Bulletin of Electrical Engineering and Informatics
Vol. 10, No. 1, February 2021, pp. 160~170
ISSN: 2302-9285, DOI: 10.11591/eei.v10i1.2405

A novel technique for speech encryption based on k-means clustering and quantum chaotic map

Amal Hameed Khaleel, Iman Qays Abduljaleel
Department of Computer Science, Basrah, Iraq

Article Info

Article history:

Received Mar 19, 2020
Revised May 24, 2020
Accepted Jun 11, 2020

Keywords:

K-means clustering
Quality metric
Quantum chaotic map
Scrambling
Speech encryption

ABSTRACT

In information transmission such as speech information, higher security and confidentiality are specially required. Therefore, data encryption is a pre-requisite for a secure communication system to protect such information from unauthorized access. A new algorithm for speech encryption is introduced in this paper. It depends on the quantum chaotic map and k-means clustering, which are employed in keys generation. Also, two stages of scrambling were used: the first relied on bits using the proposed algorithm (binary representation scrambling BiRS) and the second relied on k-means using the proposed algorithm (block representation scrambling BlRS). The objective test used statistical analysis measures (signal-to-noise-ratio, segmental signal-to-noise-ratio, frequency-weighted signal-to-noise ratio, correlation coefficient, log-likelihood ratio) applied to evaluate the proposed system. Via MATLAB simulations, it is shown that the proposed technique is secure, reliable and efficient to be implemented in secure speech communication, as well as also being characterized by high clarity of the recovered speech signal.

Corresponding Author:

Amal Hameed Khaleel,
Department of Computer Science,
Basrah University,
Basrah, Iraq.
Email: amal_albahrany@yahoo.com

1. INTRODUCTION

Speech communications are becoming increasingly common, so the need to have a high level of protection is rising dramatically [1]. Speech contains a lot of repetition compared to written text, which makes it difficult to provide security. Therefore, the information transmitted in telephone networks and radio communication techniques must be confidential. The signal of speech includes bits which have negative and positive values. Speech signs can be represented in two forms: analog and digital. The digital form is represented numerically in the analog form, where the sign is composed of ones and zeros. In analog representation, the waveform can describe the amplitude and frequency of the signal [2]. Because the encryption methods are complex, they need to be applied on flexible platforms to meet real-time speech encryption requirements.

Therefore, many researchers are interested in the encryption techniques of speech. For example, Hongjun and Xingyuan [3] used the Chebyshev map. Kaur and Sekhon [1] used algorithms to encrypt the original signal four times, using a different algorithm at each level, where the output of one level forms the input of the next level. Anees [4] used chaotic systems and tent map, and a kind of Lorenz system. Akgül et al. [5] used a non-linear equation to increase security in audio data encryption, with dimension discrete-time of chaotic systems. Agarwal et al. [6] developed a voice encryption system using an advanced

encryption standards (AES) algorithm as a real-time software application. Duta et al. [2] used algorithms divided into two main categories: asymmetric ciphers (NTRU and RSA) and symmetric ciphers (AES), to compare and evaluate based on several criteria including (DSP) implementations of three voice encryption algorithms in real time. Alroubaie *et al.* [7] suggested speech encryption using synchronised fixed-point chaotic map-based stream ciphers (SFPCM-SC). Kordov [8] proposed the encryption algorithm dependent on exemplary symmetric models utilising a pseudo-random number generator formed with modified rotation conditions and a chaotic map. In this work, we propose a new algorithm based on the chaotic algorithm for permutation (Tent Map) to scramble the values in the binary representation of the signal, and then use a clustering algorithm for data mining (k-means) to scramble the blocks of the signal and generate a secondary key for the encryption. Finally, we use the quantum logistic map to generate a primary key.

This paper is organised as follows. Section 2 introduces the proposed methods considered in this paper. Section 3 outlines our research method and the steps of the implementation of the algorithms. In section 4 we discuss the results of the implementations of the algorithms and compare them with previous research. Finally, section 5 concludes the paper.

2. THE PROPOSED METHOD

The proposed method depended on scrambling and encryption techniques. We employed the quantum chaotic map and k-means clustering in keys generation, as well as the proposed algorithms (BiRS) and (BlRS) for scrambling and encryption of speech signals. Also, the Fractional Fourier Transform (FrFT) was employed in proposed speech encryption and decryption algorithms

2.1. Speech scrambling techniques

Speech scrambling is used in converting the speech information to another, unintelligible form. There are various types of analog speech scrambling methods such as the following: time domain, frequency domain, bi-dimensional (time-frequency), transform-domain (TD) [9], and amplitude domain [2]. In our paper, we used tent map in the time domain for scrambling of speech signals.

2.2. Quantum chaotic logistic system

The chaotic system has many dynamic properties that make this very interesting for crypto apps [10]. This paper uses the skewed tent map in the key generation procedure [11]. It is usually defined as (1).

$$F(e_i, r_i) = \begin{cases} \frac{r_i}{e} & r_i = [0, e) \\ \frac{(1-r_i)}{(1-e)} & r_i = (e, 1] \end{cases} \quad (1)$$

where (e_i,r_i) are the system variables and the initial state assigned to a specific map, respectively. Using the frequency F (e, r), we can obtain values that have chaotic sequences that may be between [0, 1].

