IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received November 27, 2019, accepted December 28, 2019, date of publication December 31, 2019,
date of current version January 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2019.2963329

An Audio Encryption Algorithm Based on DNA Coding and Chaotic System

XINGYUAN WANG AND YINING SU
School of Information Science and Technology, Dalian Maritime University, Dalian 116026, China

Corresponding authors: Xingyuan Wang (xywang@dlmu.edu.cn) and Yining Su (2081604413@qq.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61672124, in part by the Password Theory Project of the 13th Five-Year Plan National Cryptography Development Fund under Grant MMJJ20170203, in part by the Liaoning Province Science and Technology Innovation Leading Talents Program Project under Grant XLYC1802013, and in part by the Key Research and Development Projects of Liaoning Province under Grant 2019020105-JH2/103.

ABSTRACT Transferring multimedia files like audio is a common problem with information security. Therefore, various encryption technologies are needed to protect these contents. This paper proposes a new audio encryption scheme that provides a high degree of security. The novelty of this scheme is the use of chaotic systems and DNA coding to confuse and diffuse audio data. The initial value of the chaotic system is controlled by the hash value of the audio, making the chaotic trajectory unpredictable. Comparison experiments using different types of audio show that the algorithm works well and is secure enough to withstand many common attacks and can be recommended for multi-channel audio processing.

INDEX TERMS Audio encryption, chaotic system, confusion and diffusion, DNA coding.

I. INTRODUCTION

The fast updating of mobile devices and multimedia technologies have increasingly high requirements on security technology. Audio, video and image are indispensable in life. Especially audio data plays an important role in multimedia data [1], [2]. If it is not reliably encrypted, the privacy of the user will be exposed and will have a significant impact. Therefore, the study of audio encryption algorithms has become an important research issue. In this paper, a chaotic encryption algorithm is proposed, chaotic encryption is mainly used for text and images. Because chaos has the characteristics of initial value sensitivity, no periodicity, pseudo-randomness, and ergodicity of chaotic sequences, chaotic systems are more effectively applied to practical applications [3]–[5]. Earlier, Yang *et al.* proposed an encrypted audio scheme using quantum [6]. Later, Sheu proposed a speech encryption method based on fractional chaotic system [7]. The researchers tried to combine chaos with other methods for audio encryption. For example, Sathiyamurthi *et al.* proposed using four chaotic maps to generate chaotic sequence encrypted audio [8]. Mohammed *et al.* changed two chaotic maps to scramble speech [9]. George *et al.* proposed a PWLCM mapping and cellular automaton encryption algorithm [10]. Ganesh Babu *et al.* proposed a high-dimensional chaotic system for audio encryption [11].

DNA is an important component of biological cells, and DNA molecules have supersize parallelism and huge storage space, so DNA encryption technology plays an important role in information security [12]–[14]. DNA encryption is commonly used for images and text, Such as Chai *et al.* applying DNA coding to color image encryption [15], [16]. Jain and Bhatnagar [17] and Shyamasree and Anees [18] have innovatively applied DNA coding to audio encryption, but still have shortcomings such as high correlation and low-speed encryption. Therefore, combining with the advantages of DNA coding and chaos, this paper proposes an effective and secure audio encryption algorithm.

In this paper, an audio encryption based on DNA coding and chaotic system is proposed. Firstly, the PWLCM system is used to generate chaotic sequences. Their initial values are generated by SHA-256 which is highly dependent on plaintext. Second, the binary of the audio is cyclically shifted to achieve global scrambling. Finally, the DNA matrix generated by dynamic coding is XORed with the key DNA matrix generated by the chaotic sequence to achieve diffusion, dynamic decoding to get encrypted audio.

The rest of the paper is organized as follows. Section 2 briefly introduces PWLCM system, DNA coding method.

The associate editor coordinating the review of this manuscript and approving it for publication was Alessia Saggese.

Section 3 details the audio encryption process. Section 4 deals with simulation experiments and safety analysis. Finally, a short conclusion is given in Section 5.

II. ALGORITHM FOUNDATION

A. DNA CODING METHOD

In biology, a DNA sequence consists of four nucleotides, adenine (A), thymine (T), cytosine (C), and guanine (G) form the major component of a nucleotide. The four bases are encoded using binary 00, 01, 10, and 11, according to the complementary pairing rules of DNA, A is paired with T, and C is paired with G, and a total of 8 coding schemes are obtained. When encrypting audio, each sample point is converted to binary and then represented by four codes. Therefore, the encryption and decryption rules of audio are shown in Table 1.

TABLE 1. DNA coding rules.

Rule	1	2	3	4	5	6	7	8
00	A	A	T	T	C	C	G	G
01	C	G	C	G	A	T	A	T
10	G	C	G	C	T	A	T	A
11	T	T	A	A	G	G	C	C

For example, a sample value is 0.7809536, which is transformed into 7805936 by a transform, and the binary sequence is represented as 11101110010101000000000. When the binary is not an even sequence, add 0 to 011011101010100000000 0 in front of the sequence, encode it according to the DNA rule, and use rule 1 to get CTCTAGGGAAAA.

The XOR rule used by this algorithm for DNA sequences is similar to the traditional XOR rule. Therefore, the DNA XOR rule is shown in Table 2.

TABLE 2. DNA XOR operation.

$\oplus$	A	G	C	T
A	A	T	C	G
G	G	A	T	C
C	C	G	A	T
T	T	C	G	A

B. PWLCM SYSTEM

Since the average density function of the Logistic map is unevenly distributed, the resulting values are less balanced. Since PWLCM has good balance, this algorithm mainly refers to this system to generate the required random sequence. The corresponding equation of the system is

$$x_i = F(x_{i-1}, \eta) = \begin{cases} x_{i-1}/\eta, & 0 \le x_{i-1} \le \eta \\ (x_{i-1} - \eta)/(0.5 - \eta), & \eta \le x_{i-1} \le 0.5 \\ F(1 - x_{i-1}, \eta), & 0.5 \le x_{i-1} \le 1. \end{cases} \quad (1)$$

The control parameter of the equation is $\eta \in (0, 0.5)$, and the range of the equation value is $x_{i-1} \in (0, 1)$. When η is in the range of values, the system will exhibit chaotic state.

III. ENCRYPTION ALGORITHM

Fig. 1 is a flowchart of the entire encryption process. The decryption process is similar to the encryption process. By performing the opposite operation on the encrypted audio, the original audio can be recovered.

The algorithm proposed mainly includes the scrambling and diffusion stages in this paper. Assume that the encrypted voice signal f has a size of $M \times 1$. The detailed steps are as follows:

A. KEY AND SYSTEM INITIAL VALUE GENERATION

Because SHA-256 is one of the most widely used and safe algorithms in the world. At the same time, small changes can make a huge change in the hash value. The SHA-256 algorithm can convert the information of the signal into a 256-bit hash value to obtain the required key K. The parameter $\eta_1, \eta_2, \eta_3, \eta_4, \eta_5, \eta_6, \eta_7$ is designated as a key, and the average value σ of the speech signal f is calculated to obtain a control parameter γ (as a key). The control parameters u and η_8 are further calculated.

$$\gamma = \sigma/\max(f), \quad (2)$$

$$u = \mathrm{mod}((\gamma + \eta_1), 1), \quad (3)$$

$$\eta_8 = (\eta_1 + \eta_2 + \eta_3 + \eta_4 + \eta_5 + \eta_6 + \eta_7)/7. \quad (4)$$

The 256-bit key is divided into 8 groups, which can be expressed as $K : (k_1, k_2, \ldots, k_{32})$. This grouping is then used to calculate the initial state variables of the chaotic map.

$$a_1 = (k_1 \oplus k_2 \oplus k_3 \oplus k_4 \oplus k_5 \oplus k_6)/255, \quad (5)$$

$$a_2 = (k_7 \oplus k_8 \oplus k_9 \oplus k_{10} \oplus k_{11} \oplus k_{12})/255, \quad (6)$$

$$a_3 = (k_{13} \oplus k_{14} \oplus k_{15} \oplus k_{16} \oplus k_{17} \oplus k_{18})/255, \quad (7)$$

$$a_4 = (k_{19} \oplus k_{20} \oplus k_{21} \oplus k_{22} \oplus k_{23} \oplus k_{24})/255, \quad (8)$$

$$a_5 = (k_{25} \oplus k_{26} \oplus k_{27} \oplus k_{28} \oplus k_{29} \oplus k_{30}) + (k_{31} \oplus k_{32})/255. \quad (9)$$

B. SCRAMBLING

Since the sample points are decimals, each sample point in the speech f is multiplied and multiplied by a multiple a (determined by the speech signal) into a positive integer to obtain a new speech signal f_1 . Then convert the sample values of f_1 into binary, and set the binary to have c bits, and obtain a new matrix f_2 whose size is $M \times c$.

