ADVANCES IN MEMRISTOR AND MEMRISTOR-BASED APPLICATIONS

EDITED BY: Jun Mou, Shaobo He and Chunlai Li
PUBLISHED IN: Frontiers in Physics

frontiers Research Topics

ISSN 1664-8714
ISBN 978-2-83250-194-8
DOI 10.3389/978-2-83250-194-8

About Frontiers

Frontiers is more than just an open-access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers Journal Series

The Frontiers Journal Series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the Frontiers Journal Series operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to Quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews.
Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view.
By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the Frontiers Journals Series: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area! Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers Editorial Office: frontiersin.org/about/contact

ADVANCES IN MEMRISTOR AND MEMRISTOR-BASED APPLICATIONS

Topic Editors:
Jun Mou, Dalian Polytechnic University, China
Shaobo He, Central South University, China
Chunlai Li, Hunan Institute of Science and Technology, China

Citation: Mou, J., He, S., Li, C., eds. (2022). Advances in Memristor and Memristor-Based Applications. Lausanne: Frontiers Media SA.
doi: 10.3389/978-2-83250-194-8

Table of Contents

frontiers | Frontiers in Physics

TYPE Editorial
PUBLISHED 30 August 2022
DOI 10.3389/fphy.2022.1005216

OPEN ACCESS

EDITED AND REVIEWED BY
Alex Hansen,
Norwegian University of Science and Technology, Norway

*CORRESPONDENCE
Jun Mou,
moujun@csu.edu.cn

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 28 July 2022
ACCEPTED 03 August 2022
PUBLISHED 30 August 2022

CITATION
Mou J (2022), Editorial: Advances in memristor and memristor-based applications.
Front. Phys. 10:1005216.
doi: 10.3389/fphy.2022.1005216

Editorial: Advances in memristor and memristor-based applications

Jun Mou*

School of Information Science and Engineering, Dalian Polytechnic University, Dalian, China

KEYWORDS

memristor, meminductor, memcapacitor, memristor-based applications, image encryption

Editorial on the Research Topic
Advances in memristor and memristor-based applications

This is my first editorial as Editor of Frontiers in Physics. I am honoured to be given this opportunity and I am looking forward to working with the Frontiers in Physics more in the future. Thank you very much to my collaborator, Professor Shaobo and Professor Chunlai Li. They have made a remarkable contribution to this Research Topic, *Advances in Memristor and Memristor-Based Applications*.

As we all know, in the past two and a half years, COVID-19 has seriously affected the development of scientific research in all aspects. However, there are still a number of scientists who overcome all difficulties and continue to work at the forefront of scientific research, their work ensures that considerable progress has been made in the study of Memristors and Memristor-Based Applications.

The memristor is an electronic component that has a memory effect on the amount of charge passing through it, and it is used in many fields. The characteristics of small size, low energy consumption and non-volatility make memristors efficient for information storage. It can be used in storage and computing fusion design, artificial intelligence and neural networks. Thus, it is very important to construct memristors and explore research in related fields for electronic information and intelligent science. And in-depth research on the memristor, meminductor and memcapacitor have been proposed, which not only improve the circuit theory, but also provide more application directions for memory elements.

Research on memristors has always been a hot issue. With the gradual deepening of research, memristor theory has become an interdisciplinary subject related to materials, physics and electronic information. Therefore, the research progress related to memristors has gradually accelerated, and a large number of valuable results have been obtained. However, there are still some problems and difficulties when disciplines cross. Therefore, we organize this Research Topic to discuss new situations and new challenges in memristors and applications based on memristors in the hope of showing some interesting research results.

In this Research Topic, 13 articles were published, these articles have been carefully reviewed and are related to memristors. Such as.

1) *The Simplest Memristor Circuit With Hyperchaos*
2) *Parameter Identification for Memristive Chaotic System Using Modified Sparrow Search Algorithm*
3) *A Chaotic System With Infinite Attractors Based on Memristor*
4) *A New Four-Dimensional Chaotic System and its Circuit Implementation*
5) *Dynamical Analysis of Two-Dimensional Memristor Cosine Map*
6) *Design of Grid Multi-Wing Chaotic Attractors Based on Fractional-Order Differential Systems*
7) *Coexistence of Multiple Attractors in a Novel Simple Jerk Chaotic Circuit With CFOAs Implementation*

These seven papers are based on memristor chaotic system design, fractional chaotic system design, system hardware implementation and parameter identification. The research of these articles is very distinctive and innovative, which provides new ideas and methods for the design and characteristic research of memristor chaotic system.

Some articles such as.

1) *A Novel 3D Image Encryption Based on the Chaotic System and RNA Crossover and Mutation*
2) *A Chaos-Based Image Encryption Scheme Using the Hamming Distance and DNA Sequence Operation*
3) *Multi-Image Encryption Algorithm for 2D and 3D Images Based on Chaotic System*
4) *A Novel Image Encryption Scheme Based on Memristive Chaotic System and Combining Bidirectional Bit-level Cyclic Shift and Dynamic DNA-level Diffusion*

These four papers focus on the application of memristor chaotic systems in secure communication, mainly in the field of image encryption. This is a research hotspot at the present stage, and the encryption algorithms proposed in these papers have good security and good application prospects.

Other papers such as.

1) *Detecting Boosting Weak Signal via A Meminductive Multistable Chaotic System*
2) *Classifying Beers With Memristor Neural Network Algorithm in a Portable Electronic Nose System*

These two papers apply memristor to signal detection and neural networks respectively, which are very novel. The application field of memristor is expanded.

Finally, I'm appealing to researchers and readers to support Frontiers in physics. Please consider submitting your best research to Frontiers in physics.

Thanks to all the staff of Frontiers in Physics, editors and reviewers.

Author contributions

The author confirms being the sole contributor of this work and has approved it for publication.

Conflict of interest

The author declares that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

frontiers
in Physics

ORIGINAL RESEARCH
published: 07 February 2022
doi: 10.3389/fphy.2022.835188

Coexistence of Multiple Attractors in a Novel Simple Jerk Chaotic Circuit With CFOAs Implementation

Qiao Wang[1,2], Zean Tian[1,3]*, Xianming Wu[4] and Weijie Tan[5]

[1]Institute of Advanced Optoelectronic Materials, Technology of School of Big Data and Information Engineering, Guizhou University, Guiyang, China, [2]College of Mathematics and Big Data, Guizhou Education University, Guiyang, China, [3]College of Computer Science and Electronic Engineering, Hunan University, Changsha, China, [4]School of Mechanical and Electrical Engineering, Guizhou Normal University, Guiyang, China, [5]State Key Laboratory of Public Bigdata, Guizhou University, Guiyang, China

A novel, simple Jerk chaotic circuit with three current feedback operational amplifiers included (CFOA-JCC) is proposed, which has a simpler circuit structure, fewer components, but higher frequency characteristics. The dynamic behaviors of CFOA-JCC are analyzed, including equilibrium, stability, Lyapunov exponent, bifurcation diagram, offset boosting, and phase diagram. Furthermore, the frequency spectrum characteristic of the ordinary op-amps Jerk chaotic circuit was compared with CFOA-JCC under the same circuit parameters, and the chaotic attractor frequency of CFOA-JCC can reach about 650 kHz, much better than that of ordinary op-amps (12 kHz). Numerical simulation shows that CFOA-JCC has coexisting attractors, verified by hardware circuit experiments.

Keywords: Jerk, chaotic, coexisting attractors, CFOA, frequency spectrum

Edited by:
Jun Mou,
Dalian Polytechnic University, China

Reviewed by:
Quan Xu,
Changzhou University, China
Chunbiao Li,
Nanjing University of Information Science and Technology, China

****Correspondence:***
Zean Tian
tianzean@hnu.edu.cn

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *14 December 2021*
Accepted: *14 January 2022*
Published: *07 February 2022*

Citation:
Wang Q, Tian Z, Wu X and Tan W (2022) Coexistence of Multiple Attractors in a Novel Simple Jerk Chaotic Circuit With CFOAs Implementation. Front. Phys. 10:835188. doi: 10.3389/fphy.2022.835188

INTRODUCTION

Chaos is a complex nonlinear phenomenon with special dynamical properties, and various chaotic systems are found and widely used in the fields of electronic communication systems, control systems, and so on [1–6]. Chaotic circuits have become an important tool for studying chaos theory due to easy observation. Proposed by Sprott based on the method of computer exhaustive, the Jerk chaotic system (a class of third-order autonomous chaotic systems) has attracted much attention because it is easy to achieve complex dynamical behavior with a simple structure. The general mathematical expression is $\dddot{x} = J(\ddot{x}, \dot{x}, x)$ [7–10]. In 2011, Sprott proposed a novel chaotic circuit of Jerk, which realizes a chaotic system with only resistance, capacitance, operational amplifier, and diode. The three first-order differential equation is as follows [11]:

.

$$\dddot{x} + \ddot{x} + x + f\left(\dot{x}\right) = 0 \tag{1}$$

The nonlinear term in the chaotic circuit can be nonlinear functions, piecewise linear functions, and mersister generation [12–14]. However, most Jerk circuit implementations use traditional voltage mode op-amps as active devices, and the gain of the designed circuit must be decreased to increase bandwidth. For the current feedback operational amplifier (CFOA), there is almost no such relationship between gain and bandwidth. Therefore, it is not necessary to design a circuit to balance the gain and bandwidth as a voltage feedback amplifier [15–17]. Therefore, using CFOA to realize chaotic circuits has become a hotspot because CFOA has better frequency gain characteristics [18–24]. It makes the circuit structure simpler and more flexible due to its good port characteristics [25–29]. There have been many reports

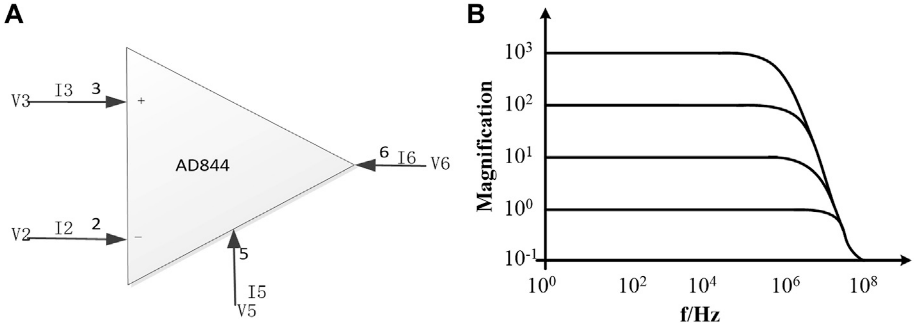


FIGURE 1 | AD844AN circuit symbol and its frequency characteristic. **(A)** AD844AN circuit symbol. **(B)** Magnification frequency characteristic curve of AD844AN.

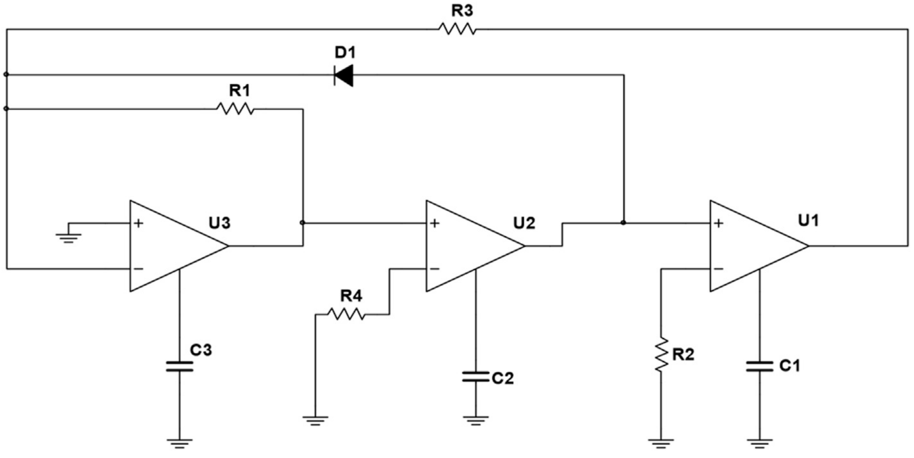


FIGURE 2 | The novel Jerk circuit with three CFOAs AD844AN.

about the coexistence attractors in recent years, but all the reports did not pay much attention to the simpler topology of the chaotic circuit. For example, in 2011, Sprott proposed a Jerk chaotic circuit with traditional voltage feedback operational amplifiers (VFOA). He did not discuss the coexistence of attractors of his system, and the circuit was not the simplest one. In 2016, Kengene et al. proposed a novel chaotic Jerk circuit, which can realize the coexistence of multiple attractors [30]. Five amplifiers and two diodes had been used in the circuit, and they discussed the frequency of the circuit. In their circuit, the center frequency can be obtained at about 4 Hz. In 2020, Xu et al. proposed a memristive Chua's circuit with attractors with two stable foci [14]. A one-dimensional multi-scroll chaotic circuit is designed with CFOA [26], but the circuit structure is complex, and there are many active devices. A grid multi-scroll chaotic circuit is designed using CFOA [27, 28], but more active devices are used with lower operating frequency. Wu proposed a chaotic circuit only using CFOA [29], but the center frequency of the chaotic signal is 250 kHz.

In this article, a novel Jerk chaotic circuit with three CFOAs (called CFOA-JCC) was designed to realize a chaotic system in Description of the CFOA-JCC. In Equilibrium Point Analysis, based on the equation of state of CFOA-JCC, the equilibrium point and the system dissipation are analyzed by solving the dimensionless equation. In Dynamical Behaviors, numerical analysis and description were done using MATLAB R2018b to perform the numerical simulation. Lyapunov exponent spectra and bifurcation diagram are plotted in adjusting the parameter regions to reveal the dynamical behaviors. The occurrence of coexisting attractors at deferent initial conditions is also discussed using phase diagrams and bifurcation. Single parameter-controlled offset boosting for variable x is simulated. In Circuit Implementation and Spectrum Characteristics, hardware experiments are performed, the phase and offset boosting diagrams in the x-y plane captured by the UNI-T2102CM oscilloscope strongly confirm the previous theoretical analysis, and then the spectrum of CFOA-JCC is compared with that of the VFOA circuit in adjusting the parameter regions of capacitance under the same value of the resistor. The conclusions are summarized in the last section.

DESCRIPTION OF THE CFOA-JCC

AD844AN is a typical CFOA with better frequency properties. Its circuit symbol is shown in **Figure 1A**. Under ideal conditions, the port characteristics of AD844AN are satisfied:

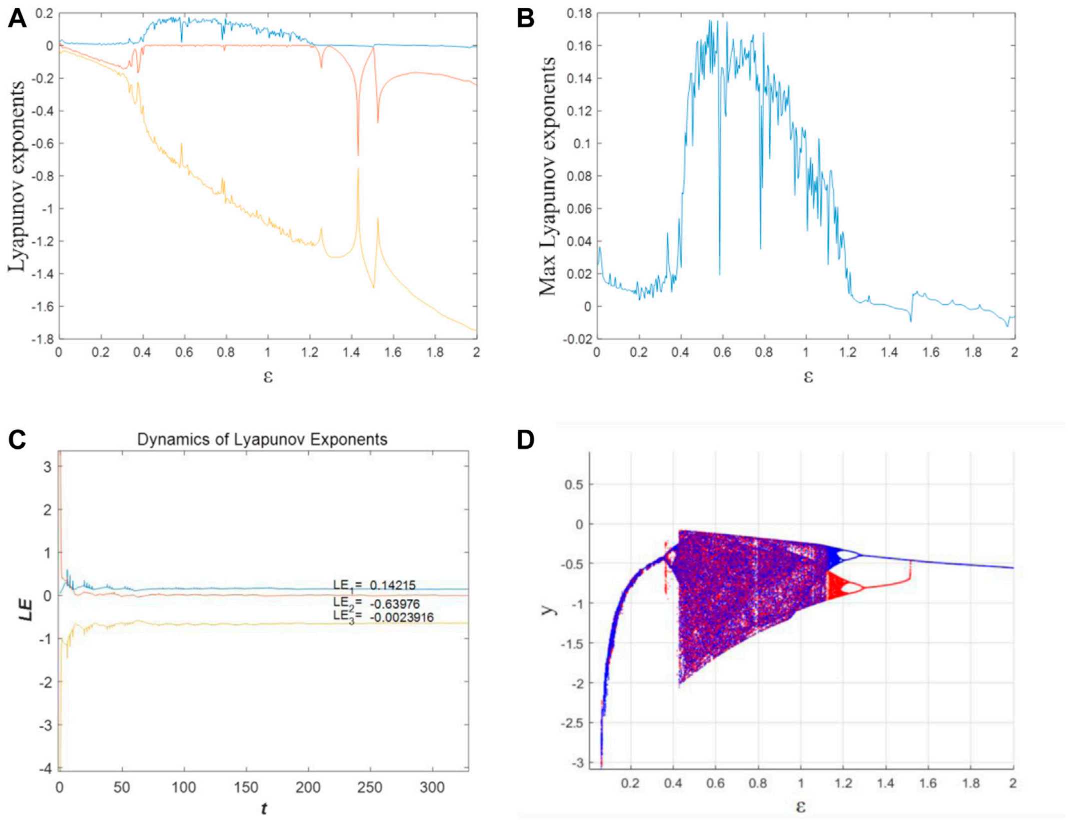


FIGURE 3 | Lyapunov exponent spectrum and bifurcation diagram at different initial values. **(A)** Lyapunov exponents at $(x_0, y_0, z_0) = (0, 0.4, 0)$. **(B)** Max Lyapunov exponent at $(x_0, y_0, z_0) = (0, 0.4, 0)$. **(C)** LEs when $\varepsilon = 0.5$ at $(x_0, y_0, z_0) = (0, 0.4, 0)$. **(D)** Bifurcation diagram at $(x_0, y_0, z_0) = (0, 0.4, 0)$ (red) and $(x_0, y_0, z_0) = (0, -0.4, 0)$ (blue).

$$\begin{cases} I_3 = 0 \\ V_2 = V_3 \\ I_2 = \pm I_3 \\ V_5 = V_6 \end{cases} \tag{2}$$

.

With the current output port, the current state of CFOA is easy to observe. As can be seen from the characteristic curve in **Figure 1B**, the gain of both op-amps is stable when the frequency is within 100 kHz. AD844AN has a better high-frequency performance than VFOA. The closed-loop bandwidth frequency at −3 dB can be expressed as $f_{cl} = \frac{1}{2\pi C_P R_f}$. The closed-loop bandwidth of the CFOA is determined by the feedback resistor R_f and the internal circuit C_p. It illustrates the importance of the feedback resistance R_f of the CFOA, so R_f can be used to adjust the frequency response of the amplifier [31].

This designed circuit configuration is shown in **Figure 2**, which includes three CFOAs; three capacitors C_1, C_2, C_3; four resistors R_1, R_2, R_3, R_4, and a silicon diode named D_1. This circuit is derived from the CFOAs-based sinusoidal oscillator [32], and a nonlinear function is obtained according to volt-ampere characteristics of the diode.

EQUILIBRIUM POINT ANALYSIS

According to Kirchhoff's Current Law and Kirchhoff's Voltage Law, the port characteristics of CFOA, and the current-voltage relations of each circuit component of the novel circuit in **Figure 2**, in terms of the three-node voltage of u_1, u_2, u_3, three coupled first-order autonomous nonlinear differential equations can be attained to express the nonlinear dynamics of the circuit. The differential equations of the proposed system can be written as

$$\begin{cases} \frac{du_1}{dt} = \frac{1}{R_2C_1}u_2 \\ \frac{du_2}{dt} = \frac{1}{R_4C_2}u_3 \\ \frac{du_3}{dt} = -\frac{1}{R_3C_3}u_1 - \frac{1}{C_3} \cdot I_S\left(e^{\frac{u_2}{V_T}} - 1\right) - \frac{1}{R_1C_3}u_3 \end{cases} \tag{3}$$

.

The voltage signals u_1, u_2, u_3 and time variable t are non-dimensional to obtain x, y, z through the following relationship [33]:

$$x \cdot 1V = u_1, y \cdot 1V = u_2, z \cdot 1V = u_3,$$
$$\alpha = \frac{R_2C_1}{R_2C_1}, \beta = \frac{R_2C_1}{R_4C_2}, \gamma = \frac{R_2C_1}{R_3C_3}, \delta = \frac{R_2C_1}{C_3}, \varepsilon = \frac{R_2C_1}{R_1C_3}, \tau = \frac{t}{R_2C_1} \tag{4}$$

When it is at room temperature 300 K, the diode reverse saturation current $I_S = 10^{-9}$ A and the voltage equivalent of temperature $V_T = 26$ mV [34]. Therefore, **Eq. 3** can be written as

$$\begin{cases} \dot{x} = \alpha y \\ \dot{y} = \beta z \\ \dot{z} = -\gamma x - \delta \cdot 10^{-9}\left(e^{\frac{y}{0.026}} - 1\right) - \varepsilon z \end{cases} \tag{5}$$

Assuming $\dot{x} = 0$, $\dot{y} = 0$, and $\dot{z} = 0$ to analyze the equilibrium point and stability, **Eq. 3** has a zero equilibrium point $P_0 = (0,0,0)$ [35].

$$\begin{cases} \alpha y = 0 \\ \beta z = 0 \\ -\gamma x - \delta \cdot 10^{-9}\left(e^{\frac{y}{0.026}} - 1\right) - \varepsilon z = 0 \end{cases} \tag{6}$$

.

The Jacobin matrix at the equilibrium point can be derived as

$$J_{(P_0)} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \beta \\ -\gamma & -0.026 \times 10^{-9}\delta & -\varepsilon \end{bmatrix} \tag{7}$$

.

The characteristic equation at the equilibrium is yielded as

$$Det\left(\boldsymbol{E\lambda} - \boldsymbol{J}_{P_0}\right) = \lambda^3 + \varepsilon\lambda^2 + 2.6 \times 10^{-11}\beta\delta \cdot \lambda + \beta\gamma = 0 \tag{8}$$

The Routh–Hurwitz conditions for the above cubic polynomial **Eq. 8** are given by [36–39]

$$\begin{cases} \gamma > 0 \\ \beta\gamma > 0 \\ E \times 2.6 \times 10^{-11}\beta\delta - \beta\varepsilon > 0 \end{cases} . \tag{9}$$

Obviously, the first and second conditions are satisfied in this system, but the third condition is only satisfied when ε and δ take very large values. Then, the system can be stable. Nevertheless, the system is unstable when the value of ε and δ is small, and the above characteristic equation has both positive and negative roots.

The circuit parameters in **Figure 2** are given as $R_1 = R_2 = R_3 = R_4 = 1\ k\Omega$ and $C_1 = C_2 = C_3 = 100nF$. Therefore, the normalized parameters can be calculated by **Eq. 4** as $\alpha = 1, \beta = 1, \gamma = 1, \delta = 1$, and $\varepsilon = 1$, so that the characteristic equation at the equilibrium is yielded as

$$Det\left(E\lambda - J_{P_0}\right) = \lambda^3 + \lambda^2 + 2.6 \cdot 10^{-11} \cdot \lambda + 1 = 0 \tag{10}$$

.

Three eigenvalues can be obtained from **Eq. 10**: λ1 = −1.4665571, λ2 = 0.232786 + 0.792552i, and λ3 = 0.232786 + 0.792552i. Among them, λ2 and λ3 are complex roots, and λ1 is a negative real root. From the three eigenvalues, it can be seen that P0 (0, 0, 0) is an unstable saddle point. Consequently, system (3) may produce chaotic behaviors [40–42]. System (3) is a three-dimensional system, and the divergence of its vector field can be described as

$$\nabla V = \frac{\partial \dot{x}}{\partial x} + \frac{\partial \dot{y}}{\partial y} + \frac{\partial \dot{z}}{\partial z} = -\varepsilon = -1 \tag{11}$$

.

It shows that the system is dissipative. When t→∞, each volume element of the trajectory of the system shrinks to 0 at an exponential rate of −1, and its gradual motion will be fixed on an attractor [43–46].

DYNAMICAL BEHAVIORS

With fixed parameters $\alpha = 1, \beta = 1, \gamma = 1, \delta = 1$, and the initial values (x0, y0, z0) = (0, 0.4, 0) and (x0, y0, z0) = (0, −0.4, 0) lying close to the equilibrium point, the Lyapunov exponent (LE) spectrum and bifurcation diagram of the system varying with parameter $\varepsilon \in$ [0.001, 2] can be obtained as shown in **Figures 3A,B,D**. In this article, all the numerical simulations are made in MATLAB R2018b, the simulation step size is 0.01, the ode45 numerical solver is used, and the simulation time is 500 s.

The system can maintain a chaotic state when the parameter ε changes within a certain range. The value of the max LE of the system is approximately 0 when $\varepsilon \in [0.001, 0.394)$, which means the system is in a period state. The bifurcation diagram shown in **Figure 3D** is represented by one or several curves. When $\varepsilon \in$ [0.0.39, 0.394), the system produces a bifurcation, which evolves from period 1 to period 2, respectively. When $\varepsilon \in [0.394, 1.2)$, there is one positive LE, which means the system is in a chaotic state. For instance, as **Figure 3C** shows, when $\varepsilon = 0.5$, the LEs can be calculated as LE_1 = 0.14215, LE_2 = −0.63976, LE_3 = −0.0023916. LE_1 is positive, and LE_2 and LE_3 are negative, meaning the system's phase volume is exponentially shrinking. According to the Kaplan–York dimension (D_{KY}), an integer j must satisfy the following requirement, $\sum_{i=1}^{j} LE_i \geq 0$, $\sum_{i=1}^{j+1} LE_i < 0$, so D_{KY} of the chaotic state can be calculated as follows:

$$D_{KY} = j + \frac{\sum_{i=1}^{j} LE_i}{\left|LE_{j+1}\right|} = 1 + \frac{\sum_{i=1}^{1} LE_i}{\left|LE_2\right|} = 2.56308 \tag{12}$$

.

In **Figure 3D**, the bifurcation diagram also proves the system is chaotic when ε ∈ [0.394, 1.2). With the increase in ε, the system returns to the periodic state when ε ≥ 1.2. From **Figure 3D**, we also can obtain that there are two different attractors when the initial value is fixed $(x_0, y_0, z_0) = (0, 0.4, 0)$ and $(x_0, y_0, z_0) = (0, -0.4, 0)$. That means there are coexisting attractors in this system.

Figure 4 shows the phase diagram of the system corresponding to different ε values. The process of the system from period bifurcation to chaos and then back to the periodic state can be clearly seen from the diagrams. The trajectories colored in red start from the initial condition $(x_0, y_0, z_0) = (0, 0.4, 0)$ and those colored in blue correspond to $(x_0, y_0, z_0) = (0, -0.4, 0)$. For the two initial conditions, it can be observed from **Figure 4** that when ε = 0.5, the chaotic attractor is obtained, as shown in **Figures 4A1, A2**. Then as ε increases, the system returns to the periodic state, but the period attractors are different under two initial conditions, which are shown in **Figures 4B1,B2,C1,C2**.

The state variable x appears once in the second equation of **Eq. 6**. Consequently, the state variable x can be easily boosted. In system (5) [47, 48], we fixed the parameters as $\alpha = 1, \beta = 1, \gamma = 1, \delta = 1$, and $\varepsilon = 1$ and added a parameter k, which can realize the offset boosting without changing the basic dynamics as the substitution of x → x + k transforms it to its original form at $k = 0$. Then, the equation can be written as

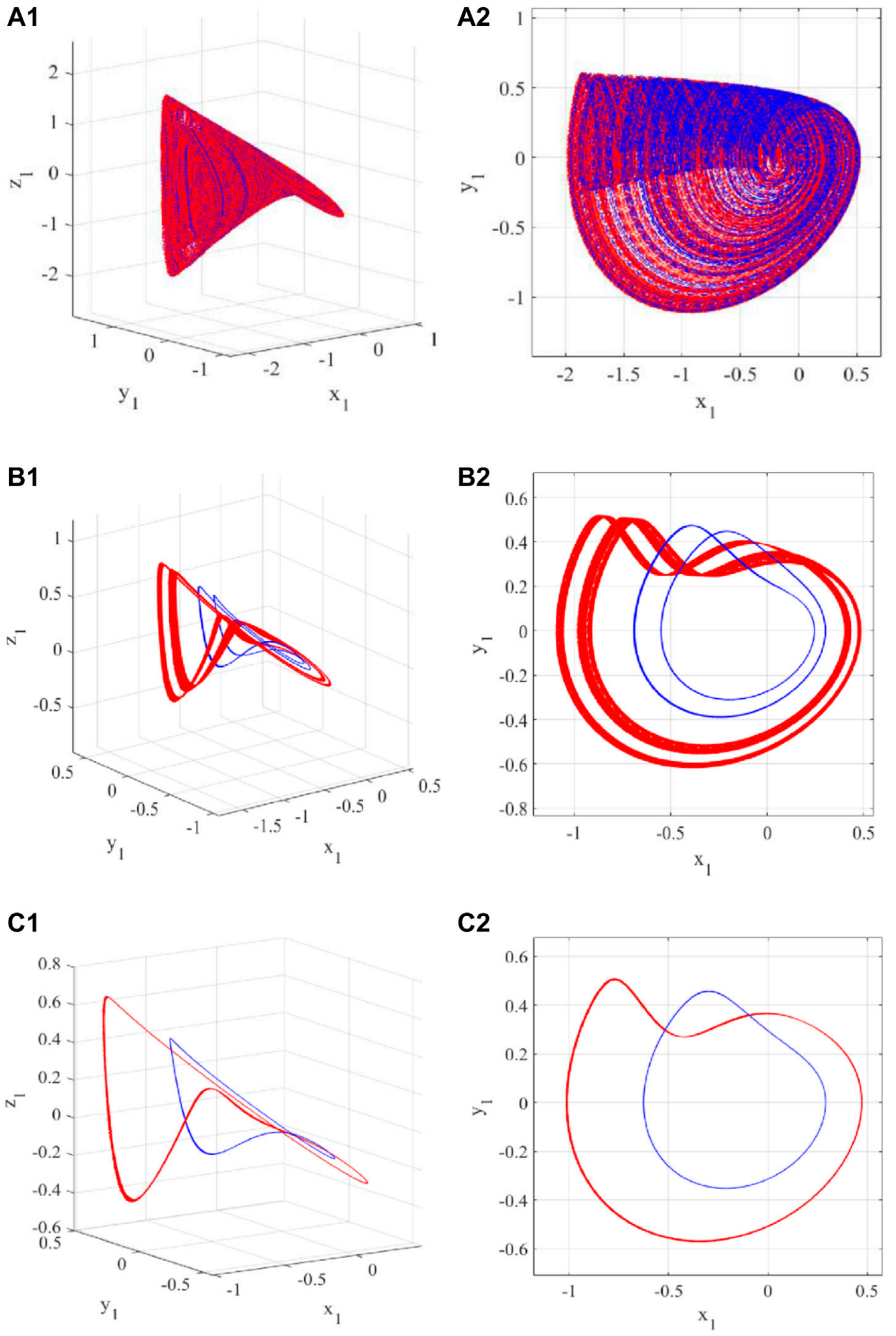


FIGURE 4 | The phase diagrams in 3D and 2D (*x*-*y* plane) with the initial conditions of $(x_0, y_0, z_0) = (0, 0.4, 0)$ (red) and $(x_0, y_0, z_0) = (0, -0.4, 0)$ (blue), and different values of ε: **(A)** 0.5; **(B)** 1.2; **(C)** 1.3.

$$\begin{cases} \dot{x} = y \\ \dot{y} = z \\ \dot{z} = -(x+k) - 10^{-9}\left(e^{\frac{y}{0.026}} - 1\right) - \varepsilon z \end{cases} \tag{13}$$

To better clarify the variable-boostable phenomenon, different k was selected to plot the phase diagrams. For the dimensionless system (13), the fixed time step of 0.001 was used for numerical simulations. The projections of attractors in the x-y plane are given in **Figures 5A,B**, where the blue, red, and yellow orbits correspond to $k = 2$ for $(x_0, y_0, z_0) = (-2, 0.4, 0)$, $k = 0$ for $(x_0, y_0, z_0) = (0, 0.4, 0)$,

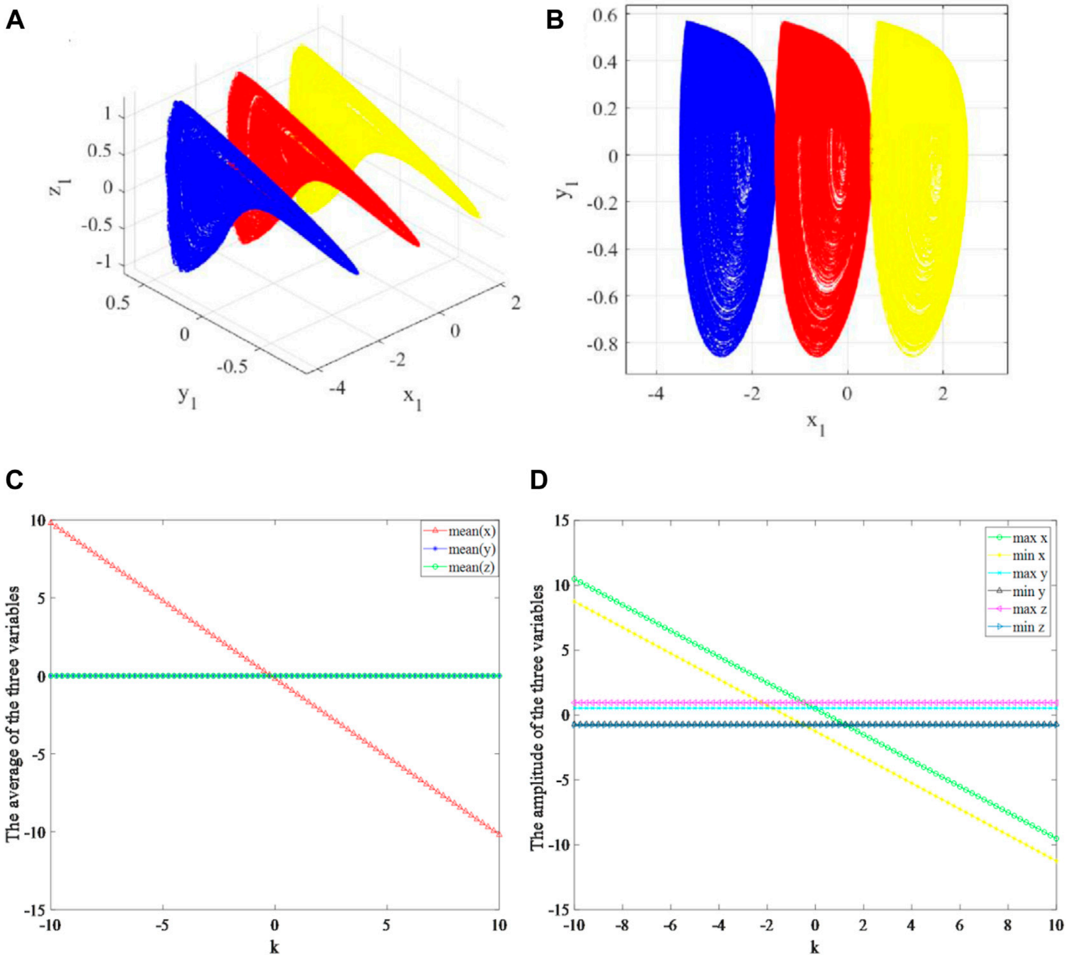


FIGURE 5 | Offset boosting behaviors of the system (13) for the constant *k*. **(A)** 3D phase diagram of the boosted attractors for *k* = 2 (blue), *k* = 0 (red), *k* = −2 (yellow). **(B)** *x*-*y* plane phase diagrams of the boosted attractors for *k* = 2 (blue), *k* = 0 (red), *k* = −2 (yellow). **(C)** The average of each variable *x*, *y*, and *z*. **(D)** the amplitude of variables *x*, *y*, and *z*.

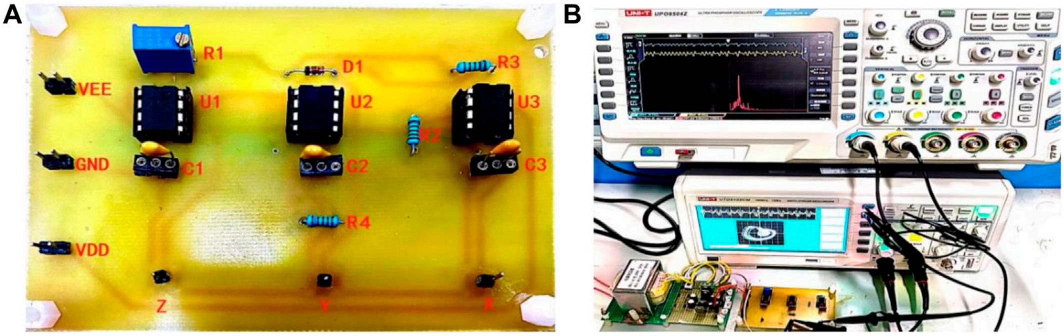


FIGURE 6 | Actual circuit implementation with COFAs. **(A)** Etched circuit board. **(B)** Actual experimental test with UNI-TUTD2102CM and UNI-T UPO9504Z.

TABLE 1 | Linear element parameters for experiments.

Parameter	Signification	Value
C_1,C_2,C_3	Resistance	10nf
R_2,R_3,R_4	Capacitance	10 k Ω
R_1	Variable resistance	0–100 k Ω

and $k = -2$ for $(x_0, y_0, z_0) = (2, 0.4, 0)$. It is found that the attractor is linearly boosted towards the x direction. Besides, we choose a relatively long time interval $\tau \in [1000,5000]$ to compute the average and amplitude of each variable [49–51]. The average and amplitude (maximum and minimum size) of x, y, and z are plotted in **Figures 5C,D**, respectively. The results show that both the average and amplitude of the variable x are boosted when adjusting k, whereas the other three variables almost keep unchanging.

CIRCUIT IMPLEMENTATION AND SPECTRUM CHARACTERISTICS

Circuit Implementation

To verify the authenticity of the simulation results, the real circuit was made to physically implement the system shown in **Figures 6A,B**, in which one precision potentiometer, three monolithic ceramic capacitors, one silicon diode 1N4148, three carbon film resistors with the error of ±5%, and three CFOAs AD844AN with operating voltages of ±12 V

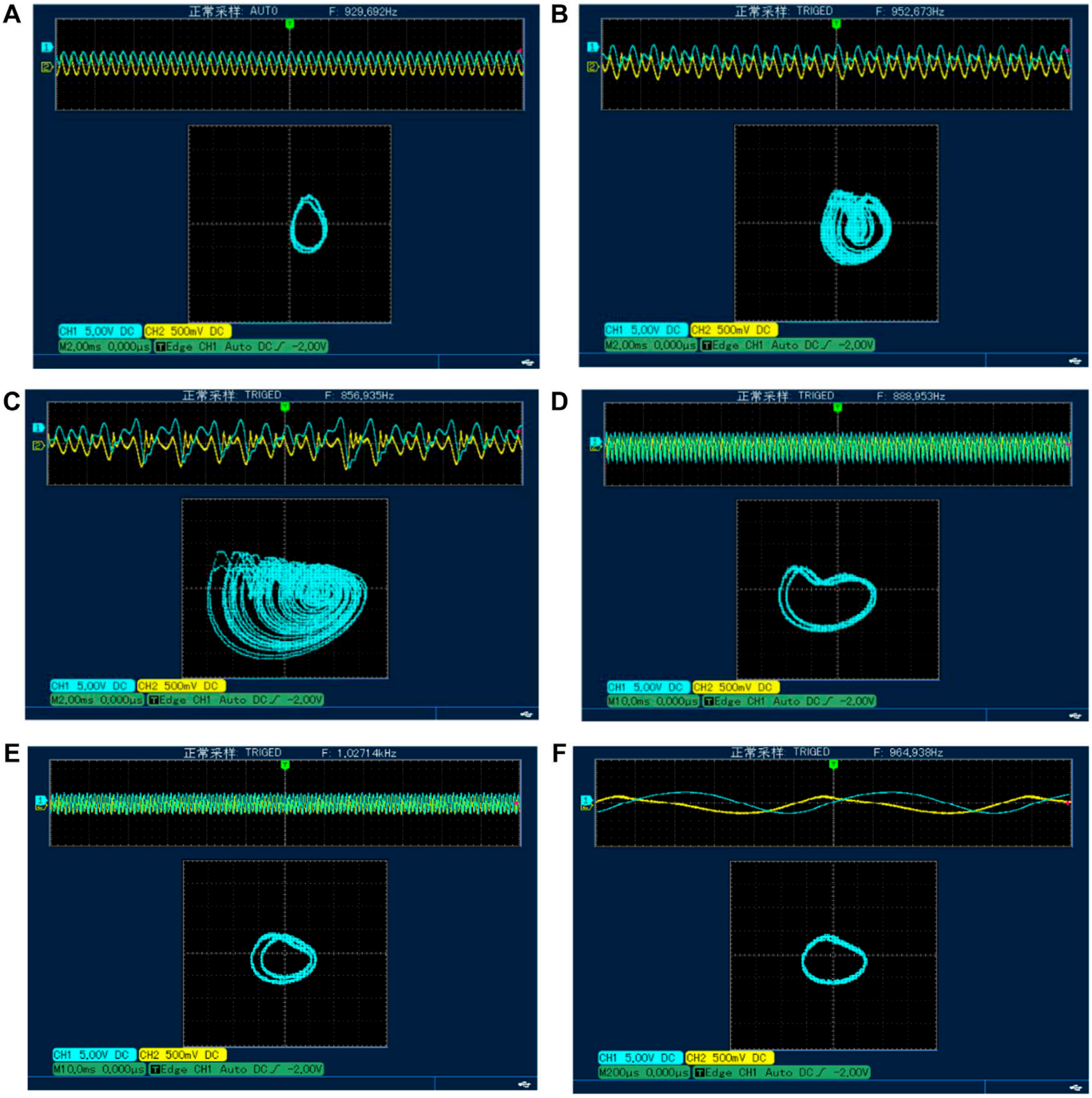


FIGURE 7 | Experiment results at different values of R1. **(A)** R1 = 34.48 kΩ. **(B)** R1 = 26.35 kΩ. **(C)** R1 = 19.8 kΩ. **(D)** R1 = 10.59 kΩ. **(E)** R1 = 8.32 kΩ. **(F)** R1 = 7.67 kΩ.

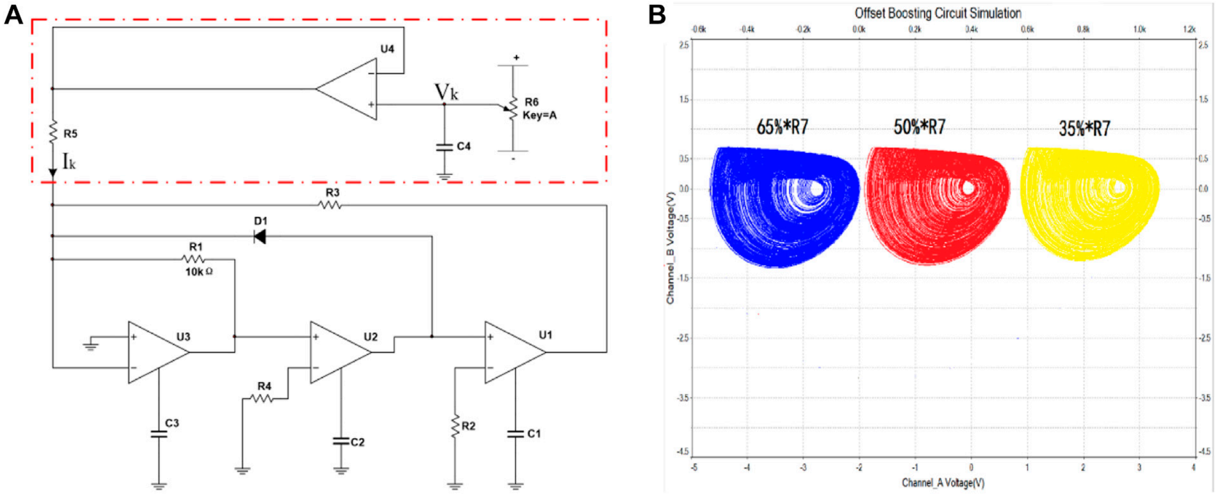


FIGURE 8 | The circuit simulation of offset boosting. **(A)** The circuit of offset boosting for variable *x*. **(B)** The attractors of offset boosting when adjusting for R7 to 35% (blue), 50% (red), and 65% (yellow).

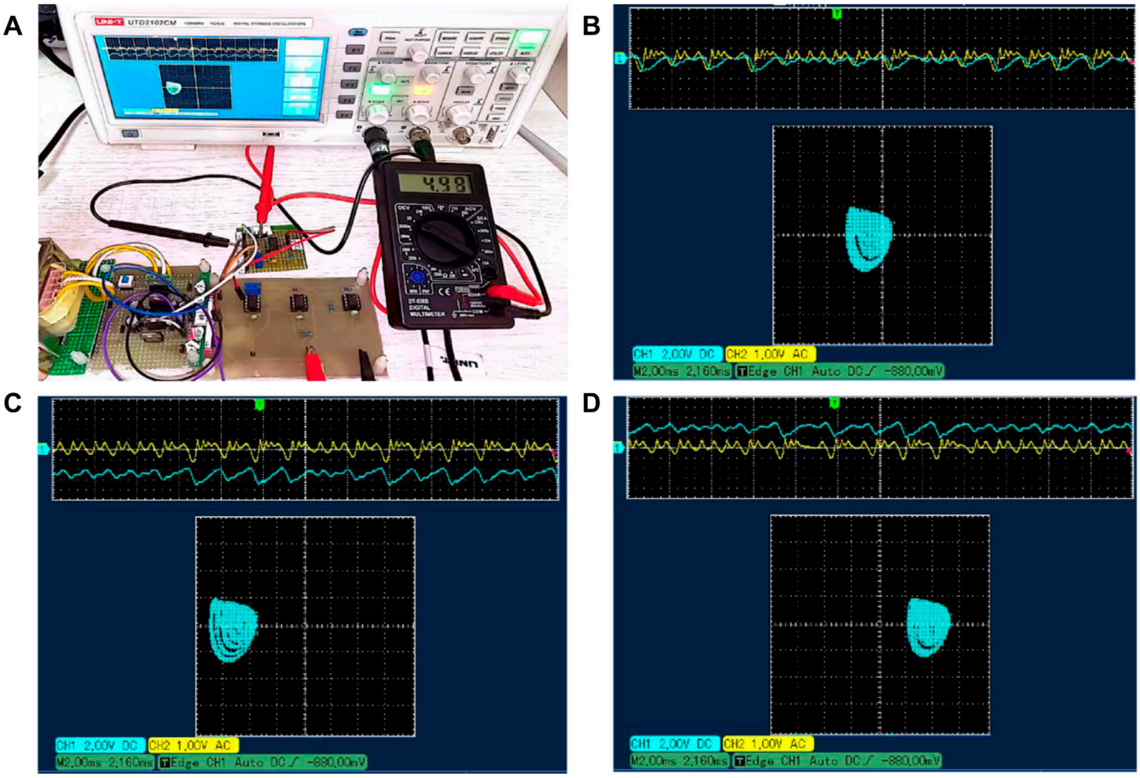


FIGURE 9 | Offset boosting with actual circuit implementation and its *x-y* plane phase diagrams in oscilloscope. **(A)** Actual experimental test. **(B)** $V_k = -4.5$ V. **(C)** $V_k = 0$ V. **(D)** $V_k = 4.5$ V.

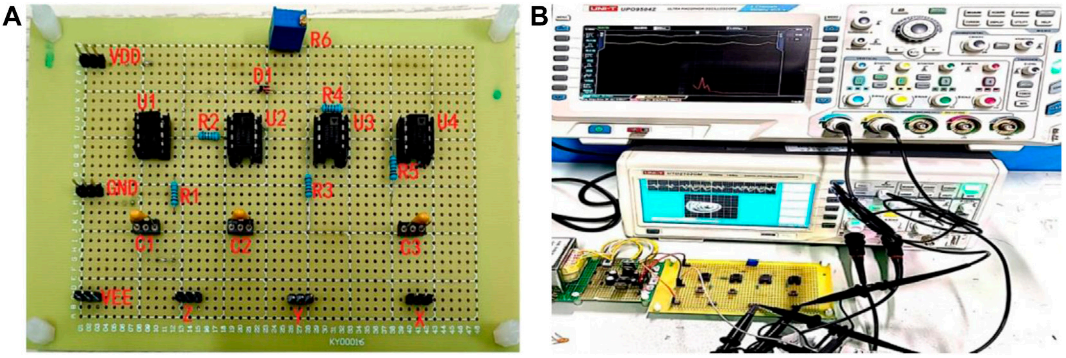


FIGURE 10 | VFOA circuit experimental test. **(A)** The experimental circuit board with AD711AN. **(B)** Actual experimental test with UNI-TUTD2102CM and UNI-T UPO9504Z.

are utilized. The circuit parameters are listed in **Table 1**. The experiment results of x-y plane phase diagrams and spectrum can be captured by the UNI-T UTD2102CM and UNI-T UPO9504Z, respectively.

According to the previous normalization formula **Eq. 4**, when the component parameters in the circuit are fixed, the value of ε depends on R_1. When $R_1 = 34.48\ \text{k}\Omega$, the value of ε is 0.29, and the system is in a period one state. Moreover, the value of R_1 continues to decrease when R_1 is adjusted to correspond to the value of ε in **Figure 4**, respectively, and the phase diagrams are captured by UNI-T UTD2102CM as shown in **Figure 7**. According to the phase diagrams, it is found that when R_1 is adjusted to a certain fixed value, the phase diagrams can be consistent with the numerical simulation results. For example, $R_1 = 19.8\ \text{k}\Omega$ and $\varepsilon = 0.5$, and its phase diagram is shown in **Figure 7C**. The system is in a chaotic state and is in the same state in **Figure 4A**. Meanwhile, it is found that when R_1 is in the range of $[10.1,\ 12.58]\ \text{k}\Omega$, the system shows periods corresponding to another period attractor, which also fully proves that the system has the phenomenon of attractors coexistence, as shown in **Figures 7D,F**.

Offset Boosting Implementation

Rewrite **Eq. 13** as follows:

$$\begin{cases} \dfrac{du_1}{dt} = \dfrac{1}{R_2C_1}u_2 \\ \dfrac{du_2}{dt} = \dfrac{1}{R_4C_2}u_3 \\ \dfrac{du_3}{dt} = -\dfrac{1}{R_3C_3}(u_1 + V_k) - \dfrac{1}{C_3}\cdot I_S\left(e^{\frac{u_2}{V_T}} - 1\right) - \dfrac{1}{R_1C_3}u_3 \end{cases} \tag{14}$$

According to the circuit diagram of **Figure 2** and **Eq. 14**, the simple Jerk circuit is built in NI Multisim 14.0, as shown in **Figure 8A**. A circuit in the red dashed box, containing a precision potentiometer, a common discharge constituting a voltage follower, and a resistor forming DC current I_k, is incorporated into the reverse port of u_1. Adjusting for R7 to 35, 50, and 65%, respectively, when $V_k = -4.5\ \text{V}$, $V_k = 0\ \text{V}$, and $V_k = 4.5\ \text{V}$, its chaotic attractors correspond to the blue, red, and yellow attractors in **Figure 8B**.

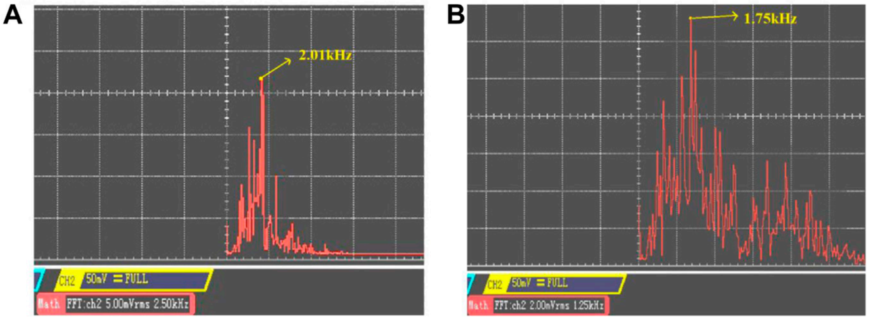
To verify the implementation ability of the simulation, we performed the experimental test as shown in **Figure 9A**. Adjusting the precision potentiometer to correspond the voltage of V_k to 4.5 , 0 , and −4.5 V, respectively, the attractor phase diagrams can be captured by the oscilloscope as shown in **Figures 9B–D**.

Frequency Spectrum Characteristics

The frequency spectrum is one of the most important characteristics of the chaotic signal. The frequency-domain characteristics of chaos are of great significance in chaotic encryption communication and automatic control [52]. In the integrator circuit, the time constant $\tau = RC$, when R is determined to be unchanged, changing the value of the capacitor C, different time constants can be obtained [53].

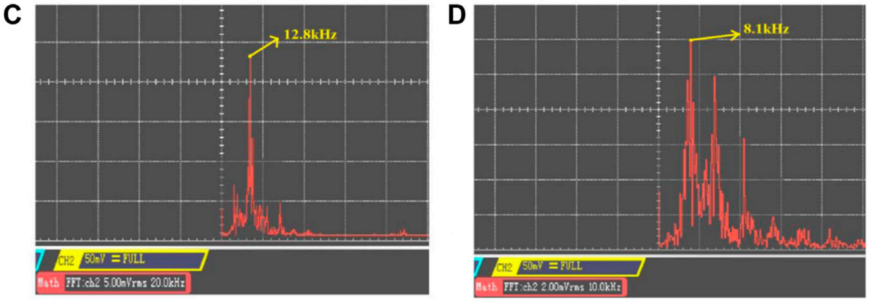
For comparison, an experimental circuit using four ordinary op-amps to realize the chaotic system [11] was made as **Figure 10** shows which includes four VFOAs AD711AN; three capacitors C_1, C_2, C_3; five resistors R_1, R_2, R_3, R_4, R_5; one precision potentiometer R_6; and one silicon diode D_1 1N4148, which provides the nonlinear function.

The specific parameters are that all the resistances are $10\ \text{k}\Omega$ in both of the two kinds of the chaotic circuit, and all capacitors are changed according to the experimental requirements at the same time. The frequency spectrums of the chaotic attractor under different time constants are obtained, as shown in **Figure 11**, during changing the value of integrating capacitors. It can be seen that when C = 100 nF, the center frequency of the two chaotic circuits is about 1.5 kHz. When C = 10 nF, the center frequency is about 12 kHz. Continuously, when C = 1 nF, the circuit with VFOAs is no longer chaotic. It can be seen from the frequency spectrum and the phase diagram that it is in a periodic state. However, CFOA-JCC is still in chaos and the center frequency of the chaotic attractor is about 120 kHz.