For this paper, we used the quantum chaotic map. Goggin et al. suggested a dissipative quantum logistic map by joining the quantum kicked towards the bath of harmonic oscillators [12, 13]. The operators used the particular well-known boson creation (α^t) and annihilation (α) operators to find a period-doubling route to the classical behaviour, as the dissipation parameter is increased. This particular chaotic map is dictated by the following equations [14]:

$$\begin{cases} x_{n+1} = r\,(x_n - |\,x_n\,|)^2 - r y_{n,} \\ y_{n+1} = -y_n\, e^{-2\beta} + e^{-\beta}\, r[(2 - x_n - x_n^*)y_n - x_n\, z_n^* - x_n^*\, z_n], \\ z_{n+1} = -z_n e^{-2\beta} + e^{-\beta}\, r[2(1 - x_n^*)z_n - 2x_n y_n - x_n]. \end{cases} \quad (2)$$

where δa represents a new quantum fluctuation about $\langle\alpha\rangle$. Effects regarding quantum corrections were produced by $\alpha = \langle\alpha\rangle + \delta\alpha$, $x = \langle\alpha\rangle$, $y = \langle\delta\alpha^t\delta\alpha\rangle$ and $z = \langle\delta\alpha\delta\alpha\rangle$; x*, z* are complex conjugates of x and z respectively. $x \in [0,1]$, $y \in [0,0.1]$, $z \in [0,0.2]$, $x^* = x$, $z^* = z$, $\beta \in [6,\infty)$, and $r \in [0,4]$.

β is the particular dissipation parameter in the specific dynamic system and the intermediate values of β in addition to yn, $z_n \neq 0$, a single-dimensional logistic map if the quantum corrections yn and $z_n \rightarrow 0$. The quantum map exhibits a period of time-doubling map to chaos [14]. Quantum chaos is based on the quantum system that displays chaotic dynamics within a certain range. This is an interesting property, which can be used in cryptography [15].

2.3. The fractional fourier transform (FrFT)

FrFT is a generalisation regarding the fourier transform (FT). The particular FrFT is visible as some sort of linear transformation, which revolves the signal through virtually any arbitrary angle into a mixed frequency-space domain [16]. FrFT is a linear operator defined as [17]:

$$X\alpha(u) = F\alpha\,(x(t)) = \int_{-\infty}^{+\infty} x(t) k_\alpha\,(t,u)dt \quad (3)$$

where kα(t,u) represents the kernel function that is defined as:

$$k\alpha(t,u) = \begin{cases} \sqrt{\frac{1-jcot\alpha}{2\pi}} & if\ \alpha\ is\ not\ multiple\ of\ \pi \\ xe^{j(u^2/2)cot\alpha}\, e^{j(t^2/2)cot\alpha - jutcsc\alpha} & \\ \delta(t-u) & if\ \alpha\ is\ a\ multiple\ of\ 2\pi \\ \delta(t+u) & if\ \alpha+\pi\ is\ a\ multiple\ of\ 2\pi \end{cases} \quad (4)$$

where δ(t): represent the Dirac function, α: represents the angle of rotation (α=2α/π), Fα: denotes the operator associated with FrFT. We get a Fourier transform if α=π/2, while we get the same signal if α=0, so the representation of the time-frequency of the signal is an intermediate value result of α: (0<α<π/2) [17].

2.4. K-means clustering

K-means clustering is one of the data mining technologies for the non-hierarchical clustering of data that classifies data as one or more groups/clusters. The algorithm works in two stages: in the first stage, the initial K centroids are randomly chosen, one for each group; in the second stage, each object from the specified input data set is associated with the block that has the closest central point [18, 19].

3. RESEARCH METHOD

In this section, we discuss the proposed speech encryption algorithm, and how to build a generator for pseudo-random numbers key.

3.1. Key generation method

In our work, we applied quantum chaotic mapping and then k-means clustering algorithms.

- Quantum chaotic map key generation

The central idea of proposal our work is to generate a quantum chaotic map key through the following steps:

Step 1: Initialise the control parameters and initialise conditions to the quantum logistic map via experimental implementation using (2) as following:

X1 = 0.45234444336
Y1 = 0.003453324566
Z1 = 0.001324523564
X* = 0.00186
Z* = 0.00398
B = 4.489
R = 3.99

Step 2: Determine the required length of the sequence (SubKey).
Step 3: Generate (2) as shown in section (2.3).
Step 4: If the length of the sequence of SubKey is not satisfied, back to Step (3), otherwise stop.

- K-means key generation

The main idea of our work to generate a k-means key consists of the following steps:

Step 1: Divide the original speech signal into blocks, each of which has a size of 64 values.
Step 2: Use the k-means algorithm in the chosen partition of speech signal saved in matrix BB of dimension (64*64) to extract 64 clusters.
Step 3: Save the centroid of these clusters in the matrix C of dimension (64*64) and save the index of each cluster in the index vector named IDX, to send it to the block representation scrambling algorithm.
Note: Matrix BB was chosen from the beginning of the blocks, where it is described as not having any zero value.
Step 4: Take the non-duplicate values in every row of the matrix C and save them in a new row. Repeat this process for each row of matrix C to create a new matrix D.

Step 5: Choose the value located in the middle of the values from each row of matrix D, and then put it in a vector (64 values). This vector represents a kmeans key that we use in our proposed algorithm for encryption as a secondary secret key.

