full scheme and stages of the extraction process are shown in Fig. 7, illustrating the logical flow from detecting hidden components in the audio signal to reconstructing the final message for decoding. For clarity and reproducibility, the detailed extraction procedure is summarized in Algorithm 2.

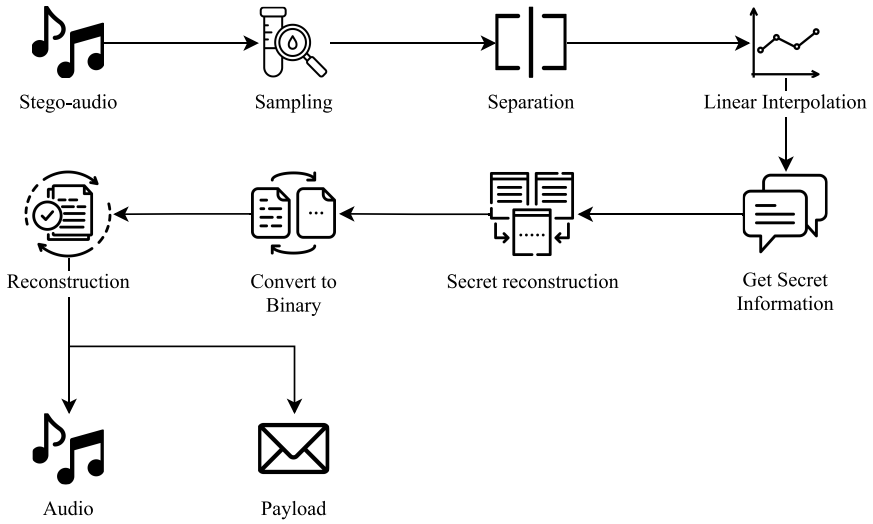

Figure 7: Extraction process.

Algorithm 2: Extraction and secret reconstruction

Require: Stego-audio shares S, threshold k
Ensure: Recovered payload M
1: Separate embedded samples from S
2: $I \leftarrow$ InterpolateEmbeddedSamples()
3: **for** each extracted share **do**
4: Recover decimal value
5: **end for**
6: $Data \leftarrow$ ReconstructSecret(k shares)
7: $M \leftarrow$ ConvertToBinary($Data$)
8: **return** M

3.2.1 Audio Sampling

This stage is the same as the initial stage in the embedding process, which starts with sampling and normalization, where each sample is added with a value of 32,768 so that all samples are in the range of 0 to 32,767. This ensured that all samples could be calculated more easily because the negative values were converted to positive values. An illustration of this process can be seen in Fig. 3.

3.2.2 Audio Separation

Next, the audio sample sequence is divided into two groups to separate the original samples from those containing the embedded payload based on their index positions. Samples with even indices (e.g., 0, 2, 4, 6, . . .) correspond to the original audio samples, whereas samples with odd indices (e.g., 1, 3, 5, 7, . . .) contain

the embedded payload. This separation step is defined in Eq. (7). The illustration of this process represents the reverse of the embedding procedure shown in Fig. 6.

$$\begin{aligned} S_k &= SS_n, \quad \text{if } n \text{ is even, } k = \frac{n}{2} \\ S''_k &= SS_n, \quad \text{if } n \text{ is odd, } k = \frac{n-1}{2} \end{aligned} \tag{7}$$

3.2.3 Linear Interpolation

Each original audio sample obtained from the previous stage needs to be added to a new sample using the linear interpolation method again to get the difference value between the new interpolated sample and the sample containing the payload by subtracting the value of the new interpolated sample from the value of the sample that has the payload embedded. The process and equations used in this stage are the same as those used in Stage 2 in the embedding process.

3.2.4 Sample Space Determination

Then, after getting a new interpolation sample, each sample will go through a calculation stage to determine how many bits can be accommodated. The process and equations used in this stage are the same as Stage 3 in the embedding process.

3.2.5 Secret Reconstruction

At this stage, the differences between the payload-carrying samples and the interpolated samples are reconstructed to obtain the payload in decimal form using Lagrange interpolation. This method is employed because it can reconstruct a unique polynomial from several known shares. The reconstruction step is defined in Eq. (8). In this formulation, $P(x)$ denotes the reconstructed interpolation polynomial, $L_i(x)$ represents the Lagrange basis polynomial, and (x_i, y_i) are the known share points. The parameter p is a prime number used in the modulo operation. The secret reconstruction process based on secret sharing is illustrated in Fig. 5.

$$\begin{aligned} P(x) &= \sum_{i=0}^{n} y_i L_i(x) \mod p \\ L_i(x) &= \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j} \mod p \end{aligned} \tag{8}$$

3.2.6 Payload Conversion

Next, the results of the previous stage in the form of a payload in decimal form need to be converted into binary form according to the number of bits that can be accommodated in each sample. This stage is the opposite of Stage 4 in the embedding process, so it has the same illustration as in the Table 1.

3.2.7 Payload and Audio Reconstruction

Finally, the payload in binary form is reconstructed back into a file with the format *.txt, while the original audio sample resulting from the second stage of the extraction process needs to be denormalized to return to the original range, namely −32,768 to 32,767, so that it can be reconstructed back into audio with a 16-bit mono channel and in the format *.wav.

4 Experimental Results

This section describes the research analysis procedure based on the experimental scenarios conducted in this study.