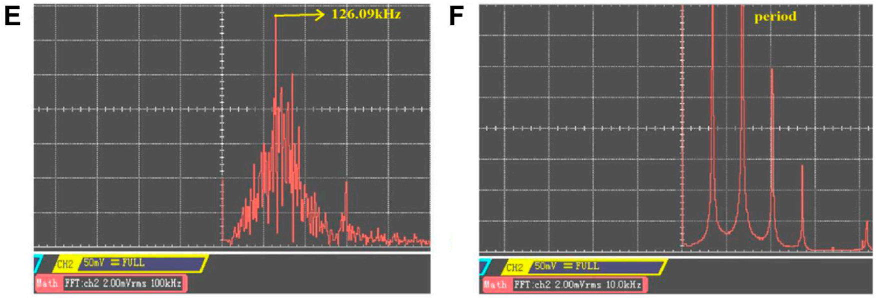
C = 0.1 uF, spectrum of CFOA-JCC.

C = 0.1 uF, spectrum of VFOA circuit



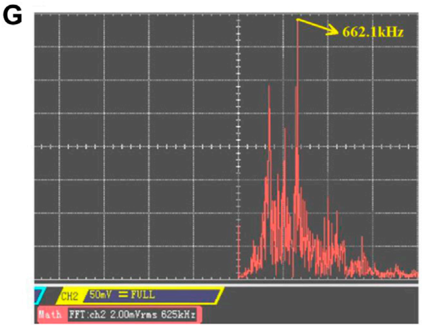
C = 10 nF, spectrum of CFOA-JCC.

C = 10 nF, spectrum of VFOA circuit.



C = 1 nF, spectrum of CFOA-JCC.

C = 1 nF, spectrum of VFOA circuit.



C = 200 pF, spectrum of CFOA-JCC

FIGURE 11 | Spectrums of the CFOA-JCC and VFOA circuit when capacitors take different values. **(A)** C = 0.1 uF, spectrum of CFOA-JCC. **(B)** C = 0.1 uF, spectrum of VFOA circuit. **(C)** C = 10 nF, spectrum of CFOA-JCC. **(D)** C = 10 nF, spectrum of VFOA circuit. **(E)** C = 1 nF, spectrum of CFOA-JCC. **(F)** C = 1 nF, spectrum of VFOA circuit. **(G)** C = 200 pF, spectrum of CFOA-JCC

What is more, the center frequency can be improved to about 650 kHz when C = 200 pF in the CFOA-JCC. Compared with the chaotic circuit based on VFOA, it can generate a high-frequency chaotic signal. Consequently, the CFOA-JCC is not only simple in structure but also can increase the operating frequency of the circuit and broaden its spectrum range.

In ideal conditions, the dominant frequency of oscillation is expected to be $f = \frac{\omega}{2\pi RC}$ and ω is the imaginary part of λ_2 and

TABLE 2 | The theoretical values of *f*in deferent capacitances.

C/pF	*f*/kHz
100,000	1.262
10,000	12.62
1,000	126.2
200	631.01

λ_3[11]. The values of *f* can be obtained as shown in **Table 2** in the case of different capacitances when all resistance values are 10 kΩ. Compering with **Figure 8**, the value of *f*in **Table 2** is extremely similar to the value in **Figure 8** experimentally observed.

CONCLUSION

The dynamics of CFOA-JCC and the electronic circuit implementation have been discussed extensively in terms of its parameters using standard nonlinear analysis techniques such as bifurcation diagrams, Lyapunov exponent plots, offset boosting, and frequency spectra. The bifurcation analysis suggests that chaos that arises in the CFOA-JCC follows the classical period-doubling when adjusting the bifurcation control parameters slowly. It is also found that the proposed CFOA-JCC exhibits the asymmetrical and striking feature of multiple coexistence attractors for a wide range of circuit parameters. A numerical simulation and circuit experimental test verified the coexistence of multiple attractors of this system. The voltage slew rate of the feedback amplifier is almost independent of frequency and has better high-frequency characteristics. Consequently, the CFOA-JCC has better high-frequency performance than the chaotic circuit designed by ordinary VFOAs, so that higher frequency chaotic attraction can be obtained. At the same time, compared with the VFOA chaotic circuit, the CFOA-JCC has a simpler topology and fewer components, conducive to system integration.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material. Further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

QW: system analysis, circuit design, and draft writing. ZT: checking the whole analysis and manuscript revision. XW: numerical analysis, circuit implementation. WT: checking the whole analysis.

FUNDING

This work was supported by the Natural Science Foundation of China (nos. 51660115, U1612442, 62061008, and 61264004), the Guizhou Province Science and Technology Plan Project (no. 20185769), and Research Projects of Innovation Group of Guizhou Provincial Department of Education (QianJiaoHeKY(2021)022).

REFERENCES

1. Lü J., Murali K., Sinha S., Leung H., Aziz-Alaoui M. Generating Multi-Scroll Chaotic Attractors by Thresholding. *Phys Lett A* (2008) 372(18):3234–9. doi:10.1016/j.physleta.2008.01.065
2. Li Z., Peng C., Tan W., Li L. An Efficient Plaintext-Related Chaotic Image Encryption Scheme Based on Compressive Sensing. *Sensors* (2021) 21(3):758. doi:10.3390/s21030758
3. Yu F., Zhang Z., Shen H., Huang Y., Cai S., Du S. FPGA Implementation and Image Encryption Application of a New PRNG Based on a Memristive Hopfield Neural Network with a Special Activation Gradient. *Chin Phys. B* (2021). doi:10.1088/1674-1056/ac3cb2
4. Xu Q., Tan X., Zhu D., Bao H., Bao B. Bifurcations to Bursting and Spiking in the Chay Neuron and Their Validation in a Digital Circuit. *Chaos, Solitons & Fractals* (2020) 141(927):110353. doi:10.1016/j.chaos.2020.110353
5. Ma C., Mou J., Li P., Liu T. Dynamic Analysis of a New Two-Dimensional Map in Three Forms: Integer-Order, Fractional-Order and Improper Fractional-Order. *Eur Phys J Spec Top* (2021) 230:1945–57. doi:10.1140/epjs/s11734-021-00133-w
6. Lu T., Chen G., Xing H. Doubling the Coexisting Attractors. *Chaos* (2019) 29: 051102–2. doi:10.1063/1.5097998
7. Sprott J. C. Some Simple Chaotic Jerk Functions. *Am J Phys* (1997) 65:537–43. doi:10.1119/1.18585
8. Sprott J. C. A New Class of Chaotic Circuit. *Phys Lett A* (2000) 266:19–23. doi:10.1016/s0375-9601(00)00026-8
9. Sprott J. C. Simple Chaotic Systems and Circuits. *Am J Phys* (2000) 68:758–63. doi:10.1119/1.19538
10. Kiers K., Klein T., Kolb J., Price S., Sprott J. C. Chaos in a Nonlinear Analog Computer. *Int J Bifurcation Chaos* (2004) 14:2867–73. doi:10.1142/s0218127404010898
11. Sprott J. C. A New Chaotic Jerk Circuit. *IEEE Trans Circuits Syst* (2011) 58: 240–3. doi:10.1109/tcsii.2011.2124490
12. Ma X., Mou J., Liu J., Ma C., Zhao X. A Novel Simple Chaotic Circuit Based on Memristor–Memcapacitor. *Nonlinear Dyn* (2020) 100:2859–76. doi:10.1007/s11071-020-05601-x
13. Xu Q., Ju Z., Ding S., Feng C., Chen M. Electromagnetic Induction Effects on Electrical Activity within a Memristive Wilson Neuron Model. *Cogn Neurodynamics* (2021). doi:10.1007/s11571-021-09764-0
14. Xu Q., Lin Y., Bao B., Chen M. Multiple Attractors in a Non-ideal Active Voltage-Controlled Memristor Based Chua's Circuit. *Chaos, Solitons & Fractals* (2016) 83:186–200. doi:10.1016/j.chaos.2015.12.007
15. Elwakil A. S., Soliman A. M. Current Conveyor Chaos Generators. *IEEE Trans Circuits Syst* (1999) 46:393–8. doi:10.1109/81.751313
16. Devices A. *AD844 Data Sheet* (1990). p. I990.
17. Wang C., Zhang J., Wang L., Shi W., Jing D. A Novel Nth-Order Voltage-Mode Universal Filter Based on CMOS CFOA. *Optik* (2016) 127:2226–30. doi:10.1016/j.ijleo.2015.11.127
18. Kilic R. On Current Feedback Operational Amplifier-Based Realization of Chua's Circuit. *Circuits Syst Signal Process* (2003) 22:475–91. doi:10.1007/s00034-003-0920-z
19. Kilic R., Yildirim F. Current-Feedback Operational Amplifier-Based Inductorless Mixed-Mode Chua's Circuits. *Int J Bifurcation chaos* (2006) 16:709–14. doi:10.1142/S0218127406015076
20. Bao B.-C., Xu J.-P., Liang Y. Inductor Current Sampled Feedback Control of Chaos in Current-Mode Boost Converter. *J Electron Sci Technology* (2008) 6:52–5.

21. Tan Ping-An T., Zhang Bo Z., Qiu Dong-Yuan Q. Stabilization of Chaotic Behaviour and Spiking Current in Thyristor with Time-Delayed Feedback Control. *wlxb* (2010) 59:5299–306. doi:10.7498/aps.59.5299
22. Ji X., Ren W., Yuan W. The Design of Multi-Scroll Chaotic Circuit Based on Current Conveyor. *Sci Technology Eng* (2013) 17.
23. Lin Y., Wang C. H., Xu H. Generation of Grid Multiscroll Chaotic Attractors Using a Current Conveyor. *J Math Sci* (2014) 201:350–60. doi:10.1007/s10958-014-1995-8
24. Joshi M., Ranjan A. Current-Controlled Chaotic Chua's Circuit Using CCCII. In: *Advances in Communication and Computational Technology. Lecture Notes in Electrical Engineering*, 668. Singapore: Springer (2020).
25. Shepherd L., et al. *Towards Ultra-low Power Bio-Inspired Processing*. London: Springer (2006).
26. Prasad S. R. A New High Frequency Realization of Chua's Chaotic Circuit Using Current Feedback Operational Amplifiers (CFOA's). *Int J Electron Comput Sci Eng* (2012) 1 (02), 223–229.
27. Wand C.-H., Yin J.-W., Lin Y. Design and Realization of Grid Multi-Scroll Chaotic Circuit Based on Current Conveyers. *Acta Phys Sin* (2012) 61. doi:10.7498/aps.61.210507
28. Lin Y., Wang C. H., Xu H. Grid Multi-Scroll Chaotic Attractors in Hybrid Image Encryption Algorithm Based on Current Conveyor. *Acta Physica Sinica* (2012) 61:514–8. doi:10.7498/aps.61.240503
29. Wu X. M., He Y. G., Yu W. X. Design and Implementation of Grid Multi-Scroll Chaotic Circuit Based on Current Feedback Operational Amplifier. *Acta Physica Sinica* (2014) 63. doi:10.7498/aps.63.180506
30. Kengne J., Njitacke Z. T., Nguomkam Negou A., Fouodji Tsostop M., Fotsin H. B. Coexistence of Multiple Attractors and Crisis Route to Chaos in a Novel Chaotic Jerk Circuit. *Int J Bifurcation Chaos* (2016) 26:1650081. doi:10.1142/s0218127416500814
31. Topaloglu S., Sagbas M., Anday F. Three-input Single-Output Second-Order Filters Using Current-Feedback Amplifiers. *AEU - Int J Electronics Commun* (2012) 66:683–6. doi:10.1016/j.aeue.2011.12.009
32. Elwakil A. S., Kennedy M. P. Chaotic Oscillators Derived from Sinusoidal Oscillators Based on the Current Feedback Op Amp. *Analog Integrated Circuits Signal Process.* (2000) 24:239–51. doi:10.1023/a:1008369810214
33. Qiwei T., et al. A Simple Inductor-free Memristive Circuit with Three Line Equilibria. *Nonlinear Dyn* (2018) 94:1585–602.
34. Se Hwan Kim S. H., Seung Hoon Lee S. H., Jang J. Fabrication and Characterization of Low-Temperature Poly-Silicon Lateral P-I-N Diode. *IEEE Electron Device Lett* (2010) 31:443–5. doi:10.1109/led.2010.2043047
35. Yan-ning F., Yi-sui S., Ji-lin Z. Non-linear Stability of the Equilibrium of a System of Mass Points. *Chin Astron Astrophysics* (2002) 26:354–62. doi:10.1016/s0275-1062(02)00077-2
36. Erawaty N., Kasbawati K., Amir A. K. Stability Analysis for Routh-Hurwitz Conditions Using Partial Pivot. *J Phys Conf Ser* (2019) 1341:062017. doi:10.1088/1742-6596/1341/6/062017
37. Peña J. M. Characterizations and Stable Tests for the Routh-Hurwitz Conditions and for Total Positivity. *Linear Algebra its Appl* (2004) 393:319–32. doi:10.1016/j.laa.2003.11.013
38. Dharma Rao N. Routh-Hurwitz Conditions and Lyapunov Methods for the Transient-Stability Problem. *Proc Inst Electr Eng UK* (1969) 116:539–47. doi:10.1049/piee.1969.0112
39. Bottema O. The Routh-Hurwitz Condition for the Biquadratic Equation. *Indagationes Mathematicae (Proceedings)* (1956) 59:403–6. doi:10.1016/s1385-7258(56)50054-6
40. Bnzar T., Lzureanu C. *A NEW 3-DIMENSIONAL SYSTEM WITH CHAOTIC BEHAVIOR* (2014).
41. Lü J., Chen G., Zhang S. Dynamical Analysis of a New Chaotic Attractor. *Int J Bifurcation Chaos* (2002) 12:1001–15. doi:10.1142/s0218127402004851
42. Marsden J. E., Ratiu T. S. *Introduction to Mechanics and Symmetry*. New York: Springer-Verlag (1999).
43. Pehlivan İ., Uyaroğlu Y. A New 3D Chaotic System with golden Proportion Equilibria: Analysis and Electronic Circuit Realization. *Comput Electr Eng* (2012) 38:1777–84. doi:10.1016/j.compeleceng.2012.08.007
44. Teng L., Iu H. H. C., Wang X., Wang X. Chaotic Behavior in Fractional-Order Memristor-Based Simplest Chaotic Circuit Using Fourth Degree Polynomial. *Nonlinear Dyn* (2014) 77:231–41. doi:10.1007/s11071-014-1286-4
45. Zhang X., Chen M., Wang Y., Tian H., Wang Z. Dynamic Analysis and Degenerate Hopf Bifurcation-Based Feedback Control of a Conservative Chaotic System and its Circuit Simulation. *Complexity* (2021) 2021:1–15. doi:10.1155/2021/5576353
46. Joshi M., Ranjan A. Investigation of Dynamical Properties in Hysteresis-Based a Simple Chaotic Waveform Generator with Two Stable Equilibrium. *Chaos, Solitons & Fractals* (2020) 134:109693 doi:10.1016/j.chaos.2020.109693
47. Li C., Sprott J. C., Liu Y., Gu Z., Zhang J. Offset Boosting for Breeding Conditional Symmetry. *Int J Bifurcation Chaos* (2018) 28(14):1850163. doi:10.1142/s0218127418501638
48. Ma C., Mou J., Xiong L., Banerjee S., Han X. Dynamical Analysis of a New Chaotic System: Asymmetric Multistability, Offset Boosting Control and Circuit Realization. *Nonlinear Dyn* (2021) 103:1–14. doi:10.1007/s11071-021-06276-8
49. Li C., Lei T., Wang X., Chen G. Dynamics Editing Based on Offset Boosting. *Chaos* (2020) 30:063124. doi:10.1063/5.0006020
50. Li C., Gu Z., Liu Z., Jafari S., Kapitaniak T. Constructing Chaotic Repellors. *Chaos, Solitons & Fractals* (2021) 142:110544. doi:10.1016/j.chaos.2020.110544
51. Li C., Peng Y., Tao Z., Sprott J. C., Ja Fari S. Coexisting Infinite Equilibria and Chaos. *Int J Bifurcation Chaos* (2021) 31. doi:10.1142/s0218127421300147
52. Zemlyanyi O. V. KEYING OF THE BROADBAND CHAOTIC SIGNAL SPECTRUM FOR DATA TRANSMISSION. *Telecommunications Radio Eng* (2016) 75 (05), 401–411. doi:10.1615/TelecomRadEng.v75.i5.20
53. Sansen W. M. C., Van Peteghem P. M. An Area-Efficient Approach to the Design of Very-Large Time Constants in Switched-Capacitor Integrators. *IEEE J Solid-State Circuits* (2003) 19:772–80. doi:10.1109/JSSC.1984.1052220

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

ORIGINAL RESEARCH
published: 07 March 2022
doi: 10.3389/fphy.2022.844966

A Novel 3D Image Encryption Based on the Chaotic System and RNA Crossover and Mutation

Ran Chu[1,2], *Shufang Zhang*[1]* *and Xianpeng Gao*[3]

[1]Information Science and Technology College, Dalian Maritime University, Dalian, China, [2]Information Science and Engineering College, Dalian Polytechnic University, Dalian, China, [3]Tokai Carbon Dalian Co., Ltd., Dalian, China

In this paper, a novel 3D image encryption based on the memristive chaotic system and RNA crossover and mutation is proposed. Firstly, the dynamic characteristics of the nonlinear system with two memristors are analyzed, including phase diagrams, Lyapunov exponential spectrums, and bifurcation diagrams. According to the merged image of three 3D images, the initial values of the memristive chaotic system are generated by SHA-256. Then the vertex coordinates are scrambled and diffused by 3D Arnold matrix and chaotic sequences. Finally, according to the dynamical encoding and decoding rules, crossover and RNA mutation are designed to confuse and diffuse the vertex coordinates. Throughout the encryption process, the Arnold matrix, RNA encoding and decoding rules, and crossover and mutation algorithms are determined by the memristive chaotic system. The experimental results verify that the proposed cryptosystem could encrypt three 3D images at the same time and resist various attacks effectively, and has good security performance.

Keywords: 3D images, memristive chaotic system, crossover, RNA mutation, 3D Arnold matrix

Edited by:
Jun Mou,
Dalian Polytechnic University, China

Reviewed by:
Fei Yu,
Changsha University of Science and Technology, China
Hongjun Liu,
University of Jinan, China

***Correspondence:**
Shufang Zhang
sfzhang@dlmu.edu.cn

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *29 December 2021*
Accepted: *18 January 2022*
Published: *07 March 2022*

Citation:
Chu R, Zhang S and Gao X (2022) A Novel 3D Image Encryption Based on the Chaotic System and RNA Crossover and Mutation. Front. Phys. 10:844966. doi: 10.3389/fphy.2022.844966

INTRODUCTION

With the development of wireless network and multimedia technology, more images are transmitted on the network. Among them, 3D printing has been widely used in medical, military, marine, and many other fields due to its available price. It can be predicted that the 3D images will be as popular as color images and videos in the future [1–3]. However, the 3D images can be downloaded easily, and 3D objects can be printed out without permission from the original providers. In order to prevent the attacks by unauthorized users, the 3D images should be encrypted before being stored and transmitted. Unlike the traditional files, the 3D image files have enormous data and special storage format, so the traditional encryption algorithms such as DES and AES cannot protect them efficiently [4–8]. Therefore, it is necessary to research high-level encryption technology for 3D images, and there are few researches on this field at present.

In the era of big data, some traditional image cryptosystems can be easily cracked by cloud computing. Therefore, it is necessary to propose a more secure image cryptosystem to meet the current requirements and protect the privacy of users. Chaotic systems have many significant features, including ergodicity, randomness, and sensitivity to initial values, which make them suitable for image encryption [9–22]. It is important to point out that the nonlinearity of memristor element enables the memristor circuits to generate chaotic signal and various chaotic attractors and exhibit abundant nonlinear phenomena [11, 23, 24]. Peng et al. [25] presented a dimensional chaotic map based on the discrete memristor, which can enlarge the hyperchaotic region and enhance complexity.

TABLE 1 | The format of the STL file

Vertex				Face	
Number	*x*-axis	*y*-axis	*z*-axis	Number	Elements in face
1	$V_{1,x}$	$V_{1,y}$	$V_{1,z}$	1	$(V_{1,x}, V_{1,y}, V_{1,z})$
2	$V_{2,x}$	$V_{2,y}$	$V_{2,z}$	2	$(V_{2,x}, V_{2,y}, V_{2,z})$
3	$V_{3,x}$	$V_{3,y}$	$V_{3,z}$	3	$(V_{3,x}, V_{3,y}, V_{3,z})$
. . .	. . .	. . .	. . .	. . .	. . .
n	$V_{n,x}$	$V_{n,y}$	$V_{n,z}$	*n*	$(V_{n,x}, V_{n,y}, V_{n,z})$

Chen et al. [26] proposed a pseudorandom number generator by three kinds of four-wing memristive hyperchaotic systems, the dynamical characteristics are sensitive and complex. Ma et al. [27] proposed a chaotic circuit based on two memristors, which has rich dynamical behaviors and is suitable for image encryption and secure communication.

With the development of the chaos theory, the fusion of chaotic system and cryptography has become an important solution to ensure image security [28–32]. Based on the chaotic systems, many image encryption schemes are designed by optical transformation [33, 34], compressive sensing [35], DNA and RNA computing [36–41], cellular automata [42], etc. Among them, RNA computing can increase information density, improve parallelism, and reduce energy consumption, so it has become a research hotspot recently [43–45]. Abbasi et al. [46] proposed a cryptosystem based on amino acid, RNA codons, and evolutionary chaotic model, and the performance of the cryptosystem is improved by biomolecules and the imperialist competition algorithm. Jarjar et al. [47] presented a method that uses several genetic and biological features of DNA and RNA to encrypt color images and model mathematical problems as biological ones. Zhang et al. [48] proposed an image cryptosystem based on hyperchaotic system and RNA operation, which could resist various attacks. However, all the researches above are based on the RNA encoding/decoding law; the calculation results between the RNA bases are easy to predict, increasing the risk of cracking. Therefore, the traditional RNA scheme could be improved.

Based on the above analyses, a novel 3D image cryptosystem is proposed in this paper. The rest of the parts are organized as follows. In *Preprocessing of the 3D Model*, the preprocessing of 3D model data is described. In *Chaotic System*, the memristive chaotic system is introduced and its dynamical characteristics are illustrated. In *Encryption Scheme*, the SHA-256 algorithm, improved 3D Arnold algorithm, and RNA level encryption algorithm are described. In *Performance Analysis*, the security performance and experimental results are evaluated through the key analysis, statistical analysis, information entropy, and robust security analysis. In *Section Conclusion*, some important conclusions are obtained.

PREPROCESSING OF THE 3D MODEL

Stereolithography (STL) is the standard technology file of current 3D printing technology, which is used to describe the triangulated surface geometry of a 3D model by vertices, faces, and unit normal. The format of the STL file is shown in **Table 1**. The x, y, and z coordinates of the vertices determine the shape of the 3D model, just like the RGB pixels of a color image file. Therefore, 3D images could be transformed into 2D objects, and the coordinates of the vertices are a protected object.

Each vertex consists of x, y, and z coordinates (floating point value of at least 96 bit). To enable fast processing with accurate results, all data should be normalized first. Considered the efficiency requirement, the Min–Max Normalization is used to normalize the coordinate data, and the scale criterion is used to rescale the coordinate data, as follows:

$$V' = R_{\min} + \frac{V - V_{\min}}{V_{\max} - V_{\min}} \times (R_{\max} - R_{\min}) \tag{1}$$

where V' is the normalized coordinate, V is the original coordinate, $V_{\min}$ and $V_{\max}$ are minimum and maximum original coordinates, and $R_{\min}$ and $R_{\max}$ are the selected range, ranging from −255 to 255.

After extracting the symbol matrix from the floating point number matrix, the positive array is obtained. Then, it is decomposed into integer matrix V_i and fractional matrix V_f.

CHAOTIC SYSTEM

Recently, a chaotic circuit based on two memristors was proposed, which has multiple attractor coexistence and state transition, defined as

$$\begin{cases} \dot{x} = \dfrac{1}{L} \cdot y - \dfrac{1}{La_1} \cdot \dfrac{x}{z} \\ \dot{y} = -\dfrac{1}{C} \cdot x - \dfrac{a_2}{C} \cdot wy \\ \dot{z} = \left(\dfrac{x}{a_1 z}\right)^2 - b_1 \\ \dot{w} = y^2 - b_2 \end{cases} \tag{2}$$

where x, y, z, and w represent the system states variables; L, C, a_1, a_2, b_1, and b_2 are system parameters. With the variation of parameters, the system has seven types of attractors. Set $L = 0.025$, $C = 0.025$, $a_1 = 5.8$, $a_2 = 0.825$, $b_1 = 1.85$, $b_2 = 10$, and the initial values are (1, 3, 1, −0.7). **Figure 1** shows the phase diagrams. The Lyapunov exponential spectrums and bifurcation diagrams are presented in **Figure 2**.

When $a_1 \in (4.5,6.5)$, $a_2 \in (0.7,0.9)$, $b_1 \in (1.7,2.2)$, and $b_2 \in (9,11)$, the bifurcation diagram shows that the system changes from periodic to chaotic, the chaotic states are widely distributed, and the system has complex dynamic characteristics, which can also be verified by the Lyapunov exponential spectrum. At least one Lyapunov exponent in a chaotic system is greater than zero, and the largest Lyapunov exponent in a Lyapunov diagram is always positive, that is, the system is in the chaotic state. These characteristics of the chaotic system make it suitable for image encryption.

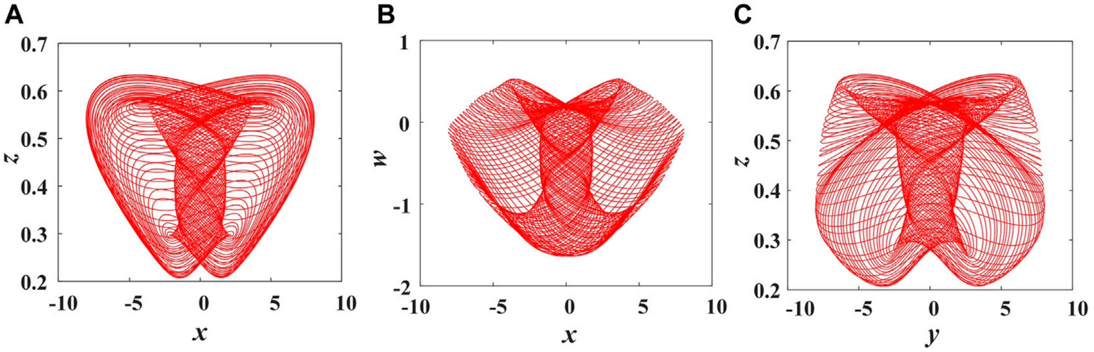


FIGURE 1 | The phase diagrams. **(A)** *x-z* plane, **(B)** *x-w* plane, **(C)** *y-z* plane.

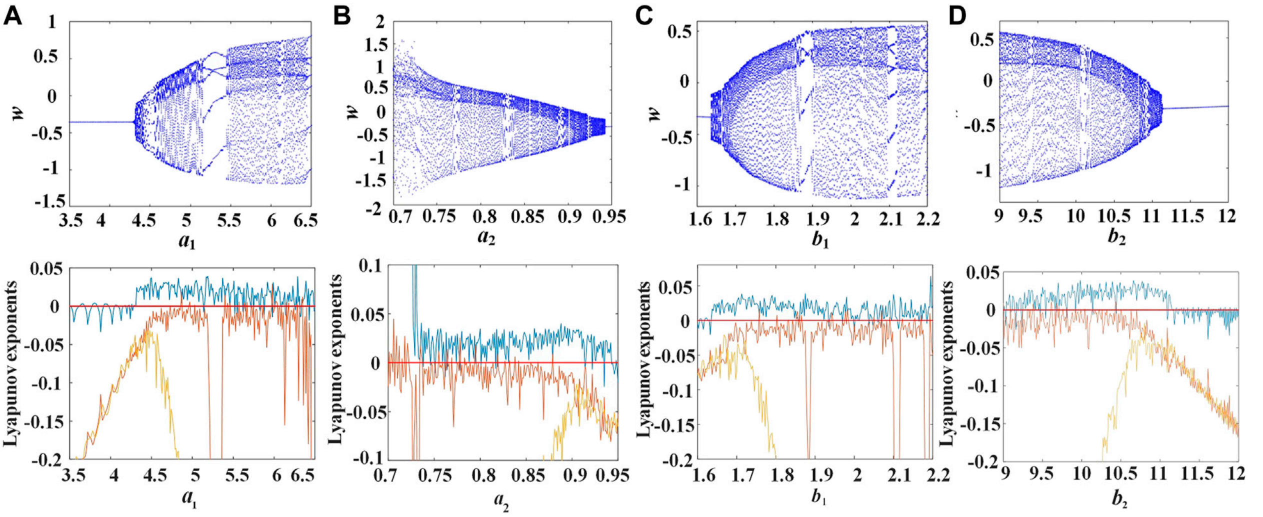


FIGURE 2 | Lyapunov exponent spectrums and bifurcation diagrams. **(A)** $a_1 \in (3.5,6.5)$, **(B)** $a_2 \in (0.7,0.95)$, **(C)** $b_1 \in (1.6,2.2)$, **(D)** $b_2 \in (9,12)$.

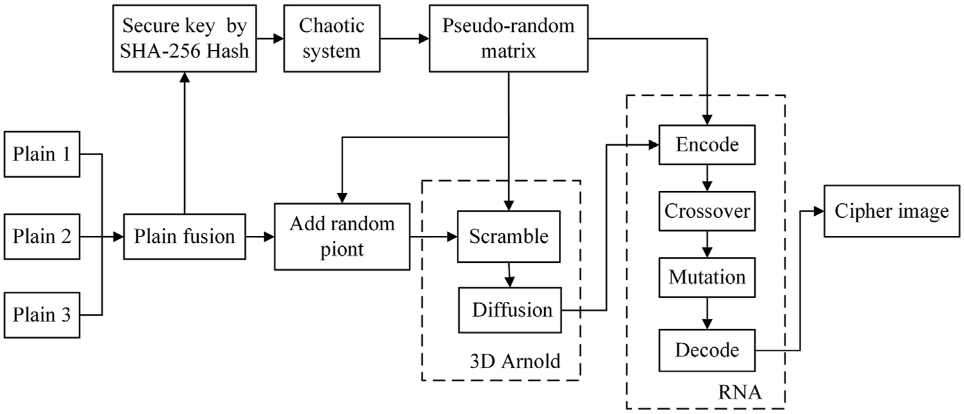


FIGURE 3 | The flowchart of encryption algorithm.

ENCRYPTION SCHEME

The encryption process is described, including the SHA-256 algorithm, improved 3D Arnold algorithm, and RNA operations. The flowchart is shown in **Figure 3**.

SHA-256 Algorithm

To improve the security, the SHA-256 is used in the encryption algorithm. Three 3D images are merged together by the method of 3D stereo arrangement. Therefore, different images and different orders will generate different hash values and then obtain different initial conditions, thus producing different chaotic sequences.

Step 1: Based on the merged image I, the 256-bit hash value is generated by the SHA-256 algorithm and divided into eight-bit blocks (total 64 group bit blocks). Thus, the stream key is obtained by XOR operation:

$$H(i) = h(2i) \oplus h(2i-1) \tag{3}$$

where $i \in (1,32)$. The slight disturbance is generated as follows

$$\begin{cases} k_1 = 10^{-15} \times \mathrm{sum}(H(1:8) \oplus H(9:16)) \\ k_2 = 10^{-15} \times \mathrm{sum}(H(17:24) \oplus H(25:32)) \\ k_3 = 10^{-15} \times \mathrm{sum}(H(1:8) \oplus H(25:32)) \\ k_4 = 10^{-15} \times \mathrm{sum}(H(9:16) \oplus H(17:24)) \\ k_5 = 10^{-15} \times \mathrm{sum}(V) \end{cases} \tag{4}$$

Step 2: Input the initial condition as the key. Then, the adjusted initial conditions are generated.

$$\begin{cases} x = x_0 + (k_1 + k_2) \\ y = y_0 + (k_2 + k_3) \\ z = z_0 + (k_3 + k_4) \\ w = w_0 + (k_4 + k_5) \end{cases} \tag{5}$$

where x, y, z, and w are adjusted initial values; x_0, y_0, z_0, and w_0 are given values without disturbance. Suppose the plaintext size is $M \times N$, according to the adjusted initial conditions, the chaotic system is iterated $(n + M \times N)$ times. To increase the sensitivity of initial values, discard the front n times values. The sequences X, Y, Z and W are generated.

$$\begin{cases} X = \mathrm{mod}(\mathrm{floor}(\mathrm{abs}(x \times 10^{15})), 256) \\ Y = \mathrm{mod}(\mathrm{floor}(\mathrm{abs}(y \times 10^{15})), 256) \\ Z = \mathrm{mod}(\mathrm{floor}(\mathrm{abs}(z \times 10^{15})), 256) \\ W = \mathrm{mod}(\mathrm{floor}(\mathrm{abs}(w \times 10^{15})), 256) \end{cases} \tag{6}$$

Thus, the chaotic sequences are affected by hash value of plaintext, which can improve the random adaptive ability.

Improved 3D Arnold Algorithm

On the basis of chaotic sequence, irrelevant vertices are added to ensure security, and then, an improved 3D Arnold cat map is performed for scrambling and diffusion. In the encryption scheme, the integer matrix is confused and diffused. In order to save the encryption time, the fractional matrix is only treated by diffusion.

Step 1: In the 3D Arnold matrix, the calculation principle of control parameters depends on the chaotic sequence, as shown in the formula:

$$\begin{cases} S_A = \lfloor a \cdot x + c \cdot y - y \cdot z \rfloor \\ S_B = \lfloor z \cdot (x-1) - b \cdot y \rfloor \\ S_C = \lfloor a \cdot y + c \cdot z - x \cdot z \rfloor \\ S_D = \lfloor x \cdot (y-1) - b \cdot z \rfloor \\ S_E = \lfloor a \cdot z + c \cdot x - x \cdot y \rfloor \\ S_F = \lfloor y \cdot (z-1) - b \cdot x \rfloor \end{cases} \tag{7}$$

where a, b, and c are specified parameters.

Step 2: The 3D Arnold matrix is obtained by multiplication of cubic matrices, as follows:

$$A = \begin{bmatrix} 1 & 0 & S_A \\ 0 & 1 & 0 \\ S_B & 0 & S_A \times S_B + 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & S_C \\ 0 & S_D & S_C \times S_D + 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & S_E & 0 \\ S_F & S_E \times S_F + 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

Step 3: The integer matrix and fractional matrix are scrambled by the 3D Arnold matrix. The scrambled coordinate positions are generated as follows:

$$\begin{bmatrix} i_A \\ j_A \\ k_A \end{bmatrix} = A \cdot \begin{bmatrix} i \\ j \\ k \end{bmatrix} \mathrm{mod} \begin{bmatrix} N_{\mathrm{row}} \\ N_{\mathrm{column}} \\ N_{\mathrm{graph}} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{9}$$

where (i, j, k) is the original position; N_{row}, N_{column}, and N_{gragh} are the number of rows, columns, and graphs. (i_A, j_A, k_A) is the scrambled position. After each vertex coordinate is shifted with the replacement vertex, the scrambled image is obtained.

Step 4: The integer matrix is diffused by the 3D Arnold matrix, as follows:

$$\begin{bmatrix} AV_i(i,j,k) \\ AV_i(i_A, j_A, k_A) \\ AV_i(i^{'}, j^{'}, k^{'}) \end{bmatrix} = A \cdot \begin{bmatrix} V_i(i,j,k) \\ V_i(i_A, j_A, k_A) \\ V_i(i^{'}, j^{'}, k^{'}) \end{bmatrix} \mathrm{mod} \begin{bmatrix} 256 \\ 256 \\ 256 \end{bmatrix} \tag{10}$$

where $V_i(i, j, k)$ and $V_i(i_A, j_A, k_A)$ are scrambled values of the previous step, and $V_i(i', j', k')$ is the diffused value of the previous round. $AV_i(i, j, k)$, $AV_i(i_A, j_A, k_A)$, and $AV_i(i', j', k')$ are the values after diffusion. After each coordinate in the matrix is transformed, the diffused image matrix I_1 is obtained.

With the improved 3D Arnold algorithm, the matrix can be scrambled and diffused simultaneously, which could save the encryption time and reduce the computational complexity.

RNA Encryption Algorithm

The RNA encryption algorithm aims to confuse and diffuse the vertex coordinates completely. The bio-inspired chaotic encryption algorithm can be divided into encoding, diffusion, crossover, mutation, and decoding stages.

Step 1: RNA coding rules. The RNA sequence is constructed by four nitrogen bases, namely, Adenine (A), Guanine (G), Uracil (U), and Cytosine (C). The complementary coding rules are shown in **Table 2**. Besides, **Table 3** shows the RNA operation rules.

Step 2: RNA diffusion. The rules of coding and operation are both determined by chaotic sequences. Convert the matrix I_1 and chaotic sequence to a binary matrix of size $M \times N \times 8$. Then,

TABLE 2 | The RNA encoding rules

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

TABLE 3 | The RNA operation rules

+	A	C	G	U	–	A	C	G	U
A	A	C	G	U	A	A	U	G	C
C	C	G	U	A	C	C	A	U	G
G	G	U	A	C	G	G	C	A	U
U	U	A	C	G	U	U	G	C	A

according to the dynamic coding rules, the plaintext is transformed into an RNA sequence of size $M \times N \times 4$. The disorder sequence is obtained by substituting base pairs under the complementary rule. Then, the diffused cipher sequence P is obtained.

Step 3: Gene crossover. Crossover, also known as gene recombination or hybridization, is the exchange of genetic material at the same location of two chromosomes to generate two new recombinant chromosomes. In 3D image encryption algorithm, the vertex coordinates are exchanged to accomplish the crossover, similar to the chromosome swapping in biology. The sequence P is divided into two sequences P_1 and P_2 of equal length. Transform the chaotic sequence $(x \oplus w)$ into a binary sequence that determines the crossover position of the unit: 1 means exchange, 0 means non-exchange, as shown in **Figure 4**. When the crossover is complete, merge the two sequences.

Step 4: RNA mutation. Codon refers to a triplet sequence of nucleotide residues in RNA that encodes a specific amino acid during protein synthesis. Mutations are certain replication errors that occur during replication, leading to the formation of new chromosomes. In 3D image cryptosystem, the changes in vertex coordinates correspond to mutation in biology. According to the chaotic sequence, two mutation units are found and defined in RNA sequence, as shown in **Figure 5**. According to the chaotic sequence, mutation units [AUC] at position 2 and [UCG] at position 4 are selected by $(z \oplus w)$. If the mutation units selected are the same, the selection will continue according to the subsequent chaotic sequence.

The codons mutated accordingly, as shown in **Figure 6**. The red and green codons convert to each other.

Step 5: RNA decoding rules. Based on chaotic sequence, decoding rules are selected, and binary matrix is obtained. Then convert it into decimal matrix, which is the final ciphertext C.

According to the RNA coding theory, each RNA gene carries certain information of the images. Dynamic RNA encoding rules and RNA decoding rules could increase the randomness of ciphertext and the resistance to statistical attacks. Crossover and mutation based on chaotic sequences could increase the unpredictability of the encryption algorithm. Therefore, a different ciphertext can be obtained by using different crossover and mutation rules and different encoding and decoding rules.

Decryption Scheme

Due to the symmetrical image cryptosystem, the cipher matrix can be decrypted and the original images can be restored using the reverse process of the encryption process, as shown in **Figure 7**. After the reverse mutation and crossover of RNA operation and the reverse scrambling and diffusion of 3D Arnold algorithm, the added random points were removed and the image was decomposed, and three original images could be obtained.

PERFORMANCE ANALYSIS

Simulation Results

In order to verify the performance of the designed cryptosystem, several experiments were carried out on 3D images. The experiments were tested on a computer: Intel Core i7 CPU 2.9GHz, RAM 32 GB. The operating system is Windows 10. The simulation software is Matlab 2016b. Set the parameters as follows: $L = 0.025$, $C = 0.025$, $a_1 = 5.8$, $a_2 = 0.825$, $b_1 = 2$, $b_2 = 10$, and the initial values are (1, 3, 1, −0.7). To test the encryption performance, three 3D images Car (39888×3), Jet (25896×3), and Rocket (51310×3) are used as encryption models. The simulation results of cryptosystem are shown in **Figure 8**. The encrypted images can completely hide the original images, and decryption process can successfully restore the original images with the keys. Based on the characteristics of 3D images, it is considered as a special color image. Therefore, the performance analysis methods of color images cryptosystem can be applied to 3D images encryption.

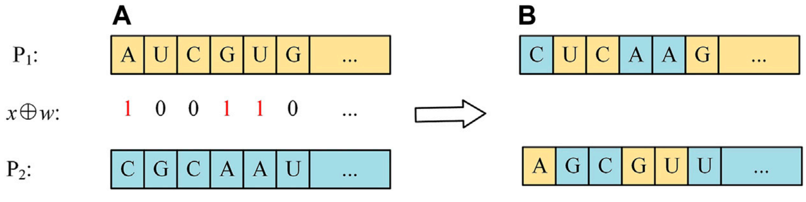


FIGURE 4 | Crossover process. **(A)** Before crossover. **(B)** After crossover.

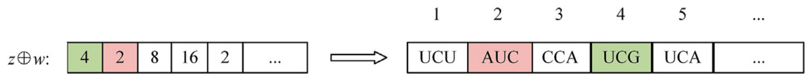


FIGURE 5 | Selection of mutant genomes.

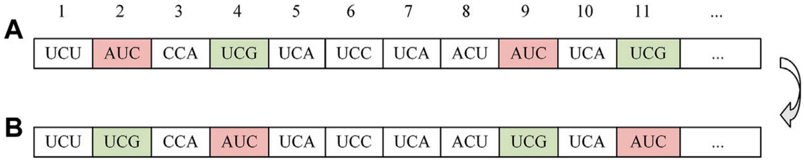


FIGURE 6 | Mutation process. **(A)** Before mutation. **(B)** After mutation.

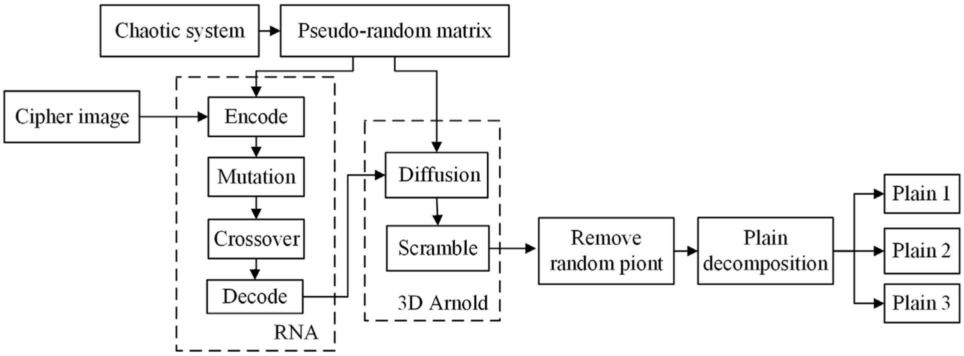


FIGURE 7 | The flowchart of decryption algorithm.

Key Space Analysis

In cryptosystem, the key space must be large enough, which is critical to the encryption algorithm. The secret key consists of 1) system parameters *L, C*, a_1, a_2, b_1, and b_2; 2) initial value of the chaotic system (*x, y, z, w*); 3) SHA-256 hash value k_1, k_2, k_3, k_4; and 4) parameter of vertex matrix k_5. With the calculation accuracy of 10^{-15}, the key space will reach 2^{747}, which exceeds the minimum requirement of 2^{100}, and could resist brute force attacks [49]. **Table 4** shows the test results of key space. Compared with other literatures, key space of this cryptosystem is larger.

Key Sensitivity Analysis

Key sensitivity is an important feature of encryption algorithm. An effective cryptosystem should be extremely sensitive to the key. The cipher matrix obtained by encrypting the same plaintext matrix should be significantly different when there is a small change in the key. In the key sensitivity test, a slight change in the key during the decryption results in three restored images that are completely different from originals, as shown in **Figure 9**.

Histogram Analysis

In order to hide the information about vertex coordinates, the ciphertext histograms should be flat and evenly distributed. The histogram results are plotted in **Figure 10**. In the figure, the plaintext and ciphertext histograms are completely different. There is a clear coordinate aggregation interval in the plaintext matrix, while the coordinates in the cipher matrix are uniformly distributed.

Moreover, the χ^2 test results can be used to quantitatively analyze whether vertex coordinates are evenly distributed. In **Table 5**, the χ^2 test results of plaintexts and ciphertexts are presented. The measured results of the proposed algorithm are less than the critical values of different probabilities (10%, 5%, and 1%), indicating that the encryption algorithm could overwrite the statistics of vertex coordinates and resist statistical attacks.

Correlation Analysis

To protect vertex coordinate information, the statistics needs to be hidden, and the correlation between adjacent vertices coordinates needs to be reduced. The relevant equations are given below:

$$\gamma_{xy} = \frac{E\{[x - E(x)] - [y - E(y)]\}}{\sqrt{D(x)D(y)}} \tag{11}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{12}$$

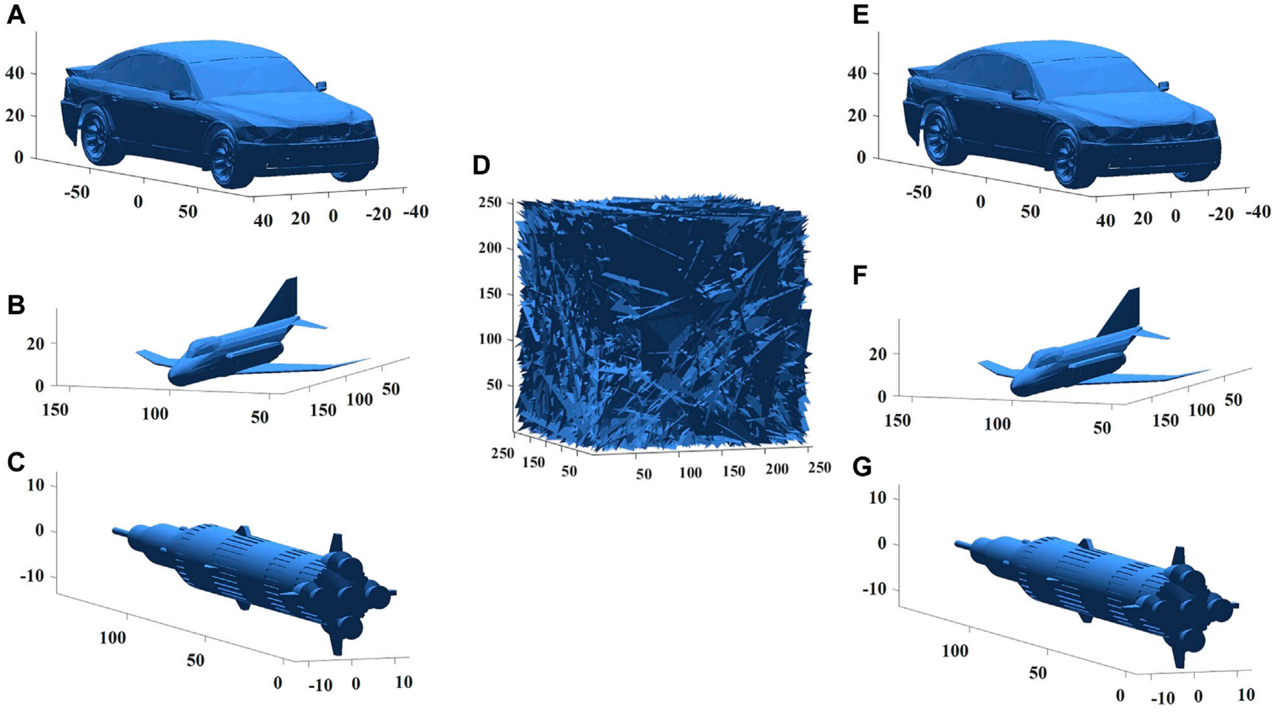


FIGURE 8 | The simulation results. **(A)** Car image. **(B)** Jet image. **(C)** Rocket image. **(D)** Cipher image. **(E)** Decrypted Car image. **(F)** Decrypted Jet image. **(G)** Decrypted Rocket image.

TABLE 4 | Key space for different algorithms

Algorithm	Proposed	Reference [7]	Reference [8]	Reference [38]	Reference [42]	Reference [50]
Key space	2^{747}	2^{599}	2^{240}	2^{280}	2^{256}	2^{224}

$$D(x) = \frac{1}{N}\sum_{i=1}^{N}[x_i - E(x)]^2 \tag{13}$$

Figures 11A,C,E show the correlation of plaintexts; the coordinate distribution is intensive, and the vertex coordinates are highly correlated. On the contrary, **Figures 11B,D,F** demonstrated that the correlation of the encryption model is low, which means that the algorithm could reduce the correlations between adjacent vertices in each direction and resist statistical attacks.

When the absolute value of the correlation coefficient is close to 1, it means that the coordinate values of the large region in the vertex matrix are small. On the contrary, if it is close to 0, the cryptosystem could confuse coordinate positions effectively. The correlation coefficient between adjacent coordinates are shown in **Table 6**. In each direction, the adjacent coordinates of the plaintexts are highly correlated and the correlation between adjacent coordinates of ciphertexts is low.

Randomness Analysis

In order to ensure the uniform distribution of cipher coordinate values, a variety of 3D images are adopted to measure the security of cipher. SP (Special Publications) 800-22 is a randomness standard test published by NIST, which contains 15 sub-tests that identify the randomness of the encryption algorithm. NIST test uses p-value magnitude and the pass rate to reflect the bit sequence randomness property of the encryption algorithm. When the p-value is greater than or equal to 0.01, it is considered that the bit sequence has passed the NIST test, that is, the bit sequence has randomness. The pass rate reflected the number of sample sequences that pass the randomness test.

With different test samples and different initial conditions, several cryptographic matrixes are obtained. Then, the cipher matrix is converted to binary sequence. The value of each cipher matrix is represented by eight bits. Thus, the length of test sample is 86,400,000 bits. The randomness test results are present in **Table 7**, with the worst set of data. The test sample can pass all 15 sub-tests, indicating that the cipher matrix has high randomness.

Information Entropy Analysis

Information entropy is used to measure the average amount of information in the whole image, and its theoretical value is 8. The

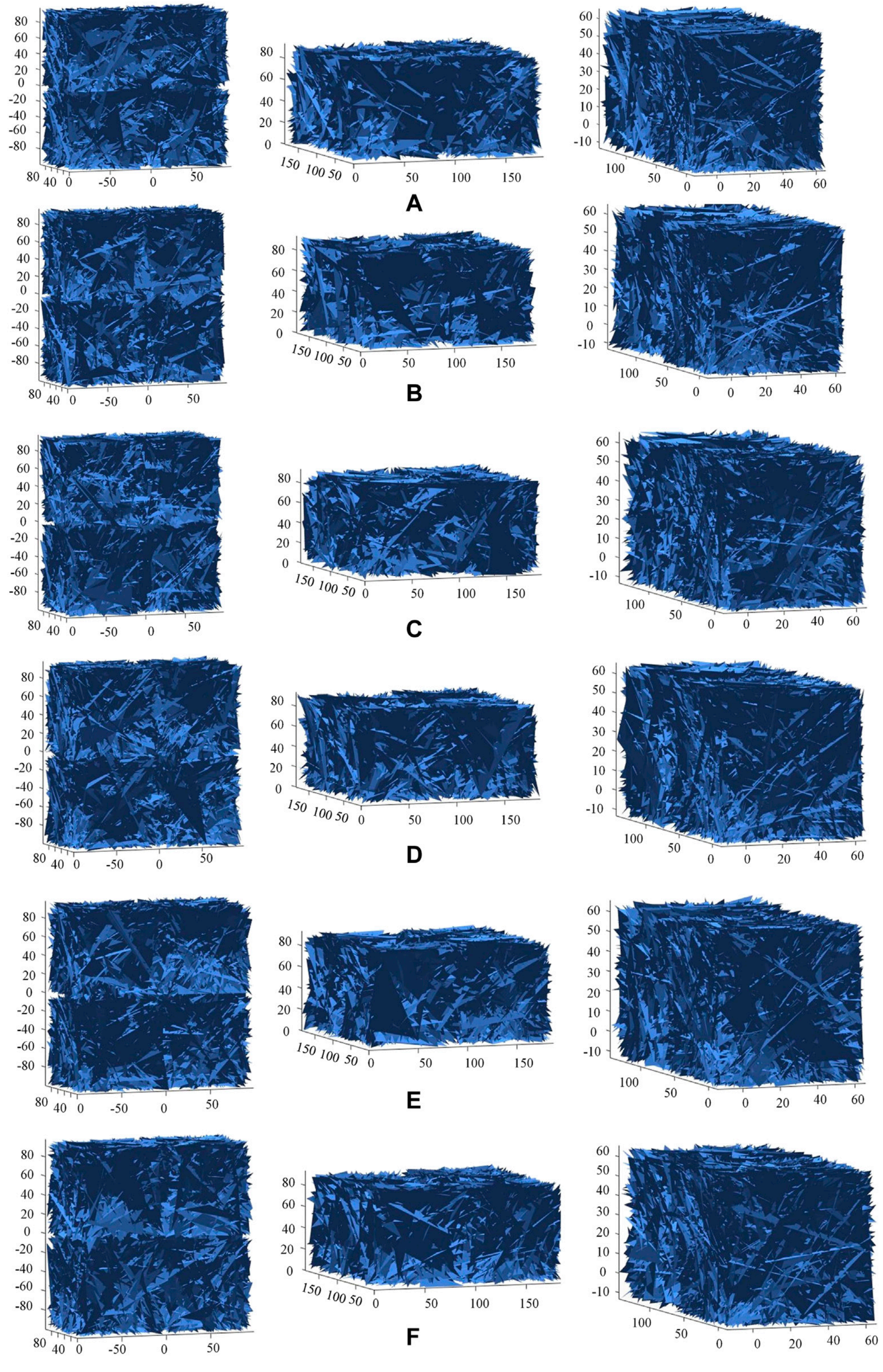


FIGURE 9 | The key sensitivity test. **(A)** a_1+10^{-15}, **(B)** a_2+10^{-15}, **(C)** b_1+10^{-15}, **(D)** b_2+10^{-15}, **(E)** $L+10^{-15}$, **(F)** $C+10^{-15}$.

more evenly coordinate values are distributed, the less effective information the information source has. The coordinates in the vertex matrix are floating-point numbers. Therefore, the positive integer less than or equal to the coordinates in vertex matrix is taken. Thus, for the vertex matrix with 256 levels, information entropy can be expressed as:

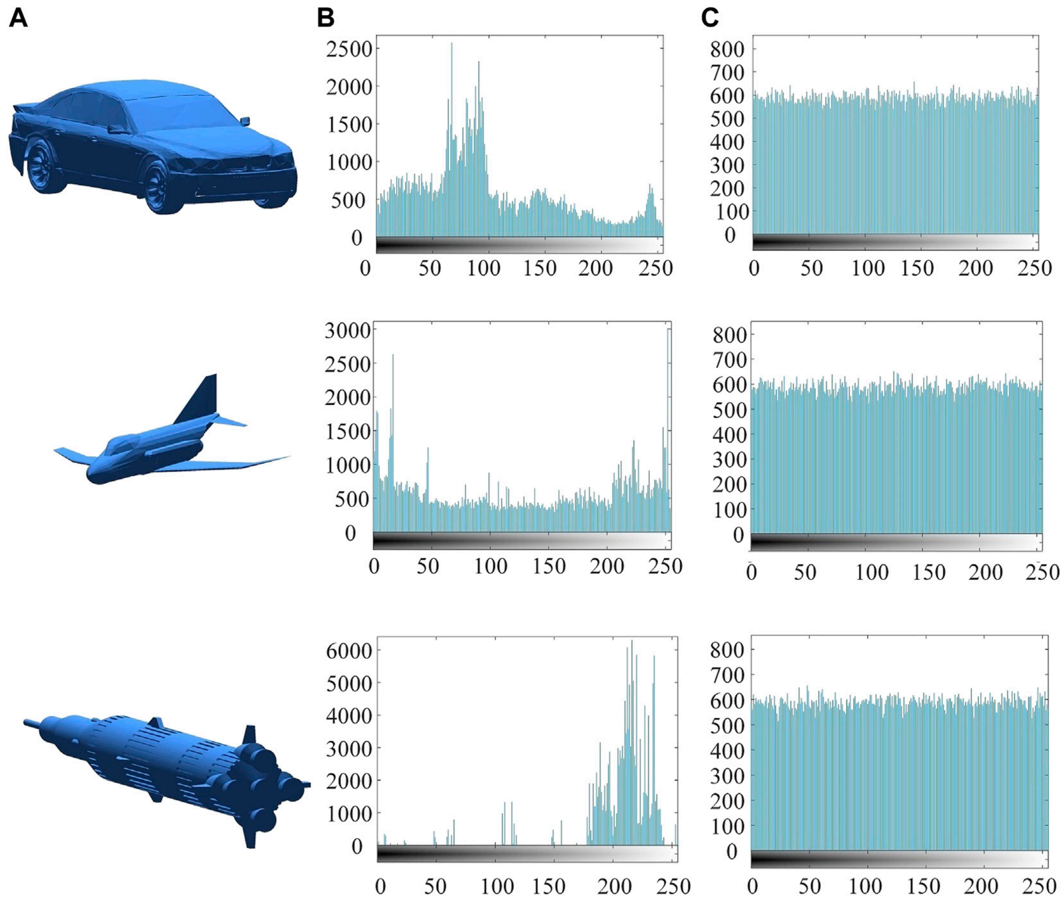


FIGURE 10 | The histogram analysis results. **(A)** Plaintext images. **(B)** Histograms of plaintext images. **(C)** Histograms of cipher images.

