Securing Audio with 3D-Chaotic Map Based Hybrid Encryption Technique

Sangram Pati[1], Minati Mishra[2] and Jayanti Rout[2,3]

[1]*Range Safety Division, PXE (DRDO), Chandipur, Balasore, Odisha, INDIA, Email: sangrampati64@gmail.com*
[2]*P.G. Department of CS, Fakir Mohan University, Balasore, Odisha, Email: minatiminu@gmail.com*
[3]*Department of CSE, C.V. Raman Global University, Bhubaneswar, Odisha*

Abstract: In today's digital life, the use of digital data is growing at a faster rate due to the introduction of evolving and progressive technologies. Currently, many tasks such as shopping, communication, education, research, etc are being performed online using Internet which may not be a secure platform for all these tasks, all the time. Security being a key requirement of data communication, developing a cryptosystem that satisfies the CIA (Confidentiality, Integrity, Availability) triad to protect sensitive user information from unauthorized users is a challenging research problem. This paper suggests an audio encryption crypto model that provides a high degree of security to audio data. The novelties of this work are, use of three different 3D Fibonacci-Lucas maps in succession for strengthening robustness of the the proposed cryptosystem against various attacks. The uses of Latin and magic squares enhances the level of confusion and diffusion. Latin square helps in nullifying the histogram analysis attack by equalising the histogram and magic square in improving the spectrogram of the encrypted media. This cryptosystem is suitable for both mono and stereo audio files and the strengths of this cryptosystem are confirmed through different testing parameter analyses, including low correlation (-0.0002), high NSCR (99.9996%), low SSNR (-20.9298dB), and low PSNR (+1.4448dB).

Keywords: Encryption, Fibonacci-Lucas transformation, Cryptosystem, Chaos, Audio encryption

1. INTRODUCTION

Before a few decades, cryptographic techniques were mostly focused on text data to secure sensitive information conveyed through emails, documents, and financial transactions. This was attributed to the characteristics of digital communication during that period, which predominantly depended on the interchange of text-based data. Nevertheless, due to the rapid advancement of technology, Internet, and the introduction of fast broadband and mobile networks, there has been a notable change in the nature of data that requires safeguarding. Currently, a significant proportion of data carried over the Internet constitute multimedia data such as encompasses images, audios and videos. The widespread use of social media platforms, online streaming services, and cloud storage has led to a rapid increase in the amount of multimedia content being shared. As a result, it has become necessary to use modern encryption techniques to protect these multimedia contents from unauthorized access. Consequently, contemporary cryptographic techniques have developed to tackle the intricacies of safeguarding multimedia data, guaranteeing the preservation of privacy and integrity in a time when data breaches and cyber threats are growing more advanced. This trend highlights the significance of strong cryptographic solutions that address not only conventional text-based data but also the varied and extensive multimedia data that has become an essential component of our digital existence.

Audio Cryptography is a technique or method where extensive algorithms are applied to audio signals. An audio signal is a representation of sound. Audio signals are available in the form of digital and analog signals. Analog signals and digital signals are occurs in electrical signal and binary representations, respectively. The lower limit frequency of audio signal is 20 Hz, whereas the upper limit frequency for our ear is 20,000 Hz or 20 kHz. To propagate secure audio or voice communication between two or more peoples for real time applications like: In online business meetings voice talk between clients and investors, In online education system voice talk between students and teacher, CBI officials, Defense and Intelligence Bureau officials etc. are more require the cryptographic algorithm of audio or voice for top secret communication. An audio file contains

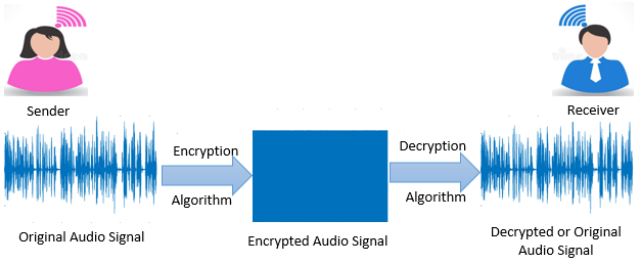

Figure 1. General Framework of Audio Cryptography

E-mail address: sangrampati64@gmail.com, minatiminu@gmail.com, routjayanti2k19@gmail.com

the sample rate or frequency, number of channels either mono or stereo, total samples or audio data, compression method (if used), and bit rate. For developing and designing a secure and fast cryptosystem for audio or voice encryption only total samples of an audio file are required. The overall process of an audio signal encryption is shown in figure 1. Initially, a sender separate frequency from audio signals and apply the encryption algorithm to the audio data or total samples of audio signal. The encrypted audio can now be sent using any insecure medium. To obtain back original audio, the receiver uses decryption algorithm and decrypts the received signal..

To develop this model of audio crypto system, we have conducted a thorough review of literature, considered the evaluation metrics, attack models, quality estimators, and efficiency measures such as Key space analysis, the avalanche effect, perceptual property, key sensitivity, differential attacks, and Statistical attacks, Sample Signal to Noise Ratio and Peak Signal to Noise Ratio, encryption and decryption times etc.

The following section gives a brief record of various cryptographic models developed and designed, from time to time, by different researchers and scientists for multimedia encryption.

2. LITERATURE REVIEW

Chaotic mapping methods have remained an important area of study in multimedia encryption. Many chaotic maps are used for developing secure crypto systems for multimedia encryption. A circle map is one such chaotic map that is used by K. Kordov along with PRNG to develope a symmetric cryptosystem [1]. According to the author, for the high level of cryptographic security, the use of one chaotic map is not enough therefore, the author in this paper, has employed two chaotic maps to strengthen the key space. The author has experimentally shown through spectrogram analysis that the frequency of the plain audio signal is destroyed after encryption. The algorithm ensures high quality of encryption by correlation NSCR values approaching zero and 100% respectively. A key space of 2^{149} is good enough to resist brute-force attack. Based on a circle map an audio encryption algorithm has been proposed by K. Kordov and L. Bonchev in [2]. According to the investigation results and discussions made by the authors, waveform analysis of encrypted audio is totally different from the original one. The level of correlation between original and encrypted audio files is close to zero which indicates the resistance statistical attack.