4.1 Experimental Setup

Before presenting the experimental scenarios, the datasets used and their quantities are first described in detail. Two main types of data are used: audio and payload. The payload dataset comprises 11 text files, with sizes ranging from 1 to 100 kb. The payload content was generated using the Lipsum text generator and stored in plain text (*.txt) format. Meanwhile, the cover audio dataset comprises 15 audio files with 16-bit mono-channel specifications and an average duration of approximately three seconds per file. The audio data were obtained from the IRMAS website [38], which provides musical recordings across multiple genres (Classical, Pop Rock, and Country–Folk) and instrument categories, including cello, acoustic guitar, piano, saxophone, and human singing voice. All audio files are in uncompressed *.wav format, enabling sample-level modification without the risk of compression-induced data loss [28]. In addition to the IRMAS dataset, an additional audio dataset is used for computational performance evaluation in Scenario 2. This dataset consists of speech audio with a duration of approximately 9 s in *.wav format, obtained from the LJSpeech dataset available on Kaggle [39]. Although the experiments employ a relatively small number of audio clips with short duration, this controlled setup was selected to ensure consistent evaluation of embedding performance.

Next, we designed three test scenarios to demonstrate the functionality and performance of the proposed method. In the first scenario, the quality of the generated stego-audio was evaluated using mean squared error (MSE) and peak signal-to-noise ratio (PSNR). These metrics compare the stego-audio signal with the corresponding original audio to analyze the impact of payload size, secret-sharing parameters (n, k), and method variations on audio imperceptibility. The similarity between the resulting stego-audio and the interpolated original audio is calculated using Eqs. (9) and (10), where N denotes the number of audio samples, S_i represents the original audio sample at index i, S_i' denotes the stego-audio sample, and 2^b corresponds to the maximum bit depth value with $b = 16$ bits. PSNR and MSE primarily measure signal distortion and may not fully reflect perceptual audio quality as perceived by human listeners. They also do not directly indicate resistance against statistical steganalysis, since high PSNR values do not necessarily imply statistical undetectability. However, they remain widely used for objective comparison in audio steganography studies.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(S_i - S_i')^2 \tag{9}$$

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

In the second scenario, we evaluate the computational performance of the proposed method in terms of execution time and memory usage. This evaluation examines how payload size and secret-sharing parameters (n, k) affect the computational cost of the embedding process. The measurement is conducted during the embedding stage, where the execution time reflects the processing duration, and memory usage represents the peak memory consumption observed during the process. Through this scenario, we aim to provide an overview of the operational requirements of the proposed method under different configurations.

In the third scenario, we compare the proposed method with several related studies using the same evaluation metrics. The comparison assesses the proposed method's performance relative to existing approaches

in terms of stego-audio quality. The analysis focuses on PSNR values reported in previous studies and compares them with the results obtained in this work across different payload sizes.

4.2 Results and Analysis

This section presents the experimental results obtained from the proposed method and analyzes their impact on stego-audio quality under different experimental scenarios. The analysis evaluates imperceptibility performance and examines how secret-sharing parameters and payload distribution affect the resulting stego-audio characteristics. The results in this section focus on quality and computational performance evaluation and do not directly represent the security performance of the proposed method.

4.2.1 Measuring the Quality of Stego-Audio Based on Secret Sharing Parameters

In this first scenario, we evaluate the stego-audio quality produced by the secret-sharing method across various (n, k) configurations. Fig. 8 shows the relationship between payload size and stego-audio quality (dB). The results indicate that stego-audio quality decreases as the payload increases. For a small payload (1 kb), the quality exceeds 113 dB, while for a larger payload (100 kb), it decreases to approximately 89–93 dB, reflecting the trade-off between payload capacity and imperceptibility. The relatively high PSNR values, particularly for smaller payload sizes, occur because the embedding process modifies interpolated samples with only small amplitude differences relative to the original 16-bit audio signal, resulting in very low numerical distortion between the cover and stego-audio. Although PSNR does not directly represent human auditory perception, these values indicate that the introduced modifications remain limited according to the objective quality metrics used.

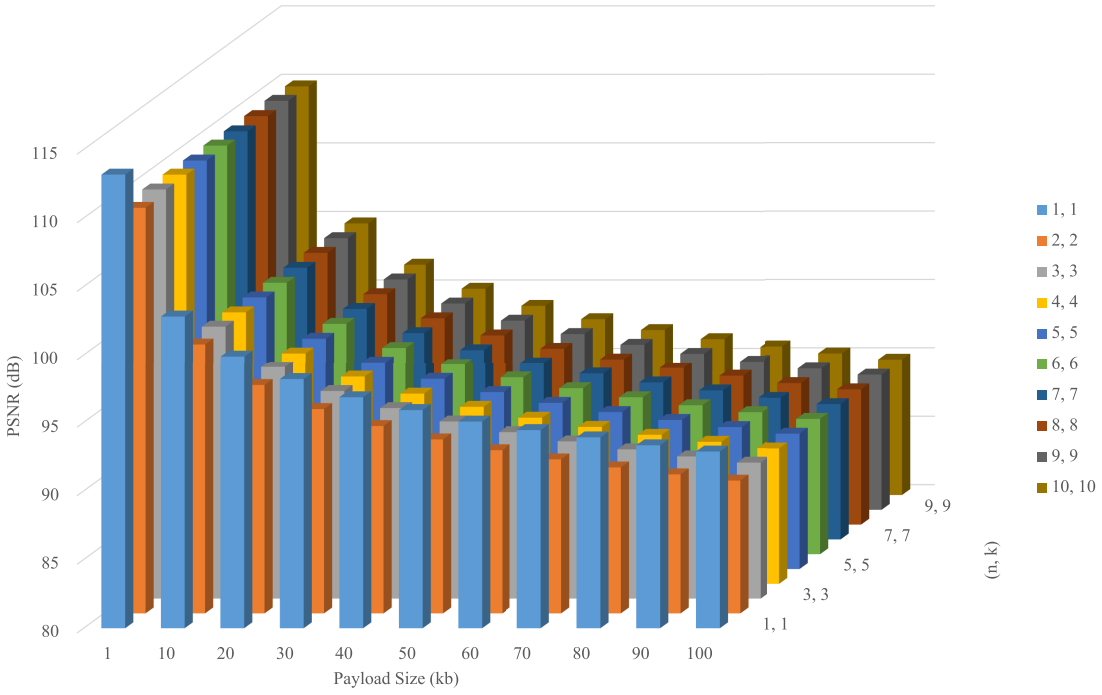

Figure 8: Comparison based on secret sharing parameters.