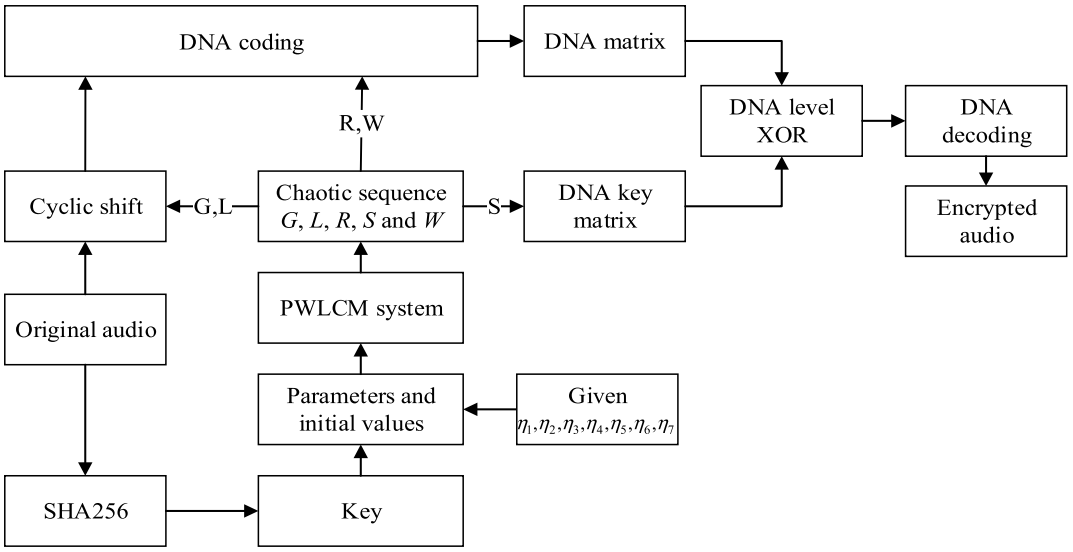

FIGURE 1. Flowchart of the encryption process.

Assuming that a signal has a value range of $[-1, 1]$, add 1 to make it range $[0, 2]$, and multiply by multiple to convert to an integer.

Step 1: Bring $x_1, x_2, x_3, x_4, x_5, \eta_2$ into the system for iteration $M + n$ times, get chaotic sequences T, B, C, D and E.

$$x = \begin{cases} x_1 = \mathrm{mod}((a_1 + u), 1) \\ x_2 = \mathrm{mod}((a_2 + u), 1) \\ x_3 = \mathrm{mod}((a_3 + u), 1) \\ x_4 = \mathrm{mod}((a_4 + u), 1) \\ x_5 = \mathrm{mod}((a_5 + u), 1) \end{cases}, \tag{10}$$

$$T = \{t_1, t_2, \ldots, t_{M+n}\}, \tag{11}$$

$$B = \{b_1, b_2, \ldots, b_{M+n}\}, \tag{12}$$

$$C = \{c_1, c_2, \ldots, c_{M+n}\}, \tag{13}$$

$$D = \{d_1, d_2, \ldots, d_{M+n}\}, \tag{14}$$

$$E = \{e_1, e_2, \ldots, e_{M+n}\}. \tag{15}$$

Step 2: The five elements t_j, b_j, c_j, d_j and e_j are randomly selected from the obtained sequences T, B, C, D and E to generate an initial value of the system used in the encryption process.

$$y = \begin{cases} y_1 = \mathrm{mod}((t_j + u), 1) \\ y_2 = \mathrm{mod}((b_j + u), 1) \\ y_3 = \mathrm{mod}((c_j + u), 1) \\ y_4 = \mathrm{mod}((d_j + u), 1) \\ y_5 = \mathrm{mod}((e_j + u), 1) \\ y_6 = (x_1 + x_2 + x_3 + x_4 + x_5)/5 \end{cases}, \tag{16}$$

Step 3: Generate random sequences G, L, R, S and W using initial value $y_1, y_2, y_3, y_4, y_5, y_6$ and parameter $\eta_3, \eta_4, \eta_5, \eta_6, \eta_7, \eta_8$.

$$G = \{g_1, g_2, \ldots, g_M\}, \tag{17}$$

$$L = \{l_1, l_2, \ldots, l_c\}, \tag{18}$$

$$R = \{r_1, r_2, \ldots, r_{Mc/2}\}, \tag{19}$$

$$S = \{s_1, s_2, \ldots, s_{Mc/2}\}, \tag{20}$$

$$W = \{w_1, w_2, \ldots, w_{Mc/2}\}. \tag{21}$$

The chaotic sequences G and L are processed to obtain the sequences G' and L'.

$$G' = \{g'_1, g'_2, \ldots, g'_M\}, \tag{22}$$

$$L' = \{l'_1, l'_2, \ldots, l'_c\}. \tag{23}$$

among them,

$$g' = \mathrm{mod}(floor(g \times 1000), c), \tag{24}$$

$$l' = \mathrm{mod}(floor(l \times 100000), M). \tag{25}$$

Step 4: Do the following for the $i(1 \leq i \leq M)$ line of the matrix f_2: If i is an even number, it is cyclically shifted g'_i times to the right; if i is an odd number, it is cyclically shifted g'_i times to the left. After all the lines are scrambled, a new matrix f_3 is obtained.

Do the following for the $j(1 \leq j \leq c)$ column of the matrix f_3: If j is even number, it is cyclically shifted up by l'_j times; If j is odd number, it is cyclically shifted down by l'_j times;

After all the columns are scrambled, a new matrix f_4 is obtained and the scrambling process ends.

C. DIFFUSION

In this section, the encoding and decoding DNA rules are generated by a chaotic system and a DNA key matrix is generated. Then use it for diffusion sorting. The specific process is as follows:

Step 1: Three chaotic sequences R and W are generated by the system, and the encoding and decoding rule sequences R_1 and R_2 are generated according to the following formula.

$$R_1 = \mathrm{mod}(floor(R \times 10000), 8), \tag{26}$$

$$R_2 = \mathrm{mod}(floor(W \times 10000), 8). \tag{27}$$

Step 2: Each sample point f_{4_i} of the scrambled matrix f_4 is selected and encoded with the corresponding rule R_{1_i} to generate a $c/2$-bit character. Where g() denotes the encoding of the sample value according to the corresponding encoding

rule, and the full text encoding is stored as the sequence f_5 of $M \times c/2$.

$$f_{5_i} = g(f_{4_i}, R_{1_i}). \tag{28}$$

Step 3: In order to change the speech sample value, a key matrix is needed to generate a DNA key matrix by S, represented by Eq. (29). Without the DNA encoding process, the DNA key matrix can be obtained directly, reducing computational complexity and encryption speed.

$$h_i = \begin{cases} A, & 0 \le s_i \le 0.25 \\ T, & 0.25 < s_i \le 0.5 \\ C, & 0.5 < s_i \le 0.75 \\ G, & 0.75 < s_i \le 1 \end{cases} \tag{29}$$

In this $H = \{h_1, h_2, \ldots, h_{Mc/2}\}$, h_i represents the ith $(1 \le i \le Mc/2)$ element.