S. Adhikari and S. Karforma have used the 2D Henon-1D Tent chaotic maps to generate a random number sequence and using this sequence as a symmetric key an audio cryptosystem has been proposed in [3]. Here on the sender side, an XOR operation is performed between the original audio file and random number sequence. On the other hand, using the same random number sequence performed an XOR operation with an encrypted file to obtain the decrypted audio file. The authors have experimentally proved that the encryption or decryption time of their method is 0.004 sec which is very less as compared to other existing methods. One more advantage of this model is spectrogram and histogram of the encrypted audio file are uniformly distributed which indicates the robustness of the encryption algorithm. Keyspace is $2^{249.1446}$. The suggested method is resistant to brute-force, differential, and statistical attacks.

Arnold Cat Map (ACM) has been extensively used for the purpose of image scrambling as well as for voice encryption [4], [5], [6]. For image processing techniques including cryptography, steganography, and watermarking M. Mishra et al. have suggested a series of 2×2 periodic maps based on the Fibonacci and Lucas series [7]. The authors have experimentally proved that the map with higher periodicity gives better security. If we know the periodicity of a unimodular map, we can easily select a map with higher periodicity for designing and developing a better cryptosystem for multimedia encryption. However, the periodicity of larger images or larger modulo does not always give higher values. To solve the minimal period $\Pi(N)$ problem of the Arnold cat map, F. J. Dyson and H. Falk have given the following formulae for ACM $k = 1, 2, 3, ...$ is [8].

$$\Pi(N) = \begin{cases} 3N, & \text{if } N = 2 \times 5^k \\ 2N, & \text{if } N = 5^k \text{ or } N = 6 \times 5^k \\ \leq \frac{12N}{7}, & \text{otherwise} \end{cases} \quad (1)$$

In [9], J. Bao and Q. Yang have suggested an algorithm to find minimal period $\Pi(N)$ of the unimodular map and its modulo N. However, two properties such as ghost and miniature for the ACM before reaching the end of the period have been discovered by B. EHRHARD in [10]. In ACM, when the values of all the elements of An(mod N) is absolute for i, $j = 1, 2, ...N - 1$ then miniatures occurred. When N is composite, most probably ghosts occurred. F. Svanstrom has explained four properties such as periodic, upside-down, ghost, and miniature of PCM (Pell's Cat Map) which is the one orientation of ACM whose determinant is -1 in [11]. The author analyzed $\Pi(N)$ of PCM and ACM are not the same and scrambling patterns are also different. In this paper, the author has formulated the PCM for k = 1, 2, 3, . . . is

$$\Pi(N) = \begin{cases} \frac{8N}{3}, & \text{if } N = 3^k \\ \frac{12N}{3}, & \text{if } N = 5^k \\ \leq \frac{24N}{11}, & \text{otherwise} \end{cases} \quad (2)$$

X. Wang and Y. Su have suggested a technique for encrypting mono or stereo channel audio using DNA coding and PWLCM (Piecewise Linear Chaotic Map) systems[12], [13]. According to the authors, "the values of logistic chaotic maps are less balanced and not evenly distributed". To overcome this problem, they have adopted the PWLCM system for generating required random sequences to balance and distribute the values. The authors have experimentally proved that their proposed algorithm is applicable for any type of voice signal encryption and is resistant to differen-

tial, statistical, and brute-force attacks with a keyspace of 2^{256}.

In [14], the authors present a new audio encryption framework that uses chaos theory and DNA encoding to overcome the limitations of traditional encryption algorithms. The symmetric encryption approach embeds chaotic properties into the encryption process, ensuring robust security while maintaining high audio fidelity. The framework includes a lightweight pseudorandom bit generator and a new scrambling algorithm. Performance metrics show a 98.28% NSCR and 33.63% UACI, demonstrating resilience against cryptographic attacks. This framework is suitable for real-time applications and secure audio communication. Maity and Dhara presents an advanced audio encryption method using SHA3-512 to compute a large 512-bit key for various parameters. It introduces a 2D Cosine Logistic Map (2DCLM) that works effectively under chaotic conditions. The method scrambles the audio signal using the hash value and decomposes it with Empirical Mode Decomposition (EMD), optimizing the process. The residuals from EMD are XOR-ed with the 2DCLM stream, and the final encrypted signal is converted back to 1D [15].

Organisation of the rest of this paper is as follows: Methods used for developing the proposed crypto model are discussed in section 3. In section 4 are discussed the testing measures used to evaluate the model. The proposed model and the experimental setups are dealt in sections 5. Section 6 deals with the results an analysis of the experiments conducted and finally, the paper is concluded with concluding remarks in section 7.

3. MATERIALS AND METHODS

In this research, We have suggested audio encryption-decryption algorithms using a series of three 3D chaotic maps and Latin and magic squares. This section explains the procedures followed in the algorithms. A detailed introduction of all the quality parameters used to check the performance of the proposed model are also given in this section.

A. 3D Unimodular Map

For my experiment I have taken three different 2D Fibonacci-Lucas Map, such as $(F11L)_2$, $(F31L)_1$, and $(F32L)_1$. In this section we have discussed how a 2D map can be extended to three dimensions(3D) map.

$$A_{2D} = \begin{bmatrix} 1 & 2 \\ 1 & 3 \end{bmatrix}$$

Step 1: Obtain the first map by fixing X axis

$$A^x_{3D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & 1 & 3 \end{bmatrix}$$

Step 2: Obtain the 2nd map by fixing the Y axis

$$A^y_{3D} = \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 1 & 0 & 2 \end{bmatrix}$$

Step 3: Obtain the 3rd map by fixing the Z axis

$$A^z_{3D} = \begin{bmatrix} 1 & 2 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The 3D unimodular map is obtained by multiplying the three basis maps. That means

$$A_{3D} = A^x_{3D} \times A^y_{3D} \times A^z_{3D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \\ 0 & 1 & 3 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 1 & 0 & 2 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 0 \\ 1 & 3 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 5 & 13 \\ 1 & 6 & 15 \\ 1 & 2 & 6 \end{bmatrix}$$

B. Magic Square

A magic square is a square array, consisting of the distinct positive integers $1, 2, 3, \ldots, n^2$, where n is the size of the magic square. In magic square all positive integers are arranged in such a way that in horizontal, vertical, and diagonal directions the sum of numbers is always same. Here n is the size of the magic square matrix, n^2 elements are are arranged in magic square. The sum of all element in each direction is same and constant value so that the value is known as the magic constant. Using equation 3 the value of magic constant can be computed.