TABLE 5 | Results of the χ^2-value test

3D model	χ^2-value (plaintext)	χ^2-value (ciphertext)	Critical value		
			$\chi^2_{0.1}$ (255)	$\chi^2_{0.05}$ (255)	$\chi^2_{0.01}$ (255)
Car	72,638.9132	246.2416	Pass	Pass	Pass
Jet	47,615.5636	260.4825	Pass	Pass	Pass
Rocket	647,556.5317	263.6170	Pass	Pass	Pass

$$H(x) = \sum_{i=0}^{N} p(x_i)\log_2 \frac{1}{p(x_i)} \tag{14}$$

Table 8 gives the information entropy of plaintexts and ciphertexts. The information entropy of original images Car, Jet, and Rocket of different sizes are 7.7123, 7.8348, and 5.929, and the information entropy of ciphertexts are significantly improved to 7.9988, 7.9988, and 7.9987, respectively. The information entropies of ciphertexts are larger than that of the plaintexts and close to the theoretical value. Compared with other algorithms, the proposed algorithm could confuse the coordinate information successfully and resist statistical attacks.

Differential Attack Analysis

When one of the original images is changed in a slight way, the encrypted images should be significantly different. NPCR and UACI could reflect the sensitivity of the original 3D images, as shown in the following formula:

$$\mathrm{NPCR} = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100\% \tag{15}$$

$$\mathrm{UACI} = \frac{1}{M \times N}\left[\sum\nolimits_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255}\right] \times 100\% \tag{16}$$

$$D(i,j) = \begin{cases} 1, C_1(i,j) \neq C_2(i,j) \\ 0, C_1(i,j) = C_2(i,j) \end{cases} \tag{17}$$

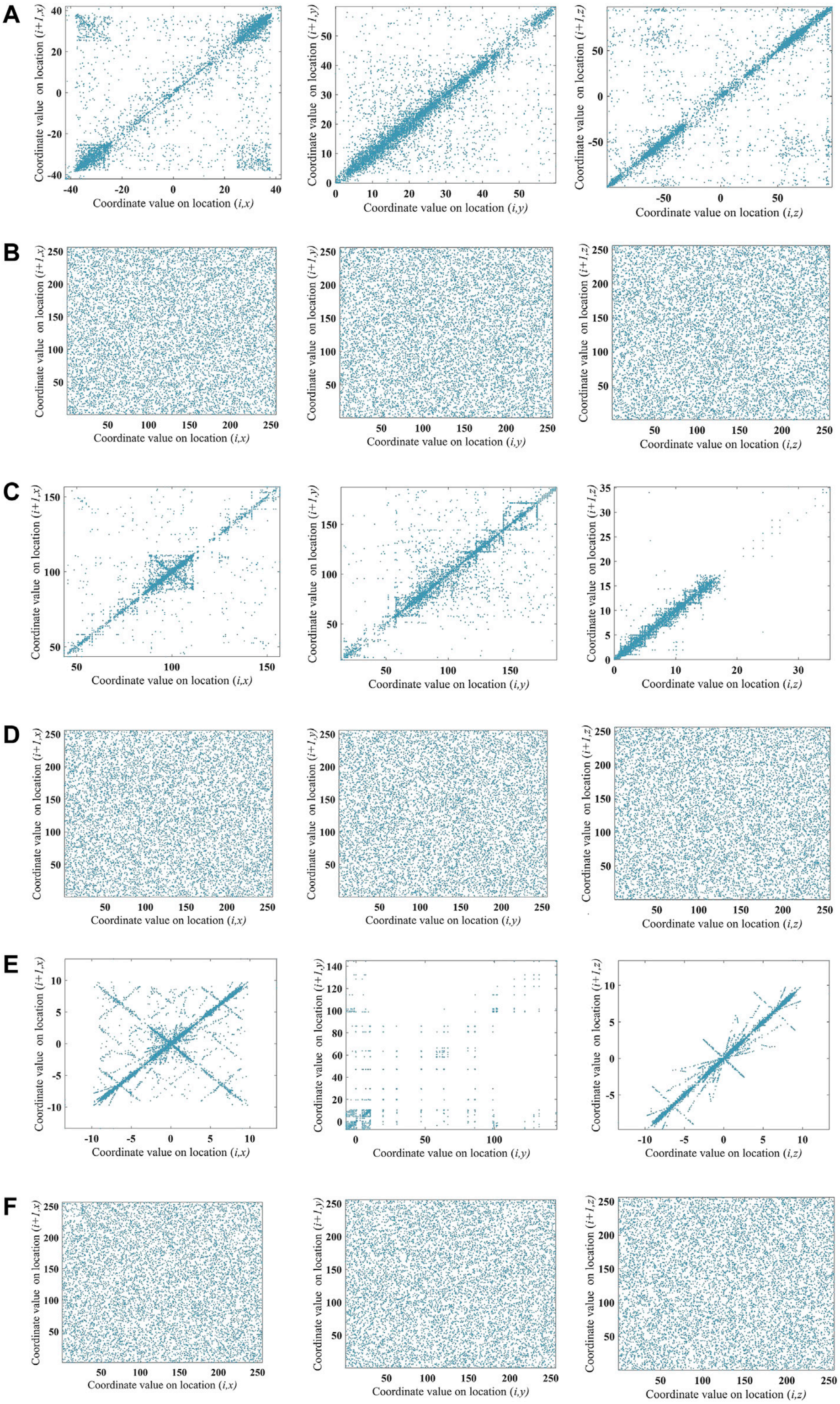


FIGURE 11 | The correlation analysis of different images. **(A)** Car plaintext. **(B)** Car ciphertext. **(C)** Jet plaintext. **(D)** Jet ciphertext. **(E)** Rocket plaintext. **(F)** Rocket ciphertext.

TABLE 6 | The correlation coefficients between the plain images and cipher images

Algorithm	3D image	*X*-direction		*Y*-direction		*Z*-direction	
		Plaintext	Ciphertext	Plaintext	Ciphertext	Plaintext	Ciphertext
Our scheme	Car	0.40657	0.010585	0.24970	0.01920	0.68542	−0.00369
	Jet	0.04589	−0.005583	0.43335	0.00810	0.56907	0.01150
	Rocket	0.72592	−0.010559	0.94497	−0.00540	0.98495	0.00864
Reference [5]	Kun jet	0.9763	0.0118	0.9878	0.0062	0.9869	0.00038
	Air early warming	0.9982	−0.0041	0.9954	0.0051	0.9975	−0.0075
	Boeing-747	0.9794	−0.0030	0.9992	0.0002	0.9815	−0.0004
Reference [8]	Terrain	0.9923	−0.0254	0.7365	−0.0097	0.9850	0.0049

TABLE 7 | NIST test results

Test		p-Value	Pass rate	Result
Frequency	—	0.249284	0.99	Pass
Block frequency	—	0.262249	0.99	Pass
Cumulative sums	Forward	0.978072	0.99	Pass
	Reverse	0.834308	0.99	Pass
Runs	—	0.009535	1	Pass
Longest run	—	0.678686	1	Pass
Binary matrix rank	—	0.319084	1	Pass
FFT	—	0.657933	0.98	Pass
Non-overlapping template matching	—	0.153763	0.97	Pass
Overlapping template matching	—	0.058984	0.96	Pass
Maurer's "universal statistical"	—	0.304126	0.99	Pass
Approximate entropy	—	0.534146	0.99	Pass
The random excursions	$x = -4$	0.378138	0.98	Pass
	$x = -3$	0.299251	1	Pass
	$x = -2$	0.350485	1	Pass
	$x = -1$	0.637119	1	Pass
	$x = 1$	0.299251	1	Pass
	$x = 2$	0.350485	0.98	Pass
	$x = 3$	0.888137	1	Pass
	$x = 4$	0.772760	0.97	Pass
Random excursions variant	$x = -9$	0.534146	1	Pass
	$x = -8$	0.213309	0.95	Pass
	$x = -7$	0.232760	0.95	Pass
	$x = -6$	0.378138	0.97	Pass
	$x = -5$	0.888137	0.95	Pass
	$x = -4$	0.637119	1	Pass
	$x = -3$	0.060239	0.98	Pass
	$x = -2$	0.949602	1	Pass
	$x = -1$	0.637119	1	Pass
	$x = 1$	0.253551	1	Pass
	$x = 2$	0.407091	0.98	Pass
	$x = 3$	0.162606	0.97	Pass
	$x = 4$	0.437274	1	Pass
	$x = 5$	0.437274	1	Pass
	$x = 6$	0.407091	1	Pass
	$x = 7$	0.949602	0.98	Pass
	$x = 8$	0.995711	0.98	Pass
	$x = 9$	0.964295	0.98	Pass
Serial	p-Value_1	0.595549	0.99	Pass
	p-Value_2	0.035174	0.99	Pass
Linear complexity	—	0.162606	1	Pass

where C_1 and C_2 represent ciphertexts before and after a coordinate value of plaintext changes by 1. When the NPCR is close to 1 and the UACI is about 0.3346, the algorithm is considered to have passed the test.

The idea values of NPCR and UACI are given [51, 52]. According to **Table 9** and **Table 10**, the mean value of NPCR and UACI of the three images is 99.6154% and 33.35%. **Figure 12** shows the NPCR and UACI calculated by randomly varying the vertex matrix values in different test model. The test results of NPCR and UACI are greater than the theoretical values with a confidence level of 0.05. A comparison of NPCR and UACI with different algorithms

TABLE 8 | The information entropy test for different algorithms

3D model	Proposed scheme				Reference [3]		Reference [8]	Reference [28]	Reference [29]
	Car	**Jet**	**Rocket**	**Lena**	**Baboon**	**Peppers**	**Terrain**	**Camera**	**Camera**
Plaintext	7.7123	7.8348	5.929	7.2544	7.6599	7.2118	6.8030	7.0097	7.0097
Ciphertext	7.9988	7.9988	7.9987	7.9959	7.9959	7.9959	7.9980	7.9968	7.9972

TABLE 9 | The NPCR results for different images

3D image	NPCR (%)	Text result		
		$NPCR^*_{0.05} = 99.5693\%$	$NPCR^*_{0.01} = 99.5527\%$	$NPCR^*_{0.001} = 99.5341\%$
Car	99.6197	Pass	Pass	Pass
Jet	99.6147	Pass	Pass	Pass
Rocket	99.6119	Pass	Pass	Pass

TABLE 10 | The UACI results for different images

3D image	UACI (%)	Text result		
		$UACI^{*+}_{0.05} = 33.6447\%$ $UACI^{*-}_{0.05} = 33.2824\%$	$UACI^{*+}_{0.01} = 33.7016\%$ $UACI^{*-}_{0.01} = 33.2255\%$	$UACI^{*+}_{0.001} = 33.7677\%$ $UACI^{*-}_{0.001} = 33.1594\%$
Car	33.3857	Pass	Pass	Pass
Jet	33.3774	Pass	Pass	Pass
Rocket	33.2855	Pass	Pass	Pass

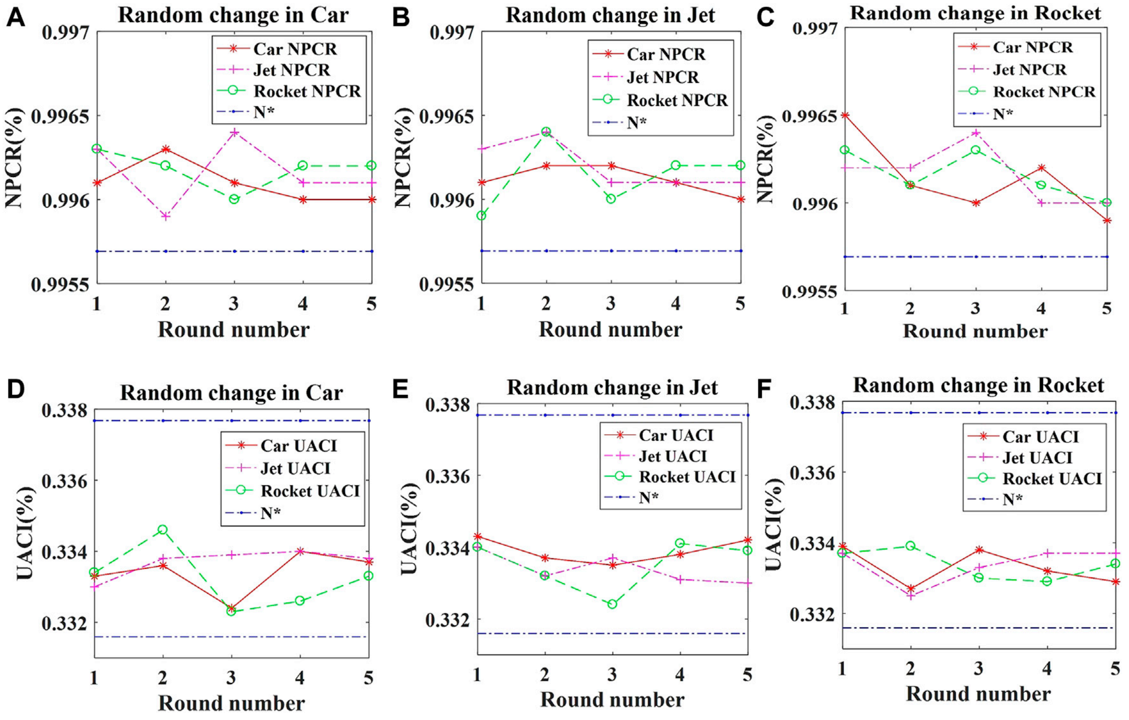


FIGURE 12 | The test results of differential attacks. **(A)** NPCR of Car change. **(B)** NPCR of Jet change. **(C)** NPCR of Rocket change. **(D)** UACI of Car change. **(E)** UACI of Jet change. **(F)** UACI of Rocket change.

TABLE 11 | Comparison of NPCR and UACI with different algorithms

Algorithm	Proposed scheme	Reference [5]	Reference [8]	Reference [28]	Reference [29]	Reference [36]	Reference [43]
NPCR (%)	99.6197	99.61	99.59	99.54	99.60	99.6060	99.6114
UACI (%)	33.3857	33.44	33.26	32.19	33.45	33.5126	33.54

is shown in **Table 11**, which proves the ability to resist differential attack.

Time Analysis

The image cryptosystem mainly includes three processes: chaotic system iteration, 3D Arnold algorithm and DNA encryption. The encryption time of images with 348,282 vertex coordinates is 30.1795 s. In Ref. [36], the encryption time of 89,271 data is 0.261795 s. In Ref. [40], the encryption time of the 256 × 256 gray image is 7.59 s. The proposed cryptosystem sacrifices the encryption time to ensure the encryption effect.

CONCLUSION

A novel 3D image encryption based on the memristive chaotic system and RNA crossover and mutation is proposed. Firstly, the chaotic system based on two memristors is analyzed by bifurcation diagrams and Lyapunov exponential spectrums. It is shown that the chaotic system has complex dynamical behaviors and large parameter space, which are suitable for image encryption. Secondly, the initial values are generated by SHA-256, so different plaintexts and merge orders will generate different keys and chaotic sequences. Besides, an improved 3D Arnold algorithm is used to confuse the vertex positions and convert coordinates. Based on dynamic RNA encoding and decoding rules, the RNA crossover and mutation algorithms are given to directly exchange codons. In addition, the cryptosystem can encrypt multiple images at the same time, which improves security and effectiveness. Finally, experimental results indicate that the cryptosystem could ensure the security of three 3D images simultaneously, and the security analysis verifies that it could resist various attacks effectively.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

RC provided the idea of algorithm, carried out the simulations, arranged the architecture and drafted the manuscript. SZ and XG supervised the work and revised the manuscript. Both authors read and approved the final manuscript.

FUNDING

This research was funded by the National Natural Science Foundation of China (No. 61231006 and No. 61501078).

REFERENCES

1. Jiang R, Zhou H, Zhang W, Yu N. Reversible Data Hiding in Encrypted Three-Dimensional Mesh Models. *IEEE Trans Multimedia* (2018) 20(1):55–67. doi:10.1109/tmm.2017.2723244
2. Peng Y, He S, Sun K. A Higher Dimensional Chaotic Map with Discrete Memristor. *AEU - Int J Elect Commun* (2021) 129:153539–7. doi:10.1016/j.aeue.2020.153539
3. Joshi AB, Kumar D, Gaffar A, Mishra DC. Triple Color Image Encryption Based on 2D Multiple Parameter Fractional Discrete Fourier Transform and 3D Arnold Transform. *Opt Lasers Eng* (2020) 133:106139–13. doi:10.1016/j.optlaseng.2020.106139
4. Pham G, Lee S-H, Kwon K-R. Interpolating Spline Curve-Based Perceptual Encryption for 3D Printing Models. *Appl Sci* (2018) 8(2):242–54. doi:10.3390/app8020242
5. Xu J, Zhao C, Mou J. A 3D Image Encryption Algorithm Based on the Chaotic System and the Image Segmentation. *IEEE Access* (2020) 8(99):145995–6005. doi:10.1109/access.2020.3005925
6. Pham N-G, Lee S-H, Lee S-H, Kwon O-H, Kwon K-R. 3D Printing Model Random Encryption Based on Geometric Transformation. *Int J Machine Learn Comput* (2018) 8(2):186–90. doi:10.18178/ijmlc.2018.8.2.685
7. Jin X, Zhu S, Xiao C, Sun H, Li X, Zhao G, et al. 3D Textured Model Encryption *via* 3D Lu Chaotic Mapping. *Sci China Inf Sci* (2017) 60(12):1–9. doi:10.1007/s11432-017-9266-1
8. Wang X, Xu M, Li Y. Fast Encryption Scheme for 3D Models Based on Chaos System. *Multimed Tools Appl* (2019) 78(23):33865–84. doi:10.1007/s11042-019-08171-2
9. An X, Qiao S. The Hidden, Period-Adding, Mixed-Mode Oscillations and Control in a HR Neuron under Electromagnetic Induction. *Chaos, Solitons & Fractals* (2021) 143:110587–18. doi:10.1016/j.chaos.2020.110587
10. Qiao S, An X-L. Dynamic Expression of a HR Neuron Model under an Electric Field. *Int J Mod Phys B* (2020) 35(02):2150024–3. doi:10.1142/s0217979221500247
11. Ma X, Mou J, Liu J, Ma C, Yang F, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor-Memcapacitor. *Nonlinear Dyn* (2020) 100(3):2859–76. doi:10.1007/s11071-020-05601-x
12. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S. Dynamics Analysis, Hardware Implementation and Engineering Applications of Novel Multi-Style Attractors in a Neural Network under Electromagnetic Radiation. *Chaos, Solitons & Fractals* (2021) 152:111350–14. doi:10.1016/j.chaos.2021.111350
13. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Jin J, et al. Design and FPGA Implementation of a Pseudo-random Number Generator Based on a Hopfield Neural Network under Electromagnetic Radiation. *Front Phys* (2021) 9:1–15. doi:10.3389/fphy.2021.690651
14. Ma C, Mou J, Xiong L, Banerjee S, Liu T, Han X. Dynamical Analysis of a New Chaotic System: Asymmetric Multistability, Offset Boosting Control and Circuit Realization. *Nonlinear Dyn* (2021) 103(3):2867–80. doi:10.1007/s11071-021-06276-8
15. Liu T, Yan H, Banerjee S, Mou J. A Fractional-Order Chaotic System with Hidden Attractor and Self-Excited Attractor and its DSP Implementation. *Chaos, Solitons & Fractals* (2021) 145(2):110791–12. doi:10.1016/j.chaos.2021.110791

16. Liu T, Banerjee S, Yan H, Mou J. Dynamical Analysis of the Improper Fractional-Order 2D-SCLMM and its DSP Implementation. *Eur Phys J Plus* (2021) 136(5):1–17. doi:10.1140/epjp/s13360-021-01503-y
17. Liu H, Zhang Y, Kadir A, Xu Y. Image Encryption Using Complex Hyper Chaotic System by Injecting Impulse into Parameters. *Appl Maths Comput* (2019) 360:83–93. doi:10.1016/j.amc.2019.04.078
18. Liu H, Kadir A, Liu J. Color Pathological Image Encryption Algorithm Using Arithmetic over Galois Field and Coupled Hyper Chaotic System. *Opt Lasers Eng* (2019) 122:123–33. doi:10.1016/j.optlaseng.2019.05.027
19. Liu H, Kadir A, Xu C. Cryptanalysis and Constructing S-Box Based on Chaotic Map and Backtracking. *Appl Maths Comput* (2020) 376:125153. doi:10.1016/j.amc.2020.125153
20. Liu H, Kadir A, Xu C. Color Image Encryption with Cipher Feedback and Coupling Chaotic Map. *Int J Bifurcation Chaos* (2020) 30(12):2050173. doi:10.1142/s0218127420501734
21. Si Y, Liu H, Chen Y. Constructing Keyed strong S-Box Using an Enhanced Quadratic Map. *Int J Bifurcation Chaos* (2021) 31(10):2150146. doi:10.1142/s0218127421501467
22. Liu H, Wang X, Kadir A. Constructing Chaos-Based Hash Function *via* Parallel Impulse Perturbation. *Soft Comput* (2021) 25(16):11077–86. doi:10.1007/s00500-021-05849-4
23. Ma C, Mou J, Li P, Liu T. Dynamic Analysis of a New Two-Dimensional Map in Three Forms: Integer-Order, Fractional-Order and Improper Fractional-Order. *Eur Phys J Spec Top* (2021) 230(7-8):1945–57. doi:10.1140/epjs/s11734-021-00133-w
24. Yu F, Li L, He B, Liu L, Qian S, Zhang Z, et al. Pseudorandom Number Generator Based on a 5D Hyperchaotic Four-wing Memristive System and its FPGA Implementation. *Eur Phys J Spec Top* (2021) 230(7-8):1763–72. doi:10.1140/epjs/s11734-021-00132-x
25. Peng Y, He S, Sun K. Chaos in the Discrete Memristor-Based System with Fractional-Order Difference. *Results Phys* (2021) 24:104106–7. doi:10.1016/j.rinp.2021.104106
26. Chen X, Qian S, Yu F, Zhang Z, Shen H, Huang Y, et al. Pseudorandom Number Generator Based on Three Kinds of Four-Wing Memristive Hyperchaotic System and its Application in Image Encryption. *Complexity* (2020) 2020:1–17. doi:10.1155/2020/8274685
27. Ma X, Mou J, Xiong L, Banerjee S, Cao Y, Wang J. A Novel Chaotic Circuit with Coexistence of Multiple Attractors and State Transition Based on Two Memristors. *Chaos, Solitons & Fractals* (2021) 152:111363–11. doi:10.1016/j.chaos.2021.111363
28. Yang F, Mou J, Ma C, Cao Y. Dynamic Analysis of an Improper Fractional-Order Laser Chaotic System and its Image Encryption Application. *Opt Lasers Eng* (2020) 129:106031–16. doi:10.1016/j.optlaseng.2020.106031
29. Li X, Mou J, Xiong L, Wang Z, Xu J. Fractional-order Double-Ring Erbium-Doped Fiber Laser Chaotic System and its Application on Image Encryption. *Opt Laser Tech* (2021) 140:107074–18. doi:10.1016/j.optlastec.2021.107074
30. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S. A New Multi-Scroll Chua's Circuit with Composite Hyperbolic tangent-cubic Nonlinearity: Complex Dynamics, Hardware Implementation and Image Encryption Application. *Integration* (2021) 81:71–83. doi:10.1016/j.vlsi.2021.05.011
31. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Du S. FPGA Implementation and Image Encryption Application of a New PRNG Based on a Memristive Hopfield Neural Network with a Special Activation Gradient. *Chin Phys. B* (2021) 31(2):20505–020505. doi:10.1088/1674-1056/ac3cb2
32. Mou J, Yang F, Chu R, Cao Y. Image Compression and Encryption Algorithm Based on Hyper-Chaotic Map. *Mobile Netw Appl* (2019) 26(5):1849–61. doi:10.1007/s11036-019-01293-9
33. Yang F, Mou J, Luo C, Cao Y. An Improved Color Image Encryption Scheme and Cryptanalysis Based on a Hyperchaotic Sequence. *Phys Scr* (2019) 94(8):085206–9. doi:10.1088/1402-4896/ab0033
34. Ravichandran D, Praveenkumar P, Balaguru Rayappan JB, Amirtharajan R. Chaos Based Crossover and Mutation for Securing DICOM Image. *Comput Biol Med* (2016) 72:170–84. doi:10.1016/j.compbiomed.2016.03.020
35. Chai X, Bi J, Gan Z, Liu X, Zhang Y, Chen Y. Color Image Compression and Encryption Scheme Based on Compressive Sensing and Double Random Encryption Strategy. *Signal Process.* (2020) 176:107684–18. doi:10.1016/j.sigpro.2020.107684
36. Zhang X, Hu Y. Multiple-image Encryption Algorithm Based on the 3D Scrambling Model and Dynamic DNA Coding. *Opt Laser Tech* (2021) 141(441):107073–16. doi:10.1016/j.optlastec.2021.107073
37. Chai X, Chen Y, Broyde LA Novel Chaos-Based Image Encryption Algorithm Using DNA Sequence Operations. *Opt Lasers Eng* (2017) 88:197–213. (Complete). doi:10.1016/j.optlaseng.2016.08.009
38. Chai X, Fu X, Gan Z, Lu Y, Chen Y. A Color Image Cryptosystem Based on Dynamic DNA Encryption and Chaos. *Signal Process.* (2019) 155(FEB.):44–62. doi:10.1016/j.sigpro.2018.09.029
39. Wang X-Y, Zhang Y-Q, Zhao Y-Y. A Novel Image Encryption Scheme Based on 2-D Logistic Map and DNA Sequence Operations. *Nonlinear Dyn* (2015) 82(3):1269–80. doi:10.1007/s11071-015-2234-7
40. Chai X, Gan Z, Lu Y, Chen Y, Han D. A Novel Image Encryption Algorithm Based on the Chaotic System and DNA Computing. *Int J Mod Phys C* (2017) 28(05):1750069–24. doi:10.1142/S0129183117500693
41. Hao J, Li H, Yan H, Mou J. A New Fractional Chaotic System and its Application in Image Encryption with DNA Mutation. *IEEE Access* (2021) 9:52364–77. doi:10.1109/ACCESS.2021.3069977
42. Martín del Rey A. A Method to Encrypt $$3$$ 3 D Solid Objects Based on Three-Dimensional Cellular Automata. In: E Onieva, I Santos, E Osaba, H Quintián, E Corchado, editors. *Lecture Notes in Computer Science*, 9121. Cham: Springer (2015). p. 427–38. doi:10.1007/978-3-319-19644-2_36
43. Devi RS, Aravind ARN, Vishal JC, Amritha D, Thenmozhi K, Rayappan JBB, et al. Image Encryption through RNA Approach Assisted with Neural Key Sequences. *Multimed Tools Appl* (2020) 79(17-18):12093–124. doi:10.1007/s11042-019-08562-5
44. Wang X, Liu L. Application of Chaotic Josephus Scrambling and RNA Computing in Image Encryption. *Multimed Tools Appl* (2021) 80(15):23337–58. doi:10.1007/s11042-020-10209-9
45. Wang X, Guan N. A Novel Chaotic Image Encryption Algorithm Based on Extended Zigzag Confusion and RNA Operation. *Opt Laser Tech* (2020) 131(6):106366–17. doi:10.1016/j.optlastec.2020.106366
46. Abbasi AA, Mazinani M, Hosseini R. Chaotic Evolutionary-Based Image Encryption Using RNA Codons and Amino Acid Truth Table. *Opt Laser Tech* (2020) 132(12):106465–13. doi:10.1016/j.optlastec.2020.106465
47. JarJar A. Two Advanced Classics Exploiting DNA and RNA Characteristics to Encrypt a Color Image. *Multimed Tools Appl* (2021) 80(16):24603–29. doi:10.1007/s11042-021-10658-w
48. Zhang D, Chen L, Li T. Hyper-Chaotic Color Image Encryption Based on Transformed Zigzag Diffusion and RNA Operation. *Entropy* (2021) 23(3):361–23. doi:10.3390/e23030361
49. Zhu C, Gan Z, Lu Y, Chai X. An Image Encryption Algorithm Based on 3-D DNA Level Permutation and Substitution Scheme. *Multimed Tools Appl* (2019) 79(11-12):7227–58. doi:10.1007/s11042-019-08226-4
50. Jin X, Wu Z, Song C, Zhang C, Li X. 3D Point Cloud Encryption through Chaotic Mapping. In: *Pacific Rim Conference on Multimedia*, 9916 (2016). p. 119–29. doi:10.1007/978-3-319-48890-5_12
51. Hua Z, Zhou Y, Huang H. Cosine-transform-based Chaotic System for Image Encryption. *Inf Sci* (2019) 480:403–19. doi:10.1016/j.ins.2018.12.048
52. Gao X, Mou J, Xiong L, Sha Y, Yan H, Cao Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn* (2022). doi:10.1007/s11071-021-07192-7

Conflict of Interest: XG was employed by the company Tokai Carbon Dalian Co., Ltd.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

The handling editor declared a shared affiliation with one of the authors RC at time of review.

frontiers | Frontiers in Physics

ORIGINAL RESEARCH
published: 27 April 2022
doi: 10.3389/fphy.2022.906138

A New Four-Dimensional Chaotic System and its Circuit Implementation

Xuan Wang, Yiran Feng and Yixin Chen*

School of Mechanical Engineering and Automation, Dalian Polytechnic University, Dalian, China

A new four-dimensional chaotic system is designed in the paper. The equilibrium point and stability of the chaotic system are analyzed, and the dynamical behaviors of the system under different parameters are analyzed by using Lyapunov exponents, Bfurcation diagram, SE and C0 complexity algorithms. The special phenomenon of the coexistence of attractors is also found. Finally, the implementation of circuit of the new system is carried out using digital signal processing (DSP) technology, and the results are consistent with the numerical simulation results, which prove the validity of the theoretical analysis. Through analysis and simulation of the system, it can be found that it has relatively rich dynamic characteristics and can be applied in areas such as confidential communication and image encryption.

Keywords: chaotic system, hidden attractor, coexistence of attractors, SE and C0 complexity, DSP implementation

Edited by:
Chunlai Li,
Hunan Institute of Science and Technology, China

Reviewed by:
Feifei Yang,
Lanzhou University of Technology, China
Li Xiong,
Hexi University, China

****Correspondence:***
Yiran Feng
fengyr@dlpu.edu.cn

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *28 March 2022*
Accepted: *11 April 2022*
Published: *27 April 2022*

Citation:
Wang X, Feng Y and Chen Y (2022) A New Four-Dimensional Chaotic System and its Circuit Implementation. Front. Phys. 10:906138. doi: 10.3389/fphy.2022.906138

1 INTRODUCTION

In 1963, Lorenz proposed the famous Lorenz chaotic system [1], which has complex dynamical behavior although it is only three-dimensional. As an important branch of nonlinear systems, chaotic systems have good prospects for applications in secure communications, image encryption and neural networks because of their unique properties [2–24]. As the research progresses, chaotic systems have been expanded to the fractional-order domain [25–33], not only in the integer-order direction [34–40]. In both integer-order and fractional-order systems, some of them have the special phenomenon of hidden attractors.

The Sil'nikov criterion considers that a chaotic system requires at least one unstable equilibrium point [41]. However, the Sil'nikov criterion is a sufficient and unnecessary condition for the emergence of chaotic phenomena. A class of systems discovered in recent years may have only stable equilibrium points, no equilibrium points or infinitely many equilibrium points, the chaotic state is still apparent, with the attractor basin not meeting the adjacent domain of the equilibrium point, such fields of attraction are called hidden attractors [42–44].

For classical systems of chaos, such as Lorenz [1], Chen [45] and Lü [46] systems, the equilibrium points of these systems are unstable and therefore generate self-excited attractors. However, if the equilibrium point of the system is unstable, then hidden attractors may be created. Since the 21st century, more studies has been done on hidden attractors. In 2010, Vagaitsev et al. [47] discovered hidden attractors in generalized Chua's circuits firstly, and since then there has been a boom in the study of hidden attractors. In 2012, G.A. Leonov et al. [48] studied hidden attractors in smooth Chua's systems and proposed an algorithm for hidden attractors localization. In 2014, Lao et al. [49] introduced a cost function in chaotic systems with hidden attractors and used it for parameter estimation in chaotic circuit systems. In 2015, Chen et al. [50] discovered the coexistence of hidden attractors by improving the classical Chua's circuit using memristors. In 2017, M. Borah et al. [51] proposed a fractional-order chaotic system and it does not have an equilibrium point, which is a system of chaos with hidden attractors. In 2019, Cang et al. [52] discovered the coexistence of hidden attractors under different parameters in a class of Lorenz-like systems. In the same year, Zhang et al. [53] introduced a multiscroll hyperchaotic system with hidden attractors based on the Jerk system. In 2020, Deng

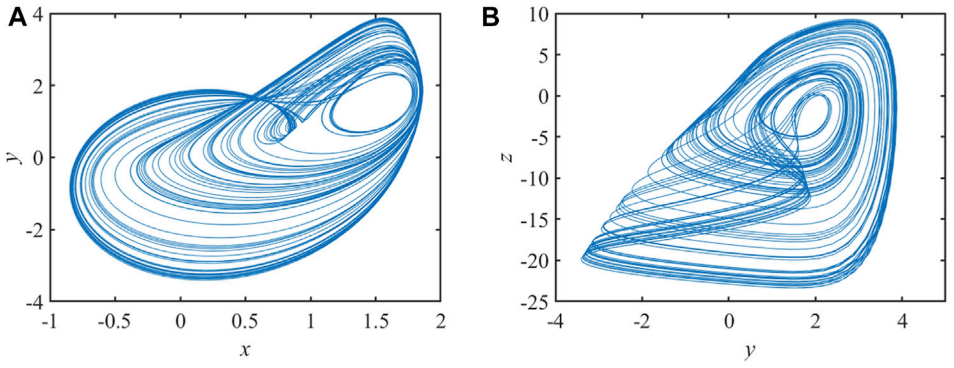


FIGURE 1 | Hidden chaotic attractor of system (1), **(A)** *x*–*y* plane **(B)** *y*–*z* plane.

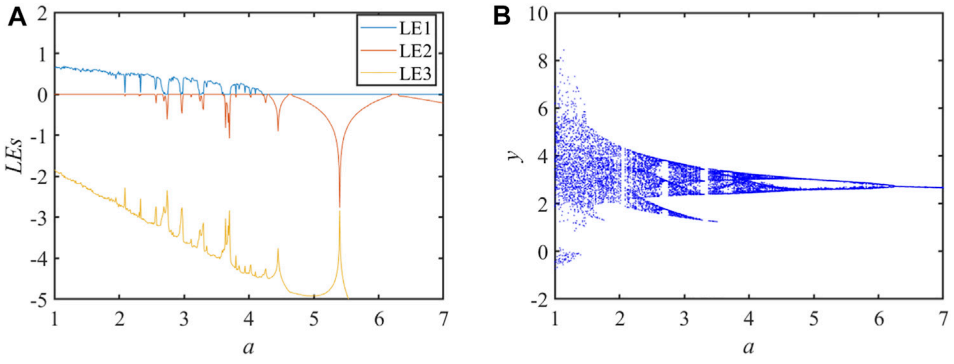


FIGURE 2 | Dynamical analysis with parameter *a*, **(A)** LEs, **(B)** BD.

TABLE 1 | The state and LEs under the change of parameter *a*.

Range	LEs	State	Range	LEs	State
1–2.69	+0 - -	Chaos	3.72–3.79	+0 - -	Chaos
2.70–2.75	0 - - -	Period	3.80–3.81	0 - - -	Period
2.76–2.95	+0 - -	Chaos	3.82–4.00	+0 - -	Chaos
2.96–2.97	0 - - -	Period	4.01–4.03	0 - - -	Period
2.98–3.23	+0 - -	Chaos	4.04–4.09	+0 - -	Chaos
3.24–3.30	0 - - -	Period	4.10–4.11	0 - - -	Period
3.31–3.59	+0 - -	Chaos	4.12–4.20	+0 - -	Chaos
3.60–3.71	0 - - -	Period	4.21–7.00	0 - - -	Period

et al. [54] proposed a multi-wing system of hidden attractors with only a single stable equilibrium point. In addition, Mou et al. [55] presented a chaotic circuit based on memristor–memcapacitor, which is a circuit system with an abundance of hidden attractors.

In addition, the coexistence of attractors is an interesting phenomenon that occurs in some chaotic systems where the parameters remain the same and the system enters different orbits and thus forms different attractors under different initial conditions due to the properties of the chaotic system itself. In dynamical systems, the presence of this phenomenon can lead to very complex behavior of the system. In the article, a novel four-dimensional chaotic system is arised from a modification of Wei system [54]. It is found that the system has stable equilibrium points and can be classified as a chaotic system with hidden attractors, while the system also has complex dynamical phenomena, including the coexistence of periodic windows and attractors. The stability, interchangeability, and agility of system can be significantly improved by using software implementation, and therefore the effectiveness of the system is verified by execution onto a DSP platform.

The remainder of this article is presented below. In **Section 2**, the mathematical model of the system is presented and the type of equilibrium points and their stability are analyzed. In **Section 3**, the dynamical behavior of the 4-dimensional system is analyzed, including Lyapunov exponent spectrum (LEs) and bifurcation diagram (BD), coexistence of attractors, SE and C0 complexity. In **Section 4**, a circuit realization of the 4-dimensional chaotic system is performed through a DSP platform. Finally, conclusions are given in **Section 5**.

2 MATHEMATICAL MODELS

2.1 Analysis of Equilibrium Points

A new chaotic system is created by inserting extra variables *w* and a constant *k* into the three-dimensional system of chaos, as follows:

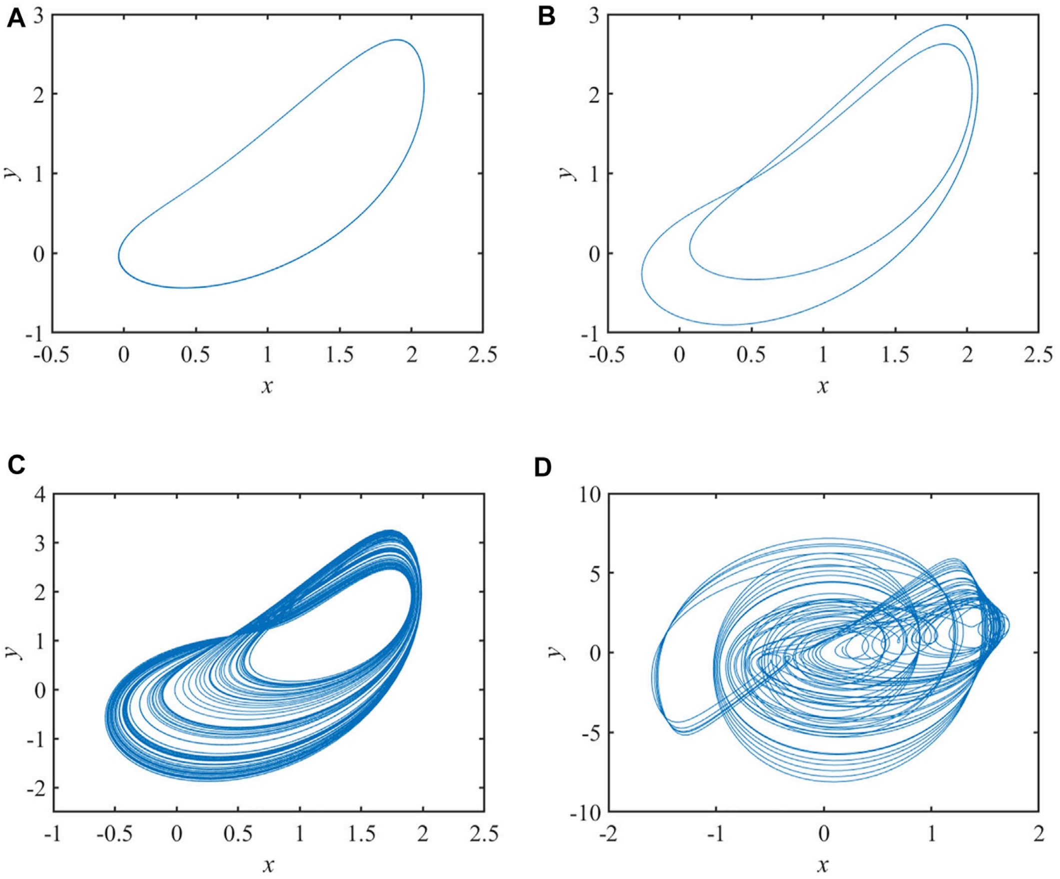


FIGURE 3 | Phase diagram of the system, **(A)** $a = 6.8$, **(B)** $a = 5.8$, **(C)** $a = 4$, **(D)** $a = 1.2$.

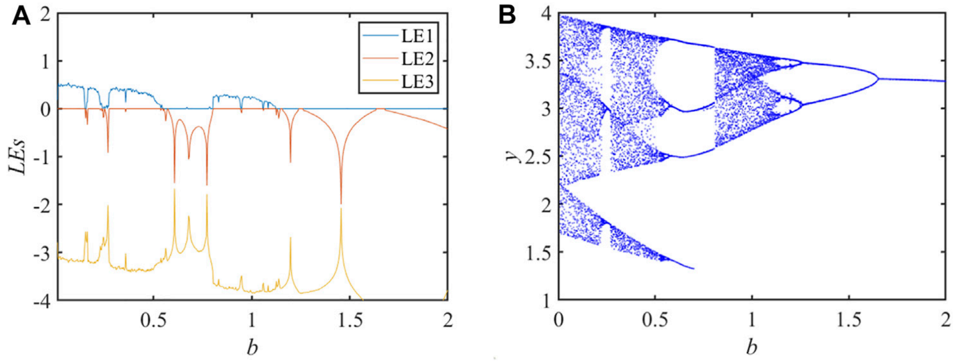


FIGURE 4 | Dynamical analysis with parameter b, **(A)** LEs, **(B)** BD.

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = -by + xz + k \\ \dot{z} = d - e^{xy} \\ \dot{w} = czw \end{cases}. \tag{1}$$

The dispersion of the system is obtained from the system dynamical equations and can be expressed as

$$\nabla V = \frac{\partial \dot{x}}{\partial x} + \frac{\partial \dot{y}}{\partial y} + \frac{\partial \dot{z}}{\partial z} + \frac{\partial \dot{w}}{\partial w}. \tag{2}$$

Setting $a = 2.6$, $b = 0.2$, $c = 5$, $d = 17$, $k = 3$, and the initial conditions as $(1, -3, 0.1, 7)$. And then

$$\nabla V = \frac{\partial \dot{x}}{\partial x} + \frac{\partial \dot{y}}{\partial y} + \frac{\partial \dot{z}}{\partial z} + \frac{\partial \dot{w}}{\partial w} = -2.3 < 0, \tag{3}$$

TABLE 2 | The state and LEs under the change of parameter *b*.

Range	LEs	State	Range	LEs	State
0.010–0.142	+0 - -	Chaos	0.786–0.798	0 - - -	Period
0.143–0.156	0 - - -	Period	0.799–0.939	+0 - -	Chaos
0.157–0.234	+0 - -	Chaos	0.940–0.945	0 - - -	Period
0.235–0.241	0 - - -	Period	0.946–1.052	+0 - -	Chaos
0.242–0.546	+0 - -	Chaos	1.053–1.057	0 - - -	Period
0.547–0.662	0 - - -	Period	1.058–1.078	+0 - -	Chaos
0.663–0.665	+0 - -	Chaos	1.079–1.081	0 - - -	Period
0.666–0.777	0 - - -	Period	1.082–1.119	+0 - -	Chaos
0.778–0.785	+0 - -	Chaos	1.120–2.000	0 - - -	Period

Eq. 2 smaller than zero, it is proved that the system is dissipative and therefore may have chaotic attractors.

Setting $\dot{x} = \dot{y} = \dot{z} = \dot{w} = 0$, it means that

$$\begin{cases} a(y-x) = 0 \\ -by + xz + k = 0. \\ d - e^{xy} = 0 \\ czw = 0 \end{cases} \tag{4}$$

It can be obtained that when $d > 1$, the system has two equilibrium points E_1 $(\sqrt{\ln d}, \sqrt{\ln d}, \frac{k}{\sqrt{\ln d}} + b, 0)$ and E_2 $(\sqrt{\ln d}, \sqrt{\ln d}, \frac{k}{\sqrt{\ln d}} + b, 0)$.

Let the set of equilibrium points be P. The Jacobian matrix of system 3) at the set of equilibrium points p is

$$J = \begin{bmatrix} -a & a & 0 & 0 \\ z & -b & x & 0 \\ -ye^{xy} & -xe^{xy} & 0 & 0 \\ 0 & 0 & cw & cz \end{bmatrix}, \tag{5}$$

and then the characteristic equation for point set

$$\lambda^4 + a_1\lambda^3 + a_2\lambda^2 + a_3\lambda + a_4 = 0, \tag{6}$$

where $a_1 = 3.3$, $a_2 = 2.2298$, $a_3 = 0.9327$, and $a_4 = 0.2589$. The eigenvalues can be calculated from the characteristic equation as $\lambda_1 = -2.5545$, $\lambda_2 = -0.5$, $\lambda_3 = -0.1227\text{-}0.4332i$, $\lambda_4 = -0.1227 + 0.4332i$, λ_1 and λ_2 are negative real roots, and λ_3 and λ_4 are a pair of complex conjugate eigenvalues, so these two equilibrium points are saddle-focus equilibrium points, which are important for chaotic systems. According to the Routh-Hurwitz criterion, the real part of all roots is negative and satisfied with

$$\Delta_1 = a_1 > 0, \tag{7}$$

$$\Delta_2 = a_1a_2 - a_0a_3 > 0, \tag{8}$$

$$\Delta_3 = a_1a_2a_3 - a_1^2a_4 - a_0a_3^3 > 0, \tag{9}$$

$$\Delta_4 = a_4\Delta_3 > 0, \tag{10}$$

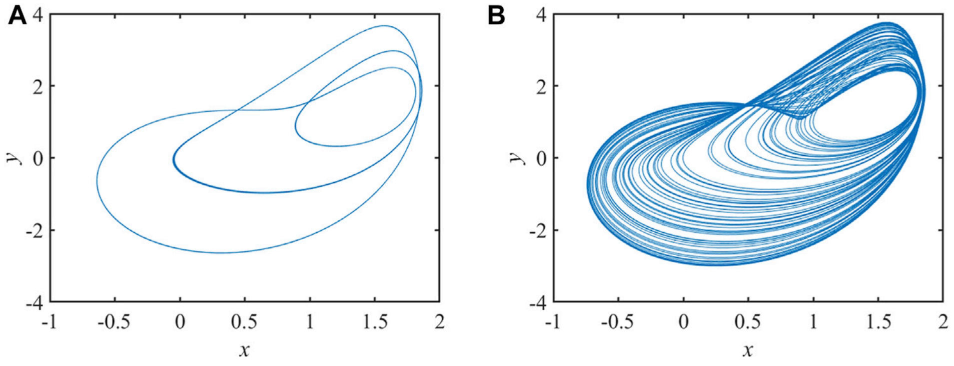


FIGURE 5 | Phase diagram of the system, **(A)** *b* = 0.70, **(B)** *b* = 0.46.

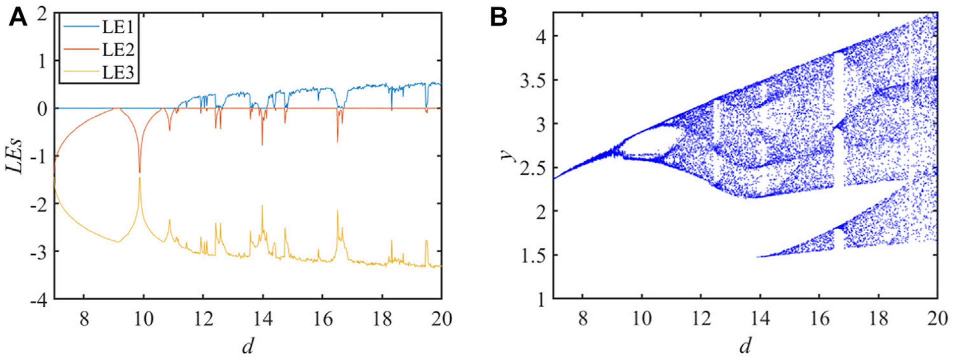


FIGURE 6 | Dynamical analysis with parameter *d*, **(A)** LEs, **(B)** BD.

TABLE 3 | The state and Les under the change of parameter *d*.

Range	LEs	State	Range	LEs	State
0–11.13	0 - - -	Period	13.65–13.84	+0 - -	Chaos
11.14–11.43	+0 - -	Chaos	13.85–14.12	0 - - -	Period
11.44–11.45	0 - - -	Period	14.13–14.37	+0 - -	Chaos
11.46–11.91	+0 - -	Chaos	14.38–14.39	0 - - -	Period
11.92–11.93	0 - - -	Period	14.40–14.73	+0 - -	Chaos
11.94–12.01	+0 - -	Chaos	14.74–14.82	0 - - -	Period
12.02–12.03	0 - - -	Period	14.83–16.48	+0 - -	Chaos
12.04–12.39	+0 - -	Chaos	16.49–16.71	0 - - -	Period
12.40–12.42	0 - - -	Period	16.72–18.30	+0 - -	Chaos
12.43–12.62	+0 - -	Chaos	18.31–18.32	0 - - -	Period
12.63–12.64	0 - - -	Period	18.33–19.45	+0 - -	Chaos
12.65–13.55	+0 - -	Chaos	19.46–19.47	0 - - -	Period
13.56–13.64	0 - - -	Period	19.48–20.00	+0 - -	Chaos

Thus, both equilibria E_1 and E_2 are asymptotically stable.

3 DYNAMICAL BEHAVIORS

3.1 Chaotic Attractor

In system (1), let $a = 2.6$, $b = 0.2$, $c = 5$, $d = 17$, $k = 3$, and the initial values are (1, −3, −0.1, 7). The phase diagrams of chaotic attractors in the system in different phase planes are shown in **Figure 1**.

3.2 Lyapunov Exponents and Bifurcation Diagram of the System

In the system, which is a four-dimensional chaotic system, the BD is analyzed in combination with the LEs to obtain the states with different parameters. In the following, parameters *a*, *b* and *d* will be set as variables with initial values of (1, −3, −0.1, 7) and step size $h = 0.01$. Only parameters *a*, *b* and *d* will be changed, the rest will remain unchanged and the state of the system will be observed.

With the parameters $a\in [1, 7]$, let $b = 0.2$, $c = 5$, $d = 17$, $k = 3$, and initial values of (1, -3, −0.1, 7), the LEs and BD of the system are shown in **Figure 2**.

The LEs can clearly see the change of the system state when the parameter *a* changes, and the BD can be obtained from the multiplicative period bifurcation into chaos, and it is known after analysis that the BD corresponds exactly to the LEs. The system states are shown in **Table 1**, which gives a clear view of the range of parameters of the system in different states.

According to the data in **Table 1**, the cyclical and chaotic states of the system under the transformation of parameter *a* are shown in **Figure 3**.