3.2. Proposed scrambling algorithm

In this algorithm, we worked to create two proposed algorithms for scrambling, one of which works at the binary representation level and the other works at the block representation level to ensure scrambling of values along with the speech signal.

- Binary representation scrambling level

In order to ensure information scrambling, we use the following algorithm, which consists of the following:

Step 1: Test all values of the speech signal. If the value is negative, convert its value to positive, and vice versa.

Step 2: Convert the speech signal to digital. This signal has several samples frequency sampling (FS), and each sample has N bits (64-bits in binary representation).

Step 3: From bit 7 to 64 in binary representation, transmitter bits locations are generated using the Tent map, with length equal to (58) bits speech samples length. For example, the tent chaotic maps given by (1) with an initial condition in our approach are:
E=.5, R=1.99, phin=.5, phi2(1)=A-(B*phin),
and the signal of digital bits is:
'00111111111000111010010100010001100111001110000001110101111110111',
after the binary scrambling stage, the digital bits will be:
'00111110011110001110100011111010011001110101100100100101101101110',
which is equal $(0.000000092797)_{10}$.
We note that the first (6 bits) did not change, because we determined the field of change from (7 bit to 64 bit) to ensure that the values are kept within the normal range of the speech signal.

Step 4: Repeat Steps (1-3) for all samples in the speech signal.

Step 5: Save the decimal values in a new signal to use it in the second stage of scrambling, as shown in Figure1:

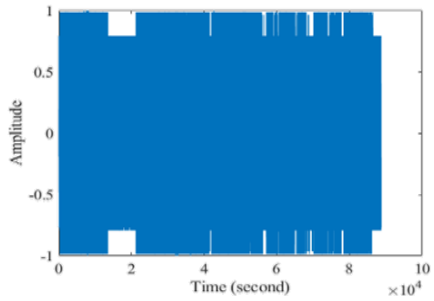


Figure 1. Scrambling speech signal using binary representation scrambling algorithm

- Blocks representation scrambling level

The speech blocks representation scrambling algorithm includes a permutation of the speech blocks in time. Our approach stages consist of the following:

Step1: Divide the signal generated by the binary representation scrambling algorithm into blocks, where each block has a size of 64 values.

Step 2: Scramble the locations of the values of each block, depending on the result IDX from the k-means clustering algorithm.

Step 3: For each block resulting from the previous step, repeat the following steps three times:
a. Divide the speech signals into two groups. The first group contains the blocks located in the odd positions of the signal, and the second group contains the blocks located in the even positions of the signal.
b. Merge the two groups into one signal by adding a block from the end of the odd group and then adding a block from the beginning of the even group.

Step 4: Save the resulting signal to use it at the next encryption stage, as shown in Figure 2:

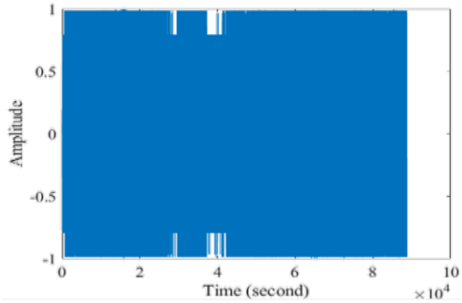


Figure 2. Scrambling signal using blocks representation scrambling algorithm

3.3. The proposed speech encryption and decryption algorithms

The proposed scheme consists of three operations: permutation of the speech signal using the tent map in the time domain; rearrangement of samples of speech signal using the k-means algorithm; encryption of speech signals. The structure of the proposed model is shown in Figure 3.

- Encryption process

The encryption process can be summarised in the following steps, as shown in Figure 3:

Step 1: Input speech signal (we used a different length of samples 3-10 ms).
Step 2: Scramble input speech signal with a tent map using the proposed binary representation scrambling algorithm.
Step 3: Subdivide the speech signal into blocks J (64 samples), then use the proposed blocks representation scrambling algorithm.
Step 4: Generate k-means key by using the k-means clustering algorithm.
Step 5: Generate mask key for each block using a quantum chaotic map. The mask key MK consists of N samples. MK(k),k=1,2,…,N.
Step 6: Separate each block signal samples into magnitude and phase using (3) of the FrFT.
Step 7: Apply XOR operation between the mask key and the magnitude samples in each block.
Step 8: Apply XOR operation between k-means keys and the result in the step above in each block.
Step 9: Apply inverse FrFT.
Step 10: Transform the encrypted speech signal from blocks to vectors and save it in the new signal.
Step 11: Transmit the encrypted speech signal