Moreover, varying the secret-sharing parameters (n, k) produces only a slight difference in stego-audio quality. For each payload size, the dB values across different configurations remain close, indicating that

share generation and threshold-based reconstruction do not introduce significant additional distortion in the evaluated metrics. Payload size shows a stronger influence on quality compared to the choice of (n, k) under the tested conditions.

It should be noted that the security aspect of secret sharing is discussed separately in Section 4.3. The threshold parameter (k) ensures that the payload can only be reconstructed when at least k valid shares are available, making unauthorized recovery more difficult as k increases. Overall, Fig. 8 indicates that the proposed method maintains high imperceptibility while providing protection for sensitive information.

4.2.2 Computational Performance Based on Secret Sharing Parameters

In this scenario, we analyze the computational performance of the proposed method in terms of execution time and memory usage under different payload sizes and secret-sharing configurations (n, k). The evaluation also includes the (1, 1) configuration as a baseline without secret sharing and a comparison with an existing method. The evaluation uses a separate speech audio dataset with a duration of approximately 9 s from the LJSpeech dataset to observe processing behavior under different input conditions. This dataset is used only for computational analysis and is not involved in the quality evaluation presented in Scenarios 1 or 3.

Table 2 shows that execution time increases with payload size across all configurations. In the baseline configuration (1, 1), the processing time ranges from approximately 2.38 to 2.67 s. For example, in the (2, 2) configuration, the processing time increases from approximately 2.48 s for a 1 kb payload to 2.73 s for a 100 kb payload. Configurations with larger n values, such as (5, 2) and (5, 3), require longer processing time compared to configurations with smaller n. Compared to Adhiyaksa et al. [24], the proposed method with (1, 1) configuration shows comparable execution time, while configurations with larger (n, k) require additional processing due to the secret-sharing mechanism.

Table 2: Execution time and memory usage under different payload sizes taken from [24] and the proposed method.

Configuration/Method	Time (s)			Memory (MB)		
	1 kb	50 kb	100 kb	1 kb	50 kb	100 kb
(1, 1)	2.38	2.43	2.67	27.02	29.76	32.56
(2, 2)	2.48	2.69	2.73	47.03	49.77	52.58
(3, 2)	3.50	3.68	3.66	62.78	65.52	68.33
(5, 2)	5.40	5.55	6.50	94.29	97.55	100.89
(5, 3)	5.39	5.59	5.61	94.29	97.55	100.89
Adhiyaksa et al. [24]	2.88	3.11	3.16	18.40	19.56	20.72

Memory usage also increases with both payload size and the number of shares. In the (2, 2) configuration, memory consumption ranges from approximately 47 to 52 MB. For configurations with larger n, such as (5, 2) and (5, 3), memory usage increases to around 94 to 101 MB. In comparison, Adhiyaksa et al. [24] requires lower memory usage, as it does not involve the secret-sharing process.

The PSNR values obtained from this scenario remain relatively similar across different configurations. Slight differences are observed for larger (n, k), but the changes are small and consistent with the behavior observed in Scenario 1. Overall, payload size affects execution time, while the number of shares has a larger impact on memory usage. Despite the additional computational cost, the processing time remains within a few seconds per audio file, indicating that the proposed method is still practical for use.

4.2.3 Quality Comparison with Previous Methods

In this scenario, we present a comprehensive comparison between the proposed method and several previous studies that have applied related methods and the same dataset, as referred to in [40–43], and [24]. This comparative analysis aims to provide in-depth insights into the proposed method's ability to maintain its security and quality levels when juxtaposed with other relevant approaches. A detailed observation of the comparison results is presented in Fig. 9, the stego-audio quality decreases as the size of the embedded payload increases. Consequently, the results indicate a negative correlation between payload size and stego-audio quality. This implies that when the payload size is increased, the audio quality of the hidden message tends to decrease, whereas a smaller payload size correlates with higher stego-audio quality. These results are consistent with the commonly observed trade-off between embedding capacity and carrier media quality.