Taking the parameter $b\in[0.01, 2]$, let $a = 2.6$, $c = 5$, $d = 17$, $k = 3$, and the initial values are (1, −3, −0.1, 7), the LEs and the

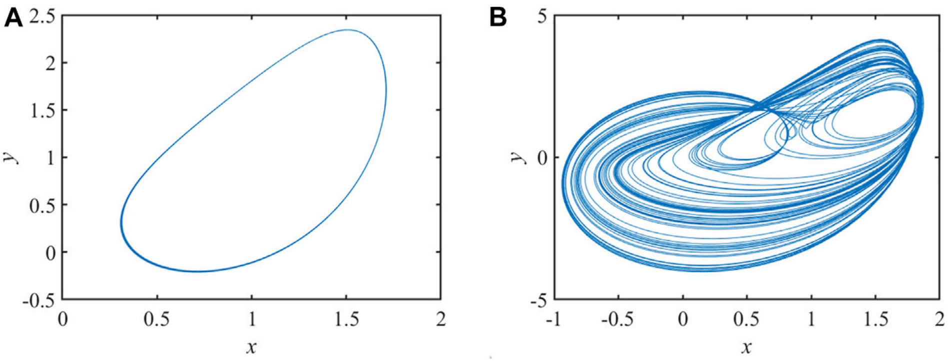


FIGURE 7 | Phase diagram of the system, **(A)** *d* = 7.0, **(B)** *d* = 19.0.

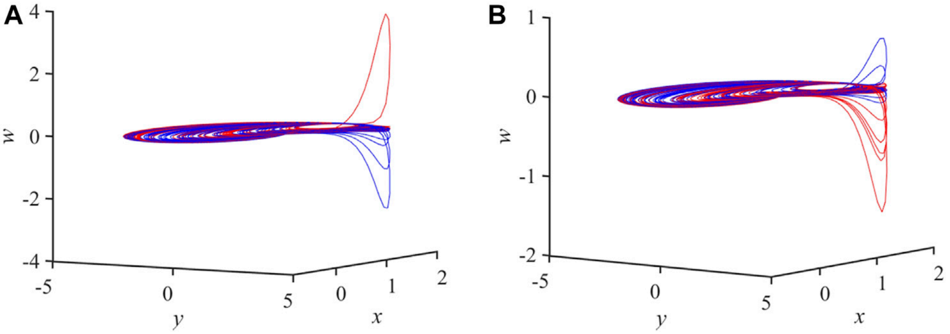


FIGURE 8 | The phase diagram of coexistence attractor, **(A)** initial values are (1, −3, −0.4, 7) and (−0.1, 0.4, −0.1,7), **(B)** initial values are (−2, 3.6, −0.1, 7) and (1, −3, −0.1, 7).

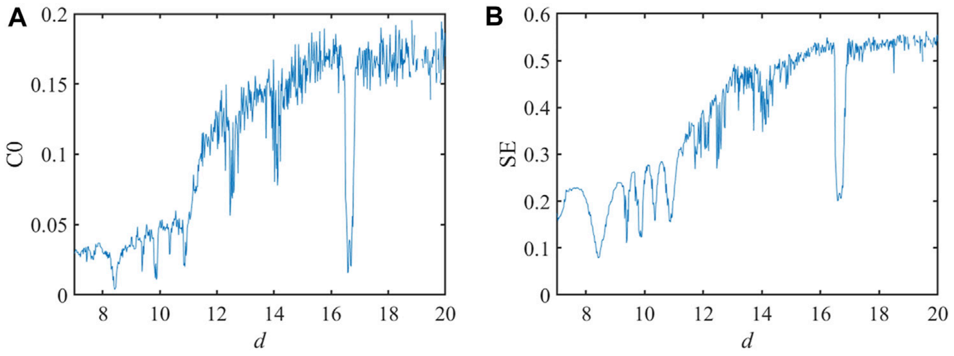


FIGURE 9 | Structural complexity curve with parameter *d*, **(A)** C0 Complexity, **(B)** SE Complexity.

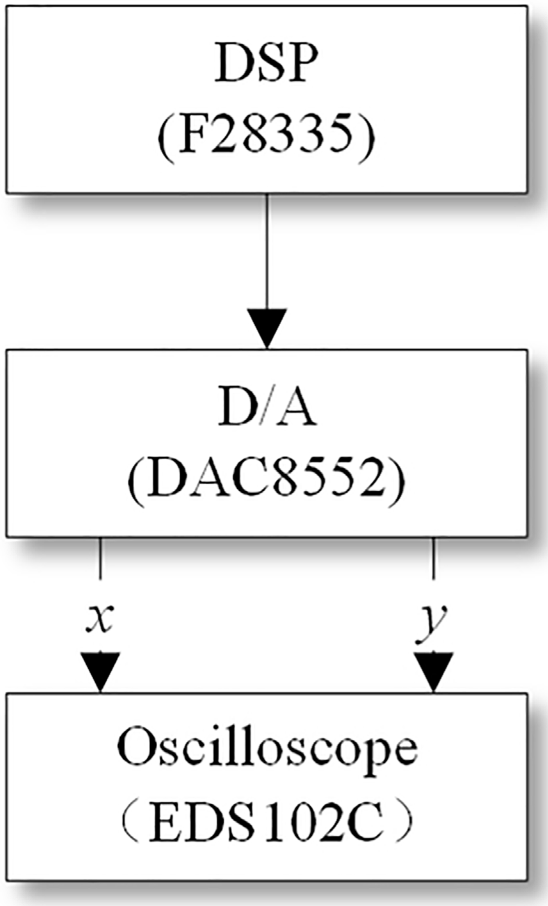


FIGURE 10 | The flow chart.

BD under this condition are displayed at **Figure 4**. As the parameter *b* varies, cyclical and chaotic regions are evident in the system.

The state table of the system as the parameters *b* change is shown in **Table 2**. It is evident that the system undergoes another period-doubling bifurcation after going through a period-doubling bifurcation into chaos. The analysis shows that the BD corresponds exactly to the LEs.

The state of the system with parameter *b* transformed is displayed at **Figure 5**.

Taking the parameter $d \in [7, 20]$, setting $a = 2.6$, $b = 0.2$, $c = 5$, $k = 3$, and initial values of (1, −3, −0.1, 7), LEs and BD for this system are displayed at **Figure 6**. The system exhibits complex dynamics of chaos and alternating periods when the parameter *d* is varied.

The state table of the system when the state varies with the parameter *d* is shown in **Table 3**. It can be seen that the chaotic range of the system is relatively large at this time, while there is a clear periodic window.

When the parameter *d* is varied, the system produces the periodic and chaotic state displayed at **Figure 7**.

3.3 Coexistence of Attractors

An interesting phenomenon that exists in chaotic systems, namely the coexistence of attractors. When the parameters of the system are constant and the initial values are varied, the trajectory of the system can gradually vary to a different state of motion. Setting $a = 2.6$, $b = 0.2$, $c = 5$, $d = 17$, $k = 3$. When the initial values are set to (1, −3, −0.4, 7) and (−0.1, 0.4, −0.1, 7), the coexistence of attractors of the system is displayed at **Figure 8A**, and when the initial values are set to (−2, 3.6, −0.1, 7) and (1, −3, −0.1, 7), the coexistence of attractors of the system is displayed at **Figure 8B**.

3.4 Complexity Analysis

Another important aspect of the study of chaotic systems is the study of their complexity. The degree of closeness of a chaotic sequence to a random sequence, as shown using the corresponding algorithm, is the complexity of a chaotic system. The higher the value of complexity, the closer the system is to a random sequence and the safer it is [56, 57]. In this paper, the SE algorithm and the C0 algorithm are used to analyse the complexity of the structure.

The SE (spectral entropy) algorithm performs a Fourier transform on the sequence and then obtains the spectral entropy value by combining the energy density in the frequency domain with the Shannon entropy. The C0

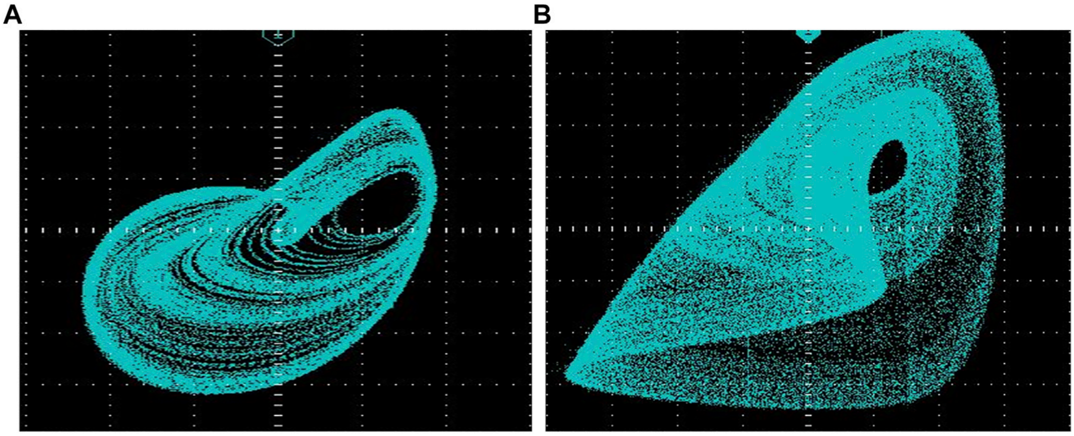


FIGURE 11 | The phase diagram of DSP simulation, **(A)** *x*–*y* plane **(B)** *y*–*z* plane.

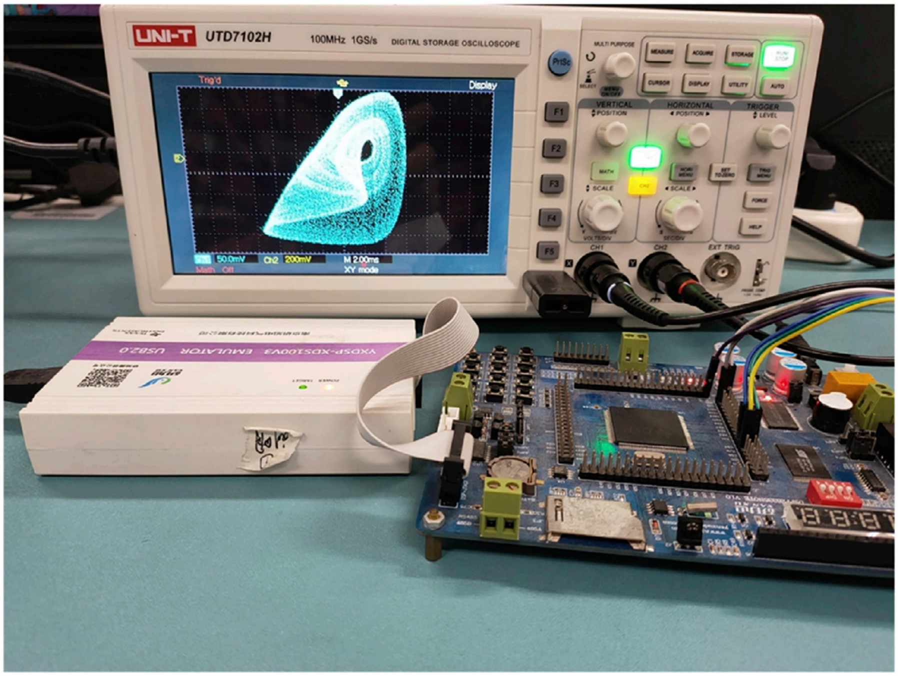


FIGURE 12 | DSP simulation physical picture.

algorithm, on the other hand, involves decomposing the sequence to produce two parts of regularity and irregularity, and then measuring the proportion of the irregular part to obtain the result.

In the section, we will analyse the complexity of the system under the variation of parameter d. When taking parameters a = 2.6, b = 0.2, c = 5, k = 3, and $d\in$ [7, 20], the results obtained are shown in **Figure 9**.

As can be seen from **Figure 9**, the SE algorithm and the C0 algorithm are highly synchronized, and the system is in the limit ring state when d < 11.30, in the chaotic state when $d\in$[11.30, 12.43], in the limit ring state when $d\in$[12.44, 12.60], in the chaotic state when $d\in$[12.61, 14.01], in the limit ring state when $d\in$[14.02, 14.17], in the chaotic state when $d\in$[14.18, 16.48], in the limit ring state when $d\in$[16.49, 16.82], and when d > 16.83, the system enters the chaotic state again. By analyzing the SE complexity diagram and the C0 complexity diagram, it can be found that the complexity is at a low point when the system is in the periodic state and increases significantly when the system enters the chaotic state, and the presented results are consistent with the LEs and the BD displayed at **Figure 6**.

4 DSP IMPLEMENTATION

Since chaotic systems are susceptible to external perturbations when implemented using analog circuits, the relevant characteristic conditions in the actual circuit are more difficult to control accurately, so they can be implemented using DSP. The DSP chip F28335, with its fast computing speed, high accuracy and low environmental impact, was chosen as the empirical platform to validate the novel system of chaos. As software can only receive digital signals, a continuous chaotic system must be discretized in order to handle the system on a DSP platform firstly. Therefore, we discretize the continuous system of chaos and transform it into a discrete chaotic sequence by the approach of fourth-order Runge-Kutta, and then write into the DSP chip using a programming language. Here a stack operation is devised to make sure the data is not corrupted as much as possible. On the output side of the DSP, a D/A converter (DAC8552) is used to convert the digital sequence into an analogue sequence which is then sent to an oscilloscope (EDS102C) so that the oscilloscope can better capture the image. The overall workflow is shown in **Figure 10**.

The parameters are given as $a = 2.6$, $b = 0.2$, $c = 5$, $d = 17$, $k = 3$, and the initial conditions as (1, −3, −0.1,7), the images are obtained as shown in **Figure 11**. By comparing with **Figure 1**, it is clear that the results obtained using DSP are consistent with the simulations carried out in MATLAB.

The f28335 chip, D/A converter and oscilloscope used for the system DSP simulation are shown in **Figure 12**.

5 CONCLUSION

The article proposes a new four-dimensional chaotic system and investigated its dynamic properties. In the study of this chaotic system, different chaotic attractors are found. By analysing the LEs, the BD and the complexity of this system, it can be understood that the system exhibits a dramatic degree of complexity in its dynamic properties as the parameters changed of the system. By numerical simulations, the peculiar phenomenon of the coexistence of chaotic attractors was observed by us. The results show that the chaotic system has very complex dynamic properties. Finally, the circuit was built and tested on a DSP platform, and a comparison of the test results with the numerical simulation results shows a high degree of consistency. The article provides a reference for the study of chaotic systems and circuit experiments, and has good prospects for applications in information encryption and secure communication.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

XW built the system model, analysed the dynamics, performed the circuit implementation and wrote the manuscript. YF and YC supervised the work and revised the manuscript. Both authors read and approved the final manuscript.

FUNDING

Dalian Young Stars of Science and Technology Project (NO. 2021RQ088).

REFERENCES

1. Lorenz EN. Deterministic Nonperiodic Flow. *J Atoms* (1963) 20. doi:10.1175/1520-0469(1963)020<0130:dnf>2.0.co;2
2. Gao X, Mou J, Xiong L, Sha Y, Yan H, Cao Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn* (2022) 108:613–36. doi:10.1007/s11071-021-07192-7
3. Feki M. An Adaptive Chaos Synchronization Scheme Applied to Secure Communication. *Chaos, Solitons & Fractals* (2003) 18(1):141–8. doi:10.1016/s0960-0779(02)00585-4
4. Zaher AA, Abu-Rezq A. On the Design of Chaos-Based Secure Communication Systems. *Commun Nonlinear Sci Numer Simulation* (2011) 16(9):3721–37. doi:10.1016/j.cnsns.2010.12.032
5. Luo C, Wang X. Chaos Generated from the Fractional-Order Complex Chen System and its Application to Digital Secure Communication. *Int J Mod Phys C* (2013) 24(04):1350025. doi:10.1142/s0129183113500253
6. Wang B, Zhang X, Dong X. Novel Secure Communication Based on Chaos Synchronization. *IEICE Trans Fundamentalsa* (2018) E101.A7:1132–5. doi:10.1587/transfun.e101.a.1132
7. Xiu C, Zhou R, Zhao S, Xu G. Memristive Hyperchaos Secure Communication Based on Sliding Mode Control. *Nonlinear Dyn* (2021) 4:1–17. doi:10.1007/s11071-021-06302-9
8. Yu F, Qian S, Chen X, Huang Y, Liu L, Shi C, et al. A New 4D Four-Wing Memristive Hyperchaotic System: Dynamical Analysis, Electronic Circuit Design, Shape Synchronization and Secure Communication. *Int J Bifurcation Chaos* (2020) 30(10). doi:10.1142/s0218127420501473
9. Kamal FM, Elsonbaty A, Elsaid A. A Novel Fractional Nonautonomous Chaotic Circuit Model and its Application to Image Encryption. *Chaos, Solitons & Fractals* (2021) 144(3):110686. doi:10.1016/j.chaos.2021.110686
10. Lin H, Wang C, Deng Q, Xu C, Deng Z, Zhou C. Review on Chaotic Dynamics of Memristive Neuron and Neural Network. *Nonlinear Dyn* (2021) 106(1):959–73. doi:10.1007/s11071-021-06853-x
11. Man Z, Li J, Di X, Sheng Y, Liu Z. Double Image Encryption Algorithm Based on Neural Network and Chaos. *Chaos, Solitons & Fractals* (2021) 152(24):111318. doi:10.1016/j.chaos.2021.111318
12. Li X, Mou J, Cao Y, Banerjee S. An Optical Image Encryption Algorithm Based on a Fractional-Order Laser Hyperchaotic System. *Int J Bifurcation Chaos* (2022) 32(03):2250035. doi:10.1142/s0218127422500353
13. Li CL, Zhou Y, Li HM, Feng W, Du JR. Image Encryption Scheme with Bit-Level Scrambling and Multiplication Diffusion. *Multimedia Tools Appl* (2021) 80(3):1–23. doi:10.1007/s11042-021-10631-7
14. Zhou Y, Li C, Li W, Li H, Feng W, Qian K. Image Encryption Algorithm with circle index Table Scrambling and Partition Diffusion. *Nonlinear Dyn* (2021) 103:2043–61. doi:10.1007/s11071-021-06206-8

15. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Du a. S. FPGA Implementation and Image Encryption Application of a New PRNG Based on a Memristive Hopfield Neural Network with a Special Activation Gradient. *Chin Phys B* (2022) 31(2):20505–020505. doi:10.1088/1674-1056/ac3cb2
16. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S. A New Multi-Scroll Chua's Circuit with Composite Hyperbolic tangent-cubic Nonlinearity: Complex Dynamics, Hardware Implementation and Image Encryption Application. *Integration* (2021) 81:71–83. doi:10.1016/j.vlsi.2021.05.011
17. Yu F, Kong X, Chen H, Yu Q, Cai S, Huang Y, et al. A 6D Fractional-Order Memristive Hopfield Neural Network and its Application in Image Encryption. *Front Phys* (2022) 10109. doi:10.3389/fphy.2022.847385
18. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Jin J, et al. Design and FPGA Implementation of a Pseudo-random Number Generator Based on a Hopfield Neural Network under Electromagnetic Radiation. *Front Phys* (2021) 9: 690651. doi:10.3389/fphy.2021.690651
19. Li Y, Li C, Zhang S, Chen GR, Zeng Z. A Self-Reproduction Hyperchaotic Map with Compound Lattice Dynamics. *IEEE Trans Ind Electron* (2022) 1. doi:10.1109/TIE.2022.3144592
20. Li Y, Li C, Zhao Y, Liu S. Memristor-type Chaotic Mapping. *Chaos* (2022) 32(2):021104. doi:10.1063/5.0082983
21. Li C, Yang Y, Yang X, Zi X, Xiao F. A Tristable Locally Active Memristor and its Application in Hopfield Neural Network. *Nonlinear Dyn* (2022) 108: 1697–717. doi:10.1007/s11071-022-07268-y
22. Gao X, Mou J, Banerjee S, Cao Y, Xiong L, Chen X. An Effective Multiple-Image Encryption Algorithm Based on 3D Cube and Hyperchaotic Map. *J King Saud Univ - Comput Inf Sci* (2022) 34(4):1535–51. doi:10.1016/j.jksuci.2022.01.017
23. Zhou S, Wang X, Zhang Y, Ge B, Wang M, Gao S. A Novel Image Encryption Cryptosystem Based on True Random Numbers and Chaotic Systems. *Multimedia Syst* (2022) 28(1):95–112. doi:10.1007/s00530-021-00803-8
24. Zhou S, Wang X, Wang M, Zhang Y. Simple Colour Image Cryptosystem with Very High Level of Security. *Chaos, Solitons & Fractals* (2020) 141:110225. doi:10.1016/j.chaos.2020.110225
25. Liu T, Banerjee S, Yan H, Mou J. Dynamical Analysis of the Improper Fractional-Order 2D-SCLMM and its DSP Implementation. *The Eur Phys J Plus* (2021) 136(5):1–17. doi:10.1140/epjp/s13360-021-01503-y
26. Ma C, Mou J, Li P, Liu T. Dynamic Analysis of a New Two-Dimensional Map in Three Forms: Integer-Order, Fractional-Order and Improper Fractional-Order. *Eur Phys J Spec Top* (2021) 230(7):1945–57. doi:10.1140/epjs/s11734-021-00133-w
27. Xiong P-Y, Jahanshahi H, Alcaraz R, Chu Y-M, Gómez-Aguilar JF, Alsaadi FE. Spectral Entropy Analysis and Synchronization of a Multi-Stable Fractional-Order Chaotic System Using a Novel Neural Network-Based Chattering-free Sliding Mode Technique. *Chaos, Solitons & Fractals* (2021) 144:110576. doi:10.1016/j.chaos.2020.110576
28. Liu T, Yan H, Banerjee S, Mou J. A Fractional-Order Chaotic System with Hidden Attractor and Self-Excited Attractor and its DSP Implementation. *Chaos, Solitons & Fractals* (2021) 145(2):110791. doi:10.1016/j.chaos.2021.110791
29. Akgül A, Rajagopal K, Durdu A, Pala MA, Boyraz ÖF, Yildiz MZ. A Simple Fractional-Order Chaotic System Based on Memristor and Memcapacitor and its Synchronization Application. *Chaos, Solitons & Fractals* (2021) 152(3): 111306. doi:10.1016/j.chaos.2021.111306
30. Xie W, Wang C, Lin H. A Fractional-Order Multistable Locally Active Memristor and its Chaotic System with Transient Transition, State Jump. *Nonlinear Dyn* (2021) 104(4):4523–41. doi:10.1007/s11071-021-06476-2
31. Jahanshahi H, Sajjadi SS, Bekiros S, Aly AA. On the Development of Variable-Order Fractional Hyperchaotic Economic System with a Nonlinear Model Predictive Controller. *Chaos, Solitons & Fractals* (2021) 144:110698. doi:10.1016/j.chaos.2021.110698
32. Mathale D, Doungmo Goufo EF, Khumalo M. Coexistence of Multi-Scroll Chaotic Attractors for a Three-Dimensional Quadratic Autonomous Fractional System with Non-local and Non-singular Kernel. *Alexandria Eng J* (2021) 60(4):3521–38. doi:10.1016/j.aej.2021.02.024
33. Yadav VK, Shukla VK, Das S. Exponential Synchronization of Fractional-Order Complex Chaotic Systems and its Application. *Chaos, Solitons & Fractals* (2021) 147(8):110937. doi:10.1016/j.chaos.2021.110937
34. Yan M, Xu H. The Multi-Scroll Hyper-Chaotic Coexistence Attractors and its Application. *Signal Processing: Image Commun* (2021) 95:116210. doi:10.1016/j.image.2021.116210
35. Sv A, As B, Atac D, Ss E. A New Multistable Plasma Torch Chaotic Jerk System, its Dynamical Analysis, Active Backstepping Control, and Circuit Design. *Backstepping Control Nonlinear Dynamical Syst* (2021) 191–214. doi:10.1016/B978-0-12-817582-8.00016-7
36. Singh S, Mathpal S, Azar AT, Vaidyanathan S, Kamal NA. Multi-switching Synchronization of Nonlinear Hyperchaotic Systems via Backstepping Control. *Backstepping Control Nonlinear Dynamical Syst* (2021) 425–47. doi:10.1016/b978-0-12-817582-8.00024-6
37. Yang J, Feng Z, Liu Z. A New Five-Dimensional Hyperchaotic System with Six Coexisting Attractors. *Qual Theor Dynamical Syst* (2021) 20(1). doi:10.1007/s12346-021-00454-0
38. Llibre J, Tian Y. The Zero-Hopf Bifurcations of a Four-Dimensional Hyperchaotic System. *J Math Phys* (2021) 62(5):052703. doi:10.1063/5.0023155
39. Li C, Li H, Xie W, Du J. A S-type Bistable Locally Active Memristor Model and its Analog Implementation in an Oscillator Circuit. *Nonlinear Dyn* (2021) 106: 1041–58. doi:10.1007/s11071-021-06814-4
40. Zhang X, Li C, Dong E, Zhao Y, Liu Z. A Conservative Memristive System with Amplitude Control and Offset Boosting. *Int J Bifurcation Chaos* (2022) 32(04): 2250057. doi:10.1142/s0218127422500572
41. Šil'nikov LP, Sil'nikov LP. A Contribution of the Problem of the Structure of an Extended Neighborhood of Rough Equilibrium State of Saddle-Focus Type. *Math USSR Sbornikmathematics Ussr-sbornik* (1970) 1010(1):9191–102. doi:10.1070/SM1970v010n01ABEH001588
42. Leonov GA, Kuznetsov NV. IWCFTA2012 Keynote Speech I - Hidden Attractors in Dynamical Systems: From Hidden Oscillation in Hilbert-Kolmogorov, Aizerman and Kalman Problems to Hidden Chaotic Attractor in Chua Circuits. *Int J Bifurcation Chaos* (2012) 23(01). doi:10.1109/iwcfta.2012.8
43. Wang N, Zhang G, Kuznetsov NV, Bao H. Hidden Attractors and Multistability in a Modified Chua's Circuit. *Commun Nonlinear Sci Numer Simulation* (2021) 92:105494. doi:10.1016/j.cnsns.2020.105494
44. Kuznetsov NV, Vagaytsev VI, Leonov GA, Seledzhi SM, Localization of Hidden Attractors in Smooth Chua's Systems (2011). 26–33. doi:10.5555/2001305.2001309
45. Ueta T, Chen G. Yet Another Chaotic Attractor. *Int J Bifurcation Chaos* (1999) 9(7):1465–6. doi:10.1142/S0218127499001024
46. Lu J, Chen G. A New Chaotic Attractor Coined. *Int J Bifurcation Chaos* (2002) 12(3):659–61. doi:10.1142/s0218127402004620
47. Vagaitsev VI, Kuznetsov NV, Leonov GA. Localization of Hidden Attractors of the Generalized Chua System Based on the Method of Harmonic Balance. *Vestnik St.Petersb Univ.Math* (2010) 43(4):242–55. doi:10.3103/s1063454110040096
48. Leonov GA, Kuznetsov NV, Vagaitsev VI. Hidden Attractor in Smooth Chua System. *Physica D Nonlinear Phenomena* (2012) 241(18): 1482–6. doi:10.1016/j.physd.2012.05.016
49. Lao S-K, Shekofteh Y, Jafari S, Sprott JC. Cost Function Based on Gaussian Mixture Model for Parameter Estimation of a Chaotic Circuit with a Hidden Attractor. *Int J Bifurcation Chaos* (2014) 24(01):1450010. doi:10.1142/s0218127414500102
50. Chen M, Li M, Yu Q, Bao B, Xu Q, Wang J. Dynamics of Self-Excited Attractors and Hidden Attractors in Generalized Memristor-Based Chua's Circuit. *Nonlinear Dyn* (2015) 81(1-2):215–26. doi:10.1007/s11071-015-1983-7
51. Borah M, Roy BK. *Hidden Attractor Dynamics of a Novel Non-equilibrium Fractional-Order Chaotic System and its Synchronisation Control*. Guwahati: IEEE (2017). p. 450–5.
52. Cang S, Yue L, Zhang R, Wang Z. Hidden and Self-Excited Coexisting Attractors in a Lorenz-like System with Two Equilibrium Points. *Nonlinear Dyn* (2019) 95(2). doi:10.1007/s11071-018-4570-x

53. Zhang X, Wang C. Multiscroll Hyperchaotic System with Hidden Attractors and its Circuit Implementation. *Int J Bifurcation Chaos* (2019) 29(09):1157–71. doi:10.1142/s0218127419501177
54. Deng Q, Wang C, Yang L. Four-Wing Hidden Attractors with One Stable Equilibrium Point. *Int J Bifurcation Chaos* (2020) 30(6):2050086. doi:10.1142/s0218127420500868
55. Ma X, Mou J, Liu J, Ma C, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor–Memcapacitor. *Nonlinear Dyn* (2020) 100(5):2859–76. doi:10.1007/s11071-020-05601-x
56. Ma X, Mou J, Xiong L, Banerjee S, Cao Y, Wang J. A Novel Chaotic Circuit with Coexistence of Multiple Attractors and State Transition Based on Two Memristors. *Chaos, Solitons & Fractals* (2021) 152:111363. doi:10.1016/j.chaos.2021.111363
57. Ma C, Mou J, Xiong L, Banerjee S, Han X. Dynamical Analysis of a New Chaotic System: Asymmetric Multistability, Offset Boosting Control and Circuit Realization. *Nonlinear Dyn* (2021) 103(6):1–14. doi:10.1007/s11071-021-06276-8

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Physics

ORIGINAL RESEARCH
published: 02 May 2022
doi: 10.3389/fphy.2022.911144

Dynamical Analysis of Two-Dimensional Memristor Cosine Map

Xintong Han[1], *Xiuguo Bi*[1]*, *Bo Sun*[1]*, *Lujie Ren*[1] *and Li Xiong*[2]

[1]*School of Science, Dalian Polytechnic University, Dalia, China,* [2]*School of Physics and Electromechanical Engineering, Hexi University, Zhangye, China*

Research on discrete memristor models applied to discrete maps deserves more in-depth discussion. In this paper, a continuous memristor is introduced and the discrete memristor model is obtained by the forward Eulerian difference algorithmic discretization. This model is coupled to a cosine map to further obtain a two-dimensional memristor cosine map. The dynamical characteristics of the memristor cosine map are investigated through numerical simulations and other analytical methods. For example, the phase diagram, the bifurcation diagram, the Lyapunov exponential spectrum and the Spectral Entropy complexity with parameters, etc., In addition, multi-stability phenomena of the system are identified. The results show that the cosine map coupled with a discrete memristor has more complex dynamical behaviors and is more suitable for applications in cryptography.

Keywords: discrete memristor, cosine map, multi-stability, memristor cosine map, SE complexity

Edited by:
Shaobo He,
Central South University, China

Reviewed by:
Fei Yu,
Changsha University of Science and Technology, China
Xiaolin Ye,
Anshan Normal University, China

****Correspondence:***
Xiuguo Bi
bixg@dlpu.edu.cn
Bo Sun
sunbo_0709@126.com

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *02 April 2022*
Accepted: *19 April 2022*
Published: *02 May 2022*

Citation:
Han X, Bi X, Sun B, Ren L and Xiong L (2022) Dynamical Analysis of Two-Dimensional Memristor Cosine Map. Front. Phys. 10:911144. doi: 10.3389/fphy.2022.911144

INTRODUCTION

In 1963, the meteorologist Lorenz proposed the Lorenz model and the "butterfly effect" [1–3]. Since then, chaos theory has gradually entered the eyes of scholars. With a series of significant research results in the history of chaos, the study of chaos theory and chaos properties has become a major research hotspot [4–8]. In recent years, chaos science has interpenetrated with other disciplines, and breakthroughs have been made in chaotic synchronous control [9, 10], chaotic secure communication [11–14] and chaotic neural networks [15–18]. This has led to a wide range of applications of chaos theory in both natural and social sciences. Although some progress has been made in the study of chaos theory [19], its complexity is still not understood and how to chaoticize it is still a very interesting issue for scholars to explore further. Chaosification is the process of generating or enhancing the chaotic nature of a system. Chaosification can increase the complexity and unpredictability of a system, and making it more suitable for areas such as secure communications.

In 1971, Chua introduced the concept of memristors [20–22] and its associated theory. Unlike resistor, inductor and capacitor, memristor is the fourth fundamental element describing the relationship between charge and flux [23, 24]. The first report of the realisability of memristors from Hewlett-Packard (HP) Labs in 2008 [25]. The result quickly gained attention and caused a surge of research on memristors by scholars in various fields [26–28]. Compared to the other three fundamental components, the memristor has a very interesting property in that it is a non-linear component with a memory function [29]. It is because of the memory property of memristors that the research related to memristors reflects the cross-fertilisation of several disciplines. Memory properties have had a great impact on computer science, biology [30], neural networks [31] and communication engineering [32, 33]. The application of memristors to the construction of non-linear circuits and systems produces complex dynamical behaviors, and therefore memristors are widely used in the field

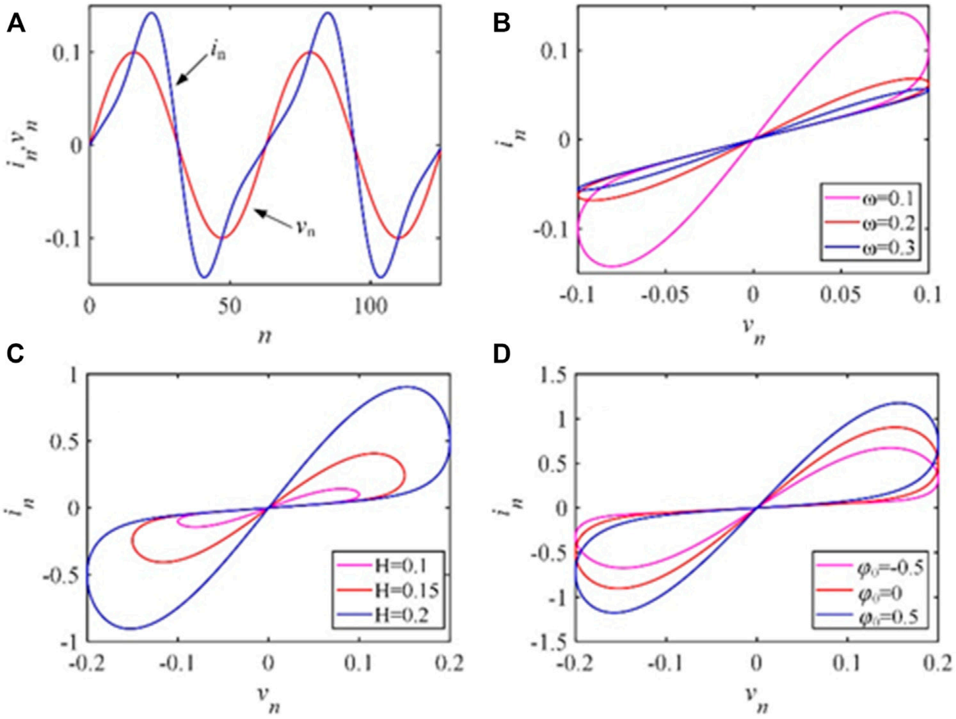


FIGURE 1 | Characteristics of the discrete memristor when being applied to a discrete sinusoidal current $i_n = H\sin(\omega n)$, where α = 0.5 and β = 0.5. **(A)** Current and voltage sequences; **(B)** H = 0.1, φ0 = 0; **(C)** ω = 0.1, φ0 = 0; **(D)** H = 0.1, ω = 0.1.

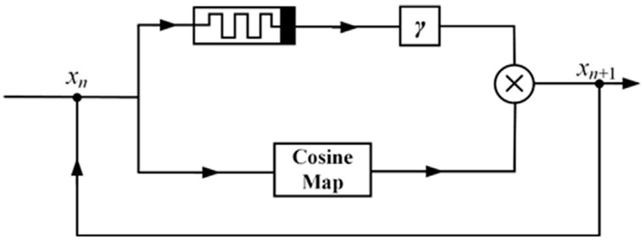


FIGURE 2 | Structural diagram of the cosine map coupled by the discrete memristor.

of chaos [34, 35]. Currently, most of the research on chaotic systems based on memristors is focused on continuous chaotic systems. For example, [36] et al. proposed a novel simple chaotic circuit, which includes a memristor, a linear inductor and a memcapacitor. [37] constructed a deoxyribonucleic acid (DNA) encoding algorithm with memristor based chaotic system. [38] et al. reported a 3-D conservative memristive system with amplitude control. However, discrete chaotic systems have the advantages of simple model building and fast iteration [39–41]. Is it possible if memristor is applied to discrete chaotic systems? Some scholars have already studied the discretization of memristors to obtain discrete memristor models [42–45]. For example, [46] et al. discussed a second-order map model based on memristor. And to enhance the chaotic performance of the 1-D maps, [47] et al. proposed a chaotification method. These research results have laid a solid foundation for the application of memristors in discrete chaotic systems. There are still great prospects for introducing discrete memristor models into discrete chaotic maps. In this paper, a new memristor cosine map is proposed based on a one-dimensional cosine map coupled with a magnetically controlled memristor model. It is shown that the memristor cosine map has richer dynamical behaviors.

The rest of the paper is structured as follows. The discretized memristor model and the memristor cosine map model based on a one-dimensional cosine map are presented in the “System Model” section. This is followed by the “Dynamics Analysis” section, which describes the dynamical behaviors of the memristor cosine map as the parameters and initial values are varied. Finally, we summarized the conclusions drawn from the research in this paper and give directions for future research.

SYSTEM MODEL

Model of Discrete Memristor

In this section, a triple magnetron memristor has been chosen, which can be expressed as

$$\begin{cases} i(t) = W(\varphi)v(t) \\ \dfrac{d\varphi(t)}{dt} = v(t), \\ W(\varphi) = \alpha + \beta\varphi^2 \end{cases} \tag{1}$$

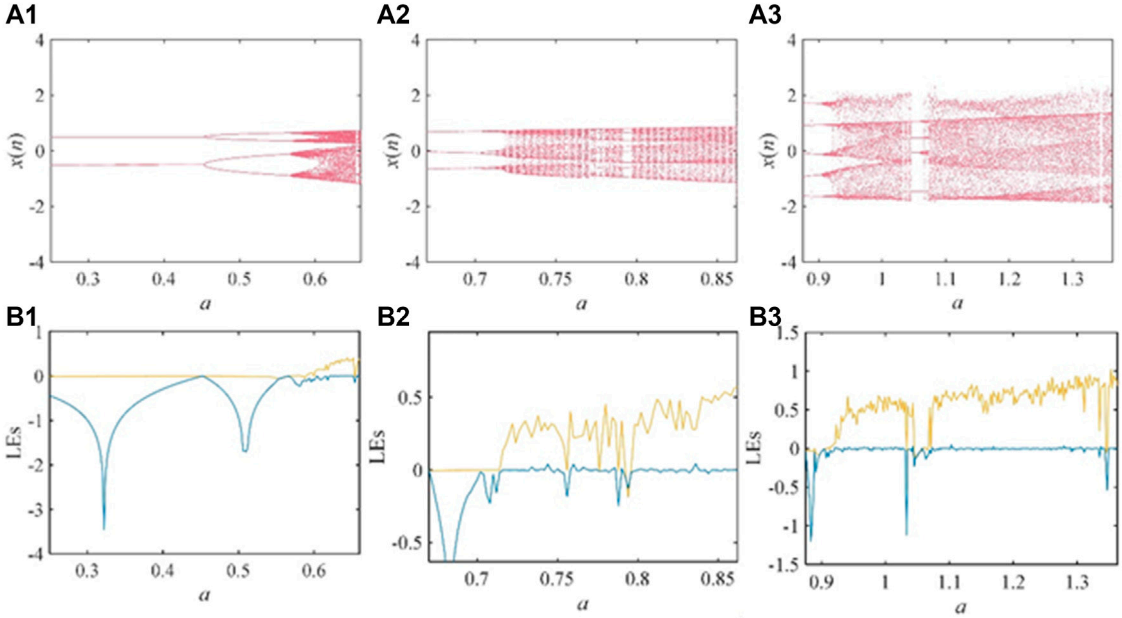


FIGURE 3 | Bifurcation and LEs of the map with parameter a. **(A1)** $a \in (0.25, 0.66)$; **(A2)** $a \in (0.67, 0.862)$; **(A3)** $a \in (0.875, 1.363)$; **(B1)** $a \in (0.25, 0.66)$; **(B2)** $a \in (0.67, 0.862)$; **(B3)** $a \in (0.875, 1.363)$.

here, $i(t)$ and $v(t)$ are the input current and the output voltage of the memristor, respectively. $W(\varphi)$ is a non-linear function about $\varphi(t)$ with the same magnitude as the conductance value, called the amnesic conductance, indicating dependence on the control of the internal variable $\varphi(t)$ of the memristor. And φ is the magnetic flux and represents the amount of internal magnetic flux change in the memristor. Then, α and β are the control parameters of the memristor.

First, discrete $i(t)$, $v(t)$, $\varphi(t)$ by sampling at unit time n to obtain i_n, v_n and φ_n respectively. Then, one can discretize the memristor model described in **Eq. 1** by using the forward Eulerian difference algorithm, the discrete memristor model can be obtained as

$$\begin{cases} i_n = (\alpha + \beta\varphi_n^2)v_n \\ \varphi_{n+1} = \varphi_n + kv_n. \end{cases} \tag{2}$$

Assuming that the sinusoidal current $i_n = H\sin(\omega n)$ is added to the model of the discrete memristor, the characteristics of the model are analyzed. Let $\alpha = 0.5$, $\beta = 0.5$, and its characteristic curve can be obtained in **Figure 1A**. In order to further analyze this discrete memristor model, the voltammetric characteristic curves of the amnestic resistor obtained by taking different ω, H, and φ_0 respectively, are shown in **Figures 1B–D**.

Memristor-Coupled Map

A cosine map with a simple structure and few parameter variables was chosen and it was defined as

$$x_{i+1} = a\cos(\pi x_i). \tag{3}$$

Based on the chaotic control method of modulation and coupling, coupling the discrete memristor model (2) with the cosine map (3), a new two-dimensional chaotic map model can be obtained. **Figure 2** is a schematic diagram of the two-dimensional map coupled by the discrete memristor, where γ is the coupling parameter. The mathematical expression is

$$\begin{cases} x_{n+1} = a\cos(\pi x_n) + \gamma x_n(\alpha + \beta y_n^2) \\ y_{n+1} = y_n + ex_n, \end{cases} \tag{4}$$

where, a is the control parameter of the cosine map (3), α and β are the control parameters of the discrete memristor model (2), and e is the step length.

DYNAMICAL ANALYSES

Bifurcation and Lyapunov Exponent Analysis

In order to analyze the dynamical characteristics of this discrete chaotic map, the bifurcation diagram and the Lyapunov exponential spectrum of the map with respect to the parameter a are plotted as shown in **Figure 3**. Set the other parameters $\gamma = -0.4$, $e = 0.02$, $\alpha = 0.5$ and $\beta = 0.5$, the initial value is (1, 1), the step size is 0.001. The discrete map has a number of divergent points as the parameter a varies in the range 0.25–1.363. Therefore, the range of parameter a is divided into three segments for analysis. When $a \in (0.25, 0.66)$, the bifurcation diagram and Lyapunov exponent spectrum of the discrete system are shown in **Figures 3A1,B1**. As can be seen in the figure, the system has a clear multiplicative period bifurcation behavior. The map goes from two cycles to four cycles and then into a chaotic state. The bifurcation and LEs with the parameter a when $a \in$

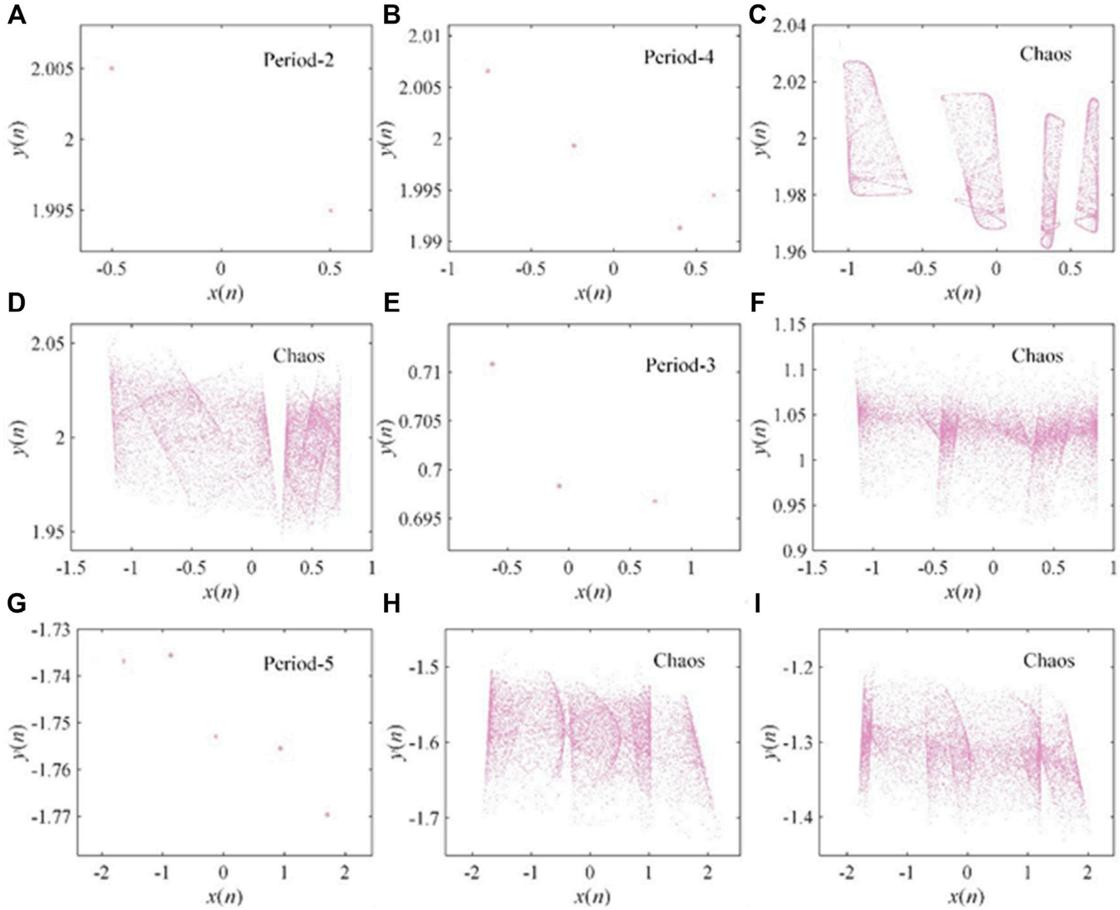


FIGURE 4 | The phase space motion trajectory of this discrete system. **(A)** $a = 0.3$; **(B)** $a = 0.5$; **(C)** $a = 0.6$; **(D)** $a = 0.65$; **(E)** $a = 0.7$; **(F)** $a = 0.85$; **(G)** $a = 0.9$; **(H)** $a = 1$; **(I)** $a = 1.2$.

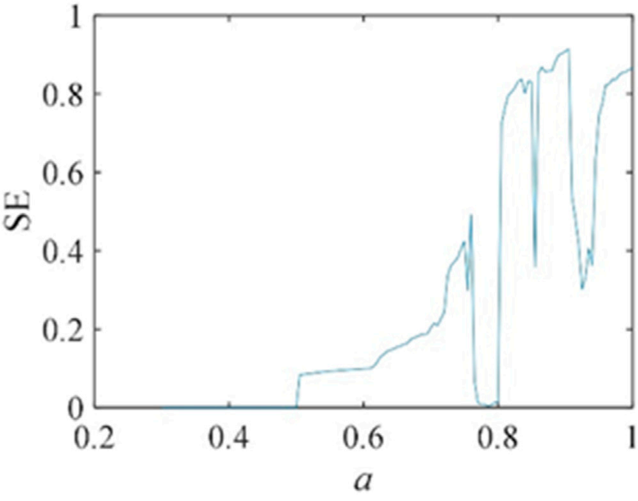


FIGURE 5 | SE complexity diagrams of the map when the parameter a varies.

(0.67, 0.862) are plotted in **Figures 3A2,B2**. The simulation results show that the two correspond well. When the parameter $a \in (0.875, 1.363)$, the bifurcation diagram and the Lyapunov exponent diagram of the chaotic map are drawn in **Figures 3A3,B3**. In addition, the trajectory of this map when taking different values of a is plotted as shown in **Figure 4**.

Spectral Entropy Complexity

The complexity of chaotic systems is a way to characterize the dynamical properties of chaotic systems. There are many algorithms to calculate the complexity value of chaotic systems, and where the complexity algorithm of the Spectral Entropy (SE) is used [48]. Its algorithm is described as follows.

First, for a pseudo-random sequence$\{x(m), m = 0, 1, 2, \ldots, M-1\}$ of length M, the DC part is removed through the following equation.

$$x(m) = x(m) - \bar{x}, \tag{5}$$

$$\bar{x} = \frac{1}{M}\sum_{m=0}^{M-1} x(m). \tag{6}$$

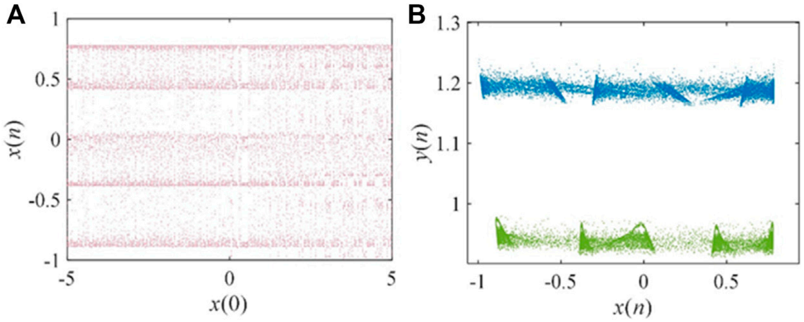


FIGURE 6 | Experimental results graph when *a* = 0.77. **(A)** Bifurcation with initial value *x* (0); **(B)** Attractors coexistence phase diagram.

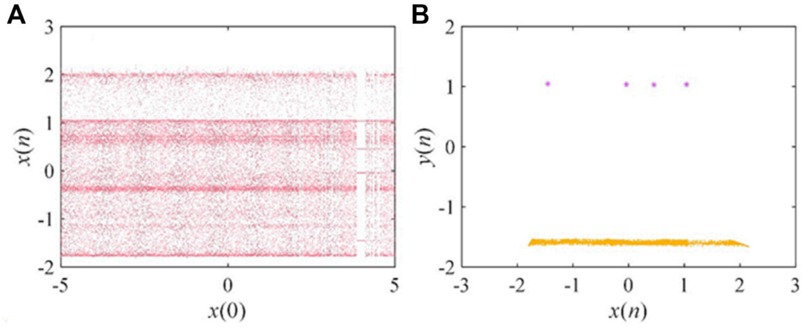


FIGURE 7 | Experimental results graph when *a* = 1.03. **(A)** Bifurcation with initial value *x* (0); **(B)** Attractors coexistence phase diagram.

Second, performing the discrete Fourier transform on the sequence $x(m)$,

$$X(k) = \sum_{m=0}^{M-1} x(m) e^{-j\frac{2\pi}{M}mk} = \sum_{m=0}^{M-1} x(m) W_M^{mk}, \tag{7}$$

Where, $k = 0, 1, 2, \ldots$, M-1.

Third, for the transformed sequence $X(k)$, take the first half for calculation. Based on the Paserval theorem, calculate the power spectrum value of one frequency point as

$$p(k) = \frac{1}{N} |X(k)|^2, \tag{8}$$

Among them, $k = 0, 1, 2, \ldots$, M/2-1, the overall power of the sequence is

$$p_{tot} = \frac{1}{M} \sum_{K=0}^{M/2-1} |X(K)|^2. \tag{9}$$

The relative power spectrum probability of the sequence is

$$P_k = \frac{p(k)}{p_{tot}} = \frac{\frac{1}{M}|X(k)|^2}{\frac{1}{M}\sum_{k=0}^{M/2-1}|X(k)|^2} = \frac{|X(k)|^2}{\sum_{k=0}^{M/2-1}|X(k)|^2}, \tag{10}$$

Fourth, calculate the spectral entropy as follows

$$se = -\sum_{k=0}^{M/2-1} p_k \ln P_k. \tag{11}$$

Finally, to facilitate the consideration, the spectral entropy is normalized

$$SE(N) = \frac{se}{\ln(M/2)}. \tag{12}$$

According to the above algorithm, the complexity curve of this discrete map with the parameter a is plotted in **Figure 5**. The vertical coordinates in the figure represent the SE complexity values corresponding to different values of a. A larger complexity value means that the sequence is closer to a random sequence and the system is more secure. Set $\gamma = -0.4$, $e = 0.02$, $\alpha = 0.5$ and $\beta = 0.5$, and the initial value remains unchanged. When $a \in (0.2, 0.5) \cup (0.76, 0.8)$, the complexity value of the discrete system is low and the system is in a periodic state. When $a \in (0.8, 0.85) \cup (0.86, 0.9) \cup (0.95, 1)$, the complexity value of the discrete system is high and the system is in a chaotic state.

Coexisting Attractors

When the system parameters are fixed, changing the initial value gives two or more different attractors coexisting. This interesting phenomenon can also be called multi-stability. In general, multi-stable systems are more sensitive to initial values and are more secure in their application to cryptography and secure communications. In this section, the phenomenon of attractors coexistence can be observed in the course of the experiment. When a = 0.77, $\gamma = -0.4$, $e = 0.01$, $\alpha = 0.5$ and $\beta = 0.5$, the bifurcation diagram of this

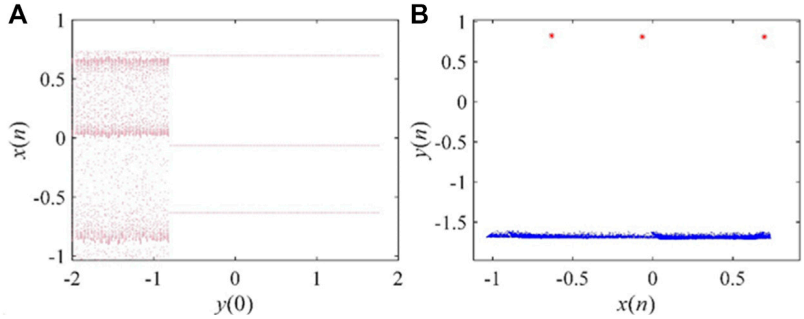


FIGURE 8 | Experimental results graph when a = 0.69. **(A)** Bifurcation with initial value y (0); **(B)** Attractors coexistence phase diagram.