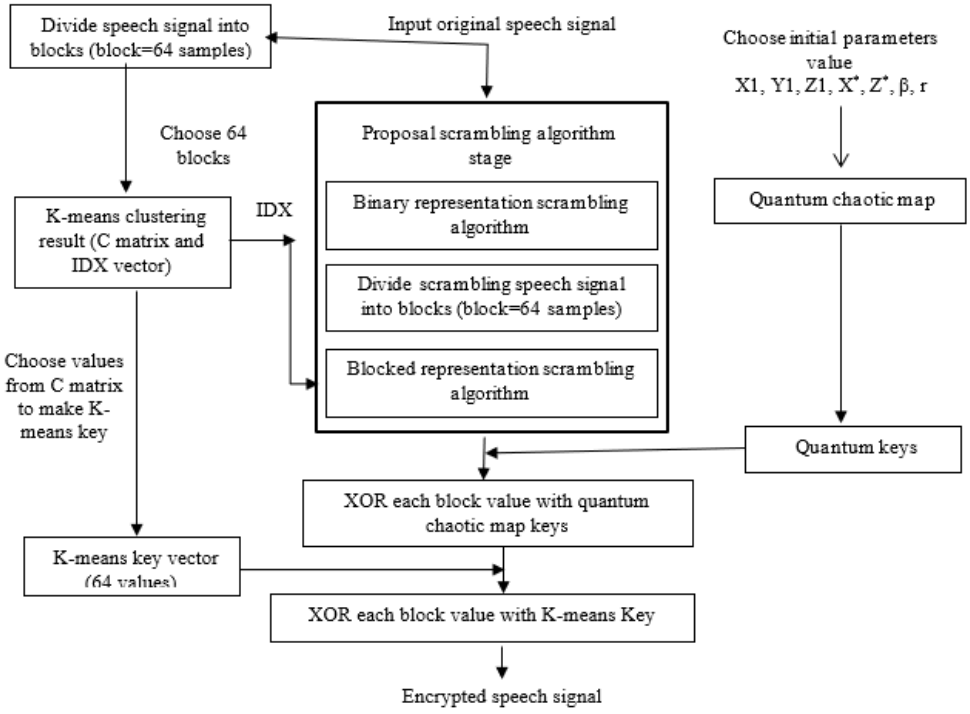


Figure 3. The flowchart of the encryption algorithm

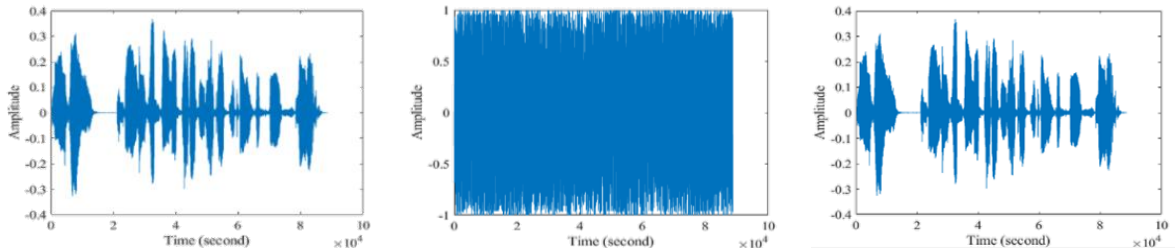
- Decryption process

The decryption process can be summarised in the following steps:

Step 1: Transform the encrypted speech signal from 1D to 2D blocks
Step 2: Receive the permutation k-means key generated using the k-means algorithm.
Step 3: Generation of mask key for each block using a quantum chaotic map. The mask key MK consists of N samples. MK (k), k=1, 2… N.
Step 4: Separate each block signal samples into phase and magnitude using the FrFT. Apply XOR operation first between the mask key and the FrFT magnitude samples, then the second XOR between the k-means key and the first XORed result.
Step 5: Apply inverse FrFT transform to each block samples.
Step 6: Apply blocks representation descrambling algorithm using k-means key.
Step 7: Transform the decrypted speech signal from 2D to 1D to get the original one.
Step 8: Apply binary representation descrambling algorithm to the original signal using a tent map.
Step 9: Save the decrypted speech signal.

4. RESULTS AND DISCUSSION

We conducted a test to evaluate our method using speech samples from a speech library as an LJ test. The library consists of 13100 short sound clips read by one speaker from seven non-fiction books. Sections vary from 1 to 10 seconds. Each audio file is PCM WAV channel 16-bit single channel at a sample rate of 22050 Hz. We performed our tests on a computer processor with Intel Core i5 second-generation speeds of 3.50 GHz, the operating system is Windows 10 (64-bit), and the simulation system is MATLAB R2018b. Figure. 4 shows an example of waveform plotting for speech signal samples in encryption and decryption.



(a) (b) (c)

Figure 4. Waveform plotting for speech signal,
(a) Original speech signal, (b) Encrypted speech signal, (c) Decrypted speech signal

4.1. Quality metrics for proposed algorithms

Several quality metrics are used to evaluate the speech cryptosystem. The purpose of these quality standards is to determine immunity algorithms encryption/decryption attacks and distortion analysis. The measures mention the value of distortion produced by the speech cryptosystem [10].

- Statistical analyses

To analyse our results statistically, we used five different measures: SNR, SNRseg, fwSNRseg, CC, and LLR [20, 21].

1. Signal-to-Noise-Ratio (SNR): To measure the level of distortion in the speech signal. When the value of SNR decreases, the quality of the e encrypted signal is higher, and when the value of SNR increases, the quality of the decrypted signal is higher.
2. Segmental Signal-to-Noise-Ratio (SNRseg): It is an improved version measure of SNR.
3. Frequency-weighed Signal-to-Noise Ratio (fwSNRseg): It is a weighted SNRseg in the frequency band that is proportional to the critical band.
4. Correlation Coefficient (CC): It is the correlation coefficient between the original signal and the distorted signal. The higher the correlation coefficient values between the original signal and the decrypted one, the better the quality of the decryption algorithm.
5. Log-Likelihood Ratio (LLR): It is a measure of the distance that can be calculated directly from the vector of the original LPC and distorted speech.