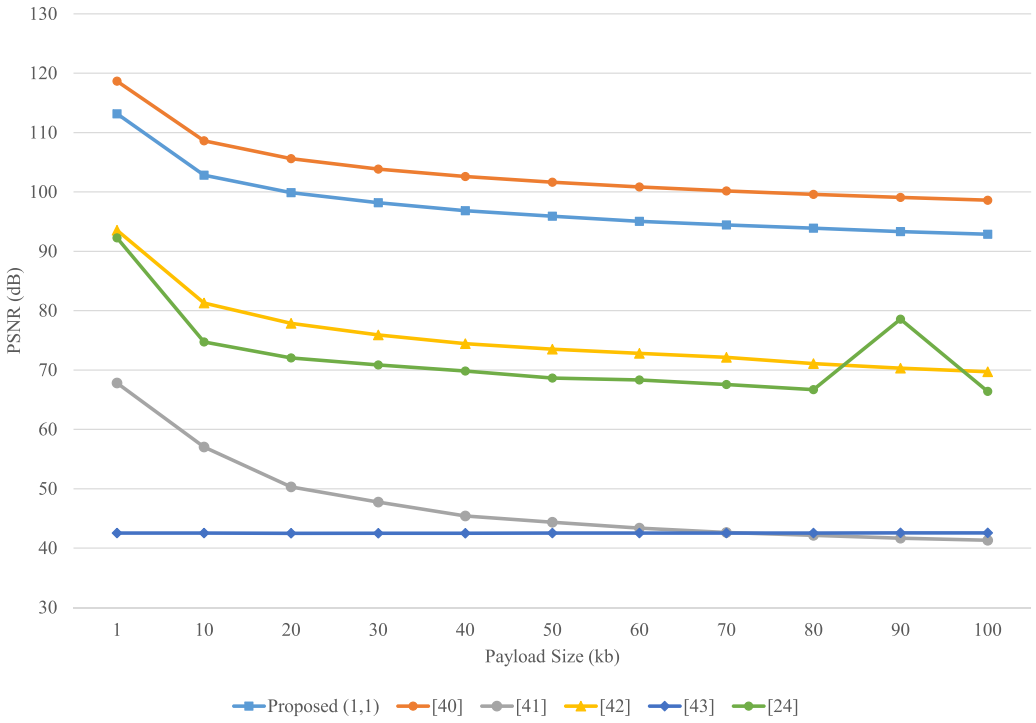

Figure 9: The average of PSNR values taken from [24,40–43], and the proposed method.

In the study conducted by [40], the highest average quality was recorded among the compared studies for every embedded payload size, amounting to 103.5690 dB. Furthermore, the PSNR value achieved was 118.9237 dB, measured in the acoustic guitar instrument within the pop-rock genre at a 1 kb payload size. This value is higher than those reported in the other compared studies, including the proposed method. Conversely, the research in [43] demonstrated the lowest average quality of 49.3151 dB across the tested payload sizes. The highest PSNR value attained by [43] was 67.5171 dB, measured in the pop-rock genre using a cello.

Meanwhile, the proposed method produced second highest after the study by [40], with an overall average PSNR of 97.8571 dB. The highest PSNR value achieved by the proposed method was 110 dB for a 1 kb payload size across all audio genres. In contrast, the lowest recorded PSNR value was 92.8753 dB at a 100 kb payload size for all audio genres. This phenomenon can be attributed to the implementation of Shamir's secret-sharing technique in the proposed method, which allows a payload to be broken down

and distributed across several audio files. Consequently, each generated stego-audio has varying quality characteristics, depending on the portion of the embedded payload and the carrier audio characteristics.

In addition to the quality evaluation, embedding capacity is also considered. A comparison with Adhiyaksa et al. [24], which uses a similar interpolation-based approach, shows that the proposed method achieves a higher capacity of approximately 396,894 bits per audio file, whereas [24] ranges from 175,000 to 184,000 bits. Based on the PSNR results in Fig. 9, the proposed method also maintains higher quality across different payload sizes. These results indicate that the proposed method can embed more data while preserving the quality of stego-audio.

4.3 Security Evaluation

This section evaluates the security aspects of the proposed method from analytical, statistical, and detector-based perspectives. The analysis includes a probabilistic security evaluation based on the secret-sharing mechanism, a statistical assessment using objective measurement metrics, and a detector-based steganalysis experiment. Security-related analysis focuses on configurations where $k > 1$, while the results in the previous section are intended for quality and computational evaluation.

4.3.1 Analytical and Probabilistic Security Analysis

The security of the proposed method aims to mitigate the risk of unauthorized parties accessing hidden information. To support the security analysis, several assumptions regarding attacker capability and share distribution are defined. The attacker is assumed to have access to the communication channel and may intercept one or more transmitted stego-audio files. However, the attacker does not know the embedding locations, the interpolation process, or the complete set of generated shares. Each stego-audio share is assumed to be transmitted or stored independently. Under these conditions, successful reconstruction of the hidden payload depends on whether the attacker can obtain a sufficient number of valid shares that satisfy the reconstruction threshold defined by the Shamir's Secret Sharing scheme.

The linear interpolation technique enables embedding to be performed on interpolated samples rather than directly on the original audio signal and not uniformly across the media, which helps reduce structural distortion and may make statistical detection more difficult for steganalysis methods that rely on distribution or pattern analysis [44]. In addition, Shamir's Secret Sharing functions as a data-sharing mechanism rather than encryption by dividing the hidden data into multiple shares, where reconstruction is only possible when a sufficient number of shares is obtained, thereby increasing the difficulty for attackers who access only partial information [45]. This combination preserves data confidentiality while introducing an additional layer of protection by fragmenting information.

Imperceptibility metrics such as PSNR are commonly used to evaluate how well the quality of the original audio signal is preserved after data embedding, which indirectly relates to signal invisibility [46]. Entropy analysis is also widely applied to examine statistical characteristics associated with resistance to detection [47]. In this study, the security discussion in this subsection focuses on analytical and probabilistic evaluation, while statistical validation based on entropy and related measurements is presented separately in the following subsection.

A probabilistic model is employed to estimate the likelihood of an attacker successfully reconstructing the hidden data. In Shamir's Secret Sharing, the secret is divided into n shares and can only be reconstructed when at least t shares are obtained, where $t \leq n$ denotes the reconstruction threshold. Let p represents the probability that an attacker intercepts an individual stego-audio share during transmission or storage.