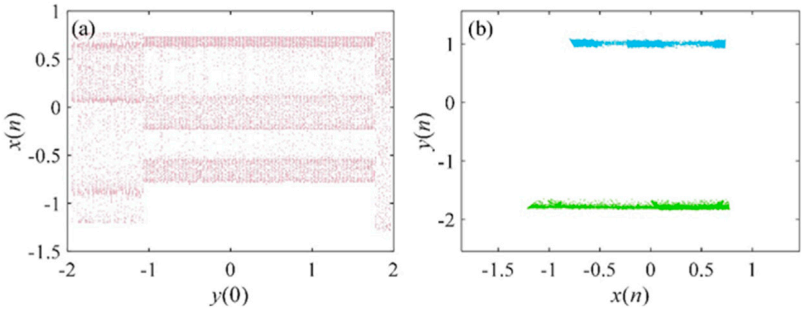


FIGURE 9 | Experimental results graph when a = 0.72. **(A)** Bifurcation with initial value y (0); **(B)** Attractors coexistence phase diagram.

map with the initial value x (0) is shown in **Figure 6A**. From the diagram, there are two different attractors for different initial values, and the phase diagram for the coexistence of the two attractors is drawn as shown in **Figure 6B**. The bifurcation diagram with the initial value x (0) is plotted in **Figure 7A** when a = 1.03, and other parameters remain unchanged. The coexistence of period four attractor and chaotic attractor can be observed in the bifurcation diagram. The plotted attractor coexistence phase diagram is shown in **Figure 7B**. In addition, let x (0) = 1, the bifurcation diagrams with the initial value y (0) are analyzed and the simulation results are shown in **Figures 8A, 9A**. Based on the two bifurcation diagrams, we take two sets of initial values (1, −1.5) and (1, 1). Then, the coexistence phase diagrams of the chaotic attractor with the period-3 attractor and the coexistence phase diagrams of two different chaotic attractors are plotted as shown in **Figures 8B**, **9B**.

CONCLUSION

In this paper, a continuous magnetron memristor was chosen and the discrete memristor model was obtained by means of the forward Eulerian difference algorithm. The discrete memristor was coupled with a one-dimensional cosine map to obtain a new two-dimensional chaotic map. The dynamical characteristics of the discrete system were analyzed by computer simulations. The results of the study show that the newly constructed discrete system has complex dynamical behaviors, higher complexity, and a higher degree of initial value sensitivity. These findings suggest that the chaoticisation method of coupling discrete memristor with discrete maps is worthy of further research, and that the map after the introduction of the discrete memristor model is more suitable for applications in areas such as secure communications. In the future, other ways of implementing this discrete system will be considered and attempts will be made to explore its practical applications.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding authors.

AUTHOR CONTRIBUTIONS

XH designed and carried out experiments, data analyzed and manuscript wrote. XB and LX made the theoretical guidance for this paper. LR carried out experiment. All authors reviewed the manuscript.

FUNDING

This work was supported by the National Natural Science Foundation of China (Grant No. 62061014).

REFERENCES

1. Adams TM, Stewart LD. Chaos Theory and Organizational Crisis: A Theoretical Analysis of the Challenges Faced by the New Orleans Police Department During Hurricane Katrina. *Public Organiz Rev* (2015) 15(3): 415–31. doi:10.1007/s11115-014-0284-9
2. Dong C, Sun K, He S, Wang H. A Hyperchaotic Cycloid Map with Attractor Topology Sensitive to System Parameters. *Chaos* (2021) 31(8):083132. doi:10.1063/5.0061519
3. Gleick J, Glazier J, Gunaratne G. Chaos: Making a New Science. *Phys Today* (1988) 41(11):79. doi:10.1063/1.2811320
4. Gu S, He S, Wang H, Du B. Analysis of Three Types of Initial Offset-Boosting Behavior for a New Fractional-Order Dynamical System. *Chaos, Solit Fractals* (2021) 143:110613. doi:10.1016/j.chaos.2020.110613
5. Han X, Mou J, Liu T, Cao Y. A New Fractional-Order 2D Discrete Chaotic Map and its DSP Implement. *Eur. Phys. J. Spec. Top.* (2021) 230(21):3913–25. doi:10.1140/epjs/s11734-021-00331-6
6. Kyriazis M. Practical Applications of Chaos Theory to the Modulation of Human Ageing: Nature Prefers Chaos to Regularity. *Biogerontology* (2003) 4(2):75–90. doi:10.1023/A:1023306419861
7. Li X, Mou J, Cao Y, Banerjee S. An Optical Image Encryption Algorithm Based on a Fractional-Order Laser Hyperchaotic System. *Int. J. Bifurc Chaos* (2022) 32:32. doi:10.1142/S0218127422500353
8. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S. A New Multi-Scroll Chua's Circuit with Composite Hyperbolic Tangent-Cubic Nonlinearity: Complex Dynamics, Hardware Implementation and Image Encryption Application. *Integration* (2021) 81:71–83. doi:10.1016/j.vlsi.2021.05.011
9. Karthikeyan A, Rajagopal K. Chaos Control in Fractional Order Smart Grid with Adaptive Sliding Mode Control and Genetically Optimized PID Control and its FPGA Implementation. *Complexity* (2017) 2017:1–18. doi:10.1155/2017/3815146
10. Vaseghi B, Pourmina MA, Mobayen S. Secure Communication in Wireless Sensor Networks Based on Chaos Synchronization Using Adaptive Sliding Mode Control. *Nonlinear Dyn*(2017) 89(9):1689–704. doi:10.1007/s11071-017-3543-9
11. Gao X, Mou J, Xiong L, Sha Y, Yan H, Cao Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn*(2022) 108(1):613–36. doi:10.1007/s11071-021-07192-7
12. He S, Sun K, Wu X. Fractional Symbolic Network Entropy Analysis for the Fractional-Order Chaotic Systems. *Phys. Scr.* (2020) 95(313pp):035220. doi:10.1088/1402-4896/ab46c9
13. Li C-L, Zhou Y, Li H-M, Feng W, Du J-R. Image Encryption Scheme with Bit-Level Scrambling and Multiplication Diffusion. *Multimed Tools Appl* (2021) 80(12):18479–501. doi:10.1007/s11042-021-10631-7
14. Zhou S, Wang X, Wang M, Zhang Y. Simple Colour Image Cryptosystem with Very High Level of Security. *Chaos, Solit Fractals* (2020) 141:110225. doi:10.1016/j.chaos.2020.110225
15. Çelik V, Demir Y. Chaotic Fractional Order Delayed Cellular Neural Network. *New Trends in Nanotechnology and Fractional Calculus Applications* (2020): 313–20. doi:10.1007/978-90-481-3293-5_27
16. Rakkiyappan R, Preethi Latha V, Zhu Q, Yao Z. Exponential Synchronization of Markovian Jumping Chaotic Neural Networks with Sampled-Data and Saturating Actuators. *Nonlinear Anal Hybrid Syst* (2017) 24:28–44. doi:10.1016/j.nahs.2016.10.004
17. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Du S. FPGA Implementation and Image Encryption Application of a New PRNG Based on a Memristive Hopfield Neural Network with a Special Activation Gradient. *Chin Phys. B* (2022) 31:020505. doi:10.1088/1674-1056/ac3cb2
18. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Jin J, et al. Design and FPGA Implementation of a Pseudo-random Number Generator Based on a Hopfield Neural Network Under Electromagnetic Radiation. *Front. Phys.* (2021) 9: 690651. doi:10.3389/fphy.2021.690651
19. He S, Sun K, Mei X, Yan B, Xu S. Numerical Analysis of a Fractional-Order Chaotic System Based on Conformable Fractional-Order Derivative. *Eur. Phys. J. Plus* (2017) 132(1):36. doi:10.1140/epjp/i2017-11306-3
20. Chua L. If It's Pinched It's a Memristor. *Semicond. Sci. Technol.* (2014) 29(10): 104001. doi:10.1088/0268-1242/29/10/104001
21. Bao H, Chen M, Wu H, Bao B. Memristor Initial-Boosted Coexisting Plane Bifurcations and its Extreme Multi-Stability Reconstitution in Two-Memristor-Based Dynamical System. *Sci. China Technol. Sci.* (2020) 63(04): 603–13. doi:10.1007/s11431-019-1450-6
22. Yu F, Kong X, Chen H, Yu Q, Cai S, Huang Y, et al. A 6D Fractional-Order Memristive Hopfield Neural Network and its Application in Image Encryption. *Front. Phys.* (2022) 10:847385. doi:10.3389/fphy.2022.847385
23. Yuan F, Wang G, Wang X. Extreme Multistability in a Memristor-Based Multi-Scroll Hyper-Chaotic System. *Chaos* (2016) 26(7):073107–519. doi:10.1063/1.4958296
24. Xiong L, Yang F, Mou J, An X, Zhang X. A Memristive System and its Applications in Red-Blue 3D Glasses and Image Encryption Algorithm with DNA Variation. *Nonlinear Dyn*(2022) 107(5):2911–33. doi:10.1007/s11071-021-07131-6
25. Huang L-L, Liu S, Xiang J-H, Wang L-Y. Design and Multistability Analysis of Five-Value Memristor-Based Chaotic System with Hidden Attractors. *Chin Phys. B* (2021) 30(10):100506. doi:10.1088/1674-1056/ac1e13
26. Ho Y, Huang GM, Li P. Dynamical Properties and Design Analysis for Nonvolatile Memristor Memories. *IEEE Trans. Circuits Syst. I* (2011) 58(4): 724–36. doi:10.1109/TCSI.2010.2078710
27. Kim H, Sah MP, Yang C, Cho S, Chua LO. Memristor Emulator for Memristor Circuit Applications. *IEEE Trans. Circuits Syst. I* (2012) 59(10):2422–31. doi:10.1109/TCSI.2012.2188957
28. Xiong L, Zhang X, Teng S, Qi L, Zhang P. Detecting Weak Signals by Using Memristor-Involved Chua's Circuit and Verification in Experimental Platform. *Int. J. Bifurc Chaos* (2020) 30(13):2050193. doi:10.1142/S021812742050193X
29. Kengne J. Periodicity, Chaos, and Multiple Attractors in a Memristor-Based Shinriki's Circuit. *Chaos Interdiscip J Nonlinea* (2015) 25:103126. doi:10.1063/1.4934653
30. Li C, Yang Y, Yang X, Zi X, Xiao F. A Tristable Locally Active Memristor and its Application in Hopfield Neural Network. *Nonlinear Dyn*(2022) 108: 1697–717. doi:10.1007/s11071-022-07268-y
31. Sun J, Han G, Zeng Z, Wang Y. Memristor-Based Neural Network Circuit of Full-Function Pavlov Associative Memory With Time Delay and Variable Learning Rate. *IEEE Trans. Cybern.* (2019) 50(7):1–11. doi:10.1109/TCYB.2019.2951520
32. Zhou Y, Li C, Li W, Li H, Feng W, Qian K. Image Encryption Algorithm with Circle Index Table Scrambling and Partition Diffusion. *Nonlinear Dyn*(2021) 103(2):2043–61. doi:10.1007/s11071-021-06206-8
33. Zhou S, Wang X, Zhang Y, Ge B, Wang M, Gao S. A Novel Image Encryption Cryptosystem Based on True Random Numbers and Chaotic Systems. *Multimed Syst* (2021) 28:95–112. doi:10.1007/s00530-021-00803-8
34. Ruan J, Sun K, Mou J, He S, Zhang L. Fractional-order Simplest Memristor-Based Chaotic Circuit with New Derivative. *Eur. Phys. J. Plus* (2018) 133(1):3. doi:10.1140/epjp/i2018-11828-0
35. Liu W, Sun K, He S. SF-SIMM High-Dimensional Hyperchaotic Map and its Performance Analysis. *Nonlinear Dyn*(2017) 89(4):2521–32. doi:10.1007/s11071-017-3601-3
36. Ma X, Mou J, Liu J, Ma C, Yang F, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor-Memcapacitor. *Nonlinear Dyn*(2020) 100(3):2859–76. doi:10.1007/s11071-020-05601-x
37. Yildirim M. DNA Encoding for RGB Image Encryption with Memristor Based Neuron Model and Chaos Phenomenon. *Microelectron J* (2020) 104:104878. doi:10.1016/j.mejo.2020.104878
38. Zhang X, Li C, Dong E, Zhao Y, Liu Z. A Conservative Memristive System with Amplitude Control and Offset Boosting. *Int. J. Bifurc Chaos* (2022) 32(04): 2250057. doi:10.1142/S0218127422500572
39. Liu T, Banerjee S, Yan H, Mou J. Dynamical Analysis of the Improper Fractional-Order 2D-SCLMM and its DSP Implementation. *Eur. Phys. J. Plus* (2021) 136(5):1–17. doi:10.1140/epjp/s13360-021-01503-y
40. Gao X, Mou J, Banerjee S, Cao Y, Xiong L, Chen X. An Effective Multiple-Image Encryption Algorithm Based on 3D Cube and Hyperchaotic Map. *J King Saud Univ - Comput Inf Sci* (2022) 34(4):1535–51. doi:10.1016/j.jksuci.2022.01.017
41. Li Y, Li C, Zhang S, Chen GR, Zeng Z. A Self-Reproduction Hyperchaotic Map With Compound Lattice Dynamics. *IEEE Trans. Ind. Electron.* (2022) 1. doi:10.1109/TIE.2022.3144592

42. Peng Y, Sun K, He S. A Discrete Memristor Model and its Application in Hénon Map. *Chaos, Solit Fractals* (2020) 137:109873. doi:10.1016/j.chaos.2020.109873
43. Deng Y, Li Y. Bifurcation and Bursting Oscillations in 2D Non-autonomous Discrete Memristor-Based Hyperchaotic Map. *Chaos, Solit Fractals* (2021) 150: 111064. doi:10.1016/j.chaos.2021.111064
44. Li C, Li H, Xie W, Du J. A S-type Bistable Locally Active Memristor Model and its Analog Implementation in an Oscillator Circuit. *Nonlinear Dyn*(2021) 106(1):1041–58. doi:10.1007/s11071-021-06814-4
45. Li Y, Li C, Zhao Y, Liu S. Memristor-Type Chaotic Mapping. *Chaos* (2022) 32(2):021104. doi:10.1063/5.0082983
46. Peng Y, He S, Sun K. Chaos in the Discrete Memristor-Based System with Fractional-Order Difference. *Results Phys* (2021) 24:104106. doi:10.1016/j.rinp.2021.104106
47. Wu C, Sun K, Xiao Y. A Hyperchaotic Map with Multi-Elliptic Cavities Based on Modulation and Coupling. *Eur. Phys. J. Spec. Top.* (2021) 230(7):2011–20. doi:10.1140/epjs/s11734-021-00126-9
48. Ye X, Mou J, Luo C, Yang F, Cao Y. Complexity Analysis of a Mixed Memristive Chaotic Circuit. *Complexity* (2018) 2018:1–9. doi:10.1155/2018/8639470

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

ORIGINAL RESEARCH
published: 10 May 2022
doi: 10.3389/fphy.2022.902500

A Chaotic System With Infinite Attractors Based on Memristor

*Junjie Wen and Jinpeng Wang**

School of Mechanical Engineering and Automation, Dalian Polytechnic University, Dalian, China

In this article, a memristor chaotic system is constructed by introducing a cosine function flux control memristor. By analyzing the balance of the system, it is found that there are coexisting attractors, and because of the periodicity of cosine function, the chaotic system has infinite coexisting attractors. The complexity analysis of Spectral Entropy (SE) and C0 is used in this paper to intuitively show the complex dynamic characteristics of the system. In addition, the introduction of paranoid propulsion also provides more possibilities for the system in engineering applications. Finally, the digital signal processing (DSP) experiment verifies the correctness of theoretical analysis and numerical analysis.

Keywords: memristor, infinite coexisting attractors, complexity analysis, offset boosting, DSP

Edited by:
Chunlai Li,
Hunan Institute of Science and Technology, China

Reviewed by:
Xiaolin Ye,
Anshan Normal University, China
Jacques Kengne,
University of Dschang, Cameroon
Baoxiang Du,
Heilongjiang University, China

****Correspondence:***
Jinpeng Wang
wangjp@dlpu.edu.cn

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *23 March 2022*
Accepted: *11 April 2022*
Published: *10 May 2022*

Citation:
Wen J and Wang J (2022) A Chaotic System With Infinite Attractors Based on Memristor.
Front. Phys. 10:902500.
doi: 10.3389/fphy.2022.902500

1 INTRODUCTION

In the 1960s, due to the birth of the Lorentz system [1], people started a period of a widespread upsurge in the study of chaos theory. Chaos is a physical phenomenon highly sensitive to initial values. Therefore, in the past decades, people have shown great interest in the creation of chaos. Chaos [2–10] has gradually developed from the study of climate to other fields, such as information science, biology, engineering, finance, and so on. So far, under the research of people, the chaotic system has been widely developed, from the original Lorentz system, to produce many different chaotic systems, including hyperchaotic system [11–17], discrete chaotic system [18–22], memristor chaotic system [23–25] and so on.

As the fourth basic circuit element, the memristor has aroused great interest in the nonlinear field since its birth. In 1971, Professor Chua made a prediction about the existence of the memristor based on the symmetrical structure of circuit elements. In 2008, HP company successfully developed a solid-state memristor that proved Chua's prediction. Therefore, the study of memristor has become a new theoretical branch. In recent years, the combination of the memristor and chaotic system has formed a new memristor chaotic system [26–32], which is also called a new research hotspot.

In addition, the multi-steady state [33–37] of dynamic systems has aroused considerable interest in the nonlinear field. When the chaotic system has the property of a periodic state, the system may have periodic attractor pairs, which are constantly shifted in the same direction, so as to achieve an infinite multi-steady state. This discovery provides a new method for studying the multi-steady state of dynamic systems in the future.

In this paper, we choose an offset boosting [38–43] chaotic system as the basis, and introduce a memristor based on the cosine function to construct a new 5-D chaotic system, which has the following properties: 1) chaos generation; 2) infinite coexistence attractor; 3) and ffset boosting. In **Section 2**, the memristor in this system is analyzed, and obtain the "8" curve of proving the memristor. In **Section 3**, the corresponding chaotic phase diagram of the system is obtained through simulation analysis, and the related dynamics analysis of different parameters of the chaotic system is carried out, including the bifurcation diagram, Lyapunov exponential spectrum, infinite coexistence attractors, offset boosting, complexity analysis. In **Section 4**, the chaotic system is simulated by digital simulation and DSP experiment platform, and the reality of the physical existence of the system is verified. Finally, the conclusion of the study is given in **Section 5**.

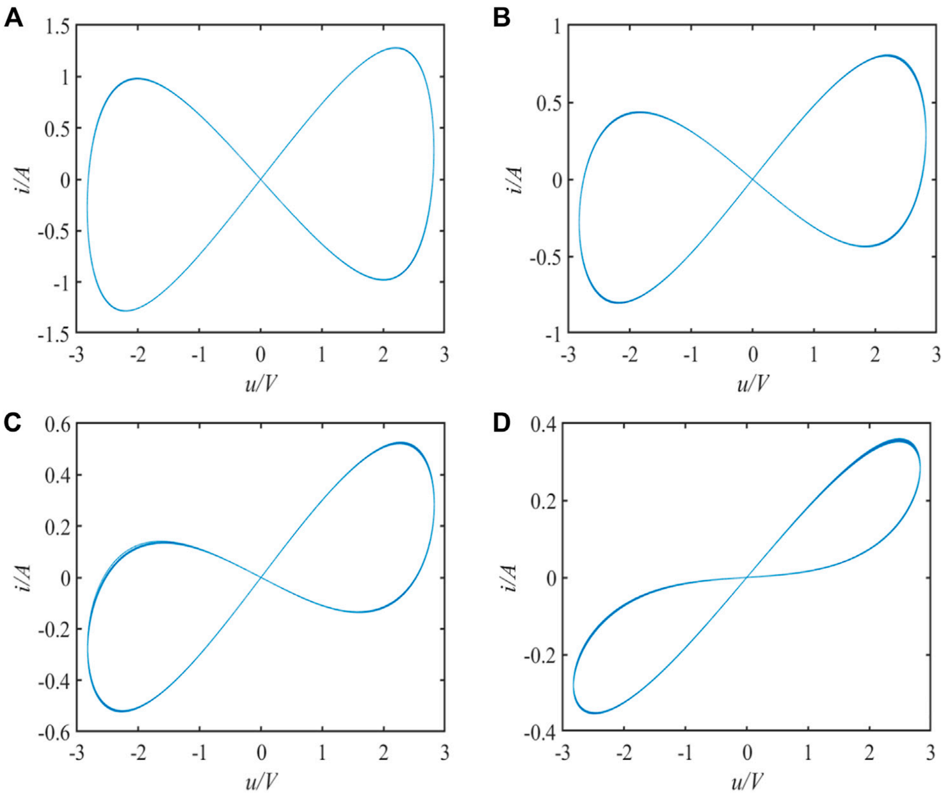


FIGURE 1 | The *u*–*i* curve of memristor: **(A)** *f* = 0.5 kHz with *u* = 3 V, **(B)** *f* = 1 kHz with *u* = 3 V, **(C)** *f* = 2 kHz with *u* = 3 V, **(D)** *f* = 5 kHz with *u* = 3 V.

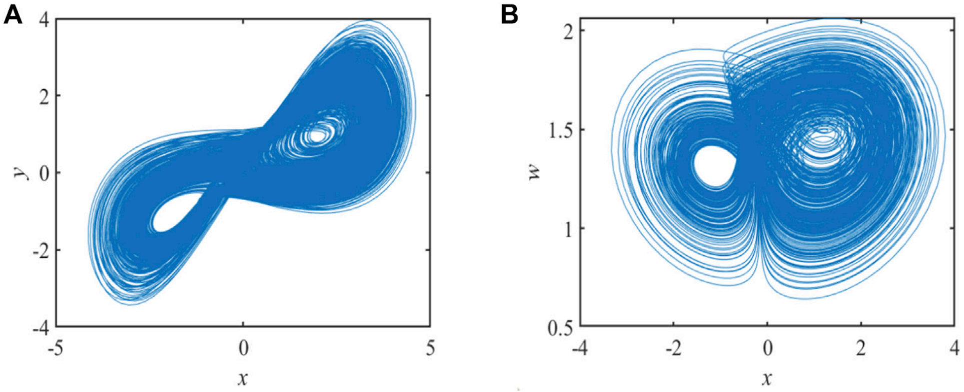


FIGURE 2 | Chaotic attractor of system (3): **(A)** *X-Y* plane, **(B)** *X-W* plane.

2 MATHEMATICAL MODEL

2.1 Memristor Model

The memristor model is expressed as follows:

$$\begin{cases} x = G(x, y, t)y \\ \dot{y} = H(x, y, t) \end{cases}, \tag{1}$$

where x is this output of the memristor, y represents the state of the memristor, and the functions G (.) and H (.) are particularly relevant to the memristor.

The flux-controlled memristor is from cosine function, and the specific equation is shown as follows:

$$\begin{cases} W(w) = \dfrac{dq(w)}{d(w)} = \cos(w) \\ i = W(w)y = \cos(w)y \\ \dfrac{d(w)}{dt} = y^2 - w \end{cases}, \tag{2}$$

Here, the following the “8” curve of the memristor is obtained by inputting different frequency parameters as shown in **Figure 1**.

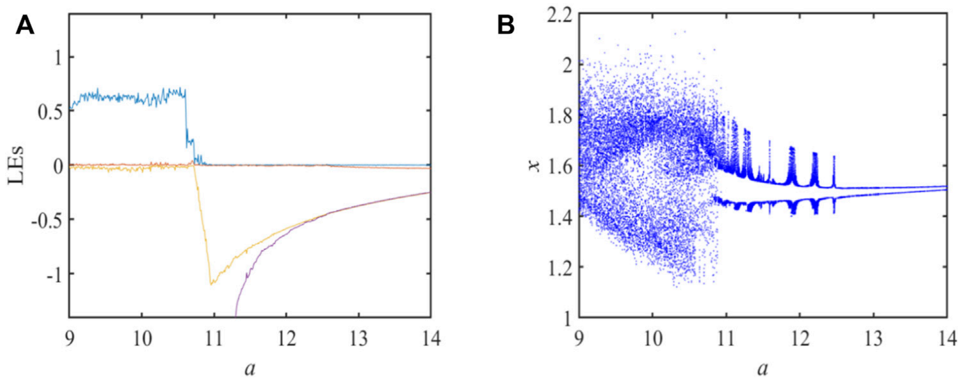


FIGURE 3 | (A) LEs, **(B)** bifurcation diagram.

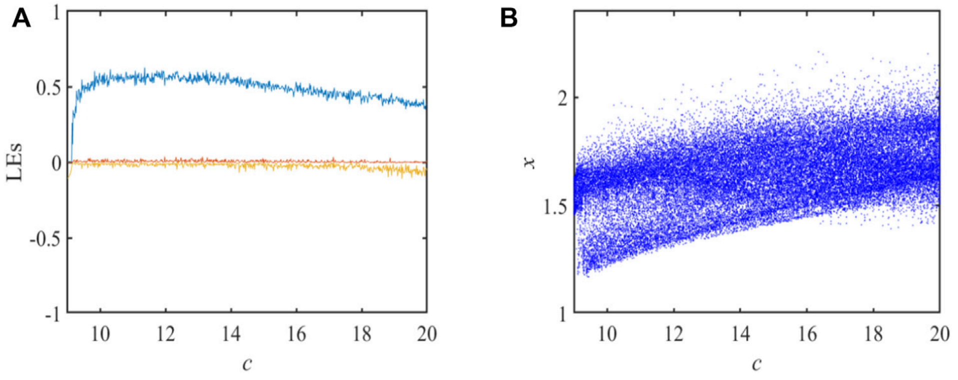


FIGURE 4 | (A) LEs, **(B)** bifurcation diagram.

TABLE 1 | System state corresponding to different parameter *a* when the original conditions is (1, 1, 1, 1, 1).

Range	LEs	State	Range	LEs	State
—	0 - - - -	Divergence	10.85	+0 - - -	Weak chaos
9.00–10.72	+0 - - -	Chaos	10.86	0 - - - -	Period
10.73	0 - - - -	Period	10.87	+0 - - -	Weak chaos
10.74–10.78	+0 - - -	Weak chaos	10.88	0 - - - -	Period
10.79	0 - - - -	Period	10.89	+0 - - -	Weak chaos
10.80–10.83	+0 - - -	Weak chaos	10.90–14.00	0 - - - -	Chaos
10.84	0 - - - -	Period	—	—	—

TABLE 2 | System state corresponding to different parameter *c* when the original conditions is (1, 1, 1, 1, 1).

Range	LEs	State	Range	LEs	State
—	0 - - - -	Divergence	9.13–20.00	+0 - - -	Chaos
9.00–9.12	0 - - - -	Period	—	—	—

TABLE 3 | System state corresponding to different parameter *d* when the original conditions is (1, 1, 1, 1, 1).

Range	LEs	State	Range	LEs	State
—	0 - - - -	Divergence	12.75–12.80	+0 - - -	Chaos
9.00–9.43	0 - - - -	Period	12.81	+0 - - -	Weak chaos
9.43–12.47	+0 - - -	Chaos	12.82–13.75	+0 - - -	Chaos
12.48	+0 - - -	Weak chaos	13.76	+0 - - -	Weak chaos
12.49–12.54	+0 - - -	Chaos	13.77–13.78	+0 - - -	Chaos
12.55	+0 - - -	Weak chaos	13.79	+0 - - -	Weak chaos
12.56–12.58	+0 - - -	Chaos	13.80–13.82	+0 - - -	Chaos
12.59	0 - - - -	Period	13.83	+0 - - -	Weak chaos
12.60–12.73	+0 - - -	Chaos	13.84–20	+0 - - -	Chaos
12.74	+0 - - -	Weak chaos	—	—	—

2.2 Equilibrium Points Set and Stability

By combining the memristor with the 4-D chaotic system, the specific expression of the chaotic system is shown below:

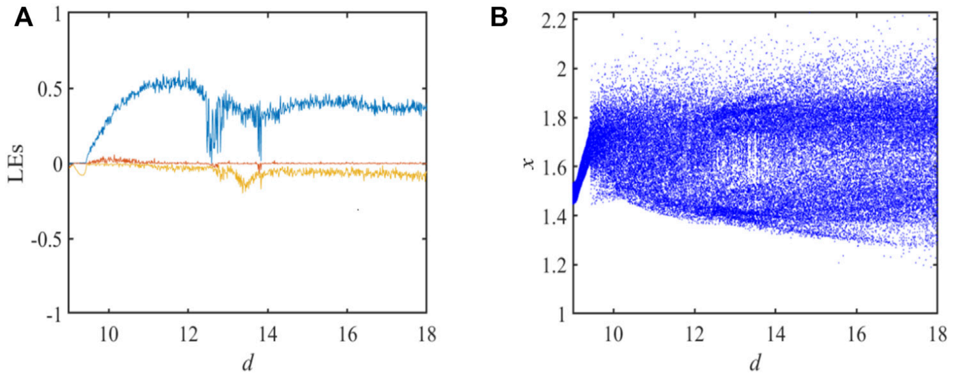


FIGURE 5 | (A) LEs **(B)** bifurcation diagram.

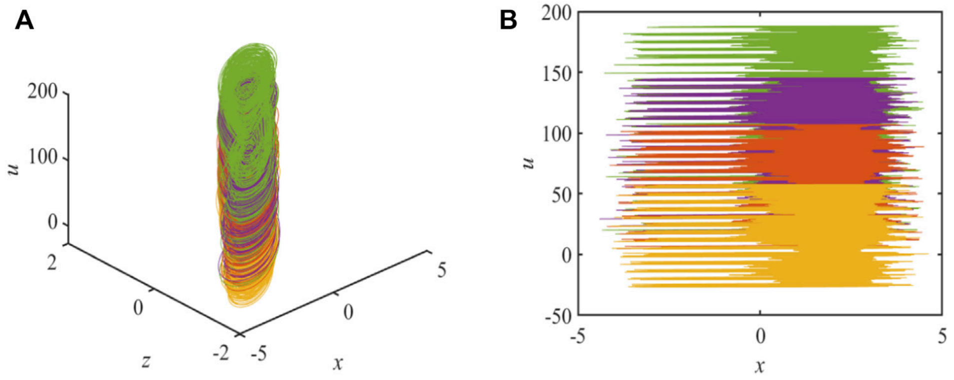


FIGURE 6 | Coexisting attractors in system (3) with $a = 9$, $b = 11$, $c = 15$, $d = 11$, $k = 1.5$, $e = 5$, $g = 1$ under original conditions $(1, 1, 1, 1, x_0)$, $x_0 = -11\pi$ $x_0 = -5\pi$ $x_0 = 0$ $x_0 = -3\pi$ are for yellow, red, purple, and green correspondingly: **(A)** coexisting attractors in x–u plane **(B)** time-domain waveform.

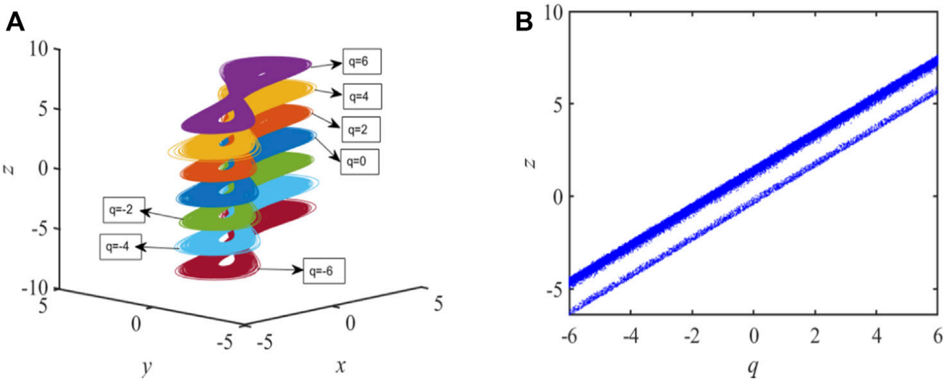


FIGURE 7 | (A) The attractor at different offsets q **(B)** bifurcation diagram.

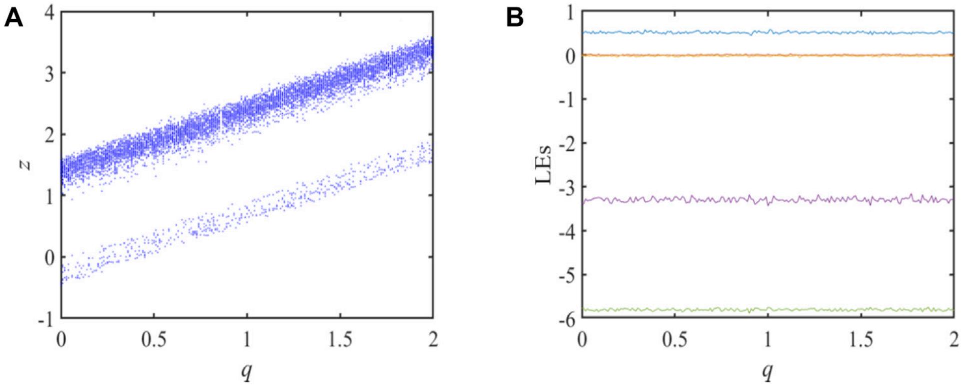


FIGURE 8 | (A) bifurcation diagram **(B)** Lyapunov exponent spectrum.

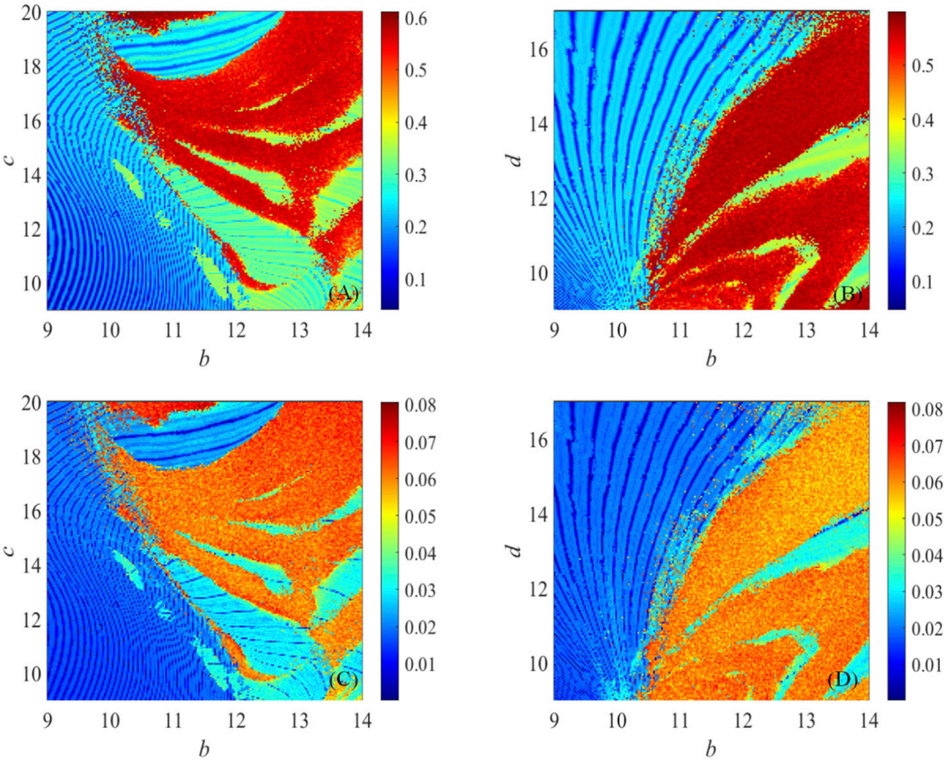


FIGURE 9 | Complexity diagram with different parameter *a d* and *e*. **(A)** Parameters *a d* in SE complexity; **(B)** parameters *a e* in SE complexity; **(C)** parameters *a d* in C_0 complexity; **(D)** parameters *a e* in C_0 complexity.

$$\begin{cases} \dot{x} = ay - bz + \cos u \\ \dot{y} = cy\cos w - x^2 z + k \\ \dot{z} = -dz + ex \\ \dot{u} = gy \\ \dot{w} = y^2 - w \end{cases} . \tag{3}$$

The divergence of the system is shown in **Eq. 4**,

$$\nabla V = \frac{\partial \dot{x}}{\partial x} + \frac{\partial \dot{y}}{\partial y} + \frac{\partial \dot{z}}{\partial z} + \frac{\partial \dot{u}}{\partial u} + \frac{\partial \dot{w}}{\partial w}, \tag{4}$$

when $a = 9, b = 11, c = 15, d = 11, k = 1.5, e = 5$, and $g = 1$, and this original conditions (1, 1, 1, 1, 1); ∇V not greater than zero; it is proved that the system is dissipative, there could be chaotic attractors in this system.

Setting $\dot{x} = \dot{y} = \dot{z} = \dot{u} = \dot{w} = 0$, then lead to

$$\begin{cases} ay - bz + \cos u = 0 \\ cy\cos w - x^2 z + k = 0 \\ -dz + ex = 0 \\ gy = 0 \\ y^2 - w = 0 \end{cases} , \tag{5}$$

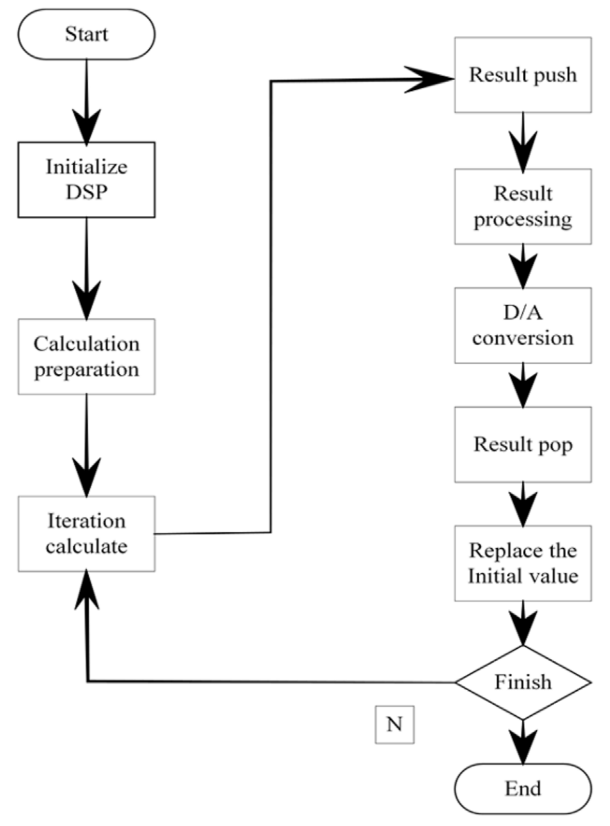


FIGURE 10 | Programming flow of DSP implementation.

from this the following equation can be obtained

$$E^{*} = \begin{cases} None, k \neq 0 \\ \left(0, 0, 0, 0, n\pi + \dfrac{\pi}{2}\right), k = 0 \end{cases}. \tag{6}$$

If $k = 0$, the equilibrium set is set to O, Jacobi matrix J_E at the equilibrium points O

$$J_E = \begin{bmatrix} -a & z & y & b & 1 \\ -z & c & -x & 0 & 0 \\ y & x & -d & 0 & 0 \\ z & 0 & x & -e & 0 \\ 0 & g & 0 & 0 & 0 \end{bmatrix}, \tag{7}$$

then, the secular equation set

$$\lambda^5 + a_1\lambda^4 + a_2\lambda^3 + a_3\lambda^2 + a_4\lambda + a_5 = 0, \tag{8}$$

where a_1 = 13.9, a_2 = -59.1, a_3 = 408.5, a_4 = 1,324.4, and a_5 = -39.5.

As shown in **Eq. 6**, the system is unstable according to the Rous criterion. Hence, when n be an any constant, and any equilibrium points within $O(0, 0, 0, 0, n\pi + \frac{\pi}{2})$, other parameters are n = 0, a = 9, b = 11, c = 15, d = 11, k = 1.5, e = 5, and g = 1, it results $a_2 = -59.1 < 0$, $a_5 = -39.5 < 0$, and λ_1 = -18.1636, λ_2 = 3.2391 + 4.7287i, λ_3 = 3.2391–4.7287i, λ_4 = -2.2397, λ_5 = 0.0296. At this point, the system is chaos.

3 NUMERICAL DIAGRAMS OF THE DYNAMICAL BEHAVIOR

3.1 Chaotic Attractor

Let a = 9, b = 11, c = 15, d = 11, k = 3, e = 5, g = 1, and original conditions (1, 1, 1, 1, 1). Through simulation of the system, the phase diagrams of different chaotic attractors are shown in **Figures 2**, **3**.

3.2 LEs and Bifurcation Diagram of the System

System (3) sets three parameters, such as a, c, and d. The specific parameter range is as follows $a \in [10, 15]$, $c \in [4, 6]$, and $d \in [3, 12]$.

After fixed the remaining parameters, set a, c, and d as variables, respectively, to determine their initial values. When analyzing the state of the system when the variable changes, the bifurcation diagram is usually combined with Lyapunov

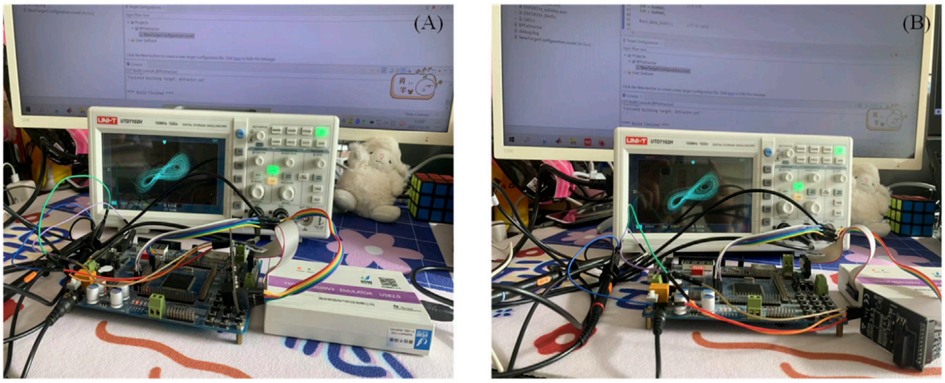


FIGURE 11 | Experimental platform for DSP implementation.

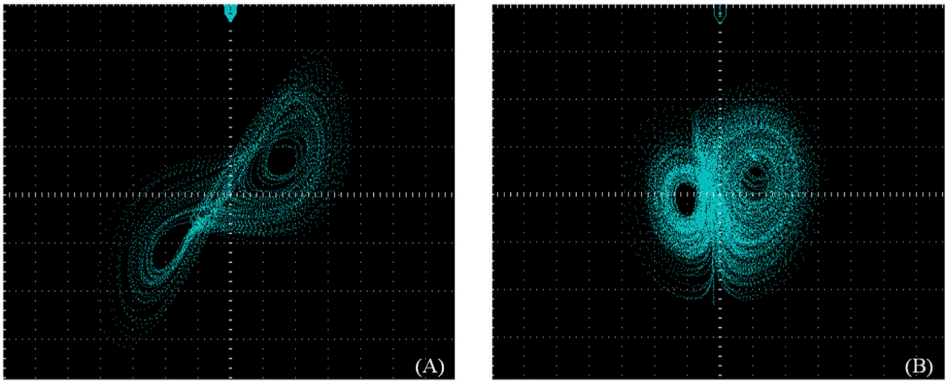


FIGURE 12 | Chaotic attractor on the DSP platform: **(A)** *X-Y* plane, **(B)** *X-W* plane.

exponential spectrum, so that the changing state of the system can be analyzed more specifically.

Set $a\in$ [9, 15], $b = 11$, $c = 15$ $d = 11$, $k = 1.5$, $e = 5$, $g = 1$. The LEs and bifurcation diagram as shown in **Figure 2** are obtained through software simulation. Meanwhile, smaller LEs are omitted below for easy observation. In the process of adjusting parameter a, it is found that the dynamical behavior of the system has complex dynamical characteristics, including a variety of states, such as period and chaos, and their mutual transformation. It can be observed from the figure that when parameter $a \in [9, 11.2]$, and in the following range, the system slowly transforms from chaos to period and remains stable. In addition, parameter a ranges the corresponding system status can be clearly understood through the **Table 1**.

Set the parameter $c\in$ [10, 30] and keep the other unchanged. When $c\in [9, 9.12]$, the system is in a period, as shown in **Figure 4**. At the same time, the system changes rapidly from period to chaos in the following range. Specific numerical changes are shown in **Table 2** and **Figure 5**.

Set $d\in$ [9, 18], and keep the other unchanged. In **Figure 6**, when $d\in [9, 9.43]$, at this point, the system LEs is 0, indicating that the system is in a periodic state, then the system changes to a chaos. When parameter $d = 12.74$ the system changes briefly to a periodic state. **Table 3** clearly shows the process of system change within the range of parameter d.

3.3 Infinite Coexisting Attractors

Coexistence attractor is a kind of phenomenon which mainly occurs in special nonlinear systems and has become a research hotspot. Because of the existence of trigonometric function in an equation, there exists the infinite coexisting attractor. The main phenomenon is that when parameters remain fixed and the initial values change, the trajectories could gradually tend to different states of motion. The phenomenon of the infinite coexisting attractor could be observed in **Figure 6**.

3.4 Offset Boosting Scheme

A new feedback state is introduced in the system to control the system flexibly so that the attractor and its attractor pool can move arbitrarily. This method is called offset boosting. By introducing a parameter q to boost the variable z, the system's attractor and its attractor pool are controlled. The improved offset-boosted system is shown in **Eq. 9**:

$$\begin{cases} x = ay - b(z-q) + \cos u \\ y = cy\cos w - x^2(z-q) + k \\ z = -d(z-q) + ex \\ u = gy \\ w = y^2 - w \end{cases} . \tag{9}$$

Let $a = 9$, $b = 11$, $c = 15$, $d = 11$, $k = 1.5$, $e = 5$, $g = 1$, when q is a constant. **Figure 7A** shows the 3D projection of the attractor with different offsets q, and with the increase of parameter q, the position of the attractor also shows a regular upward trend. **Figure 7B** shows the corresponding bifurcation diagram.

In **Figures 7A,B**, the local bifurcation diagram and the corresponding Lyapunov exponent spectrums are shown respectively. When parameter $q \in [0, 2]$, the state variable z of the system augment with the augment of offset variable q, while the attractor LEs of the system does not change. In **Figure 8B**, the attractor is offset accordingly by introducing new variables, which have considerable practical application value in engineering.

3.5 Complexity Analysis

The complexity analysis is one of the important methods to study chaotic system. In practical studies, complexity algorithms are generally introduced to measure chaotic sequences. In order to show the complexity of the chaotic system more clearly, as shown in **Figure 9B**, parameters b, c, b, and d are set respectively. When $b\in$ [10.5, 14] and $d\in$ [9, 18], it could be clearly observed that the region is dark, indicating that the system is in a chaotic state at this time. In this diagram, as the color gets darker, it also means that the system gets more complex. In addition, this multi-dimensional complexity analysis method also offers a deterministic foundation for the parameter pick of the systems.

4 DIGITAL CIRCUIT PLATFORM IMPLEMENTATION

Fast speed, high accuracy, and low environmental impact are the characteristics of the DSP chip F28335. The chaos is confirmed on the experimental platform. The D/A converter needs to simulate and convert the sequence code generated by the DSP to snatch the output sequence displayed on the corresponding range (UTD7102H). By discretization of continuous chaos, correlation can be processed on DSP platform. Then, the fourth-order Runge-Kutta method is used to transform them into discrete chaotic sequences. Finally, the iterative relation is imported into DSP by C language and a corresponding simulation is carried out. The actual operation and experiment are shown in **Figure 10**.

The parameters were set as $a = 9$, $b = 11$, $c = 15$, $d = 11$, $k = 1.5$, $e = 5$, $g = 1$. The concrete experimental objects are shown in **Figure 11**, and the attractor shown in the figure corresponds to **Figures 2A,B** one by one.

The phase diagram in **Figure 12** shown on an oscilloscope is identical to the phase diagram simulated by the computer, The experiment proves that the digital simulation circuit is correct.

5 CONCLUSION

In this article, a memristor chaotic system is proposed and its dynamic behavior is analyzed. At the same time, it is proved that the appearance of infinite coexisting attractor caused by the periodicity of a trigonometric function is correct. By introducing the variable q into the variable z, the variable z changes parallel with the control variable q, while the value of LEs remains stable. It is proved that the method of flexibly changing the original sequence by introducing a new control variable is effective, and this method is called offset boosting. Finally, the corresponding digital circuit experiment is carried out, and the correctness of the experimental results is also verified. The influence of trigonometric function on memristor chaotic system is studied in this article, which provides a new reference and idea for the future study of the interaction between the chaotic system and periodic function.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Materials, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

JW provided the idea of algorithm, carried out the simulations, arranged the architecture and drafted the manuscript. JW supervised the work and revised the manuscript. Both authors read and approved the final manuscript.

FUNDING

This research was financially supported by the Project for the National Natural Science Foundation of China (61402069), the 2017 Project for the Natural Science Foundation of Liaoning province (20170540059), and the General project of the National Social Science Fund (2019AG00482).