Step 4: In order to better hide the original information, the value of the element is further spread after scrambling. The matrix f_5 is transformed into a one dimensional sequence f_5', and the sequence f_5' and the key matrix H are XORed as follows:

$$\begin{cases} Z_1 = f_{5_i} \oplus h_{Mc/2}, & i = 1 \\ Z_i = f_{5_i} \oplus h_i \oplus Z_{i-1}, & 2 \le i \le Mc/2, \end{cases} \tag{30}$$

where f_{5_i} is the ith value of the scrambled DNA matrix f_5, and Z is the DNA matrix $M \times c$ after XOR.

After the end of the operation, Z is decoded line by line according to the decoding sequence R_2 to generate a binary matrix of $M \times c$, and the obtained matrix is converted into decimal and resume to become a voice.

At this point, the encryption process ends. The decryption process is similar to the encryption process and is the inverse of the above steps.

IV. SIMULATION RESULTS AND PERFORMANCE ANALYSIS

A. SIMULATION RESULTS

This experiment not only encrypts mono voice, but also encrypts two-channel voice. This article uses the simulation tool for matlab R2017a, also uses the ''Piano'' sound with a mono frequency of 44100HZ, the ''Human'' sound with amono frequency of 44100HZ and the ''Animal'' sound with a mono frequency of 22050HZ, and the ''Mine'' sound with a dual channel frequency of 22050HZ, the ''Car'' sound with a dual channel frequency of 11025HZ and the ''Drum'' sound with a dual channel frequency of 22050HZ. All audio is uncompressed.

The simulation results are shown in Fig. 2 and Fig. 3. Figs. 2(a)-(c) are mono original audio, corresponding to the encrypted audio as Figs. 2(d)-(f) and the decrypted audio as Figs. 2(g)-(i); Figs. 3(a)-(c) are two-channel raw audio, corresponding to the encrypted audio as Figs. 3(d)-(f) and the decrypted audio as Figs. 3(g)-(i). From the experimental results, it can be seen that the encrypted audio has not seen the original appearance. It can be seen from Fig. 2 and 3 that the original audio of different lengths has a value range of $[-1, 1]$ and their frequency values correspond to different amplitude values. After encryption, all encrypted audio values are in the range $[-1, 3]$. The amplitude value corresponding to the frequency value is between [0, 200].

B. KEY SPACE ANALYSIS

In order to resist the attacker's use of various brute force methods to crack the algorithm, the algorithm must have enough key space to ensure its own security [19]. The key contained in this algorithm is: (1) the control parameter γ related to the original audio, the slight change of the original audio will affect the encrypted audio; (2) directly specified fixed parameters $\eta_1, \eta_2, \eta_3, \eta_4, \eta_5, \eta_6, \eta_7$; (3) generate a key K based on the original audio and hash function SHA-256. Since the complexity of SHA-256 is $S_{SHA-256} = 2^{128}$, the key space is larger than the key space requirement. The algorithm meets the requirements of the cryptosystem against brute force attacks.

C. HISTOGRAM ANALYSIS

The histogram of the audio is a statistical way to describe the audio dispersion. Using algorithmic encryption is to hide its features as much as possible, so that the encrypted histogram becomes flat and can resist ciphertext attacks [20]. Figs. 4(a)-(b) shows the histogram of ''Piano'', ''Human'' and ''Animal'' three segments of audio encryption, respectively, and the corresponding encrypted histograms of Figs. 4(c)-(d) respectively. Figs. 5(a)-(b) show the histograms of the two-channel ''Mine'', ''Car'' and ''Drum'' three-stage audio encryption, respectively, and Figs. 5(c)-(d) respectively corresponding to the encrypted histogram.

D. THE ANALYSIS OF THE SPECTROGRAM

The spectrogram is the expression of three-dimensional information by two-dimensional graph, the abscissa represents time, the ordinate represents frequency, and the coordinate value represents energy value. The amount of energy is expressed in color, and the darker the color, the greater the energy. From blue to yellow to red, energy is getting stronger. Figs. 6(a)-(c) show the spectra of the original audio ''Piano'', ''Human'' and ''Animal'', and Figs. 6(d)-(f) show the encrypted audio spectrum, Figs. 6(a)-(c) show the spectra of the original audio ''Piano'', ''Human'' and ''Animal'', and Figs. 6(d)-(f) show the encrypted audio spectrum. It can be seen from the figure that the original audio spectrogram has important marks such as crossbar, chaotic and vertical lines and has a high amplitude. After encryption, the spectrogram does not see any signs and the amplitude is low.

E. PSNR TEST

PSNR is the ratio of the mean square error of the two audios to the maximum audio value [21]. For encrypted audio, the lower the PSNR, the more cryptogram contains a lot of noise, that is, the ciphertext can resist the attack. The formula