Assuming each interception occurs independently, the number of collected shares follows a binomial distribution. Accordingly, the probability of successful reconstruction is calculated as the cumulative probability of obtaining at least t shares, as expressed in Eq. (11).

$$P(k \geq t) = \sum_{k=t}^{n} \binom{n}{k} p^k (1-p)^{n-k} \tag{11}$$

In Shamir's Secret Sharing, the level of security is primarily determined by the number of shares required for reconstruction, rather than by the size of the embedded payload. The probabilistic model allows estimation of the likelihood that an attacker can successfully recover the secret data. For instance, when a secret is divided into 10 shares ($n = 10$) with a reconstruction threshold of 7 shares ($t = 7$), and the probability of intercepting each share is 20% ($p = 0.2$), the chance of obtaining enough shares remains limited. This shows that reconstruction becomes unlikely when only partial shares are accessible. In general, lower interception probability and higher threshold values reduce the possibility of successful reconstruction, providing a quantitative basis for selecting appropriate secret-sharing parameters.

Shamir's Secret Sharing is based on polynomial functions, in which reconstruction becomes deterministic once the required threshold (t) of valid shares is reached. If fewer than t shares are obtained, the original message cannot be correctly reconstructed and the resulting output generally appears random without revealing meaningful information. In contrast, when t or more valid shares are available, the secret can be recovered exactly, provided that the shares and reconstruction process are valid. Therefore, the primary difficulty for an attacker lies in obtaining the minimum number of required shares rather than performing the reconstruction itself. Although this threshold mechanism supports confidentiality, distributing and managing multiple stego-audio shares may introduce practical complexity. Future work may consider alternative strategies, such as key-based or encrypted embedding, to simplify share management.

4.3.2 Statistical Security Evaluation

In addition to analytical analysis, statistical evaluation is conducted to examine the security characteristics of the stego-audio signals. This evaluation employs Normalized Correlation (NC) and entropy measurements to analyze extraction reliability and statistical consistency after the embedding process.

Normalized Correlation (NC) measures the similarity between the original payload M and the extracted payload M', thereby evaluating reconstruction consistency after embedding and extraction. The NC value is calculated as defined in Eq. (12), where L denotes the payload length. This metric is selected because it provides a direct measure of similarity between two data sequences without relying on bit-level error counting. Values closer to 1 indicate higher similarity between the original and reconstructed payloads.

$$NC = \frac{\sum_{i=1}^{L} (M_i \cdot M_i')}{\sqrt{\sum_{i=1}^{L} M_i^2} \cdot \sqrt{\sum_{i=1}^{L} (M_i')^2}} \tag{12}$$

Meanwhile, entropy analysis is employed to observe statistical changes between the cover and stego-audio signals after embedding. The entropy value reflects the distribution of signal amplitudes and is calculated using Shannon entropy as defined in Eq. (13), where $p(x_j)$ represents the probability of occurrence of amplitude value x_j. Entropy is used as a general statistical indicator to assess whether the embedding process introduces noticeable changes in the audio signal's distribution.

$$H = -\sum_{j} p(x_j) \log_2 p(x_j) \tag{13}$$

The secret-sharing configurations evaluated in this study include (2, 2), (3, 2), (5, 2), and (5, 3). These configurations were selected to represent different combinations of total shares (n) and reconstruction thresholds (k). The (2, 2) configuration represents the basic case where all shares are required for reconstruction. Configurations such as (3, 2) and (5, 2) allow reconstruction even when some shares are unavailable, while the (5, 3) configuration applies a higher threshold requirement. These variations are used to observe whether changes in the number of shares and threshold settings influence reconstruction consistency and the statistical characteristics of the stego-audio.

The statistical evaluation results based on the NC metric are presented in Table 3. The obtained NC values for all tested configurations remain consistently close to 1, with only small differences between minimum and maximum values. This indicates that the extracted payload remains highly similar to the original payload under the evaluated conditions. Similar NC values across configurations (2, 2), (3, 2), (5, 2), and (5, 3) suggest that variations in the number of shares and reconstruction thresholds do not noticeably affect payload reconstruction when the required threshold is satisfied. This behavior is consistent with the deterministic reconstruction process of the secret-sharing scheme.

Table 3: Average Normalized Correlation (NC) values under different secret-sharing configurations.

Configuration (n, k)	Mean NC	Minimum NC	Maximum NC
(2, 2)	0.9999996	0.9999995	0.9999997
(3, 2)	0.9999996	0.9999995	0.9999997
(5, 2)	0.9999996	0.9999995	0.9999997
(5, 3)	0.9999996	0.9999995	0.9999997

The entropy comparison between cover and stego-audio signals is summarized in Table 4. The entropy values of the stego-audio remain very close to those of the corresponding cover audio for all configurations. The observed entropy differences are on the order of 10^{-4} bits, with percentage changes around 0.001%. These small variations indicate that the embedding process introduces limited changes to the overall statistical distribution of audio samples. The relatively stable entropy values across different configurations suggest that increasing the number of shares does not introduce noticeable statistical deviation under the tested scenarios.

Table 4: Average entropy comparison between cover and stego-audio signals.

Configuration (n, k)	Cover Entropy	Stego Entropy	Entropy Difference	Change (%)
(2, 2)	15.9978	15.9980	1.90×10^{-4}	0.00144
(3, 2)	15.9978	15.9980	1.89×10^{-4}	0.00143
(5, 2)	15.9978	15.9980	1.88×10^{-4}	0.00141
(5, 3)	15.9978	15.9980	1.87×10^{-4}	0.00134

Overall, the statistical evaluation shows that the proposed embedding process maintains consistent payload reconstruction while producing only limited statistical variation in the evaluated audio signals. Within the tested configurations, changes in the number of shares and reconstruction thresholds do not introduce noticeable differences in NC or entropy values. These observations describe the statistical behavior of the proposed method under the conducted experimental conditions. To provide a balanced interpretation of the results, several limitations are discussed below.