REFERENCES

1. Lorenz EN. Deterministic Nonperiodic Flow. *J Atmos Sci* (1963) 20(2):130–41. doi:10.1175/1520-0469(1963)020<0130:dnf>2.0.co;2
2. Liu T, Banerjee S, Yan H, Mou J. Dynamical Analysis of the Improper Fractional-Order 2D-SCLMM and its DSP Implementation. *Eur Phys J Plus* (2021) 136(5):506. doi:10.1140/epjp/s13360-021-01503-y
3. Ma C, Mou J, Li P, Liu T. Dynamic Analysis of a New Two-Dimensional Map in Three Forms: Integer-Order, Fractional-Order and Improper Fractional-Order. *Eur Phys J Spec Top* (2021) 230(7):1945–57. doi:10.1140/epjs/s11734-021-00133-w
4. Ma C, Mou J, Xiong L, Banerjee S, Liu T, Han X. Dynamical Analysis of a New Chaotic System: Asymmetric Multistability, Offset Boosting Control and Circuit Realization. *Nonlinear Dyn* (2021) 103(3):2867–80. doi:10.1007/s11071-021-06276-8
5. Ma X, Mou J, Liu J, Ma C, Yang F, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor-Memcapacitor. *Nonlinear Dyn* (2020) 100(3):2859–76. doi:10.1007/s11071-020-05601-x
6. Mobayen S, Fekih A, Vaidyanathan S, Sambas A. Chameleon Chaotic Systems with Quadratic Nonlinearities: An Adaptive Finite-Time Sliding Mode Control Approach and Circuit Simulation. *IEEE Access* (2021) 9:64558–73. doi:10.1109/access.2021.3074518
7. Sambas A, Vaidyanathan S, Bonny T, Zhang S, SukonoHidayat Y, et al. Mathematical Model and FPGA Realization of a Multi-Stable Chaotic Dynamical System with a Closed Butterfly-like Curve of Equilibrium Points. *Appl Sci* (2021) 11(2):788. doi:10.3390/app11020788
8. Li X, Mou J, Cao Y, Banerjee S. An Optical Image Encryption Algorithm Based on a Fractional-Order Laser Hyperchaotic System. *Int J Bifurcation Chaos* (2022) 32(03):2250035. doi:10.1142/s0218127422500353
9. Gao X, Mou J, Xiong L, Sha Y, Yan H, Cao Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn* (2022) 108:613–636. doi:10.1007/s11071-021-07192-7
10. Gao X, Mou J, Banerjee S, Cao Y, Xiong L, Chen X. An Effective Multiple-Image Encryption Algorithm Based on 3D Cube and Hyperchaotic Map. *J King Saud University-Computer Inf Sci* (2022) 34:1535–51. doi:10.1016/j.jksuci.2022.01.017
11. Chen Z, Zhao H, Chen J. A Novel Image Encryption Scheme Based on Poker Cross-Shuffling and Fractional Order Hyperchaotic System. In: *International Conference on Computer Engineering and Networks*. Springer (2020). p. 818–25. doi:10.1007/978-981-15-8462-6_94
12. Al-Azzawi SF, Al-Obeidi AS. Chaos Synchronization in a New 6D Hyperchaotic System with Self-Excited Attractors and Seventeen Terms. *Asian-european J Math* (2021) 14(05):2150085. doi:10.1142/s1793557121500856
13. Yu F, Li L, He B, Liu L, Qian S, Zhang Z, et al. Pseudorandom Number Generator Based on a 5D Hyperchaotic Four-wing Memristive System and its FPGA Implementation. *Eur Phys J Spec Top* (2021) 230(7):1763–72. doi:10.1140/epjs/s11734-021-00132-x
14. Chen X, Qian S, Yu F, Zhang Z, Shen H, Huang Y, et al. Pseudorandom Number Generator Based on Three Kinds of Four-wing Memristive Hyperchaotic System and its Application in Image Encryption. *Complexity* (2020) 2020:1–17. doi:10.1155/2020/8274685

15. Huang W, Jiang D, An Y, Liu L, Wang X. A Novel Double-Image Encryption Algorithm Based on Rossler Hyperchaotic System and Compressive Sensing. *IEEE Access* (2021) 9:41704–16. doi:10.1109/access.2021.3065453
16. Lai Q, Wan Z, Kamdem Kuate PD, Fotsin H. Dynamical Analysis, Circuit Implementation and Synchronization of a New Memristive Hyperchaotic System with Coexisting Attractors. *Mod Phys Lett B* (2021) 35(10):2150187. doi:10.1142/s0217984921501876
17. Ahmad M, Doja MN, Beg MMS. Security Analysis and Enhancements of an Image Cryptosystem Based on Hyperchaotic System. *J King Saud Univ - Comp Inf Sci* (2021) 33(1):77–85. doi:10.1016/j.jksuci.2018.02.002
18. Kumar D, Joshi AB, Singh S, Mishra VN, Rosales HG, Zhou L, et al. 6D-Chaotic System and 2D Fractional Discrete Cosine Transform Based Encryption of Biometric Templates. *IEEE Access* (2021) 9:103056–74. doi:10.1109/access.2021.3097881
19. Peng Y, He S, Sun K. A Higher Dimensional Chaotic Map with Discrete Memristor. *AEU - Int J Elect Commun* (2021) 129:153539. doi:10.1016/j.aeue.2020.153539
20. Chu Y-M, Bekiros S, Zambrano-Serrano E, Orozco-López O, Lahmiri S, Jahanshahi H, et al. Artificial Macro-Economics: A Chaotic Discrete-Time Fractional-Order Laboratory Model. *Chaos, Solitons & Fractals* (2021) 145: 110776. doi:10.1016/j.chaos.2021.110776
21. Alhadawi HS, Majid MA, Lambić D, Ahmad M. A Novel Method of S-Box Design Based on Discrete Chaotic Maps and Cuckoo Search Algorithm. *Multimed Tools Appl* (2021) 80(5):7333–50. doi:10.1007/s11042-020-10048-8
22. Dai J-Y, Ma Y, Zhou N-R. Quantum Multi-Image Compression-Encryption Scheme Based on Quantum Discrete Cosine Transform and 4D Hyper-Chaotic Henon Map. *Quan Inf Process* (2021) 20(7):1–24. doi:10.1007/s11128-021-03187-w
23. Liao K, Lei P, Tu M, Luo S, Jiang T, Jie W, et al. Memristor Based on Inorganic and Organic Two-Dimensional Materials: Mechanisms, Performance, and Synaptic Applications. *ACS Appl Mater Inter* (2021) 13(28):32606–23. doi:10.1021/acsami.1c07665
24. Peng Y, He S, Sun K. Chaos in the Discrete Memristor-Based System with Fractional-Order Difference. *Results Phys* (2021) 24:104106. doi:10.1016/j.rinp.2021.104106
25. Bao H, Hua Z, Li H, Chen M, Bao B. Discrete Memristor Hyperchaotic Maps. *IEEE Trans Circuits Syst* (2021) 68(11):4534–44. doi:10.1109/tcsi.2021.3082895
26. Xie W, Wang C, Lin H. A Fractional-Order Multistable Locally Active Memristor and its Chaotic System with Transient Transition, State Jump. *Nonlinear Dyn* (2021) 104(4):4523–41. doi:10.1007/s11071-021-06476-2
27. Akgül A, Rajagopal K, Durdu A, Pala MA, Boyraz ÖF, Yildiz MZ. A Simple Fractional-Order Chaotic System Based on Memristor and Memcapacitor and its Synchronization Application. *Chaos, Solitons & Fractals* (2021) 152:111306. doi:10.1016/j.chaos.2021.111306
28. Sun S, Yan D, Ji'e M, Du X, Wang L, Duan S. Memristor-based Time-Delay Chaotic System with Hidden Extreme Multi-Stability and Pseudo-random Sequence Generator. *Eur Phys J Spec Top* (2021) 230(18):3481–91. doi:10.1140/epjs/s11734-021-00248-0
29. Wang J, Mou J, Yan H, Liu X, Ma Y, Cao Y. A Three-Port Switch NMR Laser Chaotic System with Memristor and its Circuit Implementation. *Eur Phys J Plus* (2021) 136(11):1112. doi:10.1140/epjp/s13360-021-02097-1
30. Wang Z, Qi G. Modeling and Analysis of a Three-Terminal-Memristor-Based Conservative Chaotic System. *Entropy* (2021) 23(1):71. doi:10.3390/e23010071
31. Ma X, Mou J, Xiong L, Banerjee S, Cao Y, Wang J. A Novel Chaotic Circuit with Coexistence of Multiple Attractors and State Transition Based on Two Memristors. *Chaos, Solitons & Fractals* (2021) 152:111363. doi:10.1016/j.chaos.2021.111363
32. Li C, Yang Y, Du J, Chen Z. A Simple Chaotic Circuit with Magnetic Flux-Controlled Memristor. *Eur Phys J Spec Top* (2021) 230(7):1723–36. doi:10.1140/epjs/s11734-021-00181-2
33. Gu J, Li C, Chen Y, Iu HHC, Lei T. A Conditional Symmetric Memristive System with Infinitely many Chaotic Attractors. *IEEE Access* (2020) 8: 12394–401. doi:10.1109/access.2020.2966085
34. Yang T. Multistability in a 3D Autonomous System with Different Types of Chaotic Attractors. *Int J Bifurcation Chaos* (2021) 31(02):2150028. doi:10.1142/s0218127421500280
35. Liu B, Ye X, Chen Q. Generating Infinitely many Coexisting Attractors *via* a New 3D Cosine System and its Application in Image Encryption. *IEEE Access* (2021) 9:136292–301. doi:10.1109/access.2021.3117570
36. Cheng T, Zhang Y, Shen Y. Infinite Number of Parameter Regions with Fractal Nonchaotic Attractors in a Piecewise Map. *Fractals* (2021) 29(04):2150087. doi:10.1142/s0218348x21500870
37. Li C, Peng Y, Tao Z, Sprott JC, Jafari S. Coexisting Infinite Equilibria and Chaos. *Int J Bifurcation Chaos* (2021) 31(05):2130014. doi:10.1142/s0218127421300147
38. Sayed WS, Roshdy M, Said LA, Radwan AG. Design and FPGA Verification of Custom-Shaped Chaotic Attractors Using Rotation, Offset Boosting and Amplitude Control. *IEEE Trans Circuits Syst* (2021) 68(11):3466–70. doi:10.1109/tcsii.2021.3082271
39. Li C, Wang R, Ma X, Jiang Y. A Class of Offset Boostable 3-D Memristive System. In: Proceeding of the 2021 International Conference on Neuromorphic Computing (ICNC); Oct. 2021; Wuhan, China. IEEE (2021). p. 53–60. doi:10.1109/icnc52316.2021.9608241
40. Chen B, Xu Q, Chen M, Wu H, Bao B. Initial-condition-switched Boosting Extreme Multistability and Mechanism Analysis in a Memcapacitive Oscillator. *Front Inform Technol Electron Eng* (2021) 22(11):1517–31. doi:10.1631/fitee.2000622
41. Kong S, Li C, He S, Çiçek S, Lai Q. A Memristive Map with Coexisting Chaos and Hyperchaos*. *Chin Phys. B* (2021) 30(11):110502. doi:10.1088/1674-1056/abf4fb
42. Wen J, Feng Y, Tao X, Cao Y. Dynamical Analysis of a New Chaotic System: Hidden Attractor, Coexisting-Attractors, Offset Boosting, and DSP Realization. *IEEE Access* (2021) 9:167920–7. doi:10.1109/access.2021.3136249
43. Reich S, Sporer M, Ortmanns M. A Chopped Neural Front-End Featuring Input Impedance Boosting with Suppressed Offset-Induced Charge Transfer. *IEEE Trans Biomed Circuits Syst* (2021) 15(3):402–11. doi:10.1109/tbcas.2021.3080398

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Physics

ORIGINAL RESEARCH
published: 20 May 2022
doi: 10.3389/fphy.2022.911156

A Chaos-Based Image Encryption Scheme Using the Hamming Distance and DNA Sequence Operation

Yuwen Sha[1], *Fanling Bu*[1]*, *Hadi Jahanshahi*[2]* *and Linian Wang*[1]

[1]*School of Mechanical Engineering and Automation, Dalian Polytechnic University, Dalian, China,* [2]*Department of Mechanical and Manufacturing Engineering, University of Manitoba, Winnipeg, MA, Canada*

In this study, we introduced a new memristive chaotic system with the rich dynamic behavior, and then we proposed a chaotic-based image encryption scheme which is based on the permutation–confusion–substitution structure. In our scheme, the Hamming distance is used to design a plain-related chaotic system initial condition, and the generated chaotic sequences are assigned to permutation, diffusion, and substitution stages. In the permutation stage, an effect pixel confusion is implemented through a new permutation approach, which is a double-ended select-swap scrambling strategy. In the diffusion stage, DNA XOR operation is implemented followed by DNA triploid mutation which is introduced to enhance the strength of our encryption system. A number of experiments and extensive safety analysis have been carried out and the results fully justify that our scheme not only ensures desirable security but also has superior efficiency.

Keywords: permutation–confusion–substitution structure, hamming distance, DNA triploid mutation, memristive chaotic system, image encryption

Edited by:
Shaobo He,
Central South University, China

Reviewed by:
Shuang Zhou,
Chongqing Normal University, China
Baoxiang Du,
Heilongjiang University, China

****Correspondence:***
Fanling Bu
workingfine@163.com
Hadi Jahanshahi
jahanshahi.hadi90@gmail.com

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *02 April 2022*
Accepted: *21 April 2022*
Published: *20 May 2022*

Citation:
Sha Y, Bu F, Jahanshahi H and Wang L (2022) A Chaos-Based Image Encryption Scheme Using the Hamming Distance and DNA Sequence Operation. Front. Phys. 10:911156. doi: 10.3389/fphy.2022.911156

1 INTRODUCTION

With the advent of the Internet and multimedia, the network plays a vital role in transmitting text, images, and video files as a medium. Therefore, the need to ensure the secure transmission of information on the network has become the focus of attention in the field of secure communications. Compared with text and video files, images with the visual information have many advantages in communication and transmission, such as large amount of information, small storage space occupation, and convenient transmission, so they are the most widely used [1]. However, images without special processing face the danger of being eavesdropped and crawled by some criminals using some special tools and means and image encryption came into being in this scenario [2, 3].

In recent years, image encryption technology has been developed one after another. Previous image encryption strategies that treated the image as a stream of a binary bit such as text information and then encrypted using the well-known AES and DES technologies have not been proven to use image files [4, 5]. This is due to the large amount of image data compared to adjacent pixels which are highly correlated and so on compared to text information. In contrast, compressive sensing [6, 7], DNA operation, and [8, 9] chaos theory [10, 11] are more suitable for image encryption. Among them, the chaotic image encryption algorithm is the most popular, which is due to the internal characteristics of chaos, such as initial value sensitivity; also, nonlinear behavior can ensure that the processed image has a higher security level [9, 12–23]. For instance, Hua et al. [24] presented a gray image encryption scheme based on a 2D Logistic-Sine-coupling map. The chaotic system was generated by coupling the Logistic and Sine map. In [12], Chai et al. exploited a secure cipher image

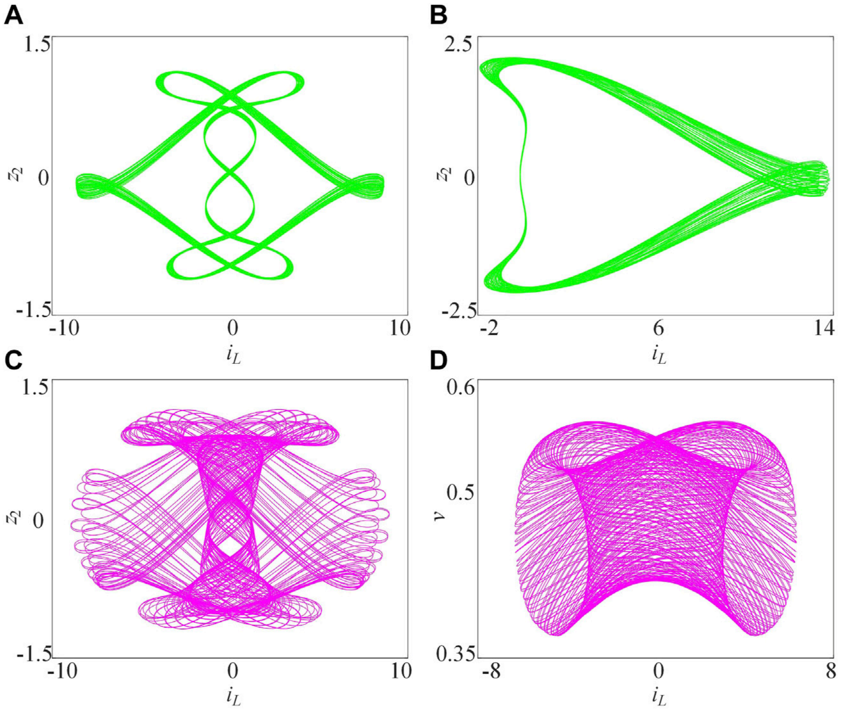


FIGURE 1 | Four kinds of attractors with different parameter values b_1 [42]: **(A)** $b_1 = -0.4$, **(B)** $b_1 = -2.58$, **(C)** $b_1 = -0.1$, and **(D)** $b_1 = -1.7$.

TABLE 1 | LE and LD values with different chaotic attractors.

Attractor	Parameter	Lyapunov Exponent	Lyapunov Dimension
I	b1 = −0.4	(0.006,0,−0.004,−0.004)	3.440
II	b1 = −2.58	(0.012,0,−0.001,−0.016)	3.660
III	b1 = −0.1	(0.004,0,−0.003,−0.010)	3.128
IV	b1 = −1.7	(0.005,0,−0.014,−8.516)	2.326

scheme using the preprocessing-permutation–diffusion structure. In this cryptosystem, the key streams were generated from a memristive hyperchaotic system. In [25], Li et al. suggested an optical cryptosystem, in which the laser hyperchaotic system was presented in a fractional-order form to improve the complexity of the chaotic sequences.

For the chaotic image cryptosystem, the security performance of the processed ciphertext is composed of many factors. One is whether the structure of the algorithm is safe enough, that is, whether to use architectures like to use the classic permutation–diffusion, permutation–substitution [26–30, 42]. Whether the initial value of the second cryptosystem is designed to be related to the plaintext image to resist known-plaintext and chosen plaintext attacks [31, 32]. Third, in the development of chaos theory, various chaotic systems are developed and they are introduced into image encryption. However, these chaotic systems are not all suitable for the field of secure communication, which may be due to their low key space and due to fewer parameters [33, 34]. In addition, the weak chaotic performance will also become the main reason to threaten the security of cryptographic systems [35–37]. Therefore, using a high-performance chaotic system to design a cryptographic scheme with a secure structure and an efficient algorithm becomes an urgent requirement [30, 38–41].

In order to improve the security and enhance the efficiency of the chaotic image cryptosystem, a memristive system with better chaos performance has been designed by our team [42], and it was introduced for image encryption in this study. First, the chaotic system with complex chaotic behavior was demonstrated through the phase diagram, Lyapunov exponents (LEs), and Lyapunov dimensions (LEs). Then, the chaotic sequences were generated for the encryption phase. Before the confusion procedure, the DNA encoding rule was provided for the image, and then a new permutation approach was designed and implemented at the DNA level. Next, we used the DNA XOR operation to perform the diffusion operation. Finally, DNA triploid mutation with superior efficiency was carried out so as to enhance the system security. The experiment results and extensive safety analysis fully justify that our cryptosystem is suitable for practical secure images due to high-security level and satisfactory encryption efficiency.

The remaining part of the study is organized as follows. In **Section 2**, we introduced a memristive chaotic system, and its dynamic behavior is evaluated by the phase diagram and Lyapunov exponent. **Section 3** shows the proposed encryption methodology. **Section 4** presents the simulation results of different size images. **Section 5** reports the security level of our encryption scheme. **Section 6** summarizes the content of this study.

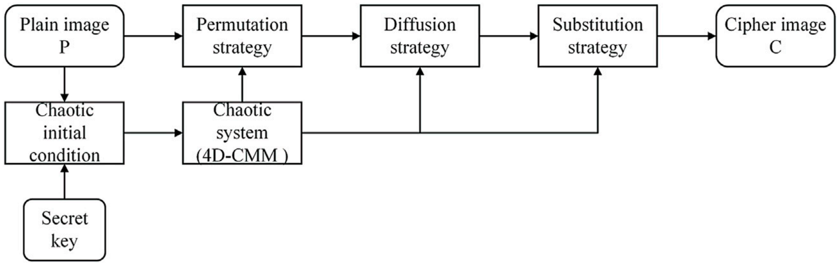


FIGURE 2 | Proposed encryption scheme flow chart.

TABLE 2 | DNA encoding and decoding rules.

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

TABLE 3 | DNA triploid mutation rules.

Ruler	1	2	3	4	5	6	7	8
A	A	A	G	C	G	A	C	A
T	G	C	T	T	T	G	T	C
C	C	T	C	A	C	C	A	T
G	T	G	A	G	A	T	G	G

2 THE NOVEL CHAOTIC CIRCUIT BASED ON TWO MEMRISTORS

Recently, a novel four-dimensional chaotic system based on memristors (4D-CMM) is proposed by our team [42] and an image cryptosystem is introduced to generate pseudo-random sequences in this study. The 4D-CMM model is defined as follows:

$$\begin{cases} \dot{i}_L = \dfrac{1}{L} \cdot v - \dfrac{1}{La_1} \cdot \dfrac{i_L}{z_1}, \\ \dot{v} = -\dfrac{1}{C} \cdot i_L - \dfrac{a_2}{C} \cdot z_2 v, \\ \dot{z}_1 = \left(\dfrac{i_L}{a_1 z_1} \right)^2 - b_1, \\ \dot{z}_2 = v^2 - b_2, \end{cases} \tag{1}$$

where a_1, a_2, b_1, b_2, L, and C represent system parameters. The set initial conditions of the system are (−1, 3, 1, and −0.7). There are four kinds of attractors with different parameter values of b_1 are shown in **Figure 1** when $a_1 = 5$, $a_2 = 0.825$, $b_2 = 10$, $L = 0.025$, and $C = 0.025$. In addition, Lyapunov exponents (LEs) and Lyapunov dimensions (LDs) with the different parameter values of b_1 are listed in **Table 1** to prove that the rich dynamic chaotic behavior exists in 4D-CMM. The results present that the 4D-CMM is chaotic and has rich dynamic behavior. Consequently, it can be introduced into the image encryption system as a key stream generator.

3 IMAGE ENCRYPTION ALGORITHM

In this section, the proposed encryption scheme flow is presented, as shown in **Figure 2**, where the structure mainly consists of four parts, namely, initial condition generation, permutation, diffusion, and substitution, where encryption operations are performed at the DNA level The whole encryption process is as follows: a plain image and key are input and the latter is used to design the initial conditions of the chaotic system to initialize the chaotic system (4D-CMM). The three pseudo-random matrices generated by 4D-CMM are used as a key steam element for the encryption process. After that, P was transformed through the permutation procedure. The diffusion and substitution procedure are carried out, in turn, finally. Next, the implementation details in the encryption process will be described.

3.1 4D-CMM Initial Conditions

Based on the Hamming distance, we designed a plain pixel–related 4D-CMM initial conditions in this subsection. The Hamming distance is derived as follows:

$$\begin{cases} H(x, y) = \sum_{i=1}^{n} h(x_i, y_i), \\ \sum_{i=1}^{n} h(x_i, y_i) \begin{cases} 0, x_i = y_i, \\ 1, x_i \neq y_i, \end{cases} \end{cases} \tag{2}$$

where x and y can be interpreted as two sequences and it can be obtained from the bit plain of image P (b_1, b_2, b_3, b_4, b_5,b_6,b_7, and b_8).

Assume that bit plain (b_1, b_2), (b_3, b_4), (b_5, b_6), and (b_7, b_8) represent W_1, W_2, W_3, and W_4, respectively. The Hamming distance $hd_1 = H(W_1, W_2)$, $hd_2 = H(W_1, W_3)$, $hd_3 = H(W_1, W_4)$, and $hd_4 = H(W_2, W_3)$. The initial conditions of 4D-CMM are obtained as follows:

$$\begin{cases} i_L(0) = x_1 + 0.5 \times hd_1 / r, \\ v(0) = y_1 + 0.6 \times hd_2 / r, \\ z_1(0) = x_2 + 0.6 \times hd_3 / r, \\ z_2(0) = y_2 + 0.5 \times hd_4 / r, \end{cases} \tag{3}$$

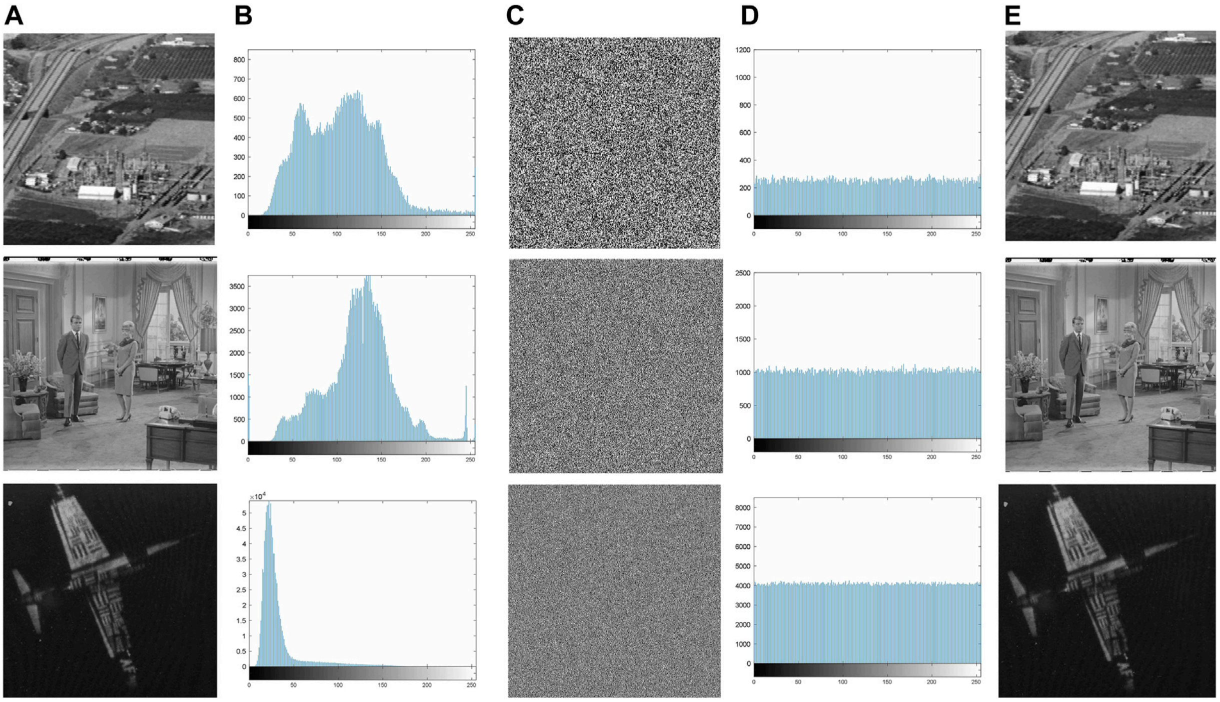


FIGURE 3 | Gray image [43] simulation results:**(A)** plain images, **(B)** histograms for **(A)**, **(C)** encrypted images, **(D)** histograms for **(B)**, and **(E)** decrypted images.

where x_1, x_2, y_1, and y_2 are secret keys and $r = M \times N$ is the number of pixels in the image P. Note that if the pixels in the image change, we can use the Hamming distance to capture the change. This means that two images can have different initial values even if they are slightly different. This feature ensures that the proposed cryptosystem can resist known-plaintext and select-plaintext attacks.

3.2 Permutation Strategy

In the permutation stage, we proposed a two-way confusion strategy to enhance the efficiency and effect of image permutation. In our scheme, the two starting points of permutation start from the head and the tail of the image, respectively, and the permutation process is similar to the floating process of bubbles, which is a nonlinear pixel swapping operation.

In the confusion process, the plain image with a size of $M \times N$ is encoded using DNA coding rule α, and then treated as a one-dimensional array P=(P (1), P (2), . . ., P (4MN)) by scanning the DNA-encoded image P in a Z-like mode. The head (positive direction) and tail (negative direction) of the image P are taken as the starting point of the two exchanges, and the exchange operation is carried out alternately in the positive and negative directions. One element is placed in the final position for each swap operation, which means that the swapped element must be in the position between the two most recently identified elements in the opposite direction. Assume that T is the position of the current element in positive exchange and T_0 represents the position to be swapped one, T_0 is obtained according to **Eqs 4**, **5**.

$$T_0 = T + kn(T), \tag{6}$$

$$kn(T) = mod\left(floor\left(abs\left(un(T) \times 10^{12}\right)\right), 4MN - T\right), \tag{7}$$

where $kn(T)$ is the key stream element used to generate confusion position T and $un = [i_L$, v, z_1, and $z_2]$ is the chaotic sequence by iterating **Eq. 1** MN times. The element swapping procedure is performed according to **Eq. 6**.

$$\begin{cases} C(X) = P(T_0) = P(T + kn(T)), \\ P(T_0) = P(T), \end{cases} \tag{8}$$

where C(X) is the confused element. Analogously, $R = MN\text{-}T$ and R_0 represent the current element and the swapped one in the negative direction. The confused element is obtained by **Eq. 7**. In addition, it can get relaxed and can be concluded that $T + R = 4\,MN$.

$$\begin{cases} R_0 = R + kn(R), \\ C(R) = C(4MN - T) = P(R_0), \\ P(R_0) = P(R), \\ kn(R) = mod\left(floor\left(abs\left(un(R) \times 10^{6}\right)\right), 4MN - R\right). \end{cases} \tag{9}$$

When the positive and negative scrambling operations are performed at the same time, the scrambled image C is obtained after 2MN operations.

3.3 Diffusion Strategy

In the diffusion phase, we need to change the pixel value and transfer the current pixel value to other pixel values as much as possible to improve the concealment of the pixel. In our scheme,

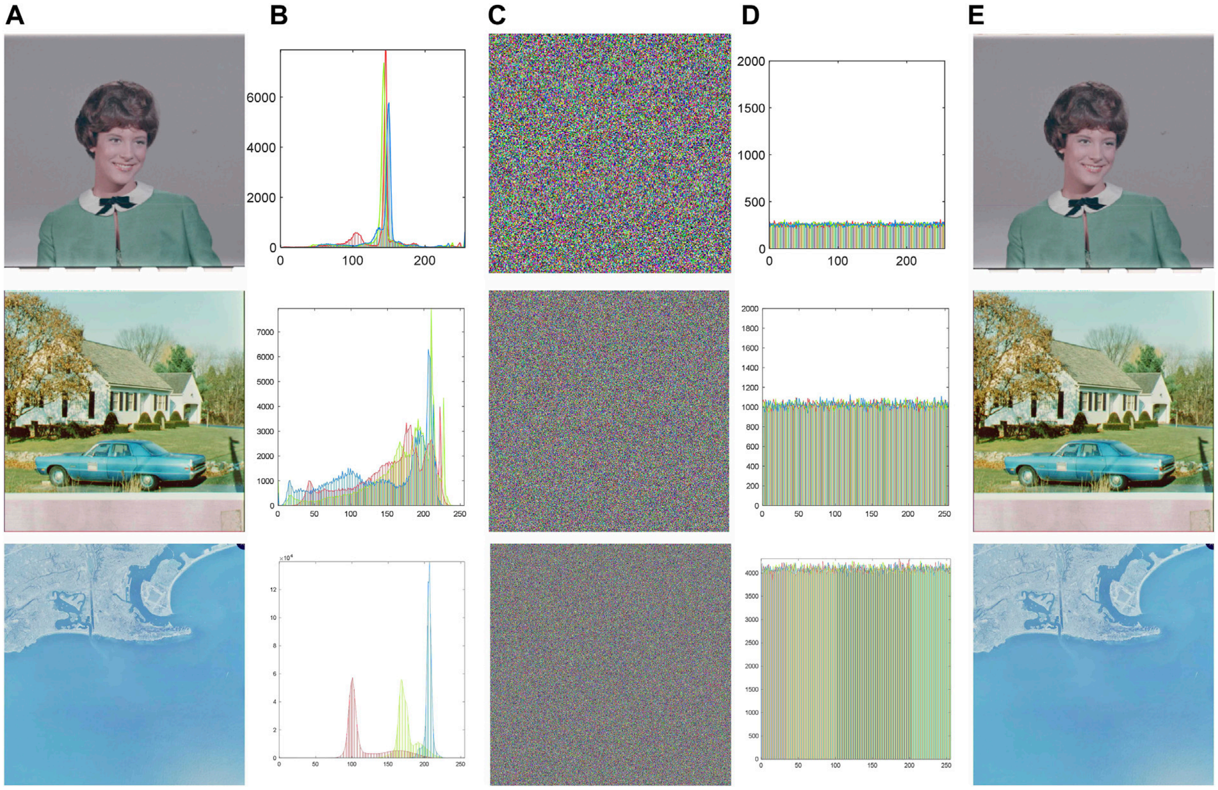


FIGURE 4 | Color image [43] simulation results: **(A)** plain images, **(B)** histograms for **(A)**, **(C)** encrypted images, **(D)** histograms for **(B)**, and **(E)** decrypted images.

TABLE 4 | Key during encryption set.

Item	Value
Hamming distance	hd_1, hd_2, hd_3, and hd_4
Chaotic system parameters	x_1, x_2, y_1, y_2, a_1, a_2, b_1, b_2, L, and C
DNA encoding/decoding rules	α and β

DNA XOR operations are introduced first and its calculation formula is shown as follows:

$$S(i) = D(i) \oplus K_1(i) \oplus S(i-1), \tag{10}$$

where D(i) represents the pixel to be processed currently, and $K_1(i)$ represents the key stream, which can be deduced from **Eq. 9**. S(i) represents the current modified pixel value, and S (i-1) represents the previous modified pixel value. When i = 1, S (0) = D(n), and $n = MN$.

$$K_1(i) = \text{mod}(un, 256) + 1. \tag{11}$$

3.4 Substitution Strategy

In order to improve the security of ciphertext, we designed a new DNA mutation rule based on the triploid mutation, which is used for the diffusion result. In triploid organisms, one DNA strand can replicate three identical DNA strands. Taking advantage of this property, new types of DNA mutations can be designed to obtain mutated single strands by adding the binary representations of the three DNAs. Assume that the current DNA single strand is "ACGT," and its binary representation is "00100111" using the coding rule 2 in **Table 2**. After DNA replication, three identical DNA strands "ACGT," "ACGT," and "ACGT" are generated. Then the mutated DNA single strand obtained is "ACTG." Similarly, when DNA coding rule 4 is used, DNA represented by "00100111″ is "TCGA", and the mutated DNA single strand obtained is "GCTA." It is pointed out that eight kinds of DNA mutation rules can be obtained under eight kinds of DNA coding rules as shown in **Table 3**, and they are in line with the principle of base complementary pairing. Note that for different four bases, triploid mutation changes only two bases at a time, but the mutated bases are represented differently under different coding rules. As a result, it is an efficient way to mutate.

The DNA mutation based on triploid is carried out as follows:

$$\kappa(i) = \text{mod}(K_1(i), 8) + 1, \tag{12}$$

where $\kappa(i)$ is the selected DNA mutation rule, and $K_1(i)$ is the key stream. After mutation was completed, the cipher E image was obtained using decoding rule β.

4 SIMULATION RESULTS

In this section, a variety of images are selected and they are mainly from the USC-SIPI database [43]. The simulation

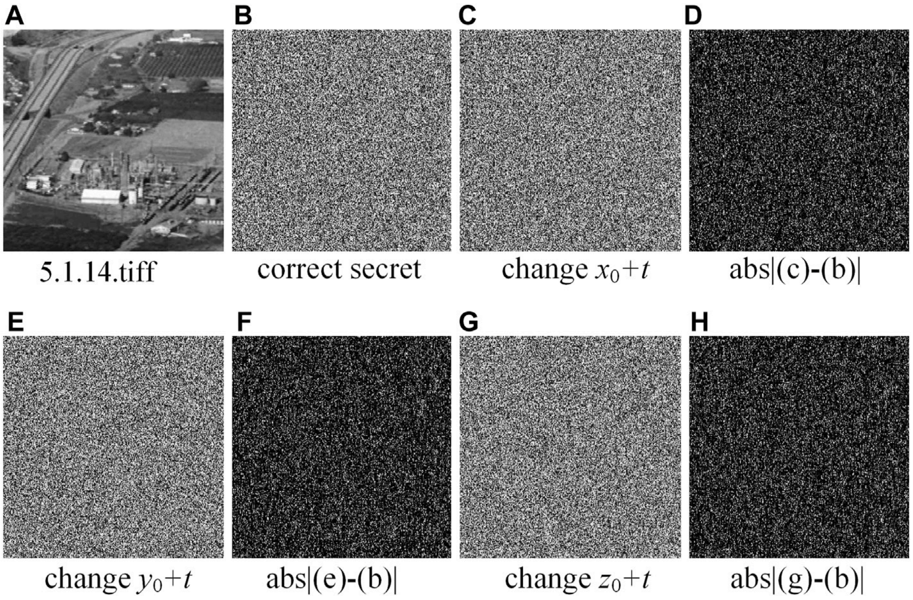


FIGURE 5 | Key sensitivity test in the ciphering stage [43].

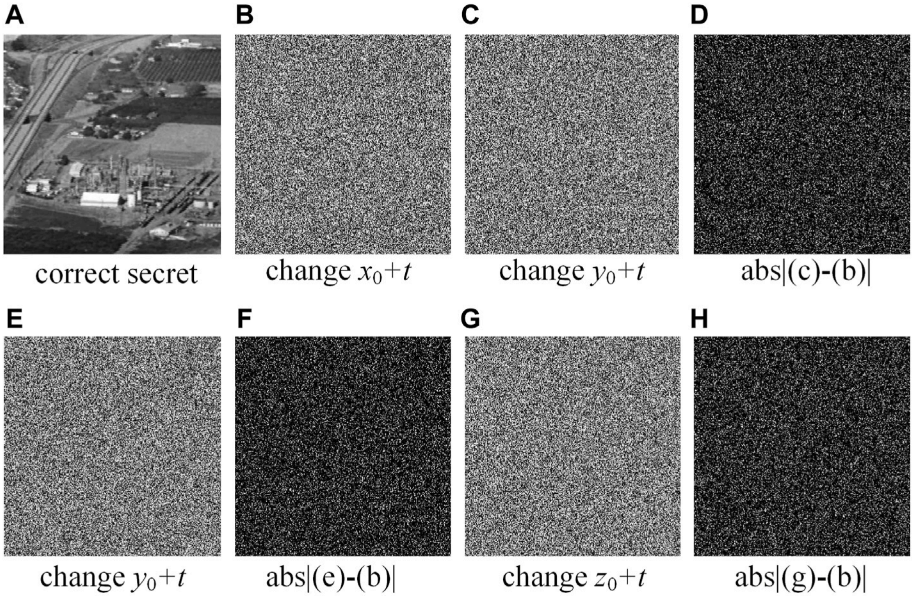
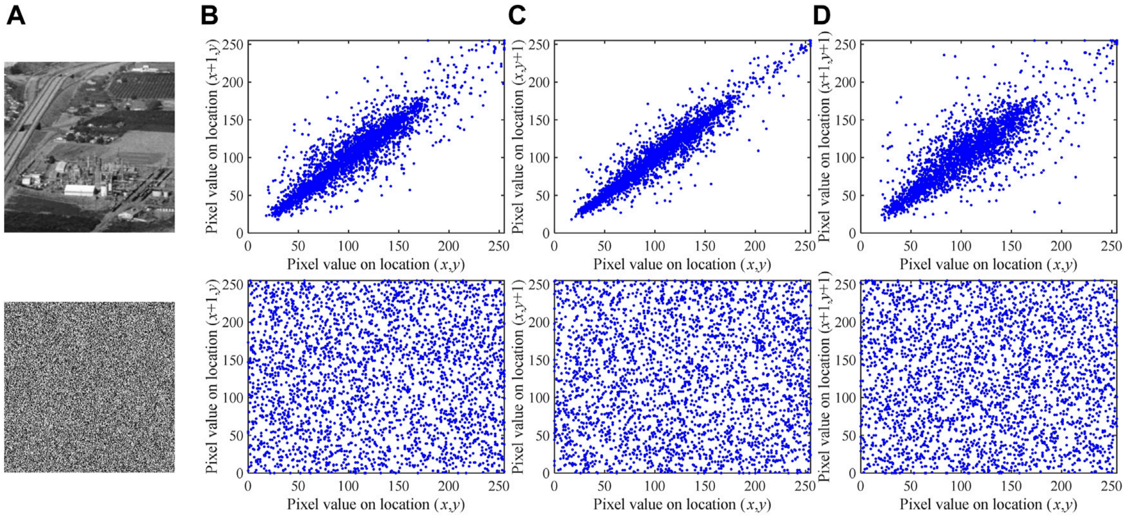


FIGURE 6 | Key sensitivity test in the deciphering stage [43].

environments are as follows: Intel Core i5-4210U CPU 1.70 GHz, Memory 4.00 G, MATLAB 2019a, and Windows 10 operation platform.

An efficient cryptographic system should be able to meet the security requirements of different types of images in secure communication. The grayscale images with different sizes are shown in **Figure 3** [43], in which the first column is the original image, the second column is the histogram of the original image, the third column is the cipher, the fourth column is the histogram of the cipher image, and the fifth is the restored image. One can observe that after all the original images are encrypted, their ciphertext images are noisy and chaotic without any visual information, which can be proved by the histogram changes of the original image during the encryption process. In addition, the original image can be accurately restored when decrypted with the correct key, as shown in **Figure 3E**. In addition, we have also conducted experiments on color images, and the satisfactory results are shown in **Figure 4** [43].



5.1.14.tiff and its cipher horizontal direction vertical direction diagonal direction

FIGURE 7 | Correlation test of image 5.1.14.tiff [43] and its cipher in three directions.

TABLE 5 | Correlation of adjacent pixels in different cipher images.

Image	Size	Correlation		
		H	V	D
5.1.09	256 × 256	0.0017	–0.0025	0.0010
5.1.10	256 × 256	–0.0034	–0.0023	0.0067
5.1.11	256 × 256	–0.0012	–0.0051	–0.0008
5.1.12	256 × 256	–0.0047	0.0053	–0.0035
5.1.13	256 × 256	0.0019	0.0028	–0.0069
5.1.14	256 × 256	0.0079	–0.0009	0.0009
5.2.08	512 × 512	0.0029	0.0011	–0.0005
5.2.09	512 × 512	–0.0037	–0.0016	0.0008
5.2.10	512 × 512	–0.0016	0.0018	–0.0015
5.3.01	1,024 × 1,024	0.0009	–0.0006	–0.0010
5.3.02	1,024 × 1,024	0.0006	–0.0013	–0.0021
7.2.01	1,024 × 1,024	0.0009	–0.0010	0.0007

TABLE 7 | Information entropy (IE) and local entropy (LSE) with different images.

Image	Size	IE	LSE	
			α = 0.001	
5.1.09	256 × 256	7.9971	7.8994	—
5.1.10	256 × 256	7.9971	7.9009	—
5.1.11	256 × 256	7.9970	7.9022	Pass
5.1.12	256 × 256	7.9972	7.8999	—
5.1.13	256 × 256	7.9976	7.9032	Pass
5.1.14	256 × 256	7.9976	7.9013	—
5.2.08	512 × 512	7.9993	7.9025	Pass
5.2.09	512 × 512	7.9993	7.9029	Pass
5.2.10	512 × 512	7.9992	7.9019	Pass
5.3.01	1,024 × 1,024	7.9998	7.9030	Pass
5.3.01	1,024 × 1,024	7.9998	7.9019	Pass
7.2.01	1,024 × 1,024	7.9998	7.9024	Pass

TABLE 6 | Correlation comparison with different scheme results using the Lena image.

Lena	Proposed	Reference [5]	Reference [12]	Reference [38]	Reference [39]
Horizontal	0.0045	0.0090	–0.00007	0.0538	–0.0016
Vertical	–0.0048	0.0010	–0.0024	0.0389	0.0057
Diagonal	0.0002	0.0013	–0.0019	0.0307	–0.0189

Information entropy.

5 SECURITY ANALYSES

5.1 Key Space Analysis

The key space consists of all the legal key combinations that can be used in the cryptosystem for encryption/decryption. A secure cryptographic system should have a large enough key space to preclude illegal decryption by implementing the brute-force attack. **Table 4** lists all the keys that appear in the proposed cryptosystem. When the precision of the computer reaches 10^{15}, the key space of the proposed cryptosystem is about $10^{15\times16} \approx 2^{797}$, which is much larger than the key space 2^{100} recommended by cryptography experts. Accordingly, the proposed cryptographic system can adequately cope with violent attacks.

5.2 Key Sensitivity

In general, key sensitivity needs to be evaluated through both encryption and decryption processes. In the encryption process, first we get the cipher using the key K and then compare it with the cipher using the key K with a weakly change $t = 10^{-15}$. In the decryption process, we also make the same adjustment to the decryption key K. In the decryption process, we also make the same adjustments to the decryption key K, and compare the differences between different decoded images. **Figure 5** [43] and

TABLE 8 | Comparison results of IE values with different scheme.

Image	Size	Proposed	Reference [5]	Reference [12]	Reference [24]	Reference [37]	Reference [38]
Lena	512 × 512	7.9994	7.9993	7.9993	7.9993	7.9994	7.9992

Figure 6 [43] show the key sensitivity test results in the encryption and decryption process, respectively. According to the difference in images, the corresponding images of different keys differ greatly, which enables our scheme to pass the key sensitivity test.

5.3 Correlation of Two Adjacent Pixels

Correlation analysis is a branch of statistical analysis, which is a quantitative analysis of the relationship between adjacent pixels in an image. The correlation r_{xy} can be derived by the following formulas:

$$r_{xy} = \frac{\operatorname{cov}(x, y)}{\sqrt{D(x)D(y)}}, \tag{13}$$

$$\operatorname{cov}(x, y) = E\{[x - E(x)][y - E(y)]\}, \tag{14}$$

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i, \tag{15}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} [x_i - E(x)]^2. \tag{16}$$

Figure 7 [43] presents 3,000 pairs of pixel values that are taken from the three directions of the plain and its cipher, namely, horizontal (H), vertical (V), and diagonal (D) directions, which constitute the distribution of coordinate points in the rectangular coordinate system. Meanwhile, we have also calculated the correlation in three channels, namely, red, green, and blue and the comparison results are presented in **Table 5**. Compared with Refs. [5, 12, 38, 39] in **Table 6**, our algorithm can more efficiently de-correlate the pixel correlation.

Information entropy (IE) can be used to measure the degree of pixel clutter and it is calculated by the following formula:

$$H(m) = -\sum_{i=0}^{255} p(m_i)\log_2 p(m_i). \tag{17}$$

For an ideal cryptographic system, the degree to which the IE calculated by the aforementioned formula approaches 8 can be used to evaluate the security of the system.

In addition, local Shannon entropy (LSE) is often used in recent years and its calculation formula is deduced as follows:

$$H_{k,T_b}(s) = \sum_{i=1}^{b} \frac{H(s_i)}{b}, \tag{18}$$

where the image is divided into b blocks with size T_b and $H(s)$ is the current block IE. When significance level $\alpha = 0.001$, IE should fall into the intervals (7.901515698 and 7.903422936).

TABLE 9 | Running time compared to the relevant encryption systems (unit: second).

Speed	Proposed	Reference [34]	Reference [38]	Reference [39]
Time (s)	1.80	5.23	1.82	2.21

Table 7 lists the IE and LSE values of different size images. The IE values of all images approach 8 and the LSE test is passed by almost all images. In addition, we implemented the algorithm comparison using image Lenna, and the compared results in **Table 8** fully certify that highly chaotic cipher images can be obtained by our algorithm.

5.4 Speed Analysis

In real-time communications, the speed at which algorithms process images is critical. A superior cryptographic system should not only have high security, but also have fast encryption and decryption ability. In our algorithm, the implementation of each scramble operation can simultaneously confuse two pixels, which come from the front and rear of the image. The time complexity is reduced from O (4*MN*) to O (2*MN*), so the running efficiency of our algorithm is improved compared with the scrambling operation of a single pixel. In addition, only the two bases are mutated during the process of mutation and the two bases can represent any two bits under different coding rules, which demonstrate the efficiency of mutation operation to a certain extent.

In the test of running speed, standard gray image Lena with the size of 512 × 512 is involved and 100 tests are carried out on it. **Table 9** shows the average of the test results through our encryption scheme and compares it with Refs. [34, 38, 39]. The experimental results show that the proposed scheme has an excellent running speed.

6 CONCLUSION

In this study, we introduced a memristor chaotic system for a chaotic-based secure communication field. The rich dynamic behavior is analyzed prior to encryption. In the process of encryption, the initial conditions of the chaotic system designed by the Hamming distance are used to enhance the plaintext correlation of cryptosystem and provide a secure key stream for encryption. In the permutation stage, we used a chaotic sequence to generate two random positions of pixels to be swapped at the beginning and end of the image to shuffle the pixel, which can enhance the pixel's confusion effect and permutation speed to a certain extent. In the diffusion stage, DNA XOR operation is implemented followed by DNA mutation is introduced to generate high-quality ciphertext images. Simulation experiments and security analysis fully show that our scheme not only ensures security but also has high efficiency.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding authors.

AUTHOR CONTRIBUTIONS

YS designed and carried out experiments, analyzed data, and wrote the manuscript. FB provided the theoretical guidance for this manuscript. LW carried out the experiment. HJ improved the algorithm. All authors reviewed the manuscript.

FUNDING

This work was supported by the Dalian Young Stars of Science and Technology Project (NO.2021RQ088).

REFERENCES

1. Talhaoui MZ, Wang X, Talhaoui A. A New One-Dimensional Chaotic Map and its Application in a Novel Permutation-Less Image Encryption Scheme. *Vis Comput* (2020) 37(7):1757–68. doi:10.1007/s00371-020-01936-z
2. Chai X, Zhi X, Gan Z, Zhang Y, Chen Y, Fu J. Combining Improved Genetic Algorithm and Matrix Semi-tensor Product (STP) in Color Image Encryption. *Signal Process.* (2021) 183:108041. doi:10.1016/j.sigpro.2021.108041
3. Movafegh Ghadirli H, Nodehi A, Enayatifar R. Color Image DNA Encryption Using mRNA Properties and Non-adjacent Coupled Map Lattices. *Multimed Tools Appl* (2020) 80(6):8445–69. doi:10.1007/s11042-020-10014-4
4. Khan JS, Ahmad J, Ahmed SS, Siddiqa HA, Abbasi SF, Kayhan SK. DNA Key Based Visual Chaotic Image Encryption. *Ifs* (2019) 37(2):2549–61. doi:10.3233/JIFS-182778
5. Mondal B, Kumar P, Singh S. A Chaotic Permutation and Diffusion Based Image Encryption Algorithm for Secure Communications. *Multimed Tools Appl* (2018) 77(23):31177–98. doi:10.1007/s11042-018-6214-z
6. Wang X, Su Y. Image Encryption Based on Compressed Sensing and DNA Encoding. *Signal Processing: Image Commun* (2021) 95:116246. doi:10.1016/j.image.2021.116246
7. Ye G, Pan C, Dong Y, Shi Y, Huang X. Image Encryption and Hiding Algorithm Based on Compressive Sensing and Random Numbers Insertion. *Signal Process.* (2020) 172:107563. doi:10.1016/j.sigpro.2020.107563
8. Patro KAK, Prasanth Jagapathi Babu M, Pavan Kumar K, Acharya B. Dual-Layer DNA-Encoding-Decoding Operation Based Image Encryption Using One-Dimensional Chaotic Map. *Adv Data Inf Sci Springer* (2020) 94 67–80. doi:10.1007/978-981-15-0694-9_8
9. Wang X, Su Y, Liu L, Zhang H, Di S. Color Image Encryption Algorithm Based on Fisher-Yates Scrambling and DNA Subsequence Operation. *Vis Comput* (2021). doi:10.1007/s00371-021-02311-2
10. Luo Y, Zhou R, Liu J, Qiu S, Cao Y. An Efficient and Self-Adapting Colour-Image Encryption Algorithm Based on Chaos and Interactions Among Multiple Layers. *Multimed Tools Appl* (2018) 77(20):26191–217. doi:10.1007/s11042-018-5844-5
11. Wu X, Wang K, Wang X, Kan H, Kurths J. Color Image DNA Encryption Using NCA Map-Based CML and One-Time Keys. *Signal Process.* (2018) 148:272–87. doi:10.1016/j.sigpro.2018.02.028
12. Chai X, Fu J, Zhang J, Han D, Gan Z. Exploiting Preprocessing-Permutation-Diffusion Strategy for Secure Image Cipher Based on 3D Latin Cube and Memristive Hyperchaotic System. *Neural Comput Applic* (2021) 33(16):10371–402. doi:10.1007/s00521-021-05797-y
13. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Du S. FPGA Implementation and Image Encryption Application of a New PRNG Based on a Memristive Hopfield Neural Network with a Special Activation Gradient. *Chin Phys. B* (2022) 31(2):020505. doi:10.1088/1674-1056/ac3cb2
14. Zhang X, Li C, Dong E, Zhao Y, Liu Z. A Conservative Memristive System with Amplitude Control and Offset Boosting. *Int J Bifurcation Chaos* (2022) 32(04):2250057. doi:10.1142/S0218127422500572
15. Li Y, Li C, Zhao Y, Liu S. Memristor-type Chaotic Mapping. *Chaos* (2022) 32(2):021104. doi:10.1063/5.0082983
16. Li Y, Li C, Zhang S, Chen GR, Zeng Z. A Self-Reproduction Hyperchaotic Map with Compound Lattice Dynamics. *IEEE Trans Ind Electron* (2022) 69(10):10564–10572. doi:10.1109/TIE.2022.3144592
17. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Jin J, et al. Design and FPGA Implementation of a Pseudo-random Number Generator Based on a Hopfield Neural Network under Electromagnetic Radiation. *Front Phys* (2021) 9:690651. doi:10.3389/fphy.2021.690651
18. Yu F, Shen H, Zhang ZZZ, Huang Y, Cai S, Du S. A New Multi-Scroll Chua's Circuit with Composite Hyperbolic tangent-cubic Nonlinearity: Complex Dynamics, Hardware Implementation and Image Encryption Application. *Integration* (2021) 81:71–83. doi:10.1016/j.vlsi.2021.05.011
19. Li C, Yang Y, Yang X, Zi X, Xiao F. A Tristable Locally Active Memristor and its Application in Hopfield Neural Network. *Nonlinear Dyn* (2022) 108:1697–717. doi:10.1007/s11071-022-07268-y
20. Li C, Li H, Xie W, Du J. A S-type Bistable Locally Active Memristor Model and its Analog Implementation in an Oscillator Circuit. *Nonlinear Dyn* (2021) 106(1):1041–58. doi:10.1007/s11071-021-06814-4
21. Yu F, Kong X, Chen H, Yu Q, Cai S, Huang Y, et al. A 6D Fractional-Order Memristive Hopfield Neural Network and its Application in Image Encryption. *Front Phys* (2022) 10:109. doi:10.3389/fphy.2022.847385
22. Ma X, Mou J, Liu J, Ma C, Yang F, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor-Memcapacitor. *Nonlinear Dyn* (2020) 100(3):2859–76. doi:10.1007/s11071-020-05601-x
23. Li X, Mou J, Cao Y, Banerjee S. An Optical Image Encryption Algorithm Based on a Fractional-Order Laser Hyperchaotic System. *Int J Bifurcation Chaos* (2022) 32(03):2250035. doi:10.1142/S0218127422500353
24. Hua Z, Jin F, Xu B, Huang H. 2D Logistic-Sine-Coupling Map for Image Encryption. *Signal Process.* (2018) 149:148–61. doi:10.1016/j.sigpro.2018.03.010
25. Li X, Mou J, Xiong L, Wang Z, Xu J. Fractional-order Double-Ring Erbium-Doped Fiber Laser Chaotic System and its Application on Image Encryption. *Opt Laser Techn* (2021) 140(3):107074. doi:10.1016/j.optlastec.2021.107074
26. Hua Z, Zhu Z, Yi S, Zhang Z, Huang H. Cross-plane Colour Image Encryption Using a Two-Dimensional Logistic Tent Modular Map. *Inf Sci* (2021) 546:1063–83. doi:10.1016/j.ins.2020.09.032
27. Gan Z-h., Chai X-l., Han D-j., Chen Y-r. A Chaotic Image Encryption Algorithm Based on 3-D Bit-Plane Permutation. *Neural Comput Applic* (2018) 31(11):7111–30. doi:10.1007/s00521-018-3541-y
28. Li CL, Zhou Y, Li HM. Image Encryption Scheme with Bit-Level Scrambling and Multiplication Diffusion[J]. *Multimedia Tools Appl* (2021) 80:18479–18501. doi:10.1007/s11042-021-10631-7
29. Zhou Y, Li C, Li W, Li H, Feng W, Qian K. Image Encryption Algorithm with circle index Table Scrambling and Partition Diffusion. *Nonlinear Dyn* (2021) 103:2043–61. doi:10.1007/s11071-021-06206-8
30. Zhou S, Wang X, Wang M, Zhang Y. Simple Colour Image Cryptosystem with Very High Level of Security. *Chaos, Solitons & Fractals* (2020) 141:110225. doi:10.1016/j.chaos.2020.110225
31. Chen J-x., Zhu Z-l., Fu C, Yu H, Zhang L-b. A Fast Chaos-Based Image Encryption Scheme with a Dynamic State Variables Selection Mechanism. *Commun Nonlinear Sci Numer Simulation* (2015) 20(3):846–60. doi:10.1016/j.cnsns.2014.06.032

32. Xiong L, Yang F, Mou J, An X, Zhang X. A Memristive System and its Applications in Red-Blue 3D Glasses and Image Encryption Algorithm with DNA Variation. *Nonlinear Dyn* (2022) 107(3):2911–33. doi:10.1007/s11071-021-07131-6
33. Yang F, An X, Xiong L. A New Discrete Chaotic Map Application in Image Encryption Algorithm. *Phys Scr* (2022) 97(3):035202. doi:10.1088/1402-4896/ac4fd0
34. Sha Y, Cao Y, Yan H, Gao X, Mou J. An Image Encryption Scheme Based on IAVL Permutation Scheme and DNA Operations. *IEEE Access* (2021) 9: 96321–36. doi:10.1109/access.2021.3094563
35. Gao X, Mou J, Xiong L, Sha Y, Yan H, Cao Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn* (2022) 108(1):613–36. doi:10.1007/s11071-021-07192-7
36. Gao X, Mou J, Banerjee S, Cao Y, Xiong L, Chen X. An Effective Multiple-Image Encryption Algorithm Based on 3D Cube and Hyperchaotic Map. *J King Saud Univ - Comp Inf Sci* (2022) 34(4):1535–51. doi:10.1016/j.jksuci.2022.01.017
37. Talhaoui MZ, Wang X, Midoun MA. Fast Image Encryption Algorithm with High Security Level Using the Bülban Chaotic Map. *J Real-time Image Proc* (2020) 18(1):85–98. doi:10.1007/s11554-020-00948-1
38. Yuan X, Zhang L, Chen J, Wang K, Zhang D. Multiple-image Encryption Scheme Based on Ghost Imaging of Hadamard Matrix and Spatial Multiplexing. *Appl Phys B* (2019) 125(9):125. doi:10.1007/s00340-019-7286-9
39. Ye H-S, Zhou N-R, Gong L-H. Multi-image Compression-Encryption Scheme Based on Quaternion Discrete Fractional Hartley Transform and Improved Pixel Adaptive Diffusion. *Signal Process.* (2020) 175:107652. doi:10.1016/j.sigpro.2020.107652
40. Xiong L, Zhang X, Teng S, Qi L, Zhang P. Detecting Weak Signals by Using Memristor-Involved Chua's Circuit and Verification in Experimental Platform. *Int J Bifurcation Chaos* (2020) 30(13):2050193. doi:10.1142/S021812742050193X
41. Zhou S, Wang X, Zhang Y. A Novel Image Encryption Cryptosystem Based on True Random Numbers and Chaotic Systems. *Multimedia Syst* (2022) 1–18. doi:10.1007/s00530-021-00803-8
42. Ma X, Mou J, Xiong L, Banerjee S, Cao Y, Wang J. A Novel Chaotic Circuit with Coexistence of Multiple Attractors and State Transition Based on Two Memristors. *Chaos, Solitons & Fractals* (2021) 152:111363. doi:10.1016/j.chaos.2021.111363
43. USC-SIPI. *USC-SIPI Image Database*. Los Angeles, CA: Signal and Image Processing Institute, University of Southern California (2013). [EB/OL]. Available at: http://sipi.usc.edu/database/ (Accessed February, 2022).