Tables 1-3 indicate the results of the proposed encryption algorithm. With a higher LLR value, lower (SNR, SNRseg, fwSNRseg) values and low (near to zero) correlation coefficient, the encrypted signal has better quality. The proposed encryption algorithm gives high values, which is a good indicator of low residual speech values that demonstrate high security in the encryption process. Also, Tables 4-6 illustrates the result of the proposed decryption algorithm. One can notice that the SNR and SNRseg scales are very high (values are positive) for each signal decrypted, while the LLR scale has a very small value with no residual clarity, and coded signals are very noisy. Correlation coefficients (CC) are close to +1, which means a high relationship between the original signal and decryption.

Table 1. The measured results of the proposed encryption system speech signal samples length 4 ms

File name	SNR	SNRseg	fwSNRseg	CC	LLR
Sample1.wav	-16.6963	-23.0400	-17.2814	-0.0028	4.3084
Sample2.wav	-16.6736	-23.6020	-17.2252	-0.0025	4.1095
Sample3.wav	-16.4604	-20.9682	-17.6003	-0.0011	4.7375
Sample4.wav	-17.1614	-22.1532	-17.6447	0.0039	4.8608
Sample5.wav	-18.4293	-24.0126	-19.4544	0.0013	4.9798

Table 2. The measured results of the proposed encryption system speech signal samples length 7 ms

File name	SNR	SNRseg	fwSNRseg	CC	LLR
Sample1.wav	-18.4586	-24.7424	-19.3823	-0.0015	4.2646
Sample2.wav	-18.3994	-24.3764	-19.6254	0.0050	4.1528
Sample3.wav	-16.2747	-21.9153	-16.8257	-0.0019	4.5313
Sample4.wav	-17.1345	-23.8644	-17.8153	0.0002	4.2187
Sample5.wav	-20.0106	-26.0432	-21.4523	0.0030	4.2764

Table 3. The measured results of the proposed encryption system speech signal samples length 10 ms

File name	SNR	SNRseg	fwSNRseg	CC	LLR
Sample1.wav	-17.1424	-22.3680	-18.0612	0.0017	4.1681
Sample2.wav	-18.1388	-24.8640	-19.4029	0.0036	3.6384
Sample3.wav	-17.5029	-23.7245	-18.1543	0.0025	4.0983
Sample4.wav	-17.8158	-23.8145	-18.7785	-0.0027	3.9557
Sample5.wav	-17.9246	-23.3474	-18.6160	-0.0001	4.0897

Table 4. The measured results of the proposed decryption algorithm speech signal samples length 4 ms,

File name	SNR	SNRseg	fwSNRseg	CC	LLR
Sample1	27.0188	34.6642	0.0077	0.9999	0.0000257
Sample2	26.3843	34.6365	0.0094	0.9999	0.0000380
Sample3	28.3853	34. 7540	0.0044	0.9999	0.0000192
Sample4	34.3380	34.6468	0.0006	0.9999	0.0000218
Sample5	41.8912	34.7370	0.0003	0.9999	0.0000177

Table 5. The measured results of the proposed decryption algorithm speech signal samples length 7 ms

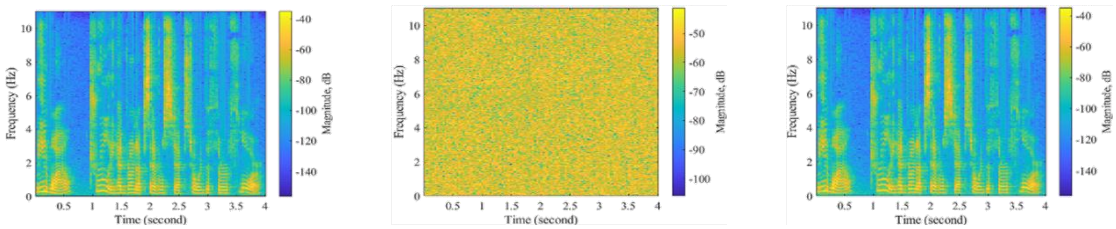
File name	SNR	SNRseg	fwSNRseg	CC	LLR
Sample1	36.1143	34.8062	0.0006	0.9998	0.0000181
Sample2	30.3567	34.8211	0.0038	0.9995	0.0000698
Sample3	35.1766	34.8129	0.0006	0.9998	0.0000284
Sample4	48.1988	34.8364	0.0000	0.9999	0.0000683
Sample5	28.0175	34.8121	0.0043	0.9998	0.0000414

Table 6. The measured results of the proposed decryption algorithm speech signal samples length 10 ms

File name	SNR	SNRseg	fwSNRseg	CC	LLR
Sample1	58.4745	34.8598	0.0000	0.9998	0.0000235
Sample2	43.0488	34.8699	0.0000	0.9999	0.0000580
Sample3	32.1390	34.8664	0.0018	0.9997	0.0000437
Sample4	70.4043	34.8689	0.0000	0.9999	0.0000634
Sample5	32.8558	34.8556	0.0027	0.9997	0.0000136

- Spectrogram analyses

A spectrogram is a tool that splits the speech sample into several blocks of the time domain, then plots each block's rapid Fourier transform and shows them all simultaneously [22]. Figure 5 shows the spectrogram layout of the origin and the signal resulting from applying the proposed encryption system.