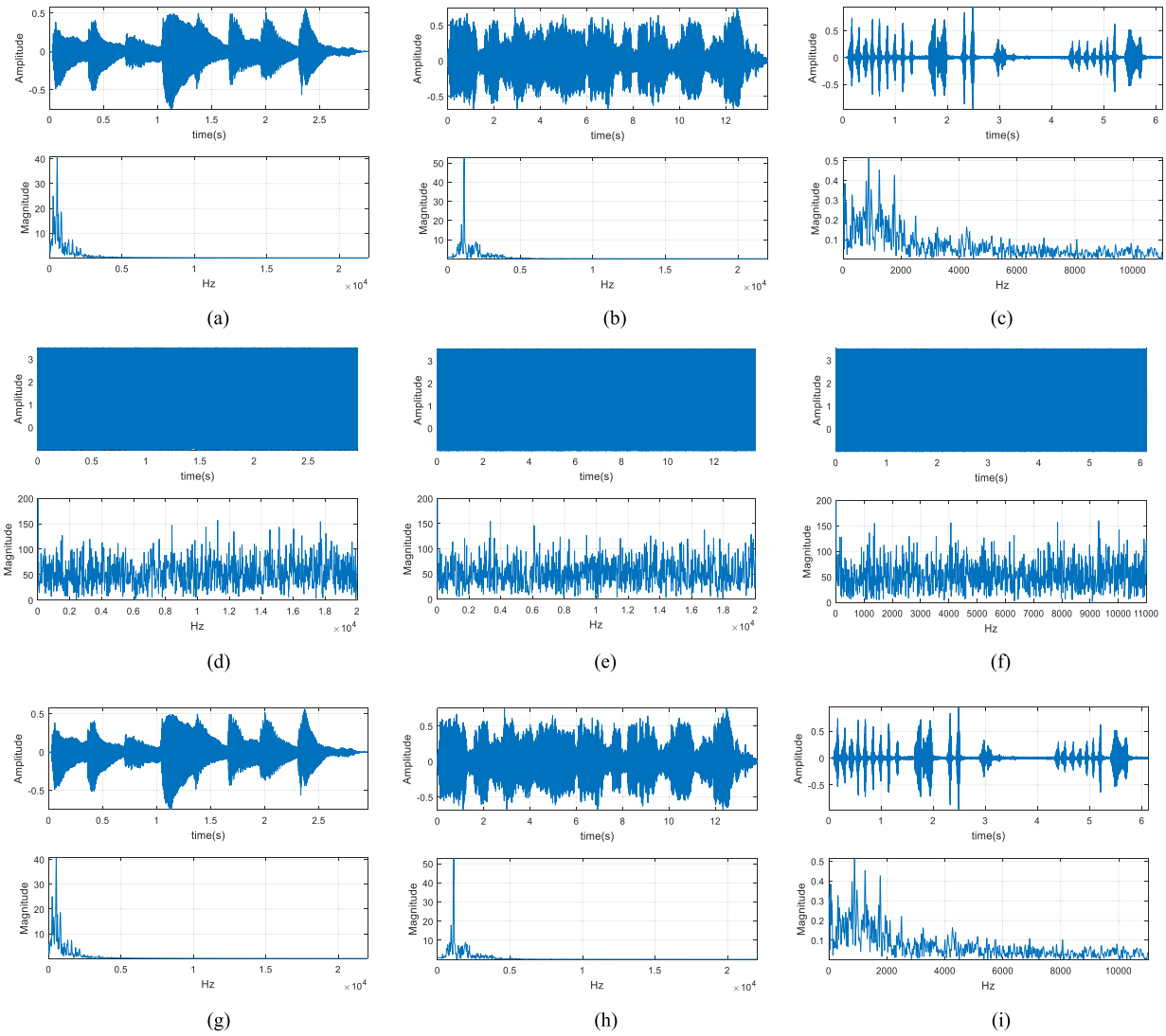

FIGURE 2. Mono audio encryption and decryption results: (a) Original audio for piano.wav, (b) Original audio for human.wav, (c) Original audio for animal.wav, (d) Encrypted audio for piano.wav, (e) Encrypted audio for human.wav, (f) Encrypted audio for animal.wav, (g) Decrypted audio for piano.wav, (h) Decrypted audio for human.wav, (i) Decrypted audio for animal.wav.

for calculating PSNR is as follows:

$$MSE = \frac{1}{M \times N} \sum_{i,j} (A[i,j] - B[i,j])^2, \tag{31}$$

$$PSNR = 10 \log_{10}(\frac{MAX^2}{MSE}). \tag{32}$$

Here, M and N represent the width and height of the audio, respectively, and (i, j) represents the position of the sample value point. A and B represent the original audio and encrypted audio, and is the maximum value in the audio. As shown in Table 3 , the PSNR of the encrypted audio is compared with the algorithms [23] and [24]. The PSNR of the algorithm is negative. The lower the values, the higher the noise level in the encrypted audio, and the easier it is to resist the attack.

TABLE 3. Encrypted audio PSNR value.

Test audio	PSNR (db)
Piano	−14.4715
Human	−12.1941
Animal	−10.1874
Mine	−5.1512
Car	−8.5007
Drum	−12.1700
Ref. [23]	4.5299
Ref. [24]	2.0000

F. CORRELATION ANALYSIS

One of the ways to evaluate the effectiveness of encryption algorithms is to calculate the correlation of adjacent samples

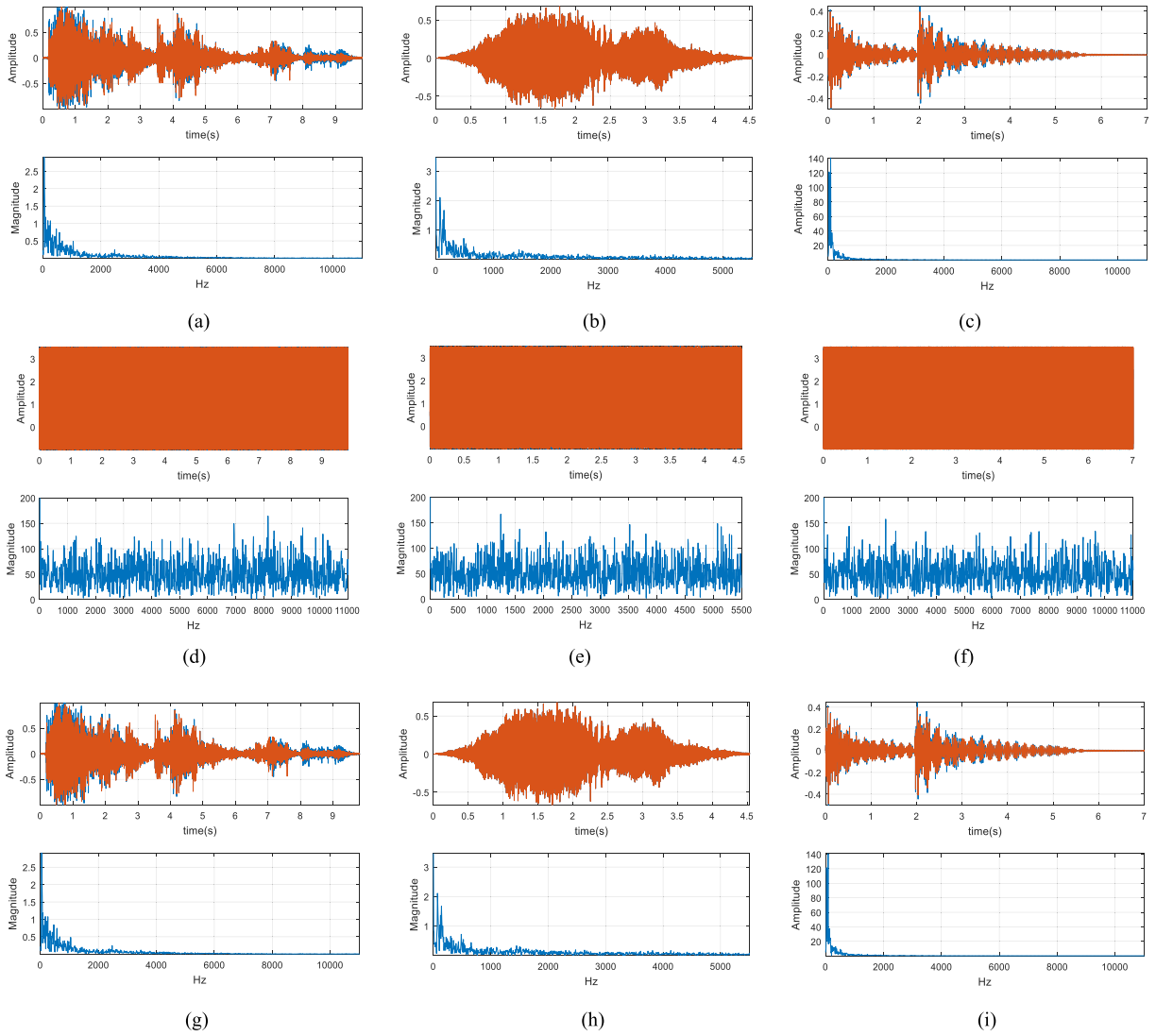

FIGURE 3. 2-channel audio encryption and decryption results: (a) Original audio for mine.wav, (b) Original audio for car.wav, (c) Original audio for drum.wav, (d) Encrypted audio for mine.wav, (e) Encrypted audio for car.wav, (f) Encrypted audio for drum.wav, (g) Decrypted audio for mine.wav, (h) Decrypted audio for car.wav, (i) Decrypted audio for drum.wav.

before and after encryption. The reason for the strong correlation between adjacent samples in the original audio is that adjacent sample values are similar or identical [22]. This algorithm reduces the correlation of the original audio, making it difficult for an attacker to obtain partial sample information by using partial ciphertext information. In order to verify that the algorithm is effective, in the audio signal, 10000 pairs of adjacent samples are selected therefrom, and the correlation of each pair is calculated according to Eq. (33).

$$r_{xy} = \frac{cov(x, y)}{\sqrt{D(x)}\sqrt{D(y)}}, \tag{33}$$

among them,

$$cov(x, y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)),$$

$$D(x) = \frac{1}{N}\sum_{i+1}^{N}(x_i - E(x))^2, \quad E(x) = \frac{1}{N}\sum_{i=1}^{N}x_i.$$