4.3.3 Detector-Based Steganalysis Evaluation

In addition to analytical and statistical analysis, a detector-based steganalysis experiment was conducted to evaluate whether the stego-audio can be distinguished from the cover audio. In this experiment, the audio was divided into segments, and statistical, spectral, and difference-based features were extracted from each segment. A Support Vector Machine (SVM) classifier was then used to perform binary classification between cover and stego audio. The evaluation uses the same speech audio dataset as in the computational scenario in 4.2.2, with payload sizes of 1, 50, and 100 kb.

The results show a detection accuracy of 47.58%, precision of 47.54%, recall of 46.77%, and F1-score of 47.15%. The AUC-ROC value of 41.62% reflects limited separability between cover and stego audio. Additionally, 5-fold cross-validation yields a mean accuracy of 48.28% (±5.10%), which is close to random classification. This observation suggests that the steganalysis model does not reliably differentiate between cover and stego audio under the current experimental setup. For comparison, Adhiyaksa et al. [24] report a detection accuracy of 39.52% and an AUC-ROC value of 34.64% under a similar evaluation setting. Overall, both methods show comparable detection performance when evaluated using the same feature-based steganalysis approach. However, the evaluation remains limited, as it does not include more advanced steganalysis methods.

4.4 Discussion and Limitations

This study was conducted under controlled experimental conditions and has several limitations. The dataset is limited to short, uncompressed audio, which may limit generalizability. The method has not been evaluated under lossy compression, signal distortion, or transcoding, as the focus is on reversible data hiding in distortion-free settings. In addition, the combination of interpolation and secret sharing introduces computational overhead that may affect real-time applicability. A detector-based steganalysis experiment has been conducted, but it is still limited to a feature-based approach. The use of multiple stego-audio shares may also introduce practical challenges in file management and distribution. These limitations provide directions for future work.

5 Conclusion

This study evaluated an audio steganography method based on Shamir's secret-sharing scheme with various configurations. The results indicate that stego-audio quality is primarily influenced by the size of the embedded payload, where larger payloads reduce audio quality, although the values remain within an acceptable range according to the objective metrics used. Variations in the (n, k) parameters have only a limited effect on quality while providing flexibility to enhance security, as higher threshold values (k) reduce the likelihood of unauthorized payload reconstruction. The proposed method demonstrates performance comparable to other studies, achieving the second-highest average PSNR among the compared methods. Statistical evaluation shows consistent payload reconstruction and only minor entropy variation between cover and stego-audio signals.

In addition, a detector-based steganalysis experiment shows that the stego-audio is not easily distinguishable from the cover audio under the evaluated feature-based approach. Nevertheless, practical implementation requires careful management of multiple stego-audio shares, and further improvements are needed to simplify the embedding process and reduce computational complexity. Future work may focus on optimizing the embedding process and extending the evaluation using more diverse datasets and steganalysis methods.

Acknowledgement: The authors express their sincere gratitude to all members of the Cyber Security Research Group, Net-Centric Computing (NCC) Laboratory, Department of Informatics, ITS, for their continuous support and insightful discussions.

Funding Statement: This research is funded by Institut Teknologi Sepuluh Nopember (ITS) and managed under the Strategic Research Grant (SRG) Type D Scheme (Contract No. 1665/PKS/ITS/2026).

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Mohammad Muzayyin Amrulloh and Tohari Ahmad; methodology, Mohammad Muzayyin Amrulloh, Tohari Ahmad; software, Mohammad Muzayyin Amrulloh; validation, Mohammad Muzayyin Amrulloh, Tohari Ahmad and Royyana Muslim Ijtihadie; formal analysis, Mohammad Muzayyin Amrulloh; investigation, Mohammad Muzayyin Amrulloh; resources, Tohari Ahmad, Royyana Muslim Ijtihadie; data curation, Mohammad Muzayyin Amrulloh; writing—original draft preparation, Mohammad Muzayyin Amrulloh; writing—review and editing, Tohari Ahmad and Royyana Muslim Ijtihadie; visualization, Mohammad Muzayyin Amrulloh; supervision, Tohari Ahmad and Royyana Muslim Ijtihadie; project administration, Tohari Ahmad; funding acquisition, Tohari Ahmad. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available in a public repository at https://its.id/m/stego-audio-dataset-py.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Li B. Secure reversible data hiding in images with scalable capacity. In: 2023 Asia Symposium on Image Processing (ASIP); 2023 Jun 15–17; Tianjin, China. p. 74–8. doi:10.1109/ASIP58895.2023.00020.
2. Zhou C, Jiang Z. Research on reversible ciphertext field information hiding algorithm based on data stream. In: 2021 Global Reliability and Prognostics and Health Management (PHM-Nanjing); 2021 Oct 15–17; Nanjing, China. p. 1–7. doi:10.1109/PHM-Nanjing52125.2021.9612841.
3. Hingmire A, Karulkar N, Mhatre R, Patil Y. A novel approach to audio steganography on audio input for secure communication. In: Proceedings of the 2023 8th International Conference on Communication and Electronics Systems (ICCES); 2023 Jun 1–3; Coimbatore, India. p. 534–8. doi:10.1109/ICCES57224.2023.10192611.
4. Pleshkova S, Bekiarski A. Generative audio steganography algorithm. In: Proceedings of the 2024 XXXIII International Scientific Conference Electronics (ET); 2024 Sep 17–19; Sozopol, Bulgaria. p. 1–4. doi:10.1109/ET63133.2024.10721560.
5. Dharani BA, Yashaswini B, Shyashyankha Reddy GR, Rajagopal SM. Multimodal steganography: a comparative analysis of LSB and DCT methods for image and audio data concealment. In: Proceedings of the 2024 IEEE 9th International Conference for Convergence in Technology (I2CT); 2024 Apr 5–7; Pune, India. p. 1–5. doi:10.1109/I2CT61223.2024.10543932.
6. Chen L, Wang R, Dong L, Yan D. Imperceptible adversarial audio steganography based on psychoacoustic model. Multimed Tools Appl. 2023;82(17):26451–63. doi:10.1007/s11042-023-14772-9.
7. Divyashree D, Bhanu KN, Anusha M. Secured communication for multimedia based steganography. In: Proceedings of the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC); 2020 Jul 2–4; Coimbatore, India. p. 856–61. doi:10.1109/ICESC48915.2020.9156063.
8. Shashi RK, Pavithra G, Manjunath TC. Developing a novel steganography concept of audio data into audio streams. In: Proceedings of the 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT); 2022 Jan 20–22; Tirunelveli, India. p. 229–33. doi:10.1109/ICSSIT53264.2022.9716566.