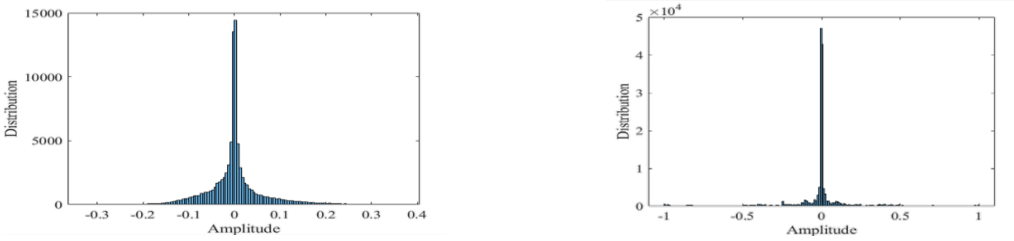


(a) (b) (c)

Figure 5. Waveform and spectrogram for speech signal,
(a) Original signal spectrogram, (b) Encrypted signal spectrogram, (c) Decrypted signal spectrogra

- Histogram analysis

Statistical analysis by histogram can be performed by examining distributions of the data. In cryptographic algorithms, the closer the distributions of encrypted data, the higher the encryption rates [22]. Figure 6 shows the histograms of the original speech signal and the signal resulting from applying the proposed encryption system.



(a) (b)

Figure 6. Histograms of the speech signal, (a) Original signal, (b) Encrypted signal

4.2. Comparison with existing schemes

A comparison is made with other techniques in order to measure the introduced scheme encryption in terms of quality metrics. The comparison is presented in Table 7. In this scheme, a large negative value of SNR, SNRseg, fwSNRseg obtained shows that the noise power is higher than the signal power, which makes it difficult to detect. Furthermore, the value of correlation in this technique is almost zero, which indicates that the original and the encrypted speech signals are uncorrelated. Finally, LLR in this technique is higher compared to others, which shows higher encryption quality. It can be concluded that in most of the quality measures, the presented speech encryption scheme outperforms other existing schemes.

Table 7. Comparison of proposed method with existing schemes

Paper	SNR (dB)	SNRseg (dB)	fwSNRseg(dB)	CC	LLR
Elshamy et al. [23]	-	-	-	0.0051	-
Al Saad and Hato [24]	-	-17.70727	-16.9303	-0.009221	-0.009221
Wahab and Mahdi [25]	-13.3816	-14.3917	-	-5.64×10-4	
Sathiyamurthi and Ramakrishnan [26]	-	-	-	0.0233	-
Hato and Shihab [21]	-13.1761			0.0040266	2.492488
Farsana and Gopakumar [27]	-24.32			0.000289	
Hato [28]	-14.55949			-0.005108	
Farsana et al. [29]	-12.76			0.0613	
Sreedharan and Eswaran [30]	-23			0.00569	
Proposed algorithm	-23.0106	-26.0432	-24.4523	0.0010	4.3764

4.3. Keyspace and sensitivity analysis

One of the important parameters for cryptographic system performance is the keyspace analysis. A perfect algorithm must have a large keyspace and be sensitive to the key value.

- Keyspace

The keyspace of the coding system must be large enough to tolerate a brute force attack. It is accepted that a keyspace larger than (2^{128}) is mathematically safe against this attack. In our system, control parameters and initial conditions are the secret keys to the encryption method. For a 10^{-7} floating-point precision, each key parameter (five initial conditions (X, Y, Z, X*, Z*), two control parameters (β, r), two permutation quantum chaotic maps and 65 initial conditions (64 centroids cluster, one index vector) can take 10^{7} possible values. Thus, the keyspace comes out as $(107)^{72} \approx 2^{1512}$, which is large enough to resist the brute force attack.

- Key sensitivity

A good encryption algorithm must be sensitive to all secret keys so that there is a large variation in the output even if there is a slight change in the keys. The sensitivity of the key means that if there is any alteration between the encryption key and the decryption key, you cannot decrypt the encrypted signal that is encrypted properly. To evaluate the key sensitivity of the proposed algorithms, we conducted numerous experiments to test the effect of changing the sensitivity of the key. We changed the control parameters (β, r) and initial conditions parameters of the quantum chaotic map (X, Y, Z, X*, Z) by a small amount for each one alone, maintaining the other parameters of the keys constant in each experiment. The results are tabulated in Table 8 and show the SNR, SNRseg, fwSNRseg, LLR and CC measures are estimated between the original signal and the signal decrypted with the changed key. Low SNR, CC value (closest CC to zero), and large LLR values prove the key sensitivity of our system. One of these tests proves that the proposed speech encryption algorithm is very sensitive to the secret key. Figure 7 shows the results of experiments to ascertain the effect of changing the sensitivity of the key.

Table 8. Test for key sensitivity of speech signal sample length (4ms)

Parameter	SNR	SNRseg	fwSNRseg	CC	LLR
β =4	-25.6540	-27.8718	-27.0788	0.2249	4.5599
r=3	-24.7482	-26.9452	-26.1561	0.2336	4.5471
X* = 0.001	-26.9748	-24.8809	-26.4561	0.2257	4.5376
Z* = 0.001	-24.7206	-25.9193	-27.1932	0.2310	4.5596
X(1)=0.46	-25.5550	-24.8536	-26.0595	0.2233	4.5532
Y(1)=0.001	-24.8482	-26.8819	-26.3298	0.2196	4.5567
Z(1)=0.002	-26.8506	-25.8861	-26.2853	0.2281	4.5427