Table 4 shows the correlation coefficients before and after ''Piano'', ''Human'' and ''Animal'' encryption. It can be seen from the table that the correlation coefficient of the original audio is mostly greater than 0.9, which means that the difference between adjacent sample values is very low. After encryption, the correlation of the encrypted audio is mostly less than 0.1. It can be seen that the correlation coefficient is very low. This algorithm is very good against statistical analysis attacks. Table 5 shows the correlation coefficients before and after the encryption of ''Mine'', ''Car'' and ''Drum''. L and R respectively represent the correlation between the left and right channels. The subscripts f and Z represent the original audio and Encrypted audio. Figs. 7(a)-(c) show the distribution of adjacent samples of the original audio ''Piano'', ''Human'' and ''Animal''. From the figure, the adjacent sample values are mostly concentrated in a certain area. After entering the encryption algorithm, as shown in Figs. 7(d)-(f), the adjacent sample values are evenly distributed. Figs. 8(a)-(c) show the distribution of adjacent

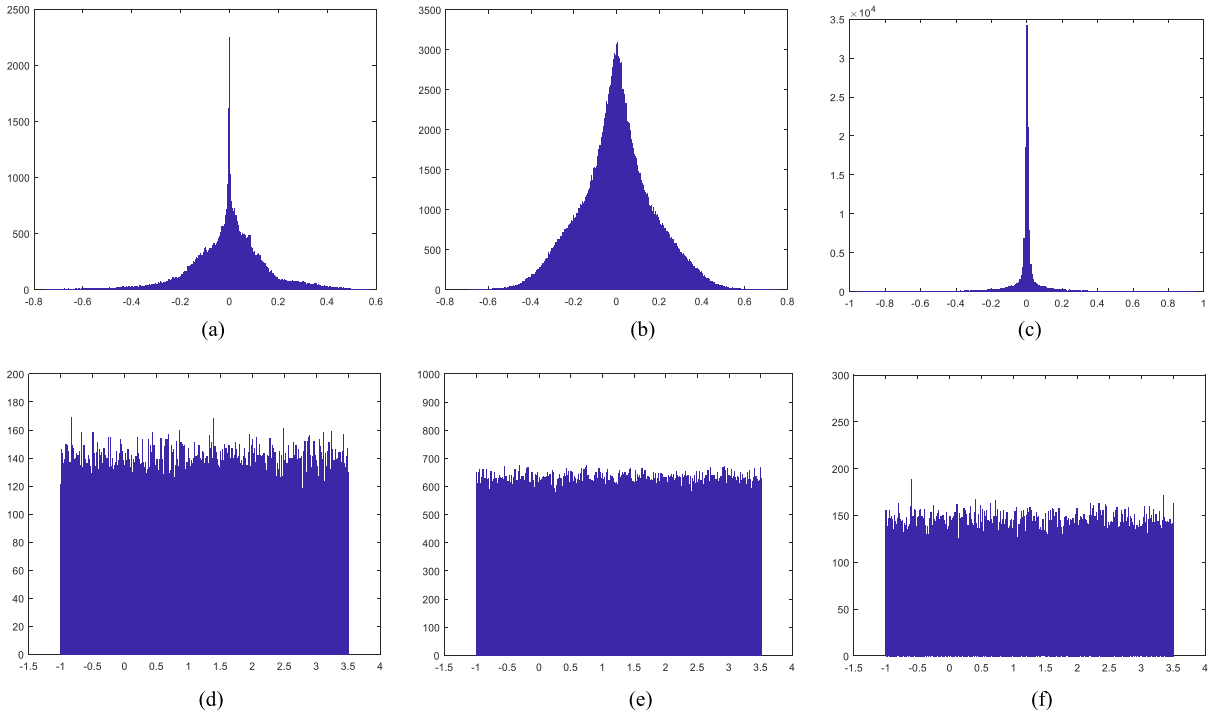

FIGURE 4. Histogram of mono original audio and encrypted audio: (a) Original audio for piano.wav, (b) Original audio for human.wav, (c) Original audio for animal.wav, (d) Encrypted audio for piano.wav, (e) Encrypted audio for human.wav, (f) Encrypted audio for animal.wav.

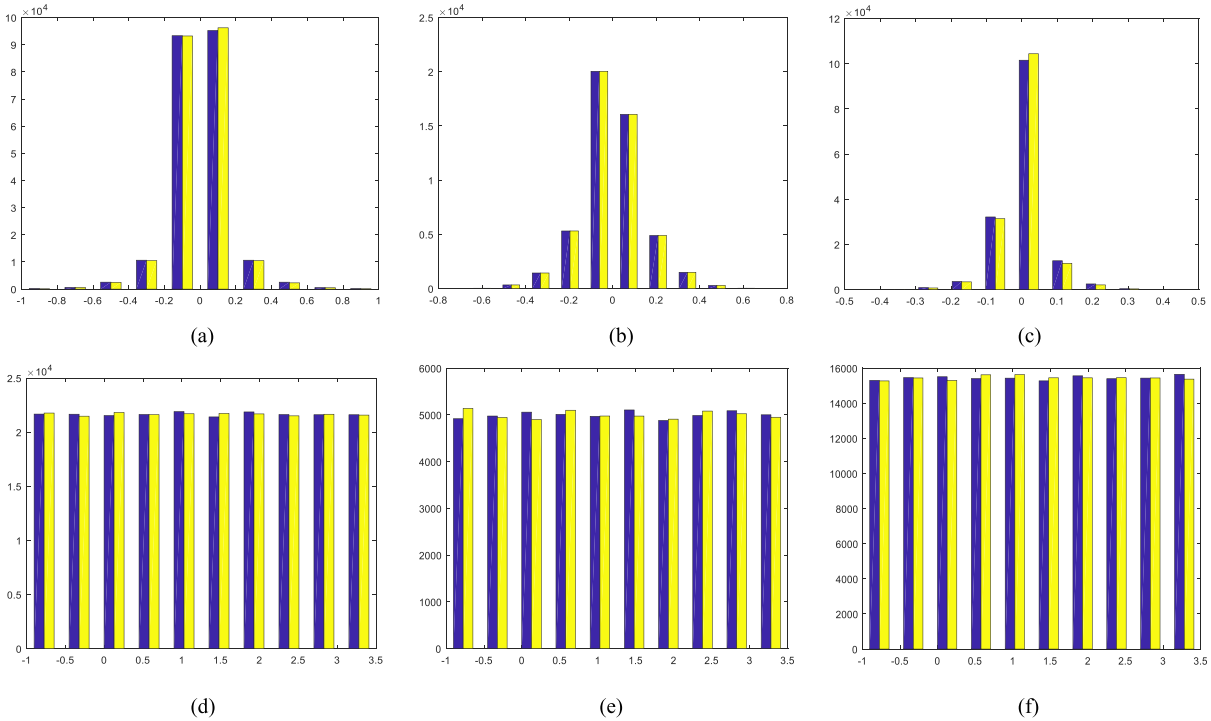

FIGURE 5. Histogram of two-channel original audio and encrypted audio: (a) Original audio for mine.wav, b) Original audio for car.wav, (c) Original audio for drum.wav, (d) Encrypted audio for mine.wav, (e) Encrypted audio for car.wav, (f) Encrypted audio for drum.wav.

samples of the left and right channels of the two channels "Mine", "Car" and "Drum", and Figs. 8(d)-(f) is the distribution of the adjacent sample values after encryption.