$$M_2(n) = \frac{1}{n}\sum_{k=1}^{n^2} k = \frac{1}{2}n(n^2+1) \tag{3}$$

Example, if n=3, a magic square consists of elements

TABLE I. Magic square consists of elements from $1, 2, 3, ...9$

8	1	6
3	5	7
4	9	2

from $1, 2, 3, ...n^2$ which is shown in table I. According to equation 3, the magic constant is

$$M_2(3) = \frac{1}{3}\sum_{k=1}^{3^2} k = \frac{1}{2}3(3^2+1) = 15$$

C. Latin Square

A Latin square is an $S \times S$ matrix filled with different numbers from set $S = \{1, 2, 3, 4, \ldots S\}$, where each number precisely occurs once in each row and column respectively. There is no repetition of numbers in the same row and column. A simple example of a 3×3 Latin square matrix is given in equation 4.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{bmatrix}_{3\times3} \tag{4}$$

In the proposed method initially, according to audio size after converting to a 2D form, if the size of the 2D audio is $\leq$256, a latin square matrix of size 256×256 or $size\ of\ 2D\ audio \times size\ of\ 2D\ audio$ is used. Otherwise,

a tiled Latin square according to size of 2D audio is considered. An example of tiled Latin square B of size 6x6 from the Latin matrix A of equation 4 is given in equation 5. For example,

$$B = \begin{bmatrix} A & A \\ A & A \end{bmatrix}_{6\times6} = \begin{bmatrix} 1 & 2 & 3 & 1 & 2 & 3 \\ 2 & 3 & 1 & 2 & 3 & 1 \\ 3 & 1 & 2 & 3 & 1 & 2 \\ 1 & 2 & 3 & 1 & 2 & 3 \\ 2 & 3 & 1 & 2 & 3 & 1 \\ 3 & 1 & 2 & 3 & 1 & 2 \end{bmatrix}_{6\times6} \tag{5}$$

In equation 6, is given a titled Latin square TL of size 512×512 from a Latin square LS of size 256×256 is given in equation 5.

$$TL = \begin{bmatrix} LS & LS \\ LS & LS \end{bmatrix}_{512\times512} \tag{6}$$

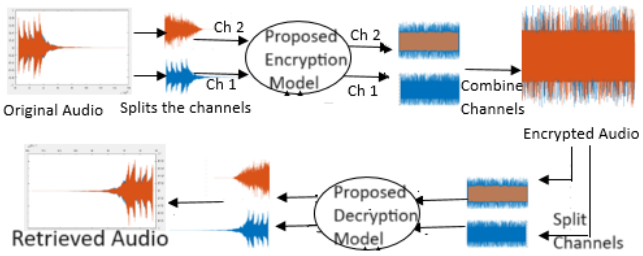

Figure 2. Block diagram of the Proposed Cryptosystem

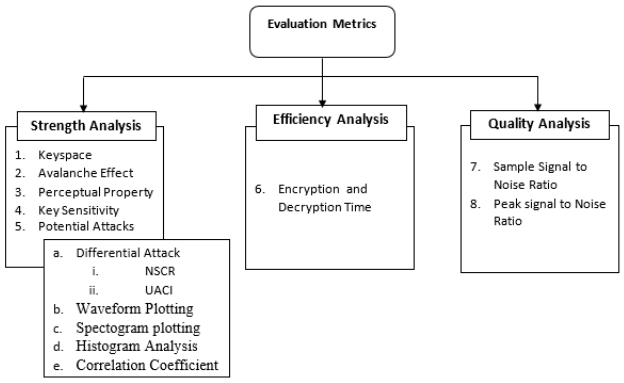

Figure 3. Test Measures

4. Evaluation Measures

The tests mentioned in figure 3 are conducted to compare strength of the suggested model. According to S. Lian et al. [16], any encryption system that withstand an attack for an hour can be considered secure for cryptographic applications. Security analysis for measuring the strength of any encryption mechanism in terms of cryptographic attacks usually includes keyspace, key sensitivity, perceptual security, and its resistance to potential attacks.

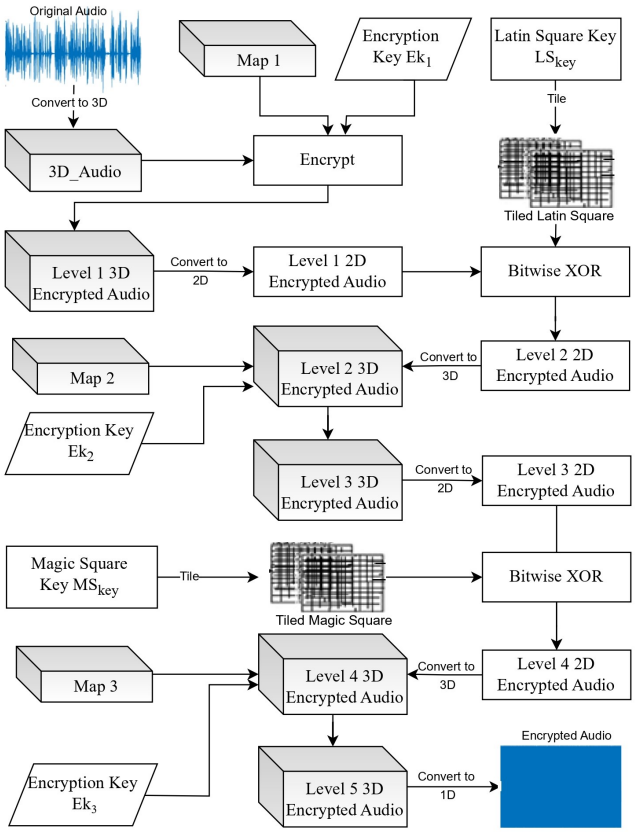

Figure 4. Block diagram of the Encryption Process for a Single Channel

1) **Key space**: In each cryptosystem, key is one of the most vital components. A secure cryptographic algorithm is incomplete without a significant key. The key space gives total number of possible keys. If K_i is a key of an algorithm, and the size of the key is 16bits, then the key will own a keyspace of 2^{16}. Given IEEE Standard for Floating-Point the precision of 64bits double variables is about 10^{-15} . A good cryptosystem would have large keyspace to resist brute-force attack.
2) **Avalanche Effect(AE)**: It is used to calculate the diffusion effect and it can be computed using equation (7).

$$AE = D(C1, C2)/N(\%) \tag{7}$$

Where, C1 and C2 are the cipher media corresponding to similar input media P1 and P2. D(C1, C2) is the hamming distance between C1 and C2. N represents the number of bits in the cipher media. A good encryption algorithm must have $AE > 50\%$. That means, if one bit is alter in the plaintext or secret key there should be alteration of at least half of the bits in the ciphertext. In other words, minimal alteration in the plaintext (input message) should produce a maximal change in the ciphertext (output message).