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Physics

ORIGINAL RESEARCH
published: 27 May 2022
doi: 10.3389/fphy.2022.912229

Detecting Boosting Weak Signal *via* A Meminductive Multistable Chaotic System

Baolin Kang[1] **and Wenjie Qin**[2]*

[1]School of Mathematics and Information Sciences, Anshan Normal University, Anshan, China, [2]School of Mathematics and Computer Science, Yunnan Minzu University, Kunming, China

In this paper, we rebuild a new meminductive chaotic circuit model based on a Wien-bridge oscillator. Due to the extreme multistability of the meminductive system, it can produce the phenomenon of many infinitely coexisting attractors. Systems that spontaneously produce coexisting oscillations are relatively rare in the study of meminductive circuit systems and are well suited as pseudo-random number generators (PRNG). In this study, a new weak signal detection model is established based on the proposed meminductive system. This detection model can detect boosting weak signals with different amplitudes. The trajectory of objects can be predicted effectively by the boosting line spectrum in the frequency domain. The experimental result shows the feasibility through which the meminductive multistable system is applied in the detection field.

Keywords: chaos, extremely multistability, weak signal detection, meminductor, equivalent circuit

Edited by:
Jun Mou,
Dalian Polytechnic University, China

Reviewed by:
Ning Wang,
Changzhou University, China
Li Xiong,
Hexi University, China

****Correspondence:***
Wenjie Qin
wenjieqin@hotmail.com

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *04 April 2022*
Accepted: *20 April 2022*
Published: *27 May 2022*

Citation:
Kang B and Qin W (2022) Detecting Boosting Weak Signal via A Meminductive Multistable Chaotic System. Front. Phys. 10:912229. doi: 10.3389/fphy.2022.912229

1 INTRODUCTION

Memristor, a nonlinear memory resistor, has been used extensively in recent studies. The concept of memristor first appeared in a paper by Chua [1]. Based on the relationship between electrical physical quantities, Chua outlines that there must be a set of relationships between the charge and the flux. In subsequent studies, Chua's team defined the resistance-like relationship between them as memristor. Until 2008, according to the conceptual model of memristor, HP laboratory manufactured the first nanoscale memristor using TiO2 material [2]. It also confirms the correctness of Chua's prediction. After further scientific certification, some researchers found that the volt-ampere characteristic curve of the memristor is a hysteresis loop due to its nonlinear characteristic [3–8]. Unlike normal volatile resistors, memristors are efficient at retaining energy, i.e. memristor is a component with a memory function. The memory properties of the memristor are a good match for the resistive components that will be used to build future artificial brains, for which it made headlines in scientific journals [9, 10]. Then, the Ventra team proposed the concept of meminductor [11], which adds better memory features to conventional storage devices.

Research on memristive circuit systems has made great progress, with continuous breakthroughs in memristive hardware [12–15]. The memristor oscillation circuit based on Chua's circuit model has become a hotspot of nonlinear system research. Due to the more nonlinear parts of the memristor, it can often produce complex dynamic behavior and higher sequence complexity. Thus, the memristive dynamical system is widely used in the related fields of physics [16–19], computer science [20–24], and cryptography [25–28]. Ye et al constructed a memristive circuit model based on Wen's bridge oscillating circuit. By replacing a memristor of the original 4D Wen's bridge oscillator, the new circuit equations with a memristor have more nonlinear terms. Based on this, the dynamic behavior of the

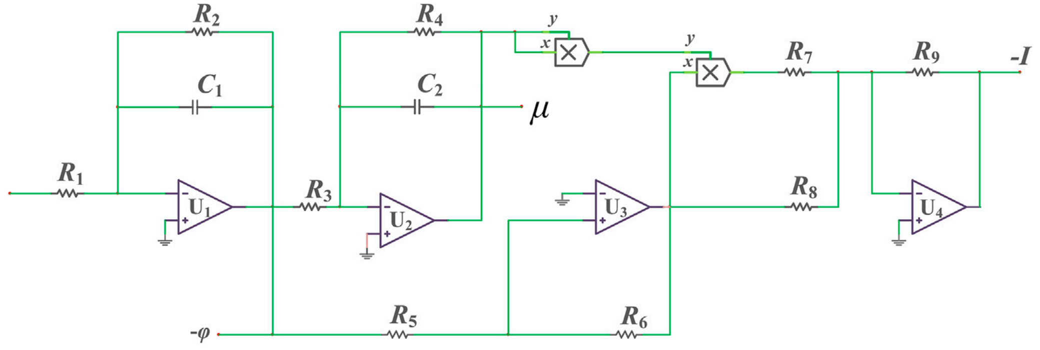


FIGURE 1 | The equivalent circuit of meminductor.

memristive circuit system becomes more complex, and there is even multi-attractor coexistence. There is even a phenomenon of multi-attractor coexistence [29]. Bao et al proposed an inductor-free memristive circuit using linear coupling of a BPF and found many coexisting infinite attractors by using numerical simulation [30]. There is a large amount of research on memristors in the existing literature. However, there are relatively few studies on meminductor, especially on the meminductive multi-stable system. Thus, it is very important to further explore complex dynamic behavior and the application of a memory-sensing system.

In this article, we focus on a novel meminductive chaotic system of extremely multistability and its application in signal detection. The article is organized as follows: in **section 2**, the meminductive equivalent circuit model is proposed. In **section 3**, the meminductive chaotic system model is built. In **section 4**, the proposed meminductive system is applied to weak signal detection. Finally, the chaotic attractors of the meminductive system are implemented by circuit experiment.

2 DESIGN OF MEMINDUCTIVE EQUIVALENT CIRCUIT

Based on the i-φ characteristics of meminductor, we have

$$i(t) = M^{-1}(\mu)\phi(t). \tag{1}$$

in the formula above, $i(t)$ and $\varphi(t)$ are the current and magnetic flux flowing through the meminductor, and $M(\mu)$ reflects the inductive resistance of the meminductor. Then, we get

$$\mu = \int_{-\infty}^{t} \varphi(\tau)d\tau. \tag{2}$$

Based on **Eqs 1**, **2**, the mathematical model of meminductor can be designed as

$$q(\mu) = a_\mu \mu + b_\mu \mu^3. \tag{3}$$

in which, a_μ and b_μ are the positive real numbers, and then we have

$$i(t) = \left(a_\mu + 3b_\mu \mu^2\right)\varphi(t). \tag{4}$$

Based on the mathematical model of the meminductor above, the equivalent circuit model of the meminductor can be designed as shown in **Figure 1**. According to the relationships of all components reflected in **Figure 1**, we can achieve the i-φ characteristic of the meminductor:

$$i(t) = \left(\frac{R_9}{R_8} + \frac{R_9}{R_7}\mu^2\right)\varphi(t). \tag{5}$$

According to **Eqs 4**, **5**, $a_\mu = R_9/R_8$ and $b_\mu = R_9/R_7$. Set the component parameters in **Figure 1** $R_1 = R_3 = R_4 = 200\text{k}\Omega$, $R_5 = R_6 = R_8 = R_9 = 10\text{k}\Omega$, $R_2 = 1\text{M}\Omega$, $R_7 = 3\text{k}\Omega$, and $C_1 = C_2 = 10\text{nF}$. By using the circuit simulation software, as shown in **Figure 2**, we can get the v-i hysteresis loop under different frequencies.

3 CONSTRUCTING A MEMINDUCTIVE CIRCUIT

3.1 Constructing a Meminductive Circuit Based on Wien-Bridge Oscillator

By building an *RLC* oscillation structure based on a Wien-bridge oscillator, a novel meminductive circuit model can be rebuilt. As reflected in **Figure 3**, the new meminductive circuit model has four resistors R_1~R_4, three capacitors, one operational amplifier, two diodes, and one meminductor.

3.2 Mathematical Model

Based on the circuit model reflected in **Figure 7**, we can obtain the circuit equations

$$\begin{cases} C_1\dfrac{du_1}{dt} = \dfrac{R_4 u_1}{R_2 R_3} - \dfrac{u_1}{R_1} - \dfrac{u_2}{R_2} - i_d \\ C_2\dfrac{du_2}{dt} = \dfrac{R_4 u_1}{R_2 R_3} - \dfrac{u_2}{R_2} \\ C_3\dfrac{du_3}{dt} = i_d - L^{-1}(\rho)\varphi \\ \dfrac{d\rho}{dt} = \varphi \\ \dfrac{d\varphi}{dt} = u_3 \end{cases}, \tag{6}$$

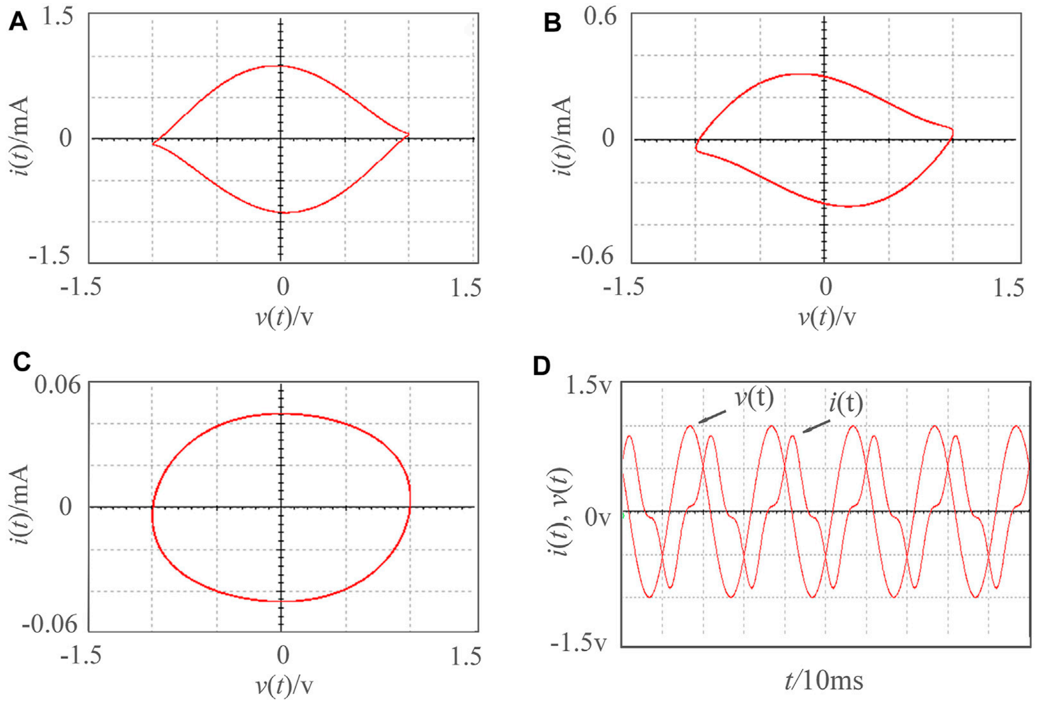


FIGURE 2 | The *v-i* hysteresis loop of a meminductive equivalent circuit under different frequencies. **(A)** $f = 50$ Hz. **(B)** $f = 70$ Hz. **(C)** $f = 180$ Hz. **(D)** waveform of $f = 50$ Hz.

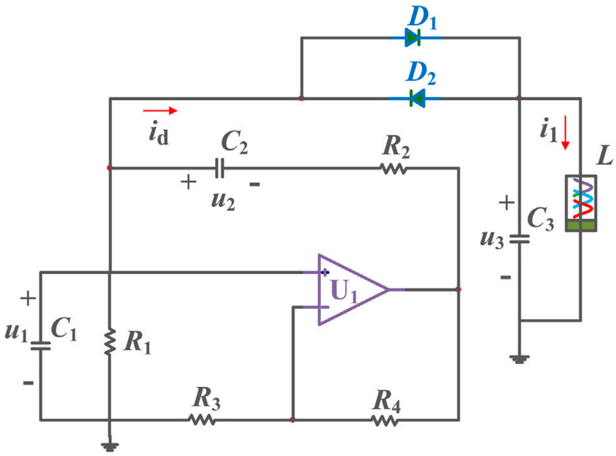


FIGURE 3 | A novel meminductive circuit based on a Wien-bridge oscillator.

in **Eq. 6**, the equations of the diodes can be expressed as

$$i_e = g_e[u_1 - u_3 + 0.5(|u_1 - u_3 - U_e| - |u_1 - u_3 + U_e|)], \quad (7)$$

in which, U_e shows the threshold voltage, g_e is the conductance. Then, let $u_1 = U_e x$, $u_2 = U_e y$, $u_3 = U_e z$, $\varphi = U_e C_1 (R_2 R_3)^{1/2} v$, $\rho = U_{th} C_1^2 R_1 R_2 w$, $i_e = U_e g_e H$, $t = C_1 (R_1 R_2)^{1/2} \tau$, $c=(R_1/R_2)^{1/2}$, $d = R_4/R_3$, $e=(R_1R_2)^{1/2} g_e$, $C_1 = C_2 = C_3$, $a = U_e C_1 R_1 R_2 a_1$, $b = U_e^3 C_1^5 R_1^3 R_2^3 b_1$, $L^{-1}(\rho) = a+3b\rho^2$, we have

$$H = x - z + 0.5(|x - z - 1| - |x - z + 1|). \quad (8)$$

Finally, we can get an equivalent mathematical model

$$\begin{cases} \dot{x} = (cd - 1/c)x - cy - eH \\ \dot{y} = cdx - cy \\ \dot{z} = eH - (a + 3bw^2)v \\ \dot{w} = v \\ \dot{v} = z \end{cases} . \quad (9)$$

According to the mathematical equations of the system (9), the dynamical behavior of the proposed system can be analyzed. When the system parameters are $a = 40$, $b = 10$, $c = 0.6$, $d = 8$, $e = 30$, the initial condition is (1,1,1,0.1,0.01). We can calculate Lyapunov exponents are $Ly_1 = 0.4403$, $Ly_2 = 0.0127$, $Ly_3 = 0$, $Ly_4 = -0.4536$, $Ly_5 = -35.18$. This indicates the proposed system is a hyperchaotic circuit system.

4 THE APPLICATION OF THE MEMINDUCTIVE SYSTEM IN WEAK SIGNAL DETECTION

4.1 The Construction of the Detection System Model

Due to the initial value sensitivity of chaotic systems, the system model may generate resonance signals with some weak signals. The detection model of the weak signal can be built based on the system (9). In the process of signal detection, the test weak signal is often mixed with the noise signals. In particular, some noise

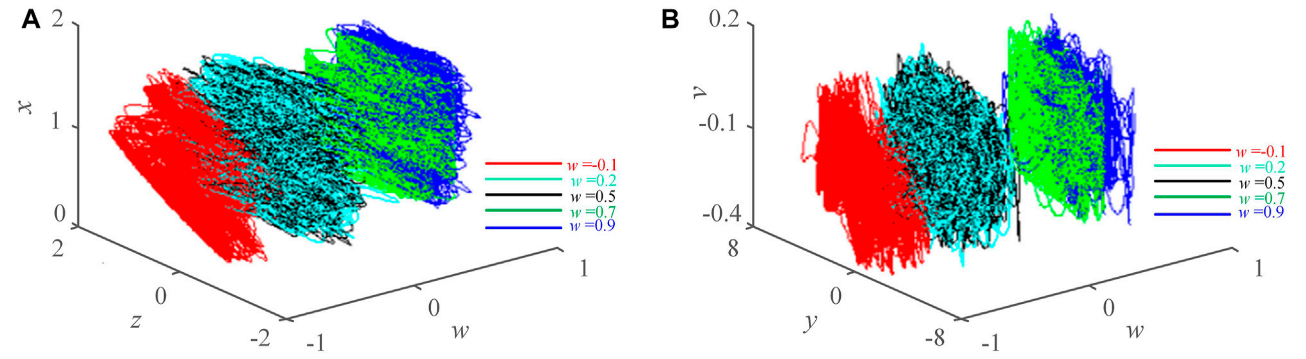


FIGURE 4 | Infinitely coexisting attractors generated by the system (10). **(A)** *x*-*z*-*w* plane. **(B)** *v*-*y*-*w* plane.

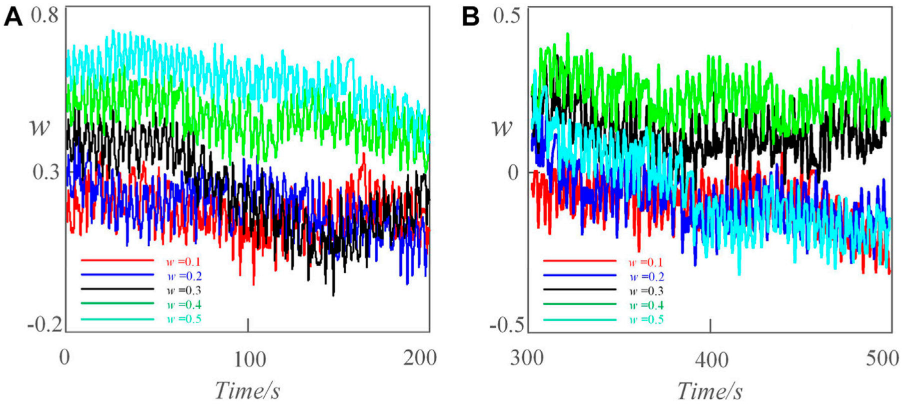


FIGURE 5 | The boosting chaos signals of the time domain. **(A)** The boost signals on *w* direction at 0–200 s. **(B)** The boost signals on *w* direction at 300–500 s.

signals have a close frequency to the test weak signal. The detection model can be expressed as

$$\begin{cases} \dot{x} = (cd - 1/c)x - cy - eH \\ \dot{y} = cdx - cy + K_1 \cos(w_1 t) + K_2 \cos(w_2 t + \varphi) \\ \dot{z} = eH - (a + 3bw^2)v \\ \dot{w} = v \\ \dot{v} = z \end{cases}, \tag{10}$$

where K_1 is the amplitude of the test weak signal, w_1 is its angular frequency. K_2 is the amplitude of the noise signal, w_2 is its angular frequency, and is its initial phase. The newly constructed detection model (10) inherits the extremely multiple stability from the meminductive system (9). Let system parameter $a = 40$, $b = 10$, $c = 0.6$, $d = 8$, $e = 30$, $C_1 = 1$, $C_2 = 2$, $w_1 = 1$, $w_2 = 2$, $\varphi = 0$, and the initial conditions is $(1,1,1,w(0),0.01)$. As shown in **Figure 4**, when $w(0) = -0.1, 0.2, 0.5, 0.7$, and 0.9 respectively, the detection model can produce infinite coexisting attractors. **Figure 5** shows the boosting signals with different amplitudes when $w(0) = 0.1, 0.2, 0.3, 0.4, 0.5$ respectively.

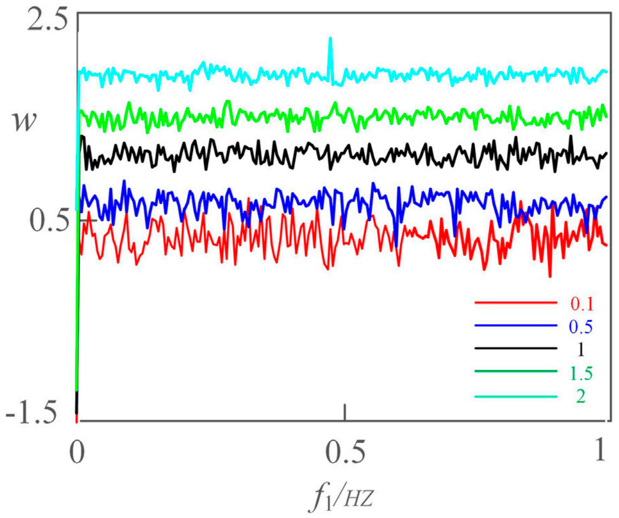


FIGURE 6 | The line spectrum of the boosting weak signal with varying amplitudes.

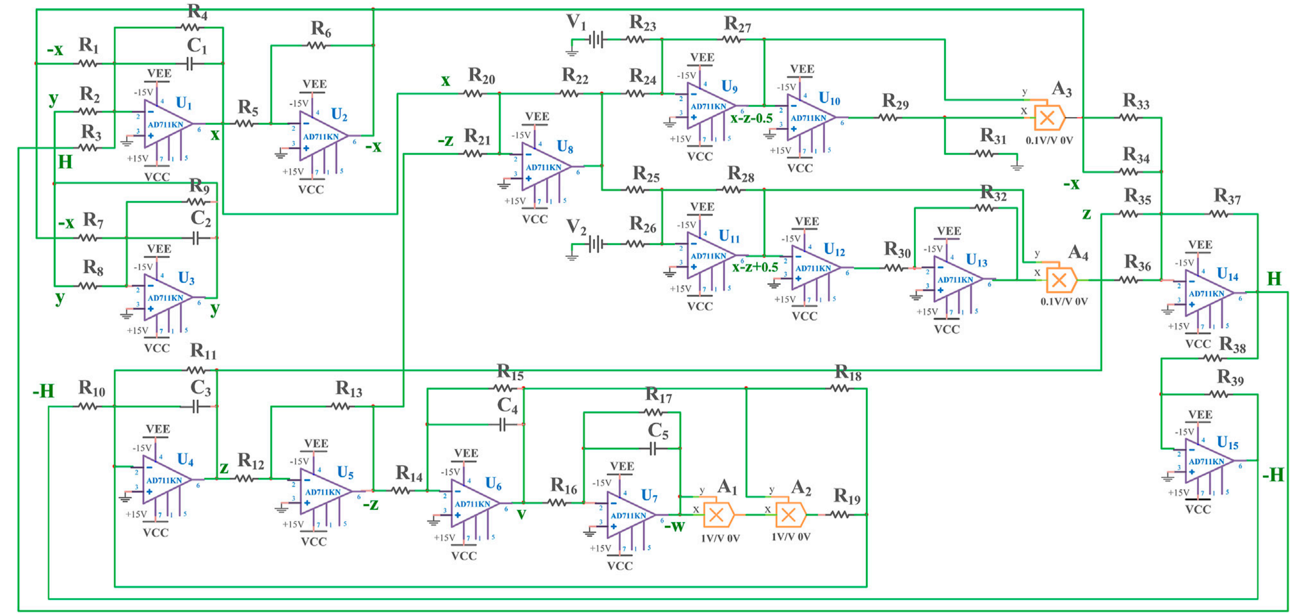


FIGURE 7 | The equivalent circuit of the proposed meminductive circuit.

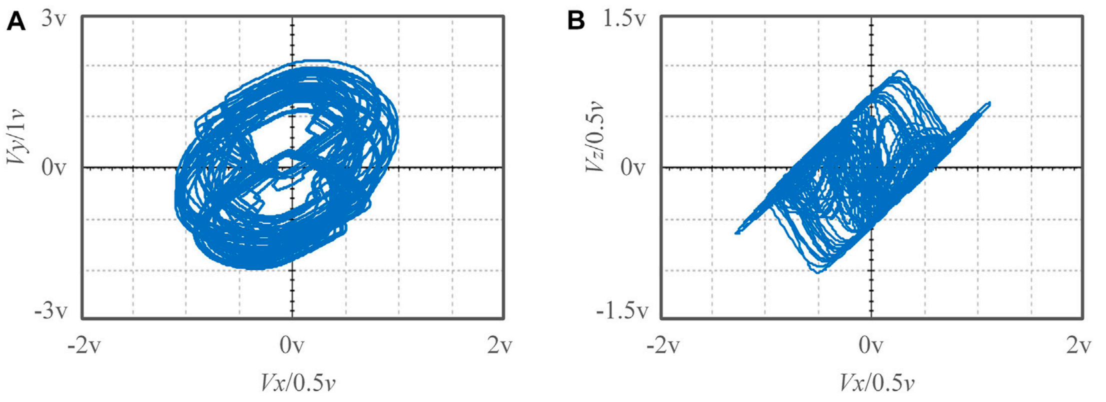


FIGURE 8 | Chaotic attractors in the circuit simulation. **(A)** V_x-V_y plane. **(B)** V_x-V_z plane.

4.2 The Extraction of the Spectral Line in the Frequency Domain

In the weak signal detection field, to understand the trajectory of the object producing the weak signal, the line spectrum in the frequency domain is often extracted. Keeping other parameters unchanged, let the initial value w = 0.1, 0.5, 1, 1.5, and 2 respectively, the line spectrum of the test signal in the frequency domain is reflected in **Figure 6**. As the initial condition w increases constantly, the corresponding amplitude of the sequence is boosting constantly, and the boosting signal can be detected within the frequency range 0–1 Hz. Due to the proposed meminductive system being extremely multistable, the useful signal can be detected within different amplitudes. Even if a detector in the amplitude range fails, the weak signals can also be detected in other amplitude ranges, which increases the fault tolerance of the detection process. These boosting signals show more useful information about the running object. Based on this key information, the trajectory of the object can be predicted.

5 REALIZATION OF THE MEMINDUCTIVE CIRCUIT SYSTEM

Doing the circuit implementation to the system (9). Based on **Eqs 8**, **9**, we have

TABLE 1 | The circuit parameters in the circuit experiment.

Resistance	Value	Resistance	Value	Resistance	Value	Others	Value
R_1	31.91 kΩ	R_{14}	100 kΩ	R_{27}	100 kΩ	C_1	100nF
R_2	166.66 kΩ	R_{15}	1 MΩ	R_{28}	100 kΩ	C_2	100nF
R_3	3.333 kΩ	R_{16}	100 kΩ	R_{29}	29 kΩ	C_3	100nF
R_4	1 MΩ	R_{17}	1 MΩ	R_{30}	129 kΩ	C_4	100nF
R_5	100 kΩ	R_{18}	2.5 kΩ	R_{31}	100 kΩ	C_5	100nF
R_6	100 kΩ	R_{19}	188.33 kΩ	R_{32}	100 kΩ	V_1	0.5 V
R_7	20.83 kΩ	R_{20}	100 kΩ	R_{33}	200 kΩ	V_2	0.5 V
R_8	166.66 kΩ	R_{21}	100 kΩ	R_{34}	100 kΩ	A_1	1
R_9	1 MΩ	R_{22}	100 kΩ	R_{35}	100 kΩ	A_2	1
R_{10}	3.333 kΩ	R_{23}	100 kΩ	R_{36}	200 kΩ	A_3	0.1
R_{11}	1 MΩ	R_{24}	100 kΩ	R_{37}	100 kΩ	A_4	0.1
R_{12}	100 kΩ	R_{25}	100 kΩ	R_{38}	100 kΩ		
R_{13}	100 kΩ	R_{26}	100 kΩ	R_{39}	100 kΩ		

$$\begin{cases} \dfrac{dX}{dT} = \dfrac{1}{R_4C_1}\left(\dfrac{R_4}{R_1}X - \dfrac{R_4}{R_2}Y - \dfrac{R_4}{R_3}H\right) \\ \dfrac{dY}{dT} = \dfrac{1}{R_9C_2}\left(\dfrac{R_9}{R_7}X - \dfrac{R_9}{R_8}Y\right) \\ \dfrac{dZ}{dT} = \dfrac{1}{R_{11}C_3}\left(-\dfrac{R_{11}}{R_{19}}W^2V - \dfrac{R_{11}}{R_{18}}V + \dfrac{R_{11}}{R_{10}}H\right), \\ \dfrac{dW}{dT} = \dfrac{R_{17}}{R_{16}C_5}V \\ \dfrac{dV}{dT} = \dfrac{R_{15}}{R_{14}C_4}Z \end{cases} \tag{11}$$

in which,

$$H = X - Z + 0.5\left(|X - Z - 1| - |X - Z + 1|\right). \tag{12}$$

Set $t = KT$, $x = MX$, $y = MY$, $z = MZ$, $w = MW$, $v = MV$, and set scale transformation factors $M = 2$ and $K = 100$, and we have

$$\begin{cases} \dfrac{dx}{dt} = \dfrac{1}{KR_4C_1}\left(\dfrac{R_4}{R_1}x - \dfrac{R_4}{R_2}y - \dfrac{R_4}{R_3}H'\right) \\ \dfrac{dy}{dt} = \dfrac{1}{KR_9C_2}\left(\dfrac{R_9}{R_7}x - \dfrac{R_9}{R_8}y\right) \\ \dfrac{dz}{dt} = \dfrac{1}{KR_{11}C_3}\left(-\dfrac{R_{11}}{4R_{19}}w^2v - \dfrac{R_{11}}{R_{18}}v + \dfrac{R_{11}}{R_{10}}H'\right), \\ \dfrac{dw}{dt} = \dfrac{1}{KR_{17}C_5}\dfrac{R_{17}}{R_{16}}v \\ \dfrac{dv}{dt} = \dfrac{1}{KR_{15}C_4}\dfrac{R_{15}}{R_{14}}z \end{cases} \tag{13}$$

where,

$$H' = x - z + 0.5\left(|x - z - 0.5| - |x - z + 0.5|\right). \tag{14}$$

In combination with the formula above, the equivalent circuit model of the system (9) can be designed as shown in **Figure 7**. Based on the equivalent circuit, as shown in **Figure 8**, the chaotic attractors can be implemented by multisim 14.0. The specific component parameters are shown in **Table 1**.

6 CONCLUSION

This proposes a new meminductive chaotic system of extreme multistability. Due to the constant shift of equilibrium with different initial values, the proposed meminductive chaotic system can generate many infinitely coexisting attractors. By introducing the model of weak signal and noise to the meminductive chaotic system, a new weak signal detection model is established. Because this detection model inherits the extreme multistability of the original meminductive system, it can detect boosting signals of different amplitudes at the same time. The spectral line in the frequency domain can effectively reflect the motion trajectory of the object generating a weak signal. Next, we will try to apply the proposed meminductive system to other related areas.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding authors.

AUTHOR CONTRIBUTIONS

BK: Conceptualization; Methodology; Data curation; Supervision; Investigation; Project administration; Writing—original draft; Writing—review and editing. WQ: Funding acquisition; Investigation; Project administration; Resources; Validation.

FUNDING

This work was supported by the National Natural Science Foundation of China (No: 12171004, 12026233, and 12026221).

REFERENCES

1. Chua L. Memristor-The Missing Circuit Element. *IEEE Trans Circuit Theor* (1971) 18:507–19. doi:10.1109/tct.1971.1083337
2. Tour JM, He T. The Fourth Element. *Nature* (2008) 453:42–3. doi:10.1038/453042a
3. Ma X, Li C, Wang R, Jiang Y, Leiet T. Memristive Computation-Oriented Chaos and Dynamics Control. *Front Phys* (2021) 27. doi:10.3389/fphy.2021.759913
4. Wu X, Wang H, He S. Localization of Hidden Attractors in Chua's System with Absolute Nonlinearity and its FPGA Implementation. *Front Phys* (2021) 06. doi:10.3389/fphy.2021.788329
5. Wang N, Zhang G, Kuznetsov NV, Bao H. Hidden Attractors and Multistability in a Modified Chua's Circuit. *Commun Nonlinear Sci Numer Simulation* (2021) 92:105494. doi:10.1016/j.cnsns.2020.105494
6. Wang N, Li C, Bao H, Chen M, Bao B. Generating Multi-Scroll Chua's Attractors via Simplified Piecewise-Linear Chua's Diode. *IEEE Trans Circuits Syst* (2019) 66:4767–79. doi:10.1109/tcsi.2019.2933365
7. Wang N, Zhang G, Bao H. Bursting Oscillations and Coexisting Attractors in a Simple Memristor-Capacitor-Based Chaotic Circuit. *Nonlinear Dyn* (2019) 97:1477–94. doi:10.1007/s11071-019-05067-6
8. Wang N, Zhang G, Bao H. A Simple Autonomous Chaotic Circuit with Dead-Zone Nonlinearity. *IEEE Trans Circuits Syst Express Briefs* (2020) 67:3052–6. doi:10.1109/tcsii.2020.3005726
9. Li C, Li H, Xie W, Du J. A S-type Bistable Locally Active Memristor Model and its Analog Implementation in an Oscillator Circuit. *Nonlinear Dyn* (2021) 106:1041–58. doi:10.1007/s11071-021-06814-4
10. Li C, Yang Y, Yang X, Zi X, Xiao F. A Tristable Locally Active Memristor and its Application in Hopfield Neural Network. *Nonlinear Dyn* (2022) 108:1697–717. doi:10.1007/s11071-022-07268-y
11. Di Ventra M, Pershin YV, Chua LO. Circuit Elements with Memory: Memristors, Memcapacitors, and Meminductors. *Proc IEEE* (2009) 97:1717–24. doi:10.1109/jproc.2009.2021077
12. Ma X, Mou J, Liu J, Ma C, Yang F, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor-Memcapacitor. *Nonlinear Dyn* (2020) 100:2859–76. doi:10.1007/s11071-020-05601-x
13. Ye X, Wang X, Gao S, Mou J, Wang Z, Yang F. A New Chaotic Circuit with Multiple Memristors and its Application in Image Encryption. *Nonlinear Dyn* (2020) 99:1489–506. doi:10.1007/s11071-019-05370-2
14. Ye X, Mou J, Luo C, Wang Z. Dynamics Analysis of Wien-Bridge Hyperchaotic Memristive Circuit System. *Nonlinear Dyn* (2018) 92:923–33. doi:10.1007/s11071-018-4100-x
15. Xiong L, Yang F, Mou J, An X, Zhang X. A Memristive System and its Applications in Red–Blue 3D Glasses and Image Encryption Algorithm with DNA Variation. *Nonlinear Dyn* (2022) 107:1–23. doi:10.1007/s11071-021-07131-6
16. Xiong L, Zhang X, Teng S, Qi L, Zhang P. Detecting Weak Signals by Using Memristor-Involved Chua's Circuit and Verification in Experimental Platform. *Int J Bifurcation Chaos* (2020) 30:2050193. doi:10.1142/s021812742050193x
17. Ye X, Wang X. Characteristic Analysis of A Simple Fractional-Order Chaotic System with Infinitely Many Coexisting Attractors and its DSP Implementation. *Phys Scr* (2020) 95:075212. doi:10.1088/1402-4896/ab8eec
18. Liu B, Ye X, Hu G. Design of a New 3D Chaotic System Producing Infinitely Many Coexisting Attractors and its Application to Weak Signal Detection. *Int J Bifurcation Chaos* (2021) 31:2150235. doi:10.1142/s0218127421502357
19. Xu S, Wang X, Ye X. A New Fractional-Order Chaos System of Hopfield Neural Network and its Application in Image Encryption", *Chaos*A New Fractional-Order Chaos System of Hopfield Neural Network and its Application in Image Encryption. *Solitons & FractalsChaos, Solitons & Fractals* (2022) 157:111889. doi:10.1016/j.chaos.2022.111889
20. Zhou Y, Li C, Li W, Li H, Feng W, Qian K. Image Encryption Algorithm with circle index Table Scrambling and Partition Diffusion. *Nonlinear Dyn* (2021) 103:2043–61. doi:10.1007/s11071-021-06206-8
21. Li C-L, Zhou Y, Li H-M, Feng W, Du J-R. Image Encryption Scheme with Bit-Level Scrambling and Multiplication Diffusion. *Multimed Tools Appl* (2021) 80:18479–501. doi:10.1007/s11042-021-10631-7
22. Li X, Mou J, Banerjee S, Cao Y. An Optical Image Encryption Algorithm Based on Fractional-Order Laser Hyperchaotic System. *Int J Bifurcation Chaos* (2022) 32:2250035. doi:10.1142/s0218127422500353
23. Gao X, Mou J, Xiong L, Sha Y, Yan H, Cao Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn* (2022) 108:613–36. doi:10.1007/s11071-021-07192-7
24. Gao X, Mou J, Banerjee S, Cao Y, Xiong L, Chen X. An Effective Multiple-Image Encryption Algorithm Based on 3D Cube and Hyperchaotic Map. *J King Saud Univ - Comp Inf Sci* (2022) 34:1535–51. doi:10.1016/j.jksuci.2022.01.017
25. Wang X, Wang X, Ma B, Li Q, Shi Y-Q. High Precision Error Prediction Algorithm Based on ridge Regression Predictor for Reversible Data Hiding. *IEEE Signal Process Lett* (2021) 28:1125–9. doi:10.1109/lsp.2021.3080181
26. Ma B, Shi YQ. A Reversible Data Hiding Scheme Based on Code Division Multiplexing. *IEEE Trans.Inform.Forensic Secur.* (2016) 11:1914–27. doi:10.1109/tifs.2016.2566261
27. Li Q, Wang X, Ma B, Wang X, Wang C, Gao S, et al. Concealed Attack for Robust Watermarking Based on Generative Model and Perceptual Loss. *IEEE Trans Circuits Syst Video Technol* (2021) 1. doi:10.1109/TCSVT.2021.3138795
28. Wang C, Ma B, Xia Z, Li J, Li Q, Shi Y-Q. Stereoscopic Image Description with Trinion Fractional-Order Continuous Orthogonal Moments. *IEEE Trans Circuits Syst Video Technol* (2022) 32:1998–2012. doi:10.1109/TCSVT.2021.3094882
29. Ye X, Wang X, Zhao H, Gao H, Zhang M. Extreme Multistability in a New Hyperchaotic Meminductive Circuit and its Circuit Implementation. *Eur Phys J Plus* (2019) 134:206. doi:10.1140/epjp/i2019-12535-0
30. Bao B, Jiang T, Xu Q, Chen M, Wu H, Hu Y. Coexisting Infinitely many Attractors in Active Band-Pass Filter-Based Memristive Circuit. *Nonlinear Dyn* (2016) 86:1711–23. doi:10.1007/s11071-016-2988-6

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Physics

ORIGINAL RESEARCH
published: 13 June 2022
doi: 10.3389/fphy.2022.901800

Multi-Image Encryption Algorithm for 2D and 3D Images Based on Chaotic System

Xinyu Gao, Miao Miao and Xiaoyang Chen*

School of Mechanical Engineering and Automation, Dalian Polytechnic University, Dalian, China

In order to accommodate multiple types of image encryption, a multi-image encryption algorithm for 2D and 3D images is designed in this paper. After recording the type and number of images, the pixels/coordinates of multiple images are stored in a cube block and are subjected to confusion and diffusion operations. The confusion step uses the random length sequence position swapping method to swap a row (column) vector of variable length with another row (column) vector of the same length at a random position. The diffusion is done by Exclusive OR to combine pixels/coordinates at different locations with different chaotic matrices. Finally, the cipher images are output one by one. Experimental simulations and security analysis verify the effectiveness and security of the algorithm.

Keywords: multi-image, 2D and 3D images, chaotic system, confusion, diffusion

Edited by:
Shaobo He,
Central South University, China

Reviewed by:
Junxin Chen,
Northeastern University, China
Zhongyun Hua,
Harbin Institute of Technology, China

****Correspondence:***
Miao Miao
miaomiao@dlpu.edu.cnmailto:
miaomiao@dlpu.edu.cn

Specialty section:
This article was submitted to
Interdisciplinary Physics,
a section of the journal
Frontiers in Physics

Received: *22 March 2022*
Accepted: *11 May 2022*
Published: *13 June 2022*

Citation:
Gao X, Miao M and Chen X (2022)
Multi-Image Encryption Algorithm for
2D and 3D Images Based on
Chaotic System.
Front. Phys. 10:901800.
doi: 10.3389/fphy.2022.901800

INTRODUCTION

Image transmission has become more convenient with the development of the Internet, but also faces greater risk. Whether in life, medical, military or commercial fields, effective image encryption is needed to guarantee the secure transmission of various images [1–6]. In addition, since the development of 3D printing technology, image transmission has been extended from 2D images to 3D images [7]. Both 2D and 3D images are being transmitted in large quantities in all corners of the network. Common single-image encryption (SIE) methods are simple and effective, but cannot handle multiple images at the same time. In order to meet the demand of images being encrypted and transmitted in large capacity, multi-image encryption (MIE) algorithms are investigated [8].

Chaotic systems have characteristics such as sensitivity to initial values, pseudo-randomness, and are suitable for use in cryptography [9–14], so there are many SIE and MIE algorithms based on chaotic systems [15–18]. The sensitivity to the initial value guarantees that the image encryption algorithm will not be easily broken when it is perturbed. The pseudo-randomness brings a rich operation to the image encryption algorithm [19–25]. Since chaotic systems play a great advantage in image encryption, using chaotic systems to design image encryption algorithms has become a popular research topic [26, 27]. In previous work on image encryption, there are many classical 1D chaotic maps and 2D chaotic systems used for image encryption. However, with the improvement of computer performance and the development of mathematics and other theories, the simple structure of 1D maps and 2D chaotic systems can be predicted by methods such as nonlinear prediction, so we are also pursuing higher dimensional, more complex and secure chaotic systems to design image encryption algorithms. In the proposed scheme, a three-dimensional memristive neuron chaotic system is used for image encryption [28]. The system is simple in structure but rich in dynamical behavior and is very suitable for use in image encryption.

Multi-image encryption algorithms have been studied since 2012 or even earlier. Most early implementations of multi-image encryption used optical methods, such as Kong et al. used wavelet transform and fractional order Fourier transform in encryption of multiple images [29, 30]; Chen et al. combined optical wavelet transform and compression sensing to compress and encrypt multiple images [31]; Huang et al. used a two-dimensional linear canonical transformation and combined it with a chaotic system to encrypt multiple images [32]. Later, chaotic maps and chaotic systems were used for multi-image encryption, and once they were used, they were widely popularized. Singh uses chaotic maps to generate chaotic random phase masks for encryption of multiple images [33]. Santo Banerjee uses a chaotic laser system to achieve simultaneous encryption of multiple images [34]. Ye combines multiple plain images with a chaotic system to obtain encrypted images, and the security of the algorithm is improved [17]. However, there are multi-image encryption algorithms dedicated to two-dimensional images or three-dimensional images, and few scholars have focused on multi-image encryption algorithms that are applicable to both two-dimensional images and three-dimensional images. In order to follow the development of 3D printing technology and the progress of communication technology, an algorithm based on chaotic system, which is applicable to both 2D multi-image encryption and 3D multi-image encryption, is proposed. In the designed encryption mechanism, either single 2D/3D image or multiple 2D/3D images can be encrypted and decrypted. The encryption algorithm uses an Fridrich-structure and uses a confusion-diffusion encryption strategy [35]. In the confusion phase, each row/column vector of each encrypted plane is split into two row/column vectors of random length to be exchanged with row/column vectors at other locations. The pseudo-randomness of the chaotic sequences guarantees that the pixels of the 2D images or the coordinates of the 3D images are sufficiently disordered. In the diffusion phase, the chaotic sequences of diffusion are determined by the pixel positions/coordinates. The chaotic sequences transformation operation greatly enhances the security of the algorithm.

The paper structure is listed hereafter. *Chaotic System* introduces the chaotic system used by the proposed algorithm. *Encryption Algorithm and Decryption Algorithm* provides a detailed description of the encryption algorithm. *Simulation Results* verifies the effectiveness of the encryption algorithm with simulation results. *Security Analysis* proves the security of the proposed algorithm using security analysis. *Conclusion* concludes the multi-image encryption algorithm work.

CHAOTIC SYSTEM

In the designed encryption scheme, a 3D memristive neuron model is used to generate chaotic sequences. The model is described as:

$$\begin{cases} \dot{x} = -x + \left(15xe^{-12.5x^2}\right)\left(15ye^{-12.5y^2}\right) + A\sin(2\pi Ft) \\ \dot{y} = -\alpha y + \alpha\left(15xe^{-12.5x^2}\right)^2 - kyz \\ \dot{z} = -y - z \end{cases}, \tag{1}$$

where α is a positive value, A is the stimulus amplitude, F is the stimulus frequency, and k is the inductive strength of the memductance. When (α, A, k, F, x_0, y_0, z_0) = (4.3, 0.24, 0.428, 1.1, 0.5, 1.55, −2), the phase trajectory of the non-autonomous chaotic system is exhibited in **Figure 1**. From **Figure 1**, chaotic system has very complex trajectories of action, indicating that this system can be applied in image encryption algorithms.

ENCRYPTION ALGORITHM AND DECRYPTION ALGORITHM

Encryption Algorithm

The encryption scheme as a whole adopts the encryption strategy of scrambling first and then diffusing. In the scrambling phase, the image data is intercepted at random lengths and swapped at random locations. During the diffusion phase, pixels or coordinates are converted using a combination of dynamic chaotic data blocks and image data blocks. After scrambling and diffusion, multiple 3D cipher images or multiple 2D cipher images are output. The encryption flowchart is displayed in **Figure 2**. The detailed encryption procedures are listed.

Step 1: Input several images of the same type and determine whether they are 2D or 3D images. If they are 2D images go to step 2, if they are 3D images go to step 3.

Step 2: The pixels of all 2D images are stored in vector V. The sizes of each 2D image are logged.

Step 3: The 3D image position coordinates are divided into an integer part and a fractional part. The fractional part is stored as a quotient and a remainder. All position coordinates are converted to integers and stored as vector V. The sizes of each 3D image are also logged.

Step 4: Convert the vector V into a cube C. Place the cube in space and mark the orientation of the cube with the x-y-z axis. The altitude of z-axis is denoted as L. The width and height of x-y plane are denoted as W and H. Taking $(W, H) = (256, 256)$ (it can also be any other appropriate value), the length of the vector V as Le, the height of the cube C can be calculated as

$$L = \mathrm{ceil}(Le/WH), \tag{2}$$

where ceil(a) is rounding a up.

Step 5: Cube C is summed and fed into Hash 256 to obtain a 64-bit hash value h. The hash value h is converted into a sequence of decimal numbers hd. Two parameters h_1 and h_2 associated with the plain images are obtained by processing the hd.

$$\begin{cases} hs(i) = hd(i) \oplus hd(sl+1-i), i = 1, \ldots, sl/2 \\ h_1 = 10^{-4} mean(hs(1\colon sl/4)) \mathrm{mod}(sl) \\ h_2 = 10^{-4} mean(hs(sl/4+1\colon sl/2)) \mathrm{mod}(sl) \end{cases}, \tag{3}$$

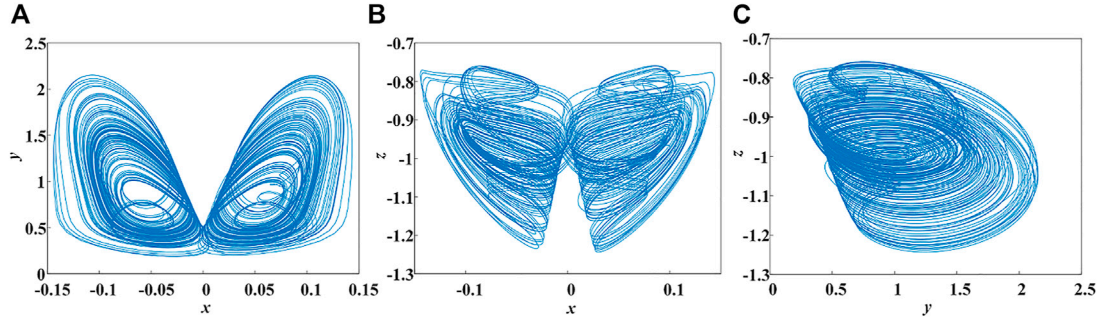


FIGURE 1 | Phase trajectories of the chaotic system **(A)** *x-y* plane **(B)** *x-z* plane **(C)** *y-z* plane.

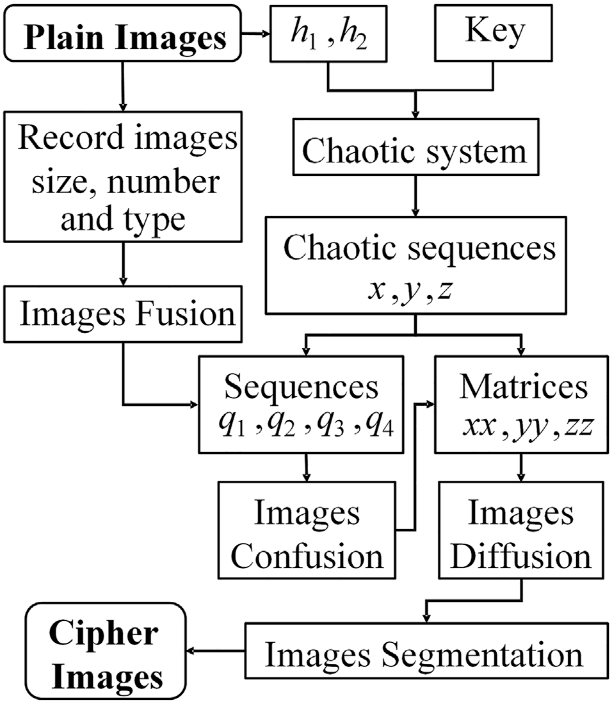


FIGURE 2 | Encryption flowchart.

where *sl* is the size of *hd*, *b*mod(*a*) yields the remainder of *b* divided by *a*.

Step 6: The parameters h_1, h_2, α, A, k, F, and the initial values x_0, y_0, z_0 are input into the chaotic system, and the chaotic sequences are obtained by iteration. After quantizing the chaotic sequences, the sequences x, y and z are obtained.

Step 7: Four sequences q_1, q_2, q_3 and q_4 are obtained using three chaotic sequences.

$$\begin{cases} q_1 = x(1\colon beta)\mathrm{mod}(beta) + 1 \\ q_2 = x(beta + 1\colon 2beta)\mathrm{mod}(beta) + 1 \\ q_3 = y(1\colon beta)\mathrm{mod}(beta) + 1 \\ q_4 = z(1\colon beta)\mathrm{mod}(beta) + 1 \end{cases}, \tag{4}$$

where *beta* is the maximum of W, H, and L.

Step 7.1: The value of the sequence q_1 is adjusted to limit it to the interval $H/4$ to $3H/4$.

$$q_1(i) = \begin{cases} q_1(i) + ceil(H/4), q_1(i) < H/4 \\ q_1(i) - floor(H/4), q_1(i) > 3H/4, i = 1, \ldots, beta. \\ q_1(i), H/4 \le q_1(i) \le 3H/4 \end{cases} \tag{5}$$

Step 7.2: Cube C is scrambled along the rows from top to bottom. Swap the positions $C(i, 1{:}q_1(j), k)$ and $C(q_2(i \times k\ \mathrm{mod}(W)+1), 1{:}q_1(j), q_3(i \times k\ \mathrm{mod}(L)+1))$, and swap the positions of $C(i, q_1(j)+1{:}H, k)$ and $C(q_3(i \times k\ \mathrm{mod}(W)+1), q_1(j)+1{:}H, q_4(i \times k\ \mathrm{mod}(L)+1))$, $i = 1, \ldots, W$; $j = 1, \ldots, H$; $k = 1, \ldots, L$.

Step 7.3: Cube C is scrambled along the columns in order from left to right. Swap the positions of $C(1{:}q_1(i)\mathrm{mod}(W), j, k)$ and $C(1{:}q_1(i)\mathrm{mod}(W), q_3(j \times k\ \mathrm{mod}(H)+1), q_4(j \times k\ \mathrm{mod}(L)+1))$, and swap the positions of $C(q_1(i)\mathrm{mod}(W)+1{:}W, j, k)$ and $C(q_1(i)\mathrm{mod}(W)+1{:}W, q_4(j \times k\ \mathrm{mod}(H)+1), q_2(j \times k\ \mathrm{mod}(L)+1))$. The cube after permutation is marked as C_1.

Step 8: Three chaotic matrices xx, yy and zz of size $W \times H$ are obtained by chaotic sequences x, y, z.

$$\begin{cases} xx = reshape(x(end - WH + 1\colon end), W, H) \\ yy = reshape(y(end - WH + 1\colon end), W, H). \\ zz = reshape(z(end - WH + 1\colon end), W, H) \end{cases} \tag{6}$$

Step 9.1. Combining the cube C_1 and the chaotic matrices, the first row and column of every plane are diffused. The diffused cube is denoted as C_2.

$$C_2(1,1,1) = C_1(1,1,1) \oplus xx(1,1), \tag{7}$$

$$C_2(1,1,k) = \begin{cases} C_1(1,1,k) \oplus xx(1,1) \oplus C_1(1,1,k-1), k\mathrm{mod}(3) = 1 \\ C_1(1,1,k) \oplus yy(1,1) \oplus C_1(1,1,k-1), k\mathrm{mod}(3) = 2, k \\ C_1(1,1,k) \oplus zz(1,1) \oplus C_1(1,1,k-1), k\mathrm{mod}(3) = 0 \end{cases} = 2, \ldots, L, \tag{8}$$

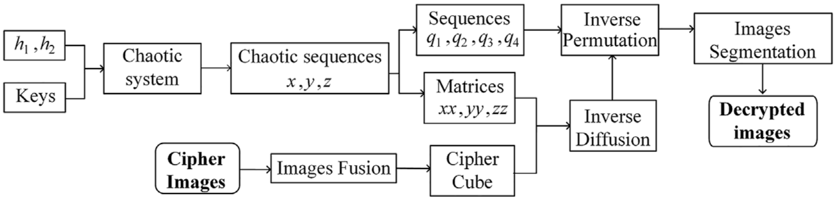


FIGURE 3 | Decryption flowchart.