9. Hardan H, Khashan OA, Alshinwan M. Edge-based data hiding and extraction algorithm to increase payload capacity and data security. Comput Mater Contin. 2025;84(1):1681–710. doi:10.32604/cmc.2025.061659.
10. Evsutin O, Melman A, Meshcheryakov R. Digital steganography and watermarking for digital images: a review of current research directions. IEEE Access. 2020;8:166589–611. doi:10.1109/ACCESS.2020.3022779.
11. Setiadi DRIM, Rustad S, Andono PN, Shidik GF. Digital image steganography survey and investigation (goal, assessment, method, development, and dataset). Signal Process. 2023;206(3):108908. doi:10.1016/j.sigpro.2022.108908.
12. Hu K, Wang M, Ma X, Chen J, Wang X, Wang X. Learning-based image steganography and watermarking: a survey. Expert Syst Appl. 2024;249(2s):123715. doi:10.1016/j.eswa.2024.123715.
13. Padmaja TS, Basha SM. A comprehensive review on steganography techniques for text, images, and audio. In: Proceedings of the 2023 IEEE Fifth International Conference on Advances in Electronics, Computers and Communications (ICAECC); 2023 Sep 7–8; Bengaluru, India. p. 1–8. doi:10.1109/ICAECC59324.2023.10560079.
14. Liu X, Shen M, Liu J, Wu Q. Image steganography with high embedding capacity based on multi-target adversarial attack. Eng Appl Artif Intell. 2025;156(3):111341. doi:10.1016/j.engappai.2025.111341.
15. Al-Yousuf FQA, Din R. Review on secured data capabilities of cryptography, steganography, and watermarking domain. Indones J Electr Eng Comput Sci. 2020;17(2):1053. doi:10.11591/ijeecs.v17.i2.pp1053-1058.
16. Mohamed KS. Data hiding: steganography and watermarking. In: New frontiers in cryptography. Cham, Switzerland: Springer; 2020. p. 89–98. doi:10.1007/978-3-030-58996-7_5.
17. Nasr MA, El-Shafai W, El-Rabaie ESM, El-Fishawy AS, El-Hoseny HM, Abd El-Samie FE, et al. A robust audio steganography technique based on image encryption using different chaotic maps. Sci Rep. 2024;14(1):22054. doi:10.1038/s41598-024-70940-3.
18. Lin Y, Liu J, Chang C, Chang C. A puzzle matrix oriented secret sharing scheme for dual images with reversibility. Signal Process. 2025;236:110056. doi:10.1016/j.sigpro.2025.110056.
19. Benhfid A, Ameur EB, Taouil Y. Reversible steganographic method based on interpolation by bivariate linear box-spline on the three directional mesh. J King Saud Univ-Comput Inf Sci. 2020;32(7):850–9. doi:10.1016/j.jksuci.2018.09.016.
20. Hema M, Saxena K, Bhagat AK, Shah V, B. J, Rajakumari R. Space reservation technique at distributed level for hiding reversible data in the encrypted images. In: Proceedings of the 2022 International Interdisciplinary Humanitarian Conference for Sustainability (IIHC); 2022 Nov 18–19; Bengaluru, India. p. 483–7. doi:10.1109/IIHC55949.2022.10060633.
21. Konduru UR, Nagarajan AP, Sri Sai CV. An improved performance of reversible data hiding in encrypted images using decision tree algorithm. Eng Appl Artif Intell. 2024;137(4):109100. doi:10.1016/j.engappai.2024.109100.
22. Zhang X, Li C, Tian L. Advanced audio coding steganography algorithm with distortion minimization model based on audio beat. Comput Electr Eng. 2023;106(6):108580. doi:10.1016/j.compeleceng.2023.108580.
23. Wang J, Wang K. A novel audio steganography based on the segmentation of the foreground and background of audio. Comput Electr Eng. 2025;123(17):110026. doi:10.1016/j.compeleceng.2024.110026.
24. Adhiyaksa FA, Amrulloh MM, Mustaqim T, Tsaniya H, Studiawan H, Shiddiqi AM. Reversible audio data hiding using samples greatest common factor and audio interpolation. In: Proceedings of the 12th Annual Computing and Communication Workshop and Conference (CCWC); 2022 Jan 26–29; Las Vegas, NV, USA. p. 659–66. doi:10.1109/CCWC54503.2022.9720763.
25. Islamy CC, Ahmad T, Ijtihadie RM. Reversible data hiding based on histogram and prediction error for sharing secret data. Cybersecurity. 2023;6(1):12. doi:10.1186/s42400-023-00147-y.
26. Amrulloh MM, Ahmad T. Fuzzy logic and the greatest common divisor on audio-based data hiding method. Int Rev Model Simul. 2022;15(3):172. doi:10.15866/iremos.v15i3.22235.
27. Samudra Y, Ahmad T. Segmentation embedding method with modified interpolation for increasing the capacity of adaptable and reversible audio data hiding. J King Saud Univ-Comput Inf Sci. 2023;35(8):101636. doi:10.1016/j.jksuci.2023.101636.