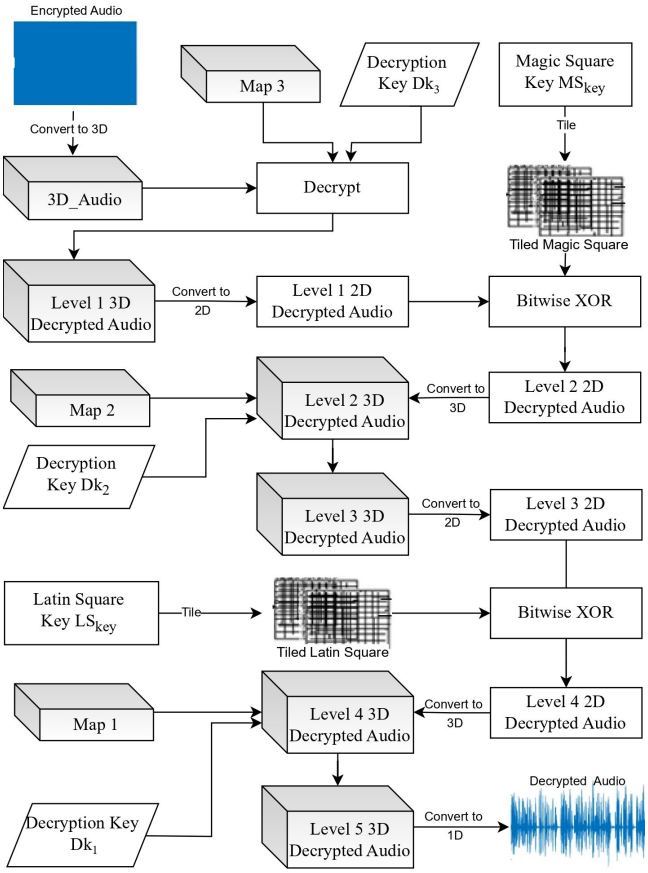

Figure 5. Block diagram of Decryption Process for a Single Channel

3) **Perceptual Property** : After encryption, if the encrypted media is not recognizable through human perception then the cryptographic system is considered to be confident in terms of perception.
4) **Key Sensitivity**: It is defined as the sensitivity of the secret key. A single bit modification in the secret must fail the decryption of the sensitive information.
5) **Potential Attacks**: A good encryption mechanism must withstand the attacks listed below.
   a) *Differential Attack*: The common measures used to examine the the resilance of a system against differential attacks are:
      i) Number of Sample Change Rate (NSCR): This test is used to examine the strength of the encryption algorithm and is calculated using Equation 8.

$$NSCR = \frac{\sum_{i=1}^{S} R_i}{S} \times 100(\%) \qquad (8)$$

Where,

$$R_i = \begin{cases} 1, x_i \neq y_i \\ 0, otherwise \end{cases} \qquad (9)$$

Where, S is number of samples. x_i and y_i are sample values of plain and encrypted media respectively. NSCR value of a good encryption algorithm is close to 100%, which indicates the security level of the encryption algorithm is high.

      ii) UACI: UACI is explained as the average intensity of difference between both files(original file and encrypted file). The UACI value can be mathematically obtained by using Equation 10.

$$UACI = \frac{1}{S}[\sum_{i=1}^{S} \frac{a_i - b_i}{2^k - 1}] \times 100 \qquad (10)$$

Where, S is the total number of samples and k bits are used to represent the media. a_i and b_i are encrypted audio file of plain and modified images respectively. For a strong encryption technique the UACI value must be closed to 33.3%.

b) *Waveform Plotting*: Waveform Plotting plots amplitude of audio signal against time. The waveform plots of some of our test audios are given in figure 6 which shows that the wave forms of the encrypted audios are completely different from those of the actual audios.
c) *Spectogram Plotting*: Another important approach for analyzing audio signals is spectogram plotting. It is 2D(Two Dimensional) graph with 3^{rd} dimension represented by different colors. In spectogram plotting, the variation of spectrum frequency of a signal against time is plotted to compare the plain and encrypted media.
d) *Histogram Analysis*: Histogram analysis is a general tool for analyzing the distribution of data values. If histogram of encrypted media is uniformly distributed, the algorithm is strong technique for audio encryption. The closed bins indicate resistance against statistical attacks.
e) *Correlation Coefficient*: This shows the interdependence of the plain and encrypted media. The level of correlation between two audio files is determined by calculating the correlation coefficient and the range is always in between -1 and 1. Correlation values between $\pm 0.7 and \pm 1$ represents strong positive or negative correlation which specifies samples of plain audio files are similar to encrypted audio files. The values in the range $\pm 0.3 and \pm 0.7$ are considered as medium positive or negative correlation, and the values in the range 0 $and \pm 0.3$ are considered as weak positive or negative correlation[17]. The correlation coefficient of a good and high-quality encryption algorithm approach zero to indicate no dependency between the plain and encrypted media. It can

be calculated using the following equation:

$$C_{mn} = \frac{Cov(m,n)}{\sqrt{D(m)}\sqrt{D(n)}} \tag{11}$$

Where,

$$D(m) = \frac{1}{S}\sum_{i=1}^{S}(m_i - \overline{m})^2 \tag{12}$$

$$D(n) = \frac{1}{S}\sum_{i=1}^{S}(n_i - \overline{n})^2 \tag{13}$$

$$Cov(m,n) = \sum_{i=1}^{S}(m_i - \overline{m})(n_i - \overline{n}) \tag{14}$$

S=number of samples
m_i, n_i = sample values
$\overline{m}$, $\overline{n}$ = mean sample values, of the plain and encrypted signals, respectively.
Cov(m,n)= covariance between plain and encrypted audio files.