G. RESISTANCE DIFFERENTIAL ATTACK ANALYSIS

Differential attacks are often used by attackers [29]. According to the change of specific plaintext encryption, the attacker

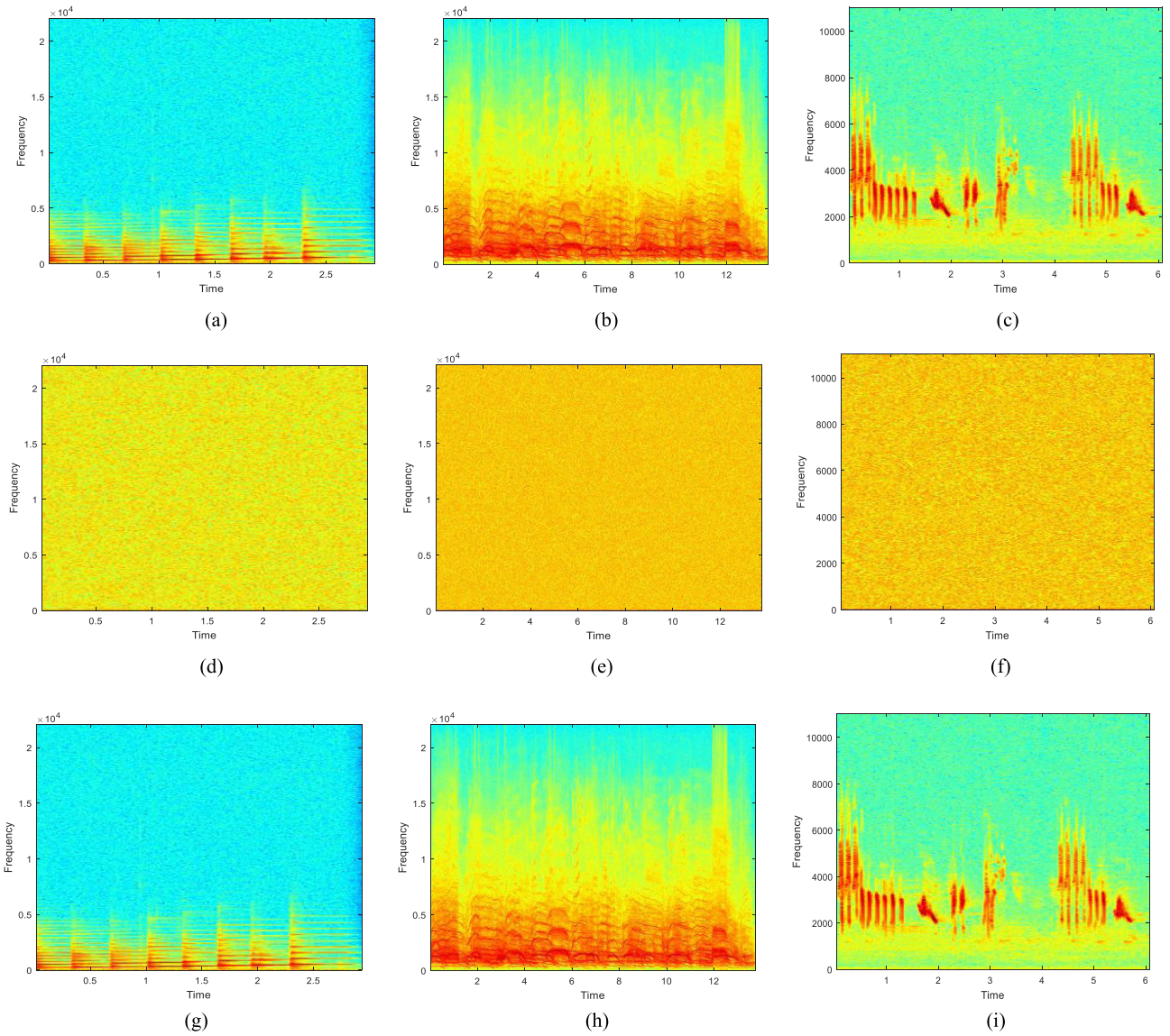

FIGURE 6. Audio spectrogram: (a) Original audio Piano spectrogram, (b) Original audio Human spectrogram, (c) Original audio Animal spectrogram, (d) Encrypted Piano spectrogram, (e) Encrypted Human spectrogram, (f) Encrypted Animal spectrogram, (g) Decrypted Piano spectrogram, (h) Decrypted Human spectrogram, (i) Decrypted Animal spectrogram.

TABLE 4. Correlation coefficient of mono audio.

Test audio	Original audio	Encrypted audio
Piano	0.9967	0.0004
Human	0.9774	−0.0005
Animal	0.6924	0.0014
Ref. [25]	0.9936	−0.1578
Ref. [26]	0.997	0.0133

finds out the relationship between the two and breaks the algorithm. If the slight change in the plaintext has a great influence on the ciphertext, the encryption algorithm can resist the differential attack. Therefore, we often calculate NSCR and UACI to measure whether the algorithm can resist differential attacks. The average NSCR and UACI values are obtained by using Eq. (34) and Eq. (35), and by reducing the value of any sample point by 0.1 at a time, the calculation is iterated 40 times at random.

$$NSCR = \frac{\sum_{i,j} D(i,j)}{L} \times 100, \tag{34}$$

$$UACI = \frac{1}{L}[\sum_{i,j} \frac{|c_1(i,j) - c_2(i,j)|}{2^k - 1}] \times 100. \tag{35}$$

Here, L represents the length of the audio, and k is the number of bits required to display the audio. c_1 and c_2 are the two ciphertexts after the original audio changes by one value. If $c_1(i,j) \neq c_2(i,j)$, then $D(i,j) = 1$, otherwise $D(i,j) = 0$. The ideal for NSCR is 100%, and the ideal value for UACI is 33.3%. It can be seen from Table 6 that the variation of the sample value is large, so the algorithm can resist differential attacks.

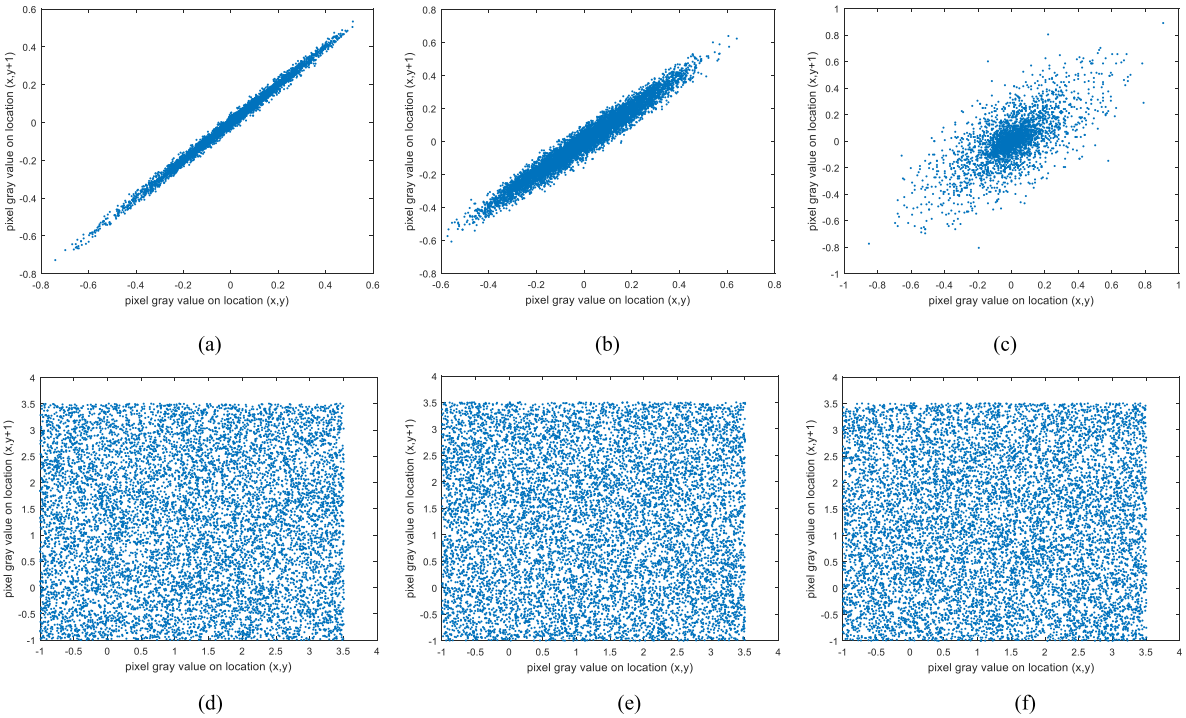

FIGURE 7. Correlation before and after mono channel audio encryption: (a) Original audio Piano correlation, (b) Original audio Human correlation, (c) Original audio Animal correlation, (d) Encrypted Piano correlation, (e) Encrypted Human correlation, (f) Encrypted Animal correlation.