28. Arthy PS, Kumar KS, Vinson Joshua S, Sahaana G, Kalpana R, David Neels Ponkumar D, et al. Secure audio steganography with reversible data hiding and encryption. In: Proceedings of the 2025 6th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI); 2025 Jan 7–8; Goathgaun, Nepal. p. 146–50. doi:10.1109/ICMCSI64620.2025.10883557.
29. Chetan M, Bhat PP, Shet V, Husenbhai SB, Bhat A. Audio watermarking using modified least significant bit technique. In: Proceedings of the 2021 International Conference on Circuits, Controls and Communications (CCUBE); 2021 Dec 23–24; Bangalore, India. p. 1–5. doi:10.1109/CCUBE53681.2021.9702715.
30. Debnath S, Mohapatra RK, Dash R. A robust secret data sharing through coverless video steganography based on average DC coefficient on bit plane segmentation. Comput Electr Eng. 2024;120(2):109766. doi:10.1016/j.compeleceng.2024.109766.
31. Firdaus DT, Jean De La Croix N, Ahmad T. AudioSecure: an open-source code to secure data using interpolation and multi-layering techniques within audio covers. Softw Impacts. 2024;22(4):100707. doi:10.1016/j.simpa.2024.100707.
32. Chattopadhyay AK, Saha S, Nag A, Nandi S. Secret sharing: a comprehensive survey, taxonomy and applications. Comput Sci Rev. 2024;51(2):100608. doi:10.1016/j.cosrev.2023.100608.
33. Hamdi AA, Eyssa AA, Abdalla MI, ElAffendi M, AlQahtani AAS, Ateya AA, et al. Improving audio steganography transmission over various wireless channels. J Sens Actuator Netw. 2025;14(6):106. doi:10.3390/jsan14060106.
34. Garno G, Rizal A, Solehudin A, Ekstanza R, Yusup D. Comparison of steganography using the discrete cosine transform method on image based bilinear, nearest neighbor and spline interpolation. JUITA J Inform. 2021;9(1):9. doi:10.30595/juita.v9i1.7302.
35. Chouhan V, Arora A. Blockchain-based secure and transparent election and vote counting mechanism using secret sharing scheme. J Ambient Intell Human Comput. 2022;14(10):14009–27. doi:10.1007/s12652-022-04108-0.
36. Wang N, Fu J, Zhang S, Zhang Z, Qiao J, Liu J, et al. Secure and distributed IoT data storage in clouds based on secret sharing and collaborative blockchain. IEEE/ACM Trans Netw. 2023;31(4):1550–65. doi:10.1109/TNET.2022.3218933.
37. Iwamura K, Kamal AAAM. Improving the security of asymmetric secret sharing scheme and its new applications. J Inf Secur Appl. 2025;93(6):104098. doi:10.1016/j.jisa.2025.104098.
38. Bosch JJ, Fuhrmann F, Herrera P. IRMAS: a dataset for instrument recognition in musical audio signals. Zenodo. 2014. doi:10.5281/zenodo.1290750.
39. Ito K, Johnson L. The LJ speech dataset. 2017 [cited 2026 Mar 14]. Available from: https://www.kaggle.com/datasets/mathurinache/the-lj-speech-dataset.
40. PejaŚ J, Cierocki L. Reversible data hiding scheme for images using gray code pixel value optimization. Procedia Comput Sci. 2021;192(4):328–37. doi:10.1016/j.procs.2021.08.034.
41. Ahmad T, Amrizal MH, Wibisono W, Ijtihadie RM. Hiding data in audio files: a smoothing-based approach to improve the quality of the stego audio. Heliyon. 2020;6(3):e03464. doi:10.1016/j.heliyon.2020.e03464.
42. Jung K, Yoo K. Data hiding method using image interpolation. Comput Stand Interfaces. 2009;31(2):465–70. doi:10.1016/j.csi.2008.06.001.
43. Bobeica A, Dragoi IC, Caciula I, Coltuc D, Albu F, Yang F. Capacity control for prediction error expansion based audio reversible data hiding. In: Proceedings of the 2018 22nd International Conference on System Theory, Control and Computing (ICSTCC); 2018 Oct 10–12; Sinaia, Romania. p. 810–5. doi:10.1109/ICSTCC.2018.8540672.
44. Martyniuk H, Kozlovskiy V, Meleshko T, Sorokun A. Method of finding cover signal for audio steganalysis calibrated methods. In: Proceedings of the 2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS); 2021 Sep 22–25; Cracow, Poland. p. 1095–100. doi:10.1109/IDAACS53288.2021.9661059.
45. Lakshmi VS, Deepthi S, Deepthi PP. Collusion resistant secret sharing scheme for secure data storage and processing over cloud. J Inf Secur Appl. 2021;60(9):102869. doi:10.1016/j.jisa.2021.102869.
46. Setiadi DRIM. PSNR vs. SSIM: imperceptibility quality assessment for image steganography. Multimed Tools Appl. 2020;80(6):8423–44. doi:10.1007/s11042-020-10035-z.
47. Ghoul S, Sulaiman R, Shukur Z. A review on security techniques in image steganography. Int J Adv Comput Sci Appl. 2023;14(6):361–85. doi:10.14569/IJACSA.2023.0140640.