6) **Encryption and Decryption Time**: It gives the amount of CPU time required for encrypting the original medium and decrypting back the ciphermedia to the original media.
7) **Sample Signal to Noise Ratio(SSNR)** : This is one of the widely used tests for determining the quality of signals. SSNR greater than 0 dB (Decibel) indicates the clarity of a signal. It can be computed as follows:

$$SSNR = 10log_{10}\frac{\sum_{i=1}^{S} x_i^2}{\sum_{i=1}^{S}[x_i - y_i]}(d\beta) \tag{15}$$

S= number of samples
x_i, y_i =the sample values of the plain audio file and encrypted audio file are respectively.
Negative SSNR values indicate then encrypted audio files are very noisy and the adopted cryptographic algorithm is destroy the clear signal from the plain audio file
8) **Peak Signal to Noise Ratio(PSNR)**: This test is used to measure the power of clear signal against the power of noise. It is more applicable for image encryption algorithm, but it can be also used for testing the quality of audio encryption algorithm. PSNR is computed as follows:

$$PSNR = 10log_{10}\frac{MAX^2}{MSE}(d\beta) \tag{16}$$

Where, MAX is the maximum possible bits of the audio stream (if the audio stream is a 16bit audio then, MAX^2 is 65,535). MSE is the Mean Squared Error between the plain and encrypted audio files, which can be defined as:

$$MSE = \frac{1}{S}\sum_{i=1}^{S}(x_i - y_i)^2 \tag{17}$$

Algorithm 1 Encryption Algorithm

Require: $Org\_Audio, Map1, Map2, Map3, Ek_1, Ek_2, Ek_3$
Ensure: Enc_audio(*Encrypted Audio*)
Latin_matrix ← Create a Latin square matrix of size 256×256.
Tiled_Latin_matrix ← Construct a tiled Latin square matrix of size 512×512 using equation 5
Magic_matrix ← Create a Magic square matrix of size 256×256.
if (the Original Audio is a Stereo Audio) **then**
    Split Channels
end if
for (Each Channel) **do**
    3D_Audio ← Convert-to-3D(Org_Audio)
    3D-Audio-Enc1 ← Scramble(3D_Audio, Map1, Ek_1)
    Audio-Enc1 ← Reshape to 2D(3D-Audio-Enc1)
    Audio-Enc2 ← XOR(Audio-Enc1, Tiled_Latin_matrix)
    3D-Audio-Enc2 ← Reshape to 3D(Audio-Enc2)
    3D-Audio-Enc3 ← Scramble(3D-Audio-Enc2, Map2, Ek_2)
    Audio-Enc3 ← Reshape to 2D(3D-Audio-Enc3)
    Audio-Enc4 ← XOR(Audio-Enc3, Tiled_Magic_matrix)
    3D-Audio-Enc4 ← Reshape to 3D(Audio-Enc4)
    3D-Audio-Enc5 ← Scramble(3D-Audio-Enc4, Map3, Ek_2).
    Enc_Audio ← Convert to 1D(3D-Audio-Enc5).
end for
Combine the Channels to get the final encrypted audio.

5. PROPOSED CRYPTOSYSTEM

The Pseudo-codes of encryption and decryption algorithms of the proposed cryptosystem are presented in algorithms 1, 2 and flow diagrams for each channel of the audio in figures 4 and 5 respectively. The overall process of the proposed model is depicted in figure 2.

A. EXPERIMENTAL SETUP

All the experiments have been executed in a laptop with Intel(R) Core(TM) i3-5005U CPU @ 2.00GHz, 8.00GB RAM and Windows 10 64-bit operating system.

The experiments have been conducted on a set of 25 self recorded audios out of which ten audios and their experimental results are given in table II. The Encryption algorithm 1 have been applied to all the recorded audios to enhance the level of strength, efficiency, and quality. The algorithms and the results have been evaluated using the evaluation matrices discussed in section 4.

6. RESULT ANALYSIS AND DISCUSSION

From figure 6, it can be concluded that there is a strong difference between the waveform plotting of the original and the encrypted files that means encrypted audio files are fully encrypted. It can be seen in figure 8 that the

TABLE II. Experimental results of 10 test audios and comparison with 5 existing methods.

Method	Test Audio	Correlation Coefficient	PSNR (in dB)	SSNR (in dB)	NSCR (in %)
[1]	file2	0.0016	4.3909	-10.6478	99.9960
[3]	A-eng-f7	0.0202	4.2145	-22.0270	99.9995
[18]	Male sound	-0.0028	-	-	99.9972
[15]	Own	0.0001	-	-	99.9982
[19]	audio4	0.0027	-	-	99.6070
Proposed	1	-0.0039	2.4396	-16.8430	**99.9996**
	2	-0.0005	2.8616	-17.0742	99.9992
	3	0.0010	2.3914	-17.2176	99.9984
	4	**-0.0002**	2.3003	-18.3677	99.9980
	5	0.0008	**1.4448**	**-20.929**8	99.9973
	6	-0.0013	1.9391	-18.8222	99.9992
	7	0.0003	1.5263	-20.7650	99.9980
	8	-0.0004	2.0525	-20.0463	99.9992
	9	-0.0007	2.1782	-18.6840	99.9980
	10	0.0013	2.5722	-17.2699	99.9980

flat color of the spectrogram and frequency of the plain audio file is destroyed in the encrypted audio file indicates the encrypted audios are completely noisy. From figure 7, it can be seen that encrypted audio files are uniformly distributed which can be withstand statistical attack. This proposed cryptosystem provides low correlation (-0.0002), high NSCR (99.9996%), low SSNR ($-20.9298dB$), and low PSNR($+1.4448dB$).

7. CONCLUSIONS

In real life, the use of digital data is growing faster due to the development of progressive technologies and securing all these digital data is of Paramount importance in Information Processing and Communication.