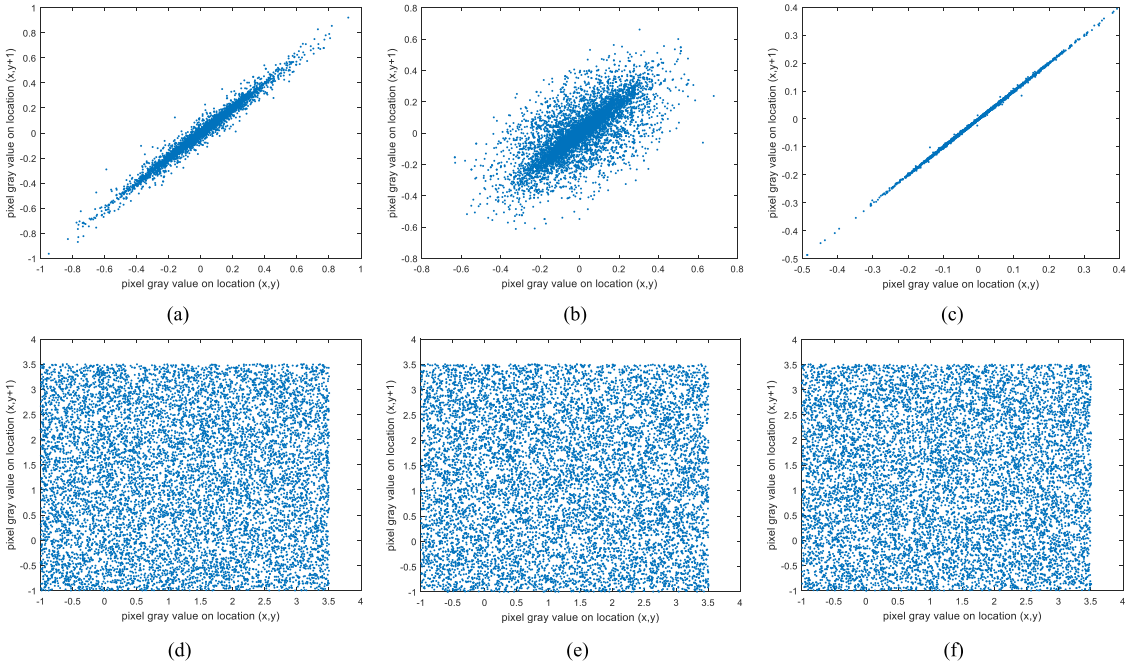

FIGURE 8. Correlation before and after dual channel audio encryption: (a) Original audio Mine correlation, (b) Original audio Car correlation, (c) Original audio Drum correlation, (d) Encrypted Mine correlation, (e) Encrypted Car correlation, (f) Encrypted Drum correlation.

H. EVALUATION OF RUNING TIME AND SPEED

Table 7 shows six different audio encryption times and operating speeds, which require approximately 0.034-0.037 encoding 1 kb data. It can be seen from the table that as the length of the audio increases, the encryption time also increases. Compared with the Ref. [14], the algorithm runs faster and has real-time application.

I. KEY SENSITIVITY ANALYSIS

In the cryptographic avalanche effect, requiring a small transformation of the input will lead to a drastic change [30]. A well-safe algorithm is extremely sensitive to keys. This paper chose "Piano" as the sample audio to verify key sensitivity. The specified key is $\eta_1 = 0.23$, $\eta_2 = 0.48$, $\eta_3 = 0.19$, $\eta_4 = 0.35$, $\eta_5 = 0.29$, $\eta_6 = 0.418$, $\eta_7 = 0.339$

TABLE 5. Correlation coefficients of two-channel audio.

Test audio	Original audio	Encrypted audio	Original audio and its encrypted audio	
	L_f/R_f	L_z/R_z	L_f/L_z	R_f/R_z
Mine	0.8739	−0.0013	0.0024	−0.0002
Car	1.0000	−0.0034	0.0029	0.0007
Drum	0.9755	−0.0013	0.0040	−0.0005
Ref. [23]	0.9999	0.0043	−0.0032	−0.0004
Ref. [27]	0.9830	0.0057	N/A	N/A

TABLE 6. NPCR and UACI of encrypted audio.

Test audio	Average value		Least value		Crest value	
	NSCR (%)	UACI (%)	NSCR (%)	UACI (%)	NSCR (%)	UACI (%)
Piano	100.0000	33.3551	100.0000	33.3551	100.0000	33.4783
Human	100.0000	33.3630	100.0000	33.2834	100.0000	33.4200
Animal	100.0000	33.3439	100.0000	33.1952	100.0000	33.5062
Mine	100.0000	33.3520	100.0000	33.2891	100.0000	33.4163
Car	100.0000	33.3782	100.0000	33.2283	100.0000	33.6024
Drum	100.0000	33.3547	100.0000	33.2750	100.0000	33.4336
Ref. [26]	98.0000	33.0000	N/A	N/A	N/A	N/A
Ref. [27]	99.9992	33.2924	N/A	N/A	N/A	N/A
Ref. [28]	99.9307	33.2705	99.8761	33.1695	100.0000	33.3917

TABLE 7. Running time and speed for encryption process.

Test audio	Total(s)	Size(kb)	Speed(s/Kb)
Piano	4.6	127	0.036
Human	20.9	594	0.035
Animal	4.6	132	0.035
Mine	14.6	423	0.034
Car	3.7	98	0.037
Drum	10.7	302	0.035
Ref. [14]	93	N/A	0.23
	4	N/A	0.25
	23	N/A	0.22

and the plaintext related key γ is $\gamma = 0.75091552734375$.

Randomly change the value of a certain key, keep the other keys unchanged, and find that the ciphertext changes greatly.

It can be seen from Table 8 that a small change to the key can cause a huge change in the encrypted audio, so the algorithm is highly sensitive to the key.

In order to verify the sensitivity of the key, this paper makes the following changes to the key:

Change 1: $\eta_1 = 0.23$ will be changed to $\eta_1 = 0.23 + 10\text{^}(-15)$

Change 2: $\eta_2 = 0.48$ will be changed to $\eta_2 = 0.48 + 10\text{^}(-15)$

Change 3: $\eta_3 = 0.19$ will be changed to $\eta_3 = 0.19 + 10\text{^}(-15)$

Change 4: $\eta_4 = 0.35$ will be changed to $\eta_4 = 0.35 + 10\text{^}(-15)$

Change 5: $\eta_5 = 0.29$ will be changed to $\eta_5 = 0.29 + 10\text{^}(-15)$

Change 6: $\eta_6 = 0.418$ will be changed to $\eta_6 = 0.418 + 10\text{^}(-15)$

Change 7: $\eta_7 = 0.339$ will be changed to $\eta_7 = 0.339 + 10\text{^}(-15)$

$$K = '84e9ec57e300987bd9669926d755f7c134f1549eea5b2639fa497a0556c5a509'.$$
$$K' = '94e9ec57e300987bd9669926d755f7c134f1549eea5b2639fa497a0556c5a509'.$$

TABLE 8. Comparison of original ciphertext and ciphertext after changing key.

Paino	Change 1	Change 2	Change 3	Change 4	Change 5	Change 6	Change 7	Change 8	Change9
NSCR (%)	100	100	100	100	87.5481	100	100	100	100
UACI (%)	33.3620	33.3869	33.4064	33.3938	32.1682	33.2731	33.4198	33.2815	33.4252

Change 8: $\eta_8 = 0.750915527343750$ will be changed to $\eta_8 = 0.750915527343750 + 10\hat{}(-15)$

Change 9: K, shown at the bottom of the previous page, will be changed to K', shown at the bottom of the previous page.