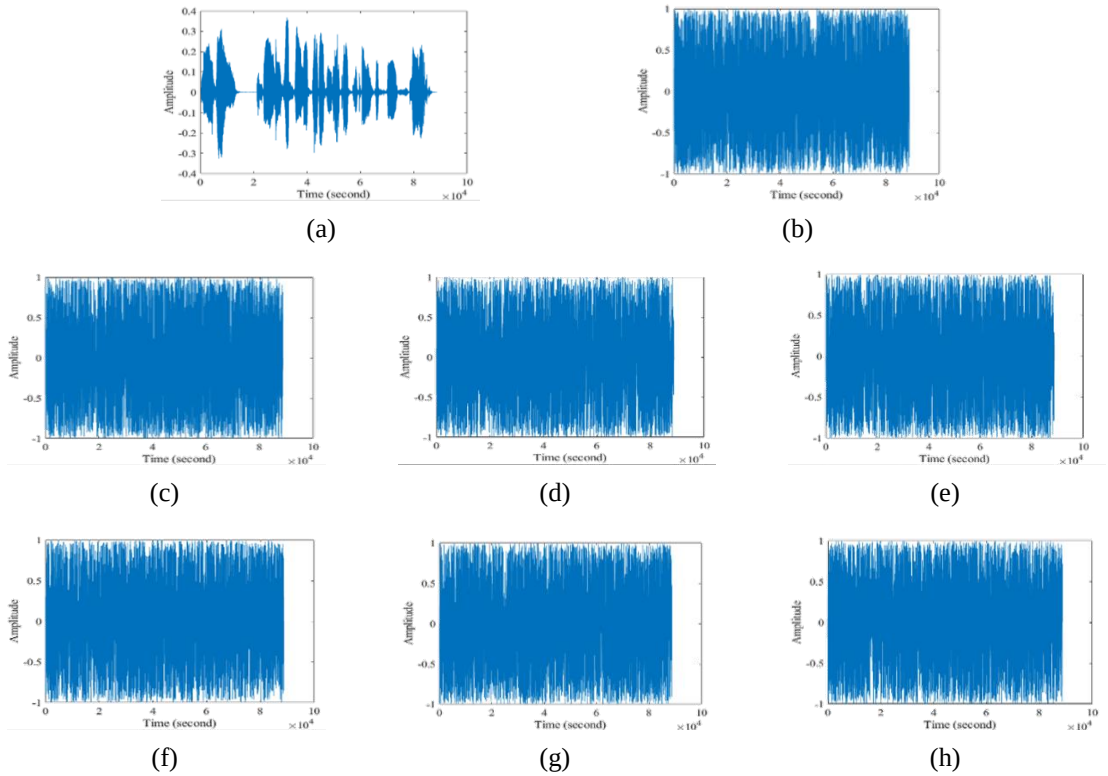


Figure 7. Results of experiments to test the effect of changing the sensitivity of the key, (a) The original signal, (b) β =4 for 4ms, (c) r=3, (d) X*, (e) Z*, (f) X1, (g) Y1, (h) Z1

5. CONCLUSION

Protecting audio systems from illegal access, alteration, and extermination is the main target of speech security. In this paper, a new method of speech encryption scheme relies on a quantum chaotic map to generate a large number of total keys as the primary key, which makes the method safe from brute force attack. Moreover, we rely on the k-means algorithm, with which we generate a secondary key for each block of samples separately, to ensure robust encryption and difficult-to-analyse keys. The scrambling and permutation increase the speech signal security and make cryptanalysis a difficult task. So our work includes two stages, the first of which is scrambling of the values by relying on their bits according to a proposed algorithm (binary representation scrambling BiRS), and the other level of the scramble relies on k-means to scramble the parts of each block according to the code that we get by using a proposed algorithm (block representation scrambling BlRS). The proposed algorithms in this paper encrypt the original signal using multiple levels (binary representation, blocks representation). This achieves good scrambling and good immunity of the speech signal to different attacks, which makes cryptanalysis difficult and increases speech safety. Key sensitivity refers to convenience for our algorithms and a high level of security, and the keyspace is so huge that it makes a brute-force attack inapplicable. The simulation results show that our algorithm returns a perfect level of the security system and a top-quality recovered speech.

REFERENCES

[1] H. Kaur and G. S. Sekhon, "A four level speech signal encryption algorithm," *International Journal of Computer Science and Communication*, vol. 3, no. 2, pp. 151-153, 2012.

[2] C. L. Duta, L. Gheorghe, N. Tapus, "Real-time DSP implementations of voice encryption algorithms," *Proceedings of the 3rd International Conference on Information Systems Security and Privacy*, pp. 439-446, 2017.

[3] L. Hongjun and W. Xingyuan, "Color image encryption based on one-time keys and robust chaotic maps," *Computers & Mathematics with Applications*, vol. 59, no. 10, pp. 3320-3327, 2010.

[4] A. Anees, "An image encryption scheme based on lorenz system for low profile applications," *3D Research*, vol. 3, no. 6, p. 24, 2015.

[5] A. Akgül, S. Kacar, and I. Pehlivan, "An audio data encryption with single and double dimension discrete-time chaotic systems discrete-time chaotic systems used in encryption applications," *Online Journal of Science and Technology*, vol. 5, no. 3, pp. 14-23, 2015.

[6] A. Agarwal, P. R. Singh, and S. Katiyar, "Secured audio encryption using AES algorithm," *International Journal of Computer Applications*, vol. 178, no. 22, pp. 29-33, 2019.