This paper suggests an audio encryption model which secures audio signals from various security attacks. The novelty of this work is the use of three different 3D Fibonacci-Lucas maps in a series. To improve the strength, and efficiency of the proposed encryption, we have implemented tiled Latin square and magic square matrices. The histogram and spectrogram analyses prove that the audio information remains completely concealed to unauthorised accesses. The strength and quality of this cryptosystem is confirmed from the testing parameters such as, low correlation (-0.0002), high NSCR (99.9996%), low SSNR (-20.9298dB), and low PSNR (+1.4448dB). This cryptosystem is suitable for mono and stereo audio files. This proposed cryptosystem ensures high quality of encryption by correlation value closed to zero and NSCR value closed to 100%. Near zero PSNR and negative SSNR values establish the high chaotic nature of the encrypted audio file making the original signals undetectable from the encrypted signals.

REFERENCES

[1] K. Kordov, "A novel audio encryption algorithm with permutation-substitution architecture," *Electronics*, vol. 8, no. 5, p. 530, 2019.

[2] K. Kordov and L. Bonchev, "Using circle map for audio encryption algorithm," *Mathematical and Software Engineering*, vol. 3, no. 2, pp. 183–189, 2017.

[3] S. Adhikari and S. Karforma, "A novel audio encryption method using henon–tent chaotic pseudo random number sequence," *International Journal of Information Technology*, vol. 13, no. 4, pp. 1463–1471, 2021.

[4] M. F. A. Elzaher, M. Shalaby, and S. H. El Ramly, "An arnold cat map-based chaotic approach for securing voice communication," in *Proceedings of the 10th International Conference on Informatics and Systems*, 2016, pp. 329–331.

[5] P. Sankhe, S. Pimple, S. Singh, and A. Lahane, "An image cryptography using henon map and arnold cat map," *Int. Res. J. Eng. Technol*, vol. 5, no. 4, pp. 1900–1904, 2018.

[6] F. Merazka, "Wideband speech encryption based arnold cat map for amr-wb g. 722.2 codec," in *Image and Signal Processing: 6th International Conference, ICISP 2014, Cherbourg, France, June 30–July 2, 2014. Proceedings 6*. Springer, 2014, pp. 658–664.

[7] M. Mishra, P. Mishra, M. Adhikary, and S. Kumar, "Image encryption using fibonacci-lucas transformation," *arXiv preprint arXiv:1210.5912*, 2012.

[8] F. J. Dyson and H. Falk, "Period of a discrete cat mapping," *The American Mathematical Monthly*, vol. 99, no. 7, pp. 603–614, 1992.

[9] J. Bao and Q. Yang, "Period of the discrete arnold cat map and general cat map," *Nonlinear Dynamics*, vol. 70, pp. 1365–1375, 2012.

[10] E. Behrends, "The ghosts of the cat," *Ergodic Theory and Dynamical Systems*, vol. 18, no. 2, pp. 321–330, 1998.

[11] F. Svanström, "Properties of a generalized arnold's discrete cat map," 2014.

[12] P. K. Naskar, S. Bhattacharyya, and A. Chaudhuri, "An audio encryption based on distinct key blocks along with pwlcm and eca," *Nonlinear Dynamics*, vol. 103, pp. 2019–2042, 2021.

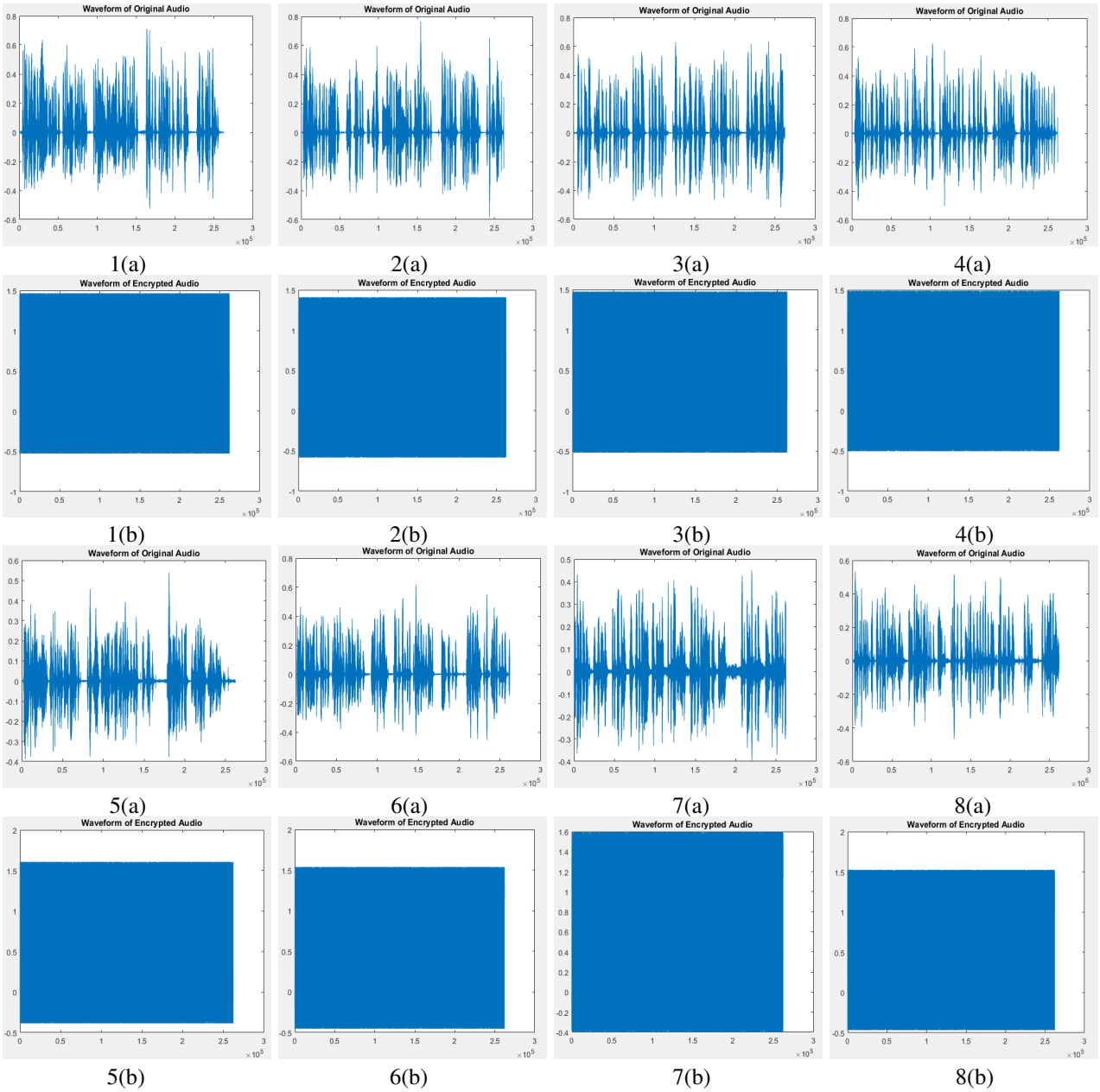

Figure 6. (1.a), (2.a), (3.a), (4.a), (5.a), (6.a), (7.a), and (8.a) are the waveform plottings of 8 original audios, their encrypted waveform plottings are (1.b), (2.b), (3.b), (4.b), (5.b), (6.b), (7.b), and (8.b) respectively.