V. CONCLUSION

Through the above various performance analysis, this paper proposes a new audio encryption scheme, combining chaos and DNA to encrypt audio. This algorithm is used for mono and two-channel audio encryption, through simulation experiments and security analysis such as histogram, correlation analysis, spectral map, key space, key sensitivity, PSNR test, differential attack, etc. Compared with the existing algorithms, the algorithm has a large key space, strong key sensitivity and can resist various attacks. Therefore, the algorithm application can be used in the field of voice encryption.

REFERENCES

[1] A. Kaur and M. K. Dutta, "An optimized high payload audio watermarking algorithm based on LU-factorization," *Multimedia Syst.*, vol. 24, no. 3, pp. 341–353, Jun. 2018.
[2] Z. Liu, J. Huang, X. Sun, and C. Qi, "A security watermark scheme used for digital speech forensics," *Multimedia Tools Appl.*, vol. 76, no. 7, pp. 9297–9317, Apr. 2017.
[3] A. Belazi, A. A. A. El-Latif, and S. Belghith, "A novel image encryption scheme based on substitution-permutation network and chaos," *Signal Process.*, vol. 128, pp. 155–170, Nov. 2016.
[4] A. Souyah and K. M. Faraoun, "An image encryption scheme combining chaos-memory cellular automata and weighted histogram," *Nonlinear Dyn.*, vol. 86, no. 1, pp. 639–653, Oct. 2016.
[5] X.-Y. Wang, S.-X. Gu, and Y.-Q. Zhang, "Novel image encryption algorithm based on cycle shift and chaotic system," *Opt. Lasers Eng.*, vol. 68, pp. 126–134, May 2015.
[6] Y.-G. Yang, J. Tian, S.-J. Sun, and P. Xu, "Quantum-assisted encryption for digital audio signals," *Optik*, vol. 126, no. 21, pp. 3221–3226, Nov. 2015.
[7] L. J. Sheu, "A speech encryption using fractional chaotic systems," *Nonlinear Dyn.*, vol. 65, nos. 1–2, pp. 103–108, Jul. 2011.
[8] P. Sathiyamurthi and S. Ramakrishnan, "Speech encryption using chaotic shift keying for secured speech communication," *EURASIP J. Audio, Speech, Music Process.*, vol. 2017, no. 1, pp. 2–11, Sep. 2017.
[9] R. S. Mohammed and S. B. Sadkhan, "Speech scrambler based on proposed random chaotic maps," in *Proc. AL-SADEQ Int. Conf. Multidisciplinary IT Commun. Sci. Appl. (AIC-MITCSA)*, May 2016.
[10] S. N. George, N. Augustine, and D. P. Pattathil, "Audio security through compressive sampling and cellular automata," *Multimedia Tools Appl.*, vol. 74, no. 23, pp. 10393–10417, Dec. 2015.
[11] S. G. Babu and P. Ilango, "Higher dimensional chaos for audio encryption," in *Proc. IEEE Symp. Comput. Intell. Cyber Secur. (CICS)*, Apr. 2013.
[12] C. T. Clelland, V. Risca, and C. Bancroft, "Hiding messages in DNA microdots," *Nature*, vol. 399, no. 6736, pp. 533–534, Jun. 1999.
[13] M. Babaei, "A novel text and image encryption method based on chaos theory and DNA computing," *Natural Comput.*, vol. 12, no. 1, pp. 101–107, Mar. 2013.
[14] H. K. Kate, J. Razmara, and A. Isazadeh, "A novel fast and secure approach for voice encryption based on DNA computing," *3D Res.*, vol. 9, no. 2, Jun. 2018.
[15] X. Chai, X. Fu, Z. Gan, Y. Lu, and Y. Chen, "A color image cryptosystem based on dynamic DNA encryption and chaos," *Signal Process.*, vol. 155, pp. 44–62, Feb. 2019.
[16] X. Chai, Z. Gan, K. Yuan, Y. Chen, and X. Liu, "A novel image encryption scheme based on DNA sequence operations and chaotic systems," *Neural Comput. Appl.*, vol. 31, no. 1, pp. 219–237, Jan. 2019.
[17] S. Jain and V. Bhatnagar, "Bit based symmetric encryption method using DNA Sequence," in *Proc. 5th Int. Conf. Next Gener. Inf. Technol. Summit (Confluence)*, Sep. 2014.
[18] C. M. Shyamasree and S. Anees, "Highly secure DNA-based audio steganography," in *Proc. Int. Conf. Recent Trends Inf. Technol. (ICRTIT)*, Jul. 2013.
[19] W. Xingyuan, F. Le, W. Shibing, C. Zhang, and Z. Yingqian, "Spatiotemporal chaos in coupled logistic map lattice with dynamic coupling coefficient and its application in image encryption," *IEEE Access*, vol. 6, pp. 39705–39724, 2018.
[20] X. Wang and S. Gao, "Image encryption algorithm for synchronously updating Boolean networks based on matrix semi-tensor product theory," *Inf. Sci.*, vol. 507, pp. 16–36, Jan. 2020.
[21] S. E. El-Khamy, N. O. Korany, and M. H. El-Sherif, "A security enhanced robust audio steganography algorithm for image hiding using sample comparison in discrete wavelet transform domain and RSA encryption," *Multimedia Tools Appl.*, vol. 76, no. 22, pp. 24091–24106, Nov. 2017.
[22] X. Wang, X. Zhu, and Y. Zhang, "An image encryption algorithm based on Josephus traversing and mixed chaotic map," *IEEE Access*, vol. 6, pp. 23733–23746, 2018.
[23] H. Liu, A. Kadir, and Y. Li, "Audio encryption scheme by confusion and diffusion based on multi-scroll chaotic system and one-time keys," *Optik*, vol. 127, no. 19, pp. 7431–7438, Oct. 2016.
[24] A. A. Tamimi and A. M. Abdalla, "An audio shuffle-encryption algorithm," presented at the Lecture Notes Eng. Comput. Sci., 2014.
[25] A. Roy and A. P. Misra, "Audio signal encryption using chaotic Hénon map and lifting wavelet transforms," *Eur. Phys. J. Plus*, vol. 132, no. 12, Dec. 2017.
[26] N. Sasikaladevi, K. Geetha, and K. N. V. Srinivas, "A multi-tier security system (SAIL) for protecting audio signals from malicious exploits," *Int. J. Speech Technol.*, vol. 21, no. 2, pp. 319–332, Jun. 2018.
[27] J. B. Lima and E. F. Da Silva Neto, "Audio encryption based on the cosine number transform," *Multimedia Tools Appl.*, vol. 75, no. 14, pp. 8403–8418, Jul. 2016.
[28] A. Ghasemzadeh and E. Esmaeili, "A novel method in audio message encryption based on a mixture of chaos function," *Int. J. Speech Technol.*, vol. 20, no. 4, pp. 829–837, Dec. 2017.
[29] P. K. Naskar, S. Paul, D. Nandy, and A. Chaudhuri, "DNA encoding and channel shuffling for secured encryption of audio data," *Multimedia Tools Appl.*, vol. 78, no. 17, pp. 25019–25042, Sep. 2019.
[30] X. Wang, L. Feng, and H. Zhao, "Fast image encryption algorithm based on parallel computing system," *Inf. Sci.*, vol. 486, pp. 340–358, Jun. 2019.

XINGYUAN WANG received the Ph.D. degree in computer software and theory from Northeast University, China, in 1999. From 1999 to 2001, he was a Postdoctoral Researcher with Northeast University. He is currently a Professor with the School of Information Computing Science, Dalian Maritime University, China. He has published three books and over 400 scientific articles in refereed journals and proceedings. His research interests include nonlinear dynamics and control, image processing, chaos cryptography, systems biology, and complex networks.

YINING SU received the bachelor's degree from the College of Mathematics and Information Science, Yantai University, China. She is currently pursuing the master's degree in computer science and technology with Dalian Maritime University. Her main research directions are chaotic encryption and image processing.