[7] Z. M. Alroubaie, M. A. Hashem, and F. S. Hasan, "FPGA design of encryption speech system using synchronized fixed-point chaotic maps based stream ciphers," *International Journal of Engineering and Advanced Technology*, vol.8, no.6, pp.1534-1541, 2019.

[8] K. Kordov, "A novel audio encryption algorithm with permutation-substitution architecture," *Electron*, vol. 8, no. 5, p. 530-1-15, 2019.

[9] N. A. Abbas and Z. H. Razaq, "Review of dct and chaotic maps in speech scrambling," *Journal of Theoretical and Applied Information Technology*, vol. 97, no. 2, pp. 569-582, 2019.

[10] M. Farouk, O. Faragallah, O. Elshakankiry, and A. Elmhalaway, "Comparison of audio speech cryptosystem using 2-D chaotic map algorithms," *Mathematics and Computer Science*, vol. 1, no. 4, pp. 66-81, 2016.

[11] E. A. Albahrani, "A new audio encryption algorithm based on chaotic block cipher," *2017 Annual Conference on New Trends in Information & Communications Technology Applications (NTICT)*, pp. 22-27, 2017.

[12] A. Zaghloul, T. Zhang, M. Amin, and A. A. Abd El-Latif, "Color encryption scheme based on adapted quantum logistic map," *Sixth International Conference on Digital Image Processing*, vol. 9159, p. 915922, 2014.

[13] M. E. Goggin, Sundaram, and P. W. Milonni, "Quantum logistic map," *Physical Review A*, vol. 41, no. 10, pp. 5705-5708, 1990.

[14] A. Akhshani, A. Akhavan, A. Mobaraki, S.-C. Lim, and Z. Hassana, "Pseudo random number generator based on quantum chaotic map," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 1, pp. 101-111, 2014.

[15] H. Liu, B. Zhao, and L. Huang, "A novel quantum image encryption algorithm based on crossover operation and mutation operation," *Multimedia Tools and* Applications, vol.78, no.14, pp. 20465-20483, 2019.

[16] B. M. Hennelly and J. T. Sheridan, "Image encryption and the fractional fourier transform," *Optik*, vol. 114, no. 6, pp. 251-265, 2003.

[17] E. Sejdić, I. Djurović, and LJ. Stanković, "," *Signal Processing*, vol. 91, no. 6, pp. 1351-1369, 2011, 2010.

[18] B. Imane, T. A. Youssef, and T. A. Mohammed, "Determination of initial centroid in K-means using PCA factor scores," *Journal of Theoretical and Applied Information Technology*, vol. 96, no. 19, pp. 6597-6606, 2018.

[19] B. Sugiantoro and M. H. Kiswanto, "Analysis email content with K-means clustering for profiling on postfix server," *Journal of Theoretical & Applied Information* Technology, vol. 95, no. 17, pp. 4103-4113, 2017.

[20] S. F. Yousif, "Speech encryption based on zaslavsky map," *Journal of Engineering and Applied Sciences*, vol. 16, no. 7, pp. 6392-639, 2019.

[21] E. Hato and D. Shihab, "Lorenz and rossler chaotic system for speech signal encryption," *International Journal of Computer Applications*, vol. 128, no. 11, pp. 25-33, 2015.

[22] M. F. A. Elzaher, M. Shalaby, Y. Kamal, and S. H. Elramly, "A speech cryptosystem based on chaotic modulation technique," vol. 4, no. 1, pp. 1-10, 2017.
[23] E. M. Elshamy, El-S. M. El-Rabaie, O. S. Faragallah, O. A. Elshakankiry, F. E. A. El-Samie, H. S. El-sayed, and S. F. El-Zoghdy, "Efficient audio cryptosystem based on chaotic maps and double random phase encoding," *International Journal of Speech Technology*, vol. 18, no. 4, pp. 619-631, 2015.
[24] S. N. AlSaad and E. Hato, "A speech encryption based on chaotic maps," *International Journal of Computer Applications*, vol. 93, no. 4, pp. 19-28, 2014.
[25] H. B. A. Wahab and S. I. Mahdi, "Modify speech cryptosystem based on shuffling overlapping blocks technique," *International Journal of Emerging Trends & Technology in Computer Science*, vol. 4, no. 2, pp. 70-75, 2015.
[26] P. Sathiyamurthi and S. Ramakrishnan, "Speech encryption using chaotic shift keying for secured speech communication," *EURASIP Journal on Audio, Speech, and Music Processing*, no. 1, pp. 1-11, 2017.
[27] F. J. Farsana and K. A. Gopakumar, "Novel approach for speech encryption: Zaslavsky map as pseudo random number generator," *Procedia Computer Science*, vol. 93, pp. 816-823, 2016.
[28] E. Hato, "Cellular automata and chaotic maps for speech signal encryption," *International Journal of Applied Information System*, vol. 9, no. 3, pp. 9-16, 2015.
[29] F. J. Farsana, K. Gopakumar AnuAssis, "Speech encryption based on two dimensional maps," *International Journal of Advanced Engineering & Science Research*, vol. 4, no. 1, pp. 101-103, 2017.
[30] S. Sreedharan and C. Eswaran, "Speech encryption using advanced encryption standard for secured communication," *International Journal of Recent Technology and Engineering*, vol. 8, no. 3, pp. 6515-6521, 2019.