[13] X. Wang and Y. Su, "An audio encryption algorithm based on dna coding and chaotic system," *IEEE Access*, vol. 8, pp. 9260–9270, 2019.

[14] M. Roy, S. Chakraborty, and K. Mali, "Audio encryption framework based on chaotic map and dna encoding," *Applied Acoustics*, vol. 224, p. 110152, 2024.

[15] A. Maity and B. C. Dhara, "An audio encryption scheme based on empirical mode decomposition and 2d cosine logistic map," *IEEE Latin America Transactions*, vol. 22, no. 4, pp. 267–275, 2024.

[16] S. Lian, J. Sun, G. Liu, and Z. Wang, "Efficient video encryption scheme based on advanced video coding," *Multimedia Tools and Applications*, vol. 38, pp. 75–89, 2008.

[17] K. Kordov, "A novel audio encryption algorithm with permutation-substitution architecture," *Electronics*, vol. 8, no. 5, p. 530, 2019.

[18] D. Shah, T. Shah, and S. S. Jamal, "Digital audio signals encryption by mobius transformation and hénon map," *Multimedia systems*, vol. 26, no. 2, pp. 235–245, 2020.

[19] M. Demirtas, "A bit-level audio encryption algorithm using a new

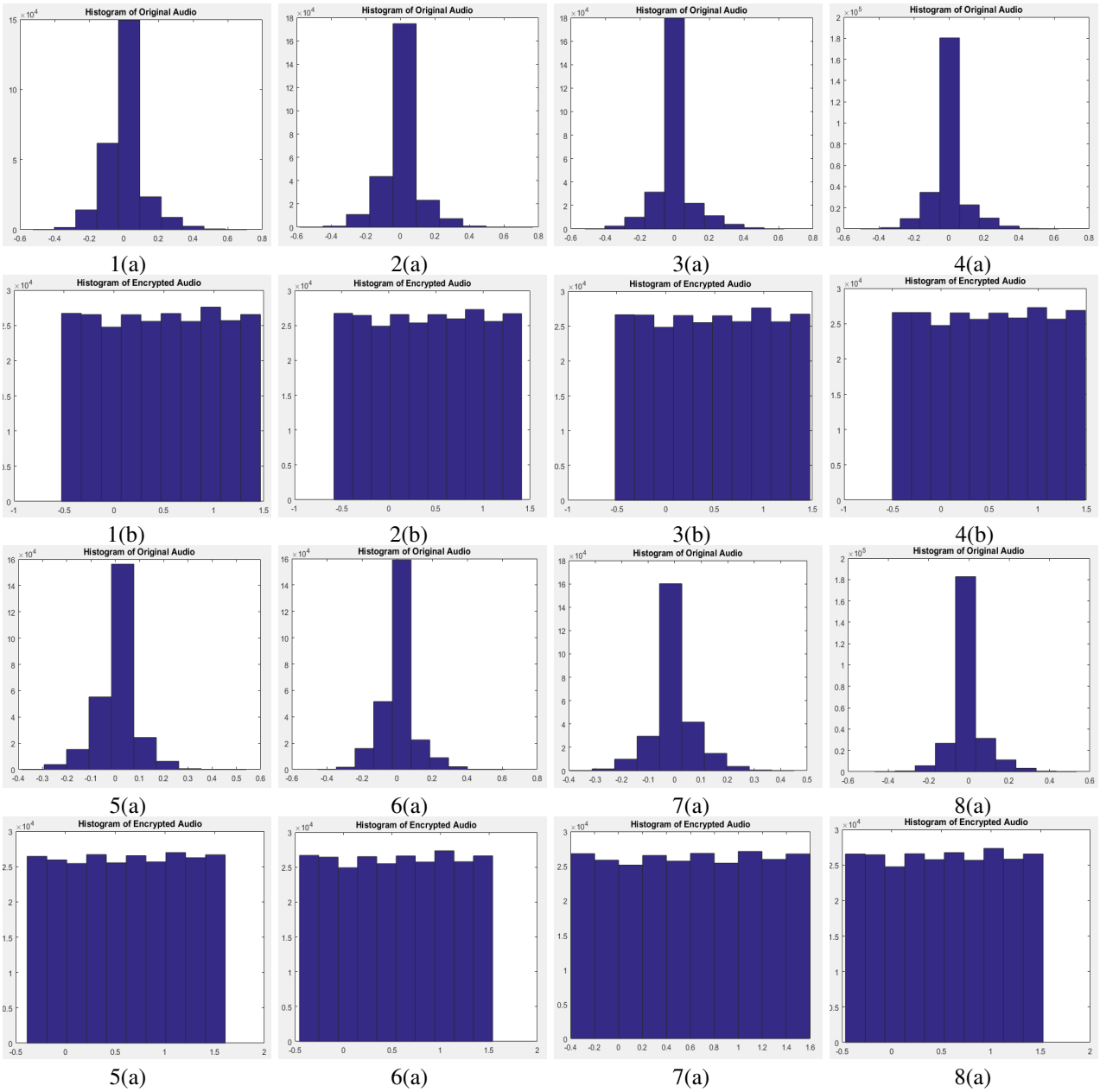

Figure 7. (1.a), (2.a), (3.a), (4.a), (5.a), (6.a), (7.a), and (8.a) are the histogram plottings of 8 original audios, their encrypted histogram plottings are (1.b), (2.b), (3.b), (4.b), (5.b), (6.b), (7.b), and (8.b)respectively.

hybrid chaotic map," *Kahramanmaraş Sütçü İmam Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 27, no. 2, pp. 601–614, 2024.

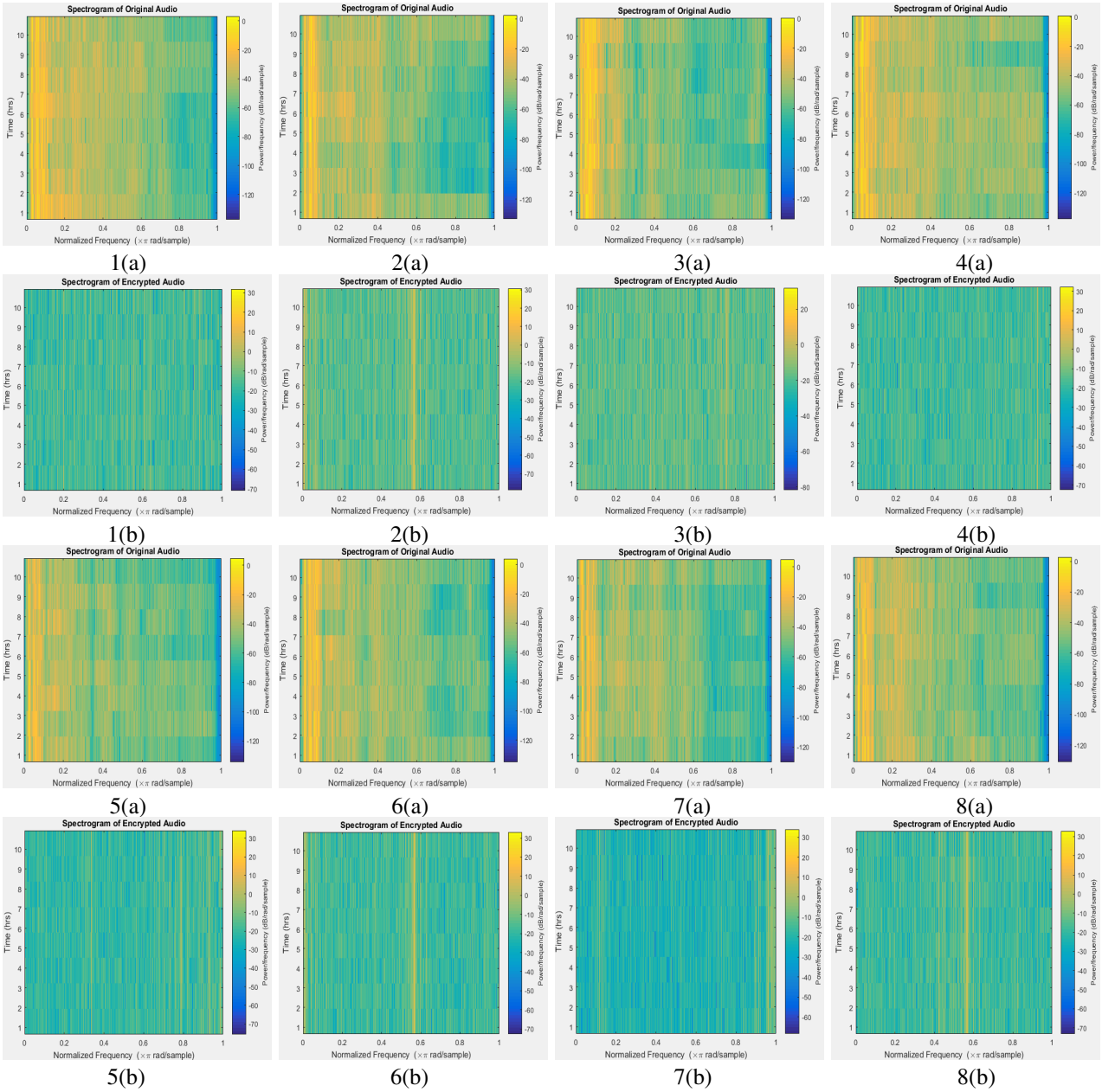

Figure 8. (1.a), (2.a), (3.a), (4.a), (5.a), (6.a), (7.a), (8.a), and (9.a) are the spectrogram plottings of 8 original audios, their encrypted spectrogram plottings are (1.b), (2.b), (3.b), (4.b), (5.b), (6.b), (7.b), and (8.b) respectively.

Algorithm 2 Decryption Algorithm

Require: *Dec_audio*, *Map*1, *Map*2, *Map*3, *Period*1, *Period*2, *Period*3
Ensure: *Dec_audio* (*Decrypted Audio*)

$DK_1 \leftarrow Period1 - Ek1$ ▷ Compute the Decryption Key1
$DK_2 \leftarrow Period2 - Ek2$ ▷ Compute the Decryption Key2
$DK_3 \leftarrow Period3 - Ek3$ ▷ Compute the Decryption Key3
Latin_matrix ← Create a Latin square matrix of size 256×256.
Tiled_Latin_matrix ← Construct a tiled Latin square matrix of size 512×512 by using equation 5
Magic_matrix ← Create a Magic square matrix of size 256×256.
Tiled_Magic_matrix ← Construct a new tiled square matrix of size 512×512 by using equation 5.
if (The Audio is a Stereo) **then**
    Split is
end if
for (Each Channel **do**
    3D_Audio ← Convert-to-3D(Enc_Audio)
    3D-Audio-Dec1 ← Scramble(3D_Audio, Map3,Dk_3)
    Audio-Dec1 ← Reshape to 2D(3D-Audio-Dec1)
    Audio-Dec2 ← XOR(Audio-Dec1, Tiled_Magic_matrix)
    3D-Audio-Dec2 ← Reshape to 3D(Audio-Dec2)
    3D-Audio-Dec3 ← Scramble(3D-Audio-Dec2, Map2, Dk_2)
    Audio-Dec3 ← Reshape to 2D(3D-Audio-Dec3)
    Audio-Dec4 ← XOR(Audio-Dec3, Tiled_Latin_matrix)
    3D-Audio-Dec4 ← Reshape to 3D(Audio-Dec4)
    3D-Audio-Dec5 ← Scramble(3D-Audio-Dec4, Map1, Dk_1).
    Dec_Audio ← Convert to 1D(3D-Audio-Dec5).
end for
Combine the Channels to get the Decrypted Audio Signal