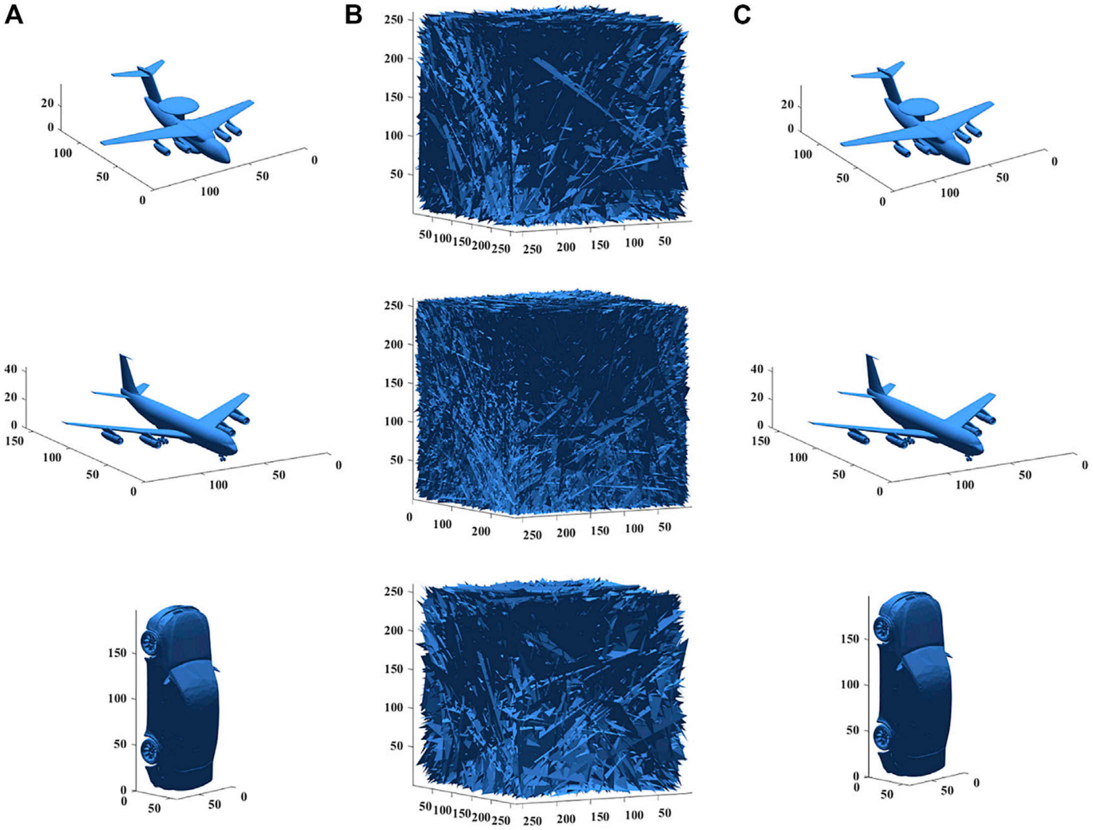


FIGURE 4 | Simulation results of 3D images **(A)** Plain images (Air picket, Boeing 747, BMW car) **(B)** Cipher images **(C)** Decrypted images.

$$C_2(1,j,k)=\begin{cases}C_1(1,j,k)\oplus xx(1,j)\oplus C_1(1,j-1,k), jk\bmod(3)=1\\ C_1(1,j,k)\oplus yy(1,j)\oplus C_1(1,j-1,k), jk\bmod(3)=2,\\ C_1(1,j,k)\oplus zz(1,j)\oplus C_1(1,j-1,k), jk\bmod(3)=0\end{cases}\begin{matrix}k=1,\ldots,L\\ j=2,\ldots,H\end{matrix}, \tag{9}$$

$$C_2(i,1,k)=\begin{cases}C_1(i,1,k)\oplus xx(i,1)\oplus C_1(i-1,1,k), ik\bmod(3)=1\\ C_1(i,1,k)\oplus yy(i,1)\oplus C_1(i-1,1,k), ik\bmod(3)=2,\\ C_1(i,1,k)\oplus zz(i,1)\oplus C_1(i-1,1,k), ik\bmod(3)=0\end{cases}\begin{matrix}k=1,\ldots,L\\ i=2,\ldots,W\end{matrix}. \tag{10}$$

Step 9.2. Diffusion is performed on the remaining part of the cube.

$$\begin{cases}CO(i,j,1)=C_2(i-1,j,1)+C_2(i,j-1,1)+C_2(i-1,j-1,1)\bmod(256)\\ i=2,\ldots,W; j=2,\ldots,H\\ C_2(i,j,1)=\begin{cases}C_1(i,j,1)\oplus xx(i,j)\oplus CO(i,j,1), ij\bmod(3)=1\\ C_1(i,j,1)\oplus yy(i,j)\oplus CO(i,j,1), ij\bmod(3)=2\\ C_1(i,j,1)\oplus zz(i,j)\oplus CO(i,j,1), ij\bmod(3)=0\end{cases}\end{cases}, \tag{11}$$

$$\begin{cases}CO(i,j,k)=C_2(i-1,j,k)+C_2(i,j-1,k)+C_2(i-1,j-1,k)\bmod(256)\\ i=2,\ldots,W; j=2,\ldots,H; k=2,\ldots,L\\ C_2(i,j,k)=\begin{cases}\begin{cases}FF(i,j)=xx(i-1,j)+xx(i,j-1)+xx(i-1,j-1)\bmod(256)\\ C_1(i,j,k)\oplus xx(i,j)\oplus CO(i,j,k)\oplus FF(i,j)\end{cases}, ijk\bmod(3)=1\\ \begin{cases}FF(i,j)=yy(i-1,j)+yy(i,j-1)+yy(i-1,j-1)\bmod(256)\\ C_1(i,j,k)\oplus yy(i,j)\oplus CO(i,j,k)\oplus FF(i,j)\end{cases}, ijk\bmod(3)=2\\ \begin{cases}FF(i,j)=zz(i-1,j)+zz(i,j-1)+zz(i-1,j-1)\bmod(256)\\ C_1(i,j,k)\oplus zz(i,j)\oplus CO(i,j,k)\oplus FF(i,j)\end{cases}, ijk\bmod(3)=0\end{cases}\end{cases}. \tag{12}$$

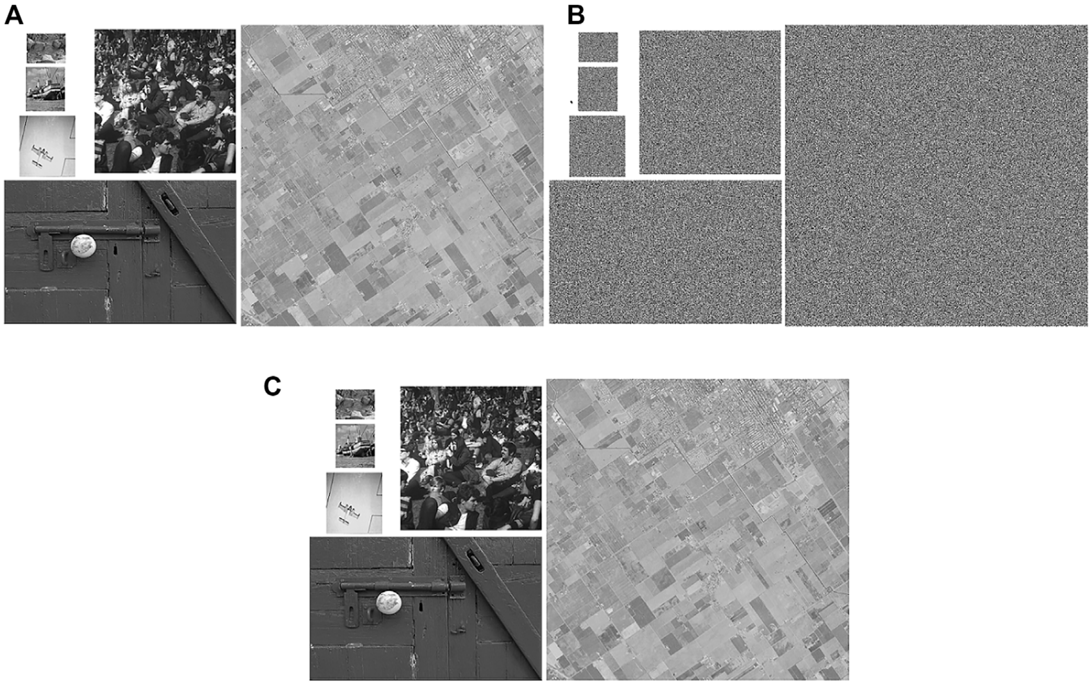


FIGURE 5 | Simulation results of 2D images **(A)** Plain images (Water, Boat, 5.1.11, Crowd, Kodimo2, 2.2.11) **(B)** Cipher images **(C)** Decrypted images.

TABLE 1 | Key space of encryption scheme.

Parameter	Key space	Parameter	Key space
α	10^{15}	y_0	10^{15}
A	10^{14}	z_0	10^{15}
k	10^{14}	h_1	10^{14}
F	10^{14}	h_2	10^{14}
x_0	10^{16}	Total	$10^{131} \approx 2^{435}$

Step 10. Based on the image size recorded in step 2 or step 3, the cube C_2 is split and output to obtain the cipher images.

Decryption Algorithm

The process of image decryption is to flip the encryption process. The cipher images and the keys are input to the decryption system, and the chaotic sequences and chaotic matrices are generated by the chaotic system for the inverse diffusion and inverse permutation of the decryption. All cipher images are fused into a cipher cube, and the exclusive OR operation is performed in the order of points, rows, columns, and planes. The inverse diffused cube is reverse permuted from right to left along the columns and from bottom to top along the rows. The decrypted images are obtained by re-partitioning the cube after the inverse permutation and reorganizing them according to the sizes and types of the original images. The specific decryption flow chart is shown in **Figure 3**.

SIMULATION RESULTS

In order to verify whether the encryption and decryption algorithms are effective, a set of 3D images and a set of 2D images are used for encryption and decryption tests respectively, and the results are presented in **Figures 4**, **5**. In **Figure 4**, the three 3D plain images (Air picket, Boeing 747, BMW car) are shown in (A), the cipher images obtained by encryption are shown in (B),

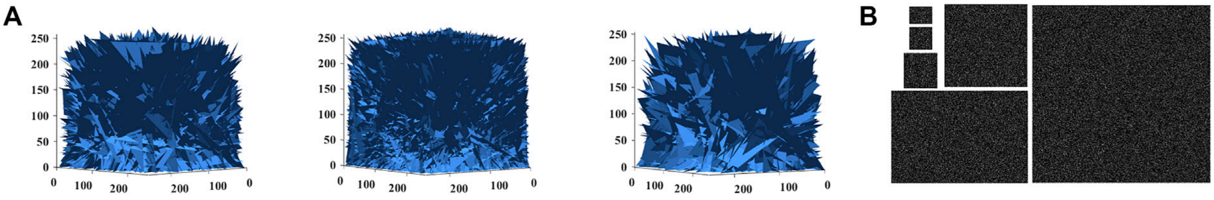


FIGURE 6 | Encryption process key sensitivity **(A)** 3D models **(B)** 2D images.

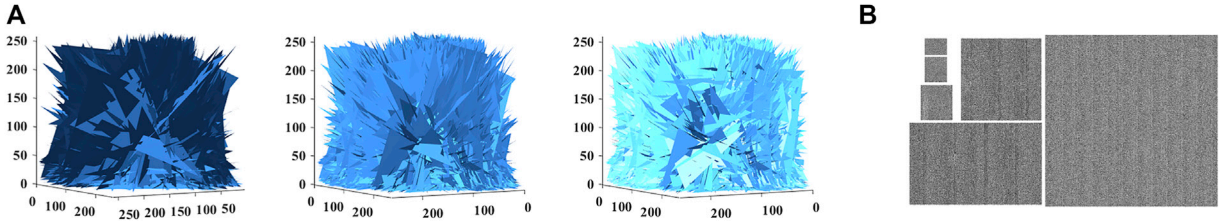


FIGURE 7 | Decryption process key sensitivity**(A)** 3D models **(B)** 2D images.

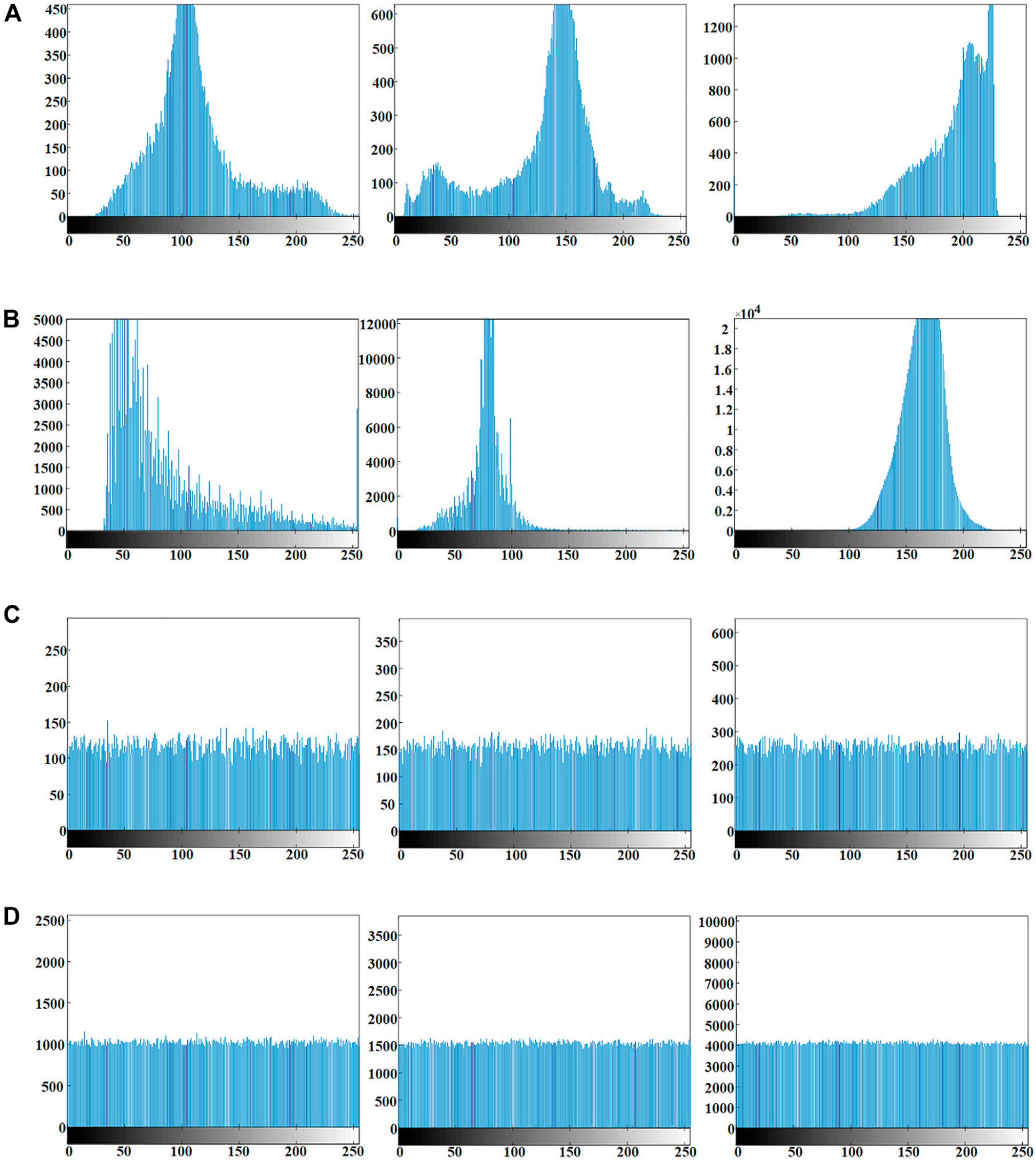


FIGURE 8 | Histogram of encryption and decryption images **(A,B)** Histogram of plain images (Water, Boat, 5.1.11, Crowd, Kodimo2, 2.2.11) **(C,D)** Histogram of cipher images.

TABLE 2 | χ^2 test results.

Image	χ^2 Value		Result		
	Plain image	Cipher image	$\chi^2_{0.01}$ (255)	$\chi^2_{0.05}$ (255)	$\chi^2_{0.1}$ (255)
Water	43758.6176	252.1685	Pass	Pass	Pass
Boat	63497.6288	229.3504	Pass	Pass	Pass
5.1.11	220848.6796	231.4259	Pass	Pass	Pass
Crowd	740064.1758	275.8242	Pass	Pass	Pass
Kodimo2	3605450.7031	283.7227	Pass	Pass	Pass
2.2.11	3352937.8725	246.2635	Pass	Pass	Pass

and the decrypted results are exhibited in (C). In **Figure 5**, the multiple 2D plain images (Water, Boat, 5.1.11, Crowd, Kodimo2, 2.2.11), the cipher images and the decrypted results are displayed in (A), (B) and (C). It can be observed from **Figure 4** and **Figure 5** that the decrypted images are consistent with the original images, while no information related to the original images is visible in the cipher images. **Figure 4** and **Figure 5** together verify that the designed multiple-image encryption scheme can encrypt the images to be protected and the decryption scheme can successfully recover the cipher images.

SECURITY ANALYSIS

Key Space and Key Sensitivity

It takes a very long time for an attacker to break the algorithm using a brute force attack when the key space is relatively large. Therefore, the probability of the encryption algorithm being breached decreases as the key space increases. When the key space is greater than 2,100, the encryption algorithm can be considered resistant to brute-force attacks [36]. The key consists of two parts: the parameters associated with the plain images and chaotic system parameters. The key sensitivity is tested by applying a small perturbation to each parameter, so that the key space for each parameter can be obtained. For example, the key remains unchanged during encryption and a perturbation of 10^{-15} is applied to parameter α during decryption, the decrypted images are observed to be noise-like images and the key space of parameter α can be determined to be 10^{15}. The key space for each parameter is tested so that the total key space is obtained, and the test results are listed in **Table 1**. From **Table 1**, the key space is obviously larger than the minimum value, so the designed encryption algorithm will not be breached by brute force attacks.

Testing of key sensitivity can be divided into encryption process and decryption process. When the encryption key is slightly changed, the obtained cipher image is $CC_2(i)$ ($i = 1, 2, \ldots, n$, n is the total number of plain images) and the cipher images obtained from the original key encryption is $C_2(i)$. Let $CC(i) = |C_2(i)\text{-}CC_2(i)|$, if the obtained image $CC(i)$ is a black 2D image or a blank 3D image, it means that the encryption process of the encryption algorithm is not sensitive enough. If the obtained image $CC(i)$ is a noise-like 2D image or a cluttered 3D image, it means that the encryption algorithm has a strong encryption process key sensitivity. When the key is slightly changed during the decryption process and then the correct decrypted image is not available, then the encryption scheme is sensitive to the key during the decryption process. The results of the key sensitivity test are shown in **Figures 6**, **7**. As can be seen in **Figure 6**, the cipher images obtained after the key is slightly perturbed during the encryption process are very different. As can be seen in **Figure 7**, the slight alteration of the decryption key leads to the inability to obtain the correct decrypted images. The sensitivity of the encryption scheme to the key is demonstrated in **Figures 6**, **7**.

Statistical Properties

Histogram and χ^2 Test

Histogram and χ^2 test are commonly used to check whether the image pixels are evenly distributed. The histogram represents the distribution of light and dark in the image, the brighter the image the more the histogram crest tends to the right. Usually, the histograms of the plain images are irregularly distributed, and the histograms of the cipher images, where the pixel distribution information is hidden, are uniformly distributed. The χ^2 test is used to test whether the assumption of uniform distribution of cipher images pixels holds. If the χ^2 test results are less than the theoretical value ($\chi^2_{0.01}$ (255) = 310.4574, $\chi^2_{0.05}$ (255) = 293.2478, $\chi^2_{0.1}$ (255) = 284.3359) at different test levels, the hypothesis is valid. To check whether the proposed encryption algorithm can hide the light and dark information of the images, the histogram and χ^2 test are presented in **Figure 8** and **Table 2**. From **Figure 8**, the histograms of plain images vary, but the histograms of cipher images are uniformly distributed. The data in **Table 2** verify the findings in **Figure 8**. Combined with **Figure 8** and **Table 2**, the proposed encryption scheme can well hide the light and dark information of the images and avoid the statistical information of the images from being exploited by attackers.

Correlation and Correlation Coefficient

Correlation generally refers to the adjacent pixel correlation of an image, mainly for 2D images. For 3D images, it is used to detect the correlation between adjacent coordinates. The encryption algorithm should have the ability to break the correlation between the adjacent pixels of the 2D images or the adjacent coordinates of the 3D images, so that the adjacent pixel/coordinates cipher images tend to be uncorrelated with each other. The correlation and correlation coefficients of the

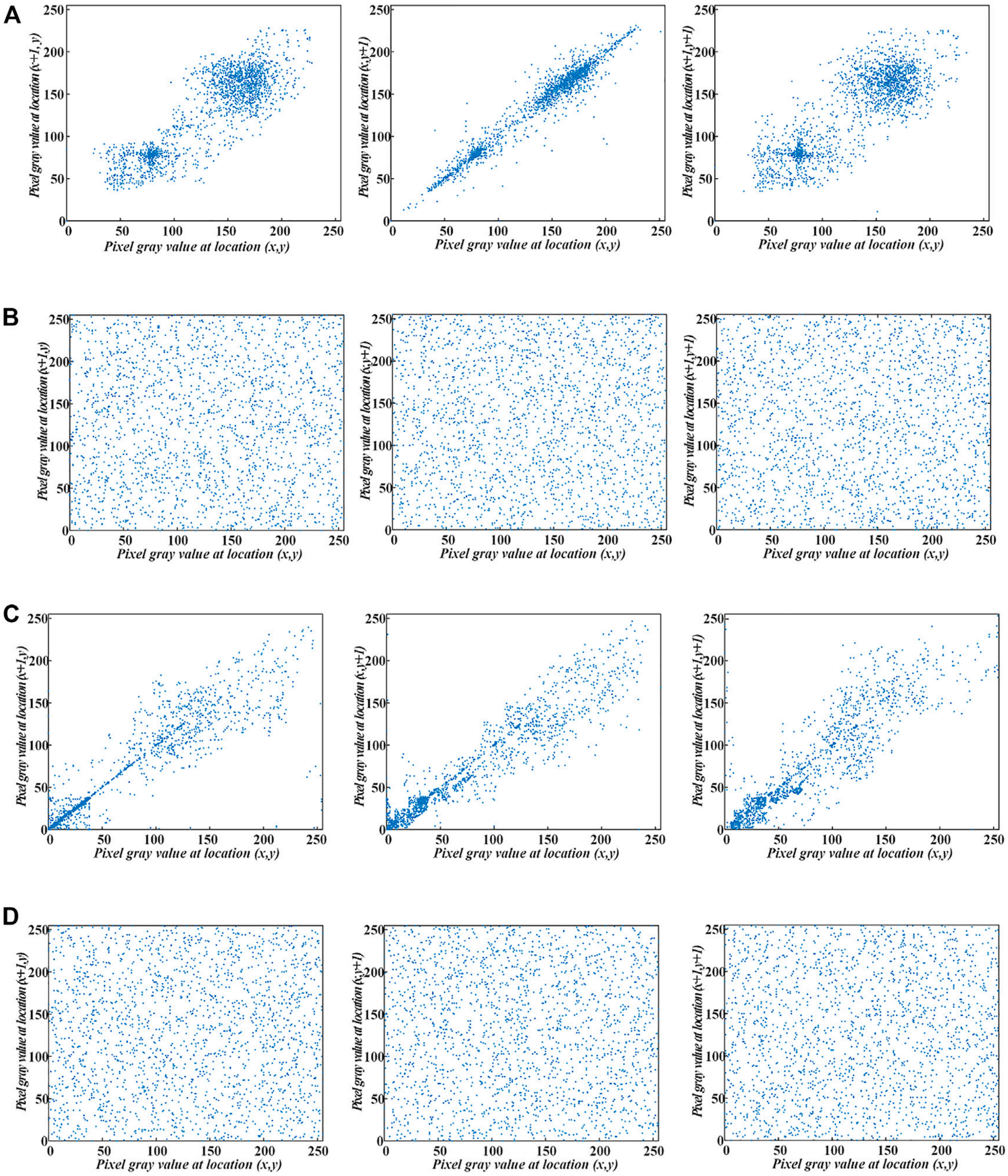


FIGURE 9 | Correlation of different images **(A)** Correlation of 2D images plain cube **(B)** Correlation of 2D images cipher cube **(C)** Correlation of 3D images plain cube **(D)** Correlation of 3D images cipher cube.

TABLE 3 | Correlation coefficient of the plain images and cipher images.

Image	2D images cube		3D images cube	
	Plain	Cipher	Plain	Cipher
Horizontal	0.7288	−0.0243	0.6703	−0.0055
Vertical	0.9750	0.0072	0.6608	−0.0027
Diagonal	0.6928	0.0035	0.6618	−0.0031

TABLE 4 | Information entropy and local entropy test results.

Entropy	Image	Plain cube	Cipher cube
Information entropy	2D images	7.0985	7.9999
	3D images	7.1736	7.9999
Local entropy	2D images	6.8302	7.9028
	3D images	7.5172	7.9030

TABLE 5 | Comparison with other algorithms on entropy.

Algorithm	Information entropy	Algorithm	Local entropy
Reference [38]	7.9998	Reference [40]	7.9034
Reference [39]	7.9993	Reference [42]	7.9025
Reference [40]	7.9973	Reference [17]	7.9125
Reference [26]	7.9998	Reference [44]	7.9028
Reference [41]	7.9994	Ref. 7 [45]	7.9028
Reference [43]	7.9992	Reference [46]	7.9029
Proposed	7.9999	Proposed	7.9029

TABLE 6 | Differential attack test results.

Image	NPCR(%)	UACI(%)
Air picket	99.6133	33.4674
Boeing 747	99.6150	33.4625
BMW car	99.6144	33.4863
Water	99.6449	33.4898
Boat	99.6191	33.4528
5.1.11	99.6262	33.4709
Crowd	99.6159	33.4506
Kodimo2	99.6156	33.4620
2.2.11	99.6144	33.4525
Average	99.6199	33.4661

TABLE 7 | Comparison with other algorithms on differential attack.

Algorithm	NPCR(%)	UACI(%)
Reference [38]	99.6060	33.5126
Reference [40]	99.6082	33.4701
Reference [41]	99.6199	33.4791
Reference [43]	99.5000	33.4825
Reference [17]	99.6077	33.4398
Proposed	99.6199	33.4661

plain cube and cipher cube are tested. The test results are presented in **Figure 9** and **Table 3**. As can be seen in **Figure 9**, the 2D plain image shows strong correlation with the distribution of adjacent pixels on a straight line with slope of 1. The 3D plain image also shows strong correlation. The adjacent pixels/coordinates of cipher image spread irregularly over the whole area and show no correlation. The correlation coefficients in **Table 3** verify the adjacent pixel/coordinate distributions in **Figure 9**, where the correlation coefficients are larger for plain images and significantly reduced and very close to 0 for cipher images. It can be concluded from **Figure 9**; **Table 3** that the presented image encryption scheme can effectively hide the correlation of the plain images.

Information Entropy and Local Entropy

Information entropy is used to measure the amount of information carried by an image. The greater the information entropy of an image, the greater the amount of information it carries, and then the more confusing the image is from a visual point of view. The local entropy of an image can reflect the local features of the image. The information entropy and local entropy are calculated as in **Eq. 13**, and the test results are shown in **Table 4**. From **Table 4**, it can be seen that the information entropy of the cipher images is significantly increased compared to that of the plain images, which are close to the theoretical value of 8. When the test level is 0.5, the theoretical range of local entropy is (7.901515698, 7.903037329) [37]. In **Table 4**, the local entropy of the cipher images all fall within the theoretical range. The comparison between the proposed algorithm and other algorithms regarding information entropy and local entropy is listed in **Table 5** [17, 26, 38–46]. From **Tables 4**, **5**, the designed encryption algorithm is able to mask the effective information of the plain images so that the cipher images do not contain readable information.

$$\begin{cases} H = -\sum_{i=0}^{255} p(i)\log_2 p(i) \\ H_{k,T_B}(S,L) = \sum_{i=1}^{k} \frac{H(S_{T_B},L)}{k} \end{cases}, \tag{13}$$

where $p(i)$ is the probability of gray value i, $H(S_{TB}, L)$ is the information entropy of the non-overlapping image block S_{TB}, L is the image gray level, $k = 30$ is the number of image blocks, and $T_B = 1936$ is the pixels of each image block.

Differential Attack

A differential attack is an attacker who analyzes the change in the cipher images caused by a change in the plain images to attack the encryption algorithm. The number of pixels change rate (NPCR) and the unified average changing intensity (UACI) are used as differential attack test metrics. The plain images are encrypted normally to get a set of original cipher images, and the plain images are slightly changed to get a new set of cipher images. The difference between the two sets of cipher images is measured by NPCR and UACI. The theoretical values of NPCR and UACI are calculated to be 99.6094 and 33.4635%, respectively [47]. A suitable encryption algorithm should have an NPCR greater than or equal to 99.6094% and the larger the better, and the UACI should be close to 33.4635%. To check the ability of the proposed encryption algorithm to resist the differential attack, NPCR and UACI are shown in **Table 6**. From **Table 6**, it can be seen that both 2D images and 3D images are used for encryption and the proposed algorithm is resistant to differential attacks. The differential test results of different algorithms are presented in **Table 7** [17, 38, 40, 41, 43], where the proposed encryption algorithm is as good as other algorithms in resisting differential attacks.

CONCLUSION

In this paper, an encryption scheme that can be used for both 2D multiple images and 3D multiple images is proposed. The proposed encryption and decryption algorithm can successfully encrypt and decrypt images regardless of whether the input images are multiple 2D images or multiple 3D images, and regardless of whether the input images are of the same size. The length of the chaotic system iteration is determined by the total size of the multiple images, and the chaotic sequences used for diffusion are determined by the position index of the images. In the algorithm design, the characteristics of the three-dimensional chaotic system are fully utilized, and all chaotic sequences are mobilized to fully participate into the encryption operation. The simulation results of multi-image encryption and decryption verify the ability of the proposed algorithm to encrypt and decrypt images. At the same time, the strong key sensitivity provides a guarantee for the algorithm to resist brute force attacks; the statistical test results demonstrate the ability of the algorithm to resist statistical attacks; the test results of differential attack and the comparison results illustrate that the encryption algorithm can effectively resist the differential attack. The experimental simulations and security tests together prove the effectiveness and practicality of the proposed encryption scheme, which has great application potential.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

XG provided the idea of algorithm, carried out the simulations, arranged the architecture and drafted the manuscript. MM and XC supervised the work and revised the manuscript. Both authors read and approved the final manuscript.

REFERENCES

1. Zhou S, Wang X, Zhang Y, Ge B, Wang M, Gao S. A Novel Image Encryption Cryptosystem Based on True Random Numbers and Chaotic Systems. *Multimedia Syst* (2022) 28(1):95–112. doi:10.1007/s00530-021-00803-8
2. Yu F, Kong X, Chen H, Yu Q, Cai S, Huang Y, et al. A 6D Fractional-Order Memristive Hopfield Neural Network and its Application in Image Encryption. *Front Phys* (2022) 10, 847385. (in English). doi:10.3389/fphy.2022.847385
3. Xiong L, Yang F, Mou J, An X, Zhang X. A Memristive System and its Applications in Red-Blue 3D Glasses and Image Encryption Algorithm with DNA Variation. *Nonlinear Dyn* (2022) 107(3):2911–33. doi:10.1007/s11071-021-07131-6
4. Zhou Y, Li C, Li W, Li H, Feng W, Qian K. Image Encryption Algorithm with circle index Table Scrambling and Partition Diffusion. *Nonlinear Dyn* (2021) 103(2):2043–61. doi:10.1007/s11071-021-06206-8
5. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Du S. FPGA Implementation and Image Encryption Application of a New PRNG Based on a Memristive Hopfield Neural Network with a Special Activation Gradient. *Chin Phys. B* (2022) 31:020505. doi:10.1088/1674-1056/ac3cb2
6. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S. A New Multi-Scroll Chua's Circuit with Composite Hyperbolic tangent-cubic Nonlinearity: Complex Dynamics, Hardware Implementation and Image Encryption Application. *Integration* (2021) 81:71–83. doi:10.1016/j.vlsi.2021.05.011
7. Xu J, Zhao C, Mou J, "A 3D Image Encryption Algorithm Based on the Chaotic System and the Image Segmentation," *IEEE Access* (2020) 8, 1–13. doi:10.1109/access.2020.3005925
8. Zhou S, Wang X, Wang M, Zhang Y. Simple Colour Image Cryptosystem with Very High Level of Security. *Chaos, Solitons Fractals* (2020) 141:110225. doi:10.1016/j.chaos.2020.110225
9. Ma C, Mou J, Li P, Liu T. Dynamic Analysis of a New Two-Dimensional Map in Three Forms: Integer-Order, Fractional-Order and Improper Fractional-Order. *Eur Phys J Spec Top* (2021) 230(7):1945–57. doi:10.1140/epjs/s11734-021-00133-w
10. Ma X, Mou J, Liu J, Ma C, Yang F, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor-Memcapacitor. *Nonlinear Dyn* (2020) 100(3):2859–76. doi:10.1007/s11071-020-05601-x
11. Liu T, Banerjee S, Yan H, Mou J. Dynamical Analysis of the Improper Fractional-Order 2D-SCLMM and its DSP Implementation. *The Eur Phys J Plus* (2021) 136(5):1–17. doi:10.1140/epjp/s13360-021-01503-y
12. Ma C, Mou J, Xiong L, Banerjee S, Han X. Dynamical Analysis of a New Chaotic System: Asymmetric Multistability, Offset Boosting Control and Circuit Realization. *Nonlinear Dyn* (2021) 103(6):1–14. doi:10.1007/s11071-021-06276-8
13. Li C, Yang Y, Yang X, Zi X, Xiao F. A Tristable Locally Active Memristor and its Application in Hopfield Neural Network. *Nonlinear Dyn* (2022) 108(2): 1697–717. doi:10.1007/s11071-022-07268-y
14. Xiong L, Zhang X, Teng S, Qi L, Zhang P. Detecting Weak Signals by Using Memristor-Involved Chua's Circuit and Verification in Experimental Platform. *Int J Bifurcation Chaos* (2020) 30(13):2050193. doi:10.1142/s021812742050193x
15. Hua Z, Zhang K, Li Y, Zhou Y. Visually Secure Image Encryption Using Adaptive-Thresholding Sparsification and Parallel Compressive Sensing. *Signal Process.* (2021) 183:107998. doi:10.1016/j.sigpro.2021.107998
16. Yousif B, Khalifa F, Makram A, Takieldeen A. A Novel Image Encryption/decryption Scheme Based on Integrating Multiple Chaotic Maps. *AIP Adv* (2020) 10(7):075220. doi:10.1063/5.0009225
17. Ye H-S, Zhou N-R, Gong L-H. Multi-image Compression-Encryption Scheme Based on Quaternion Discrete Fractional Hartley Transform and Improved Pixel Adaptive Diffusion. *Signal Process.* (2020) 175:107652. doi:10.1016/j.sigpro.2020.107652
18. Tang Z, Song J, Zhang X, Sun R. Multiple-image Encryption with Bit-Plane Decomposition and Chaotic Maps. *Opt Lasers Eng* (2016) 80:1–11. doi:10.1016/j.optlaseng.2015.12.004
19. Li X, Mou J, Cao Y, Banerjee S. An Optical Image Encryption Algorithm Based on a Fractional-Order Laser Hyperchaotic System. *Int J Bifurcation Chaos* (2022) 32(03):2250035. doi:10.1142/s0218127422500353
20. Yang F, An X, xiong L. A New Discrete Chaotic Map Application in Image Encryption Algorithm. *Phys Scr* (2022) 97(3):035202. doi:10.1088/1402-4896/ac4fd0
21. Zhang X, Li C, Dong E, Zhao Y, Liu Z. A Conservative Memristive System with Amplitude Control and Offset Boosting. *Int J Bifurcation Chaos* (2022) 32(04): 2250057. doi:10.1142/s0218127422500572
22. Li Y, Li C, Zhao Y, Liu S. Memristor-type Chaotic Mapping. *Chaos* (2022) 32(2):021104. doi:10.1063/5.0082983
23. Li Y, Li C, Zhang S, Chen G, Zeng Z. A Self-Reproduction Hyperchaotic Map with Compound Lattice Dynamics. *IEEE Trans Ind Electron* (2022) 69: 10564–72. doi:10.1109/TIE.2022.3144592
24. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Jin J, et al. Design and FPGA Implementation of a Pseudo-random Number Generator Based on a Hopfield Neural Network under Electromagnetic RadiationEnglish). *Front Phys* (2021) 9:690651. doi:10.3389/fphy.2021.690651
25. Li C, Li H, Xie W, Du J. A S-type Bistable Locally Active Memristor Model and its Analog Implementation in an Oscillator Circuit. *Nonlinear Dyn* (2021) 106(1):1041–58. doi:10.1007/s11071-021-06814-4
26. Sahasrabuddhe A, Laiphrakpam DS. Multiple Images Encryption Based on 3D Scrambling and Hyper-Chaotic System. *Inf Sci* (2021) 550:252–67. doi:10.1016/j.ins.2020.10.031

27. Rakheja P, Vig R, Singh P. Double Image Encryption Using 3D Lorenz Chaotic System, 2D Non-separable Linear Canonical Transform and QR Decomposition. *Opt Quant Electron* (2020) 52(2):103. doi:10.1007/s11082-020-2219-8
28. Bao BC, Zhu YX, Jun MA, Bao H, Huagan WU, Chen M. Memristive Neuron Model with an Adapting Synapse and its Hardware Experiments. *SCIENCE CHINA Technol Sci* (2021) 64(5):11. doi:10.1007/s11431-020-1730-0
29. Kong Dezhao 孔., Shen Xueju 沈., Lin Chao 林., Gao Yuchen 高. Multi-Image Encryption Based on Wavelet Transform and Fractional Fourier Transform. 激光与光电子学进展 (2013) 50(9):091002. doi:10.3788/lop50.091002
30. Kong D, Shen X, Xu Q, Xin W, Guo H. Multiple-image Encryption Scheme Based on Cascaded Fractional Fourier Transform. *Appl Opt* (2013) 52(12): 2619–25. doi:10.1364/ao.52.002619
31. Chen X-D, Liu Q, Wang J, Wang Q-H. Asymmetric Encryption of Multi-Image Based on Compressed Sensing and Feature Fusion with High Quality Image Reconstruction. *Opt Laser Tech* (2018) 107:302–12. doi:10.1016/j.optlastec.2018.06.016
32. Huang Z-J, Cheng S, Gong L-H, Zhou N-R. Nonlinear Optical Multi-Image Encryption Scheme with Two-Dimensional Linear Canonical Transform. *Opt Lasers Eng* (2020) 124:105821. doi:10.1016/j.optlaseng.2019.105821
33. Singh N, Sinha A. Chaos Based Multiple Image Encryption Using Multiple Canonical Transforms. *Opt Laser Tech* (2010) 42(5):724–31. doi:10.1016/j.optlastec.2009.11.016
34. Banerjee S, Mukhopadhyay S, Rondoni L. Multi-image Encryption Based on Synchronization of Chaotic Lasers and Iris Authentication. *Opt Lasers Eng* (2012) 50(7):950–7. doi:10.1016/j.optlaseng.2012.02.009
35. Aparna H, Bhumijaa B, Santhiyadevi R, Vaishanavi K, Sathanarayanan M, Rengarajan A, et al. Double Layered Fridrich Structure to Conserve Medical Data Privacy Using Quantum Cryptosystem. *J Inf Security Appl* (2021) 63: 102972. doi:10.1016/j.jisa.2021.102972
36. Gao X, Mou J, Banerjee S, Cao Y, Xiong L, Chen X. An Effective Multiple-Image Encryption Algorithm Based on 3D Cube and Hyperchaotic Map. *J King Saud Univ - Comp Inf Sci* (2022) 34:1535–51. doi:10.1016/j.jksuci.2022.01.017
37. Wu Y, Zhou Y, Saveriades G, Agaian S, Noonan JP, Natarajan P. Local Shannon Entropy Measure with Statistical Tests for Image Randomness. *Inf Sci* (2013) 222:323–42. doi:10.1016/j.ins.2012.07.049
38. Zhang X, Hu Y. Multiple-image Encryption Algorithm Based on the 3D Scrambling Model and Dynamic DNA Coding. *Opt Laser Tech* (2021) 141: 107073–88. doi:10.1016/j.optlastec.2021.107073
39. Gao X, Mou J, Xiong L, Sha Y, Yan H, Cao Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn* (2022) 108:613–36. doi:10.1007/s11071-021-07192-7
40. Zarebnia M, Parvaz R. Dynamical 2D and 3D Image Encryption Method by Hybrid System Based on Cat Map and Wavelet Transform. *Optik* (2020) 219: 165148. doi:10.1016/j.ijleo.2020.165148
41. Patro KAK, Acharya B. An Efficient Dual-Layer Cross-Coupled Chaotic Map Security-Based Multi-Image Encryption System. *Nonlinear Dyn* (2021) 104(3): 2759–805. doi:10.1007/s11071-021-06409-z
42. Huang W, Jiang D, An Y, Liu L, Wang X, "A Novel Double-Image Encryption Algorithm Based on Rossler Hyperchaotic System and Compressive Sensing, *IEEE Access* (2021) 9, 41704–16. doi:10.1109/access.2021.3065453
43. Zhang L, Zhang X. Multiple-image Encryption Algorithm Based on Bit Planes and Chaos. *Multimedia Tools Appl* (2020) 79(29-30):20753–71. doi:10.1007/s11042-020-08835-4
44. Xian Y, Wang X. Fractal Sorting Matrix and its Application on Chaotic Image Encryption. *Inf Sci* (2021) 547:1154–69. doi:10.1016/j.ins.2020.09.055
45. Alawida M, Teh JS, Samsudin A, Alshoura WH. An Image Encryption Scheme Based on Hybridizing Digital Chaos and Finite State Machine. *Signal Process.* (2019) 164:249–66. doi:10.1016/j.sigpro.2019.06.013
46. Hua Z, Zhou Y. Image Encryption Using 2D Logistic-Adjusted-Sine Map. *Inf Sci* (2016) 339:237–53. doi:10.1016/j.ins.2016.01.017
47. Yue W, Noonan JPSos Agaian. NPCR and UACI Randomness Tests for Image Encryption. *J Selected Areas Telecommunications* (2011) 1(2):31–38.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Physics

ORIGINAL RESEARCH
published: 15 June 2022
doi: 10.3389/fphy.2022.904200

The Simplest Memristor Circuit With Hyperchaos

Xingce Liu[1] *and Jinpeng Wang*[2]*

[1]*School of Mechanical Engineering and Automation, Dalian Polytechnic University, Dalian, China,* [2]*The School of Information Science and Engineering, Dalian Polytechnic University, China*

In recent years, with the in-depth study of the memristor, meminductor, and memcapacitor, the fourth basic element has been developed vigorously. The chaotic circuit including the meminductor, memcapacitor, and memristor is designed in this study. The equation of state for the chaotic system is obtained according to Kirchhoff's volt–current law, and the mathematical model of the chaotic system is obtained through dimensionless processing. The stability of the equilibrium point is analyzed in detail, and the dynamic behaviors of the system are analyzed by traditional methods such as LEs and bifurcation diagram. Moreover, some special phenomena exist in the system, such as state transition and coexistence of attractors. Finally, the circuit is implemented by DSP to prove the realizability of chaotic circuit.

Keywords: memristor, meminductor, memcapacitor, coexisting attractors, state transition, DSP implementation

OPEN ACCESS

Edited by:
Chunlai Li,
Hunan Institute of Science and Technology, China

Reviewed by:
Fei Yu,
Changsha University of Science and Technology, China
Shuang Zhou,
Chongqing Normal University, China

***Correspondence:**
Jinpeng Wang
wangjp@dlpu.edu.cn

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *25 March 2022*
Accepted: *19 April 2022*
Published: *15 June 2022*

Citation:
Liu X and Wang J (2022) The Simplest Memristor Circuit With Hyperchaos. Front. Phys. 10:904200. doi: 10.3389/fphy.2022.904200

INTRODUCTION

A chaotic system is a kind of complex nonlinear dynamic system, and its motion has pseudo-randomness. The orbit moves in a specific range but never repeats. Chaos is often described as a specific phenomenon with an infinite period or a phenomenon similar to random behavior. According to the growing understanding of chaos, people found that chaos includes not only integer-order chaotic system but also fractional-order chaotic system [1–7].In addition, people usually study chaos by constructing chaotic circuits. The nonlinear element is an important part of the nonlinear system.

The memristor is a kind of nonlinear element [8]. Professor Chua presented a special circuit element and suggested that it could be used to describe the relationship between charge value and magnetic flux in 1971 and it is called a memristor [9]. From then on, Professor Chua proposed the concept of memristor for several decades, and no memristor element has been found in practice. So research on the fourth basic element has been stalled. Until 2008, the HP Laboratory in the United States claimed to have completed the hardware realization of memristor components on the nanoscale for the first time, which eventually led to booming research on the fourth type of components [10, 11]. Many research studies on chaotic circuits are carried out from this [12–16]. However, at present, because the hardware implementation of the memristor in the real sense can only be completed at the laboratory level, the application research of the memristor is mainly based on its mathematical model or the circuit equivalent model.

Through continuous research on the memristor, in 2009, Professor Chua presented the meminductor and memcapacitor, two memory elements, which further deepen the research on nonlinear systems. Although the basic principles of meminductor and memcapacitor components have been understood, there is no material that can be used to manufacture these two basic components in nature, so the hardware implementation of the meminductor and memcapacitor is still at a standstill until now. However, due to its unique memory characteristics, many theoretical analyses and research studies on meminductor and memcapacitor components have been completed. The chaotic circuits based on the meminductor and memcapacitor are rarely reported [17–20].

Because of their distinctively dynamic characteristics, the memristor, meminductor, and memcapacitor have been used in chaotic systems. In 2008, a memristor was used to replace Chua's diode to design the chaotic system, and numerical simulation analysis and circuit implementation were carried out [21]. In 2012, Professor Chua first proposed a simple-series chaotic circuit based on a memristor and implemented it. The circuit consists only of a memristor, an inductor, and a capacitor in series [22]. In 2013, Professor Xu modified the simplest chaotic circuit. By connecting the inductor, memristor, and capacitor in parallel, he proposed a minimalist parallel chaotic circuit and simulated it [23]. In 2019, Professor Yuan constructed the simplest parallel chaotic circuit by connecting the memcapacitor, meminductor, and memristor in parallel. Its dynamic characteristics were not only analyzed but also the circuit simulation was carried out [24]. The chaotic system based on the memristor and other memory elements has better application in a neural network, chaotic secure communication, and image encryption due to its excellent dynamic characteristics [25–39].

In this study, a chaotic system including the meminductor, memcapacitor, and memristor is proposed. Because of the special dynamic characteristics of the memristor, meminductor, and memcapacitor, compared with the general chaotic oscillator, it has more dynamic characteristics [40–43]. Finally, digital circuits are less affected by the environment and have greater flexibility in parameter changes. Therefore, a complete set of dynamic analyses and implementation of a chaotic oscillator is carried out by DSP in this study to prove the realizability of the circuit [44–52].

The rest of the study follows according to the abovementioned background. In the second section, the model of the memcapacitor, memristor, and meminductor is presented. In the third section, the state equation is determined by the constructed chaotic circuit, the stability of the equilibrium point is analyzed, and the dynamical characteristics of the system are analyzed through the traditional way. In the fourth section, the circuit is implemented by DSP. In the fifth section, the relevant work of this study is summarized.

MODELS OF THE MEMCAPACITOR, MEMINDUCTOR, AND MEMRISTOR

Model of the Memristor

This study uses the model of the classic memristor proposed by Professor Chua [53]. The equation of state is shown below:

$$\begin{cases} V_M = a\left(y^2 - 1\right)i_M \\ \dot{y} = i_M - by - i_M y \end{cases}, \tag{1}$$

In the equation, i_{M} represents the current of the memristor, V_{M} represents the voltage of the memristor, b and a represent the parameters, and y represents the state variables of the memristor.

An AC power supply with a frequency of 10 Hz is assumed and an effective value of 10 A is input at both ends of the memristor. Through numerical simulation of the memristor, we can obtain its *v-i* memory curve, as shown in **Figure 1A**. The simulation results show that its characteristic curve is a compact "8″ hysteretic curve. As the frequency increases, the side lobe area of the characteristic curve decreases. This is consistent with the basic characteristics of the memristor proposed by Professor Chua.

Model of the Memcapacitor

By professor Chua's definition of the memcapacitor, it is a passive memory element [54]. According to the definition of the memcapacitor, the current passing through it can be expressed as i, the charge gathered at both ends of the memcapacitor can be expressed as q, and the time integral of charge quantity q can be expressed as σ. So, the state equation of the memcapacitor element is shown as follows:

$$\begin{cases} v(t) = C_{\mathrm{M}}^{-1}(\sigma)q(t) \\ C_{\mathrm{M}}^{-1}(\sigma) = c + d\cos(\sigma) \\ \sigma = \displaystyle\int_{t_0}^{t} q(\tau)\mathrm{d}\tau \\ q = \displaystyle\int_{t_0}^{t} i(\tau)\mathrm{d}\tau \end{cases}, \tag{2}$$

For the model of the memcapacitor mentioned above, when the input of the memcapacitor is an AC power supply with a frequency of 10 Hz and an effective value of 2A, the curve of the dynamic characteristic of the memcapacitor can be obtained through numerical simulation, as shown in **Figure 2**. This is a compact hysteresis curve with a shape similar to an inclined "8", and the side lobe area decreases with increasing frequency. This is consistent with the basic properties of memcapacitors.

Model of the Meminductor

By professor Chua's definition of the meminductor [54], its model used in this study is as follows:

$$\begin{cases} i(t) = L_{\mathrm{M}}^{-1}(\rho)\phi(t) \\ L_{\mathrm{M}}^{-1}(\rho) = e + g\rho \\ \rho = \displaystyle\int_{t_0}^{t} \phi(\tau)\mathrm{d}\tau \end{cases}, \tag{3}$$

The ρ in the equation is the integral of magnetic flux φ, where e and g represent the internal parameters of the meminductor.

A sinusoidal signal with a magnetic flux of 10 Wb and a frequency of 5 Hz is applied at both ends of the meminductor, and the characteristic curve of the meminductor element *Φ-i* is shown in **Figure 3A**. The curve of the meminductor is also a slanted hysteresis curve similar to "8". Through numerical

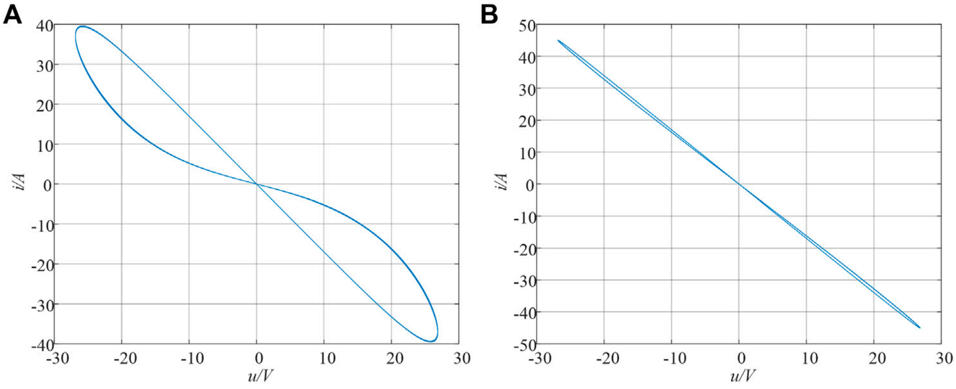


FIGURE 1 | *v-i* curve of the memristor with *b* = 2.6, *a* = 2 **(A)** *f* = 10 Hz. **(B)** *f* = 50 Hz.

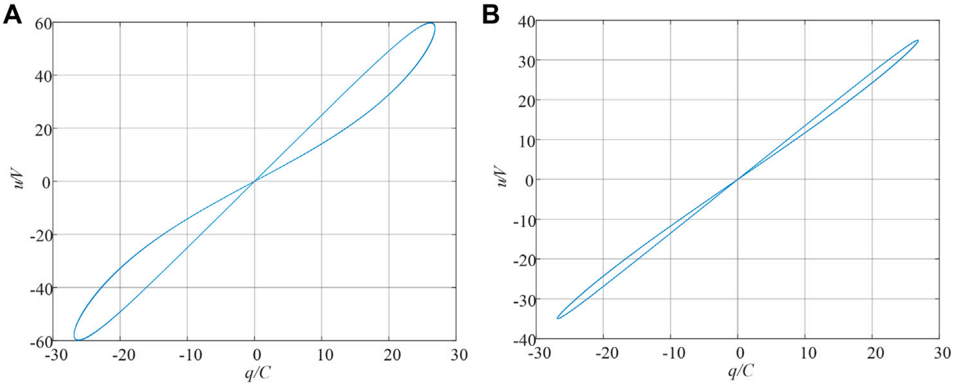


FIGURE 2 | Hysteresis curve of the memcapacitor, **(A)** *c* = 1.7, *d* = 2, *f* = 10 Hz. **(B)** *c* = 1.7, *d* = 2, *f* = 80 Hz.

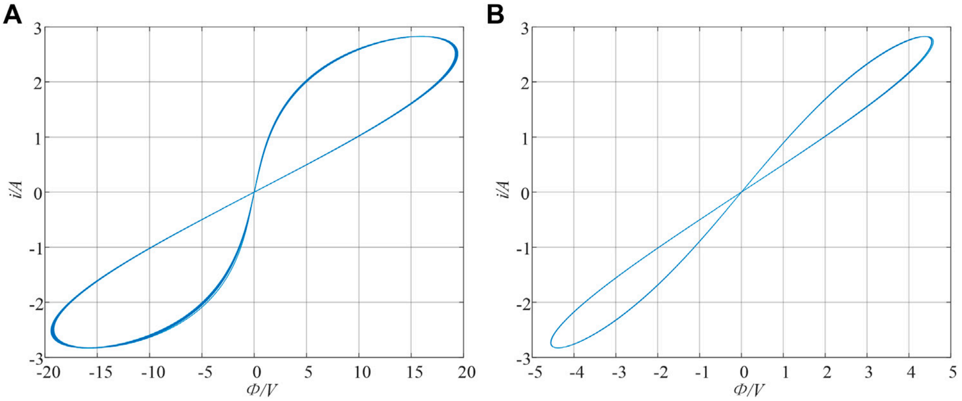


FIGURE 3 | Hysteresis curve of the meminductor, **(A)** *e* = 1, *g* = 1, *f* = 5 Hz. **(B)** *e* = 1, *g* = 1, *f* = 20 Hz.

simulation, it was found that the sidelobe area of the characteristic curve of the meminductor gradually decreased with the increase of input frequency, which was consistent with the dynamic characteristics of the meminductor proposed by Professor Chua.

A NEW SIMPLE CHAOTIC OSCILLATOR

A New Chaotic Oscillator

In this circuit, there are three nonlinear elements, namely, an independent charge-controlled memcapacitor, a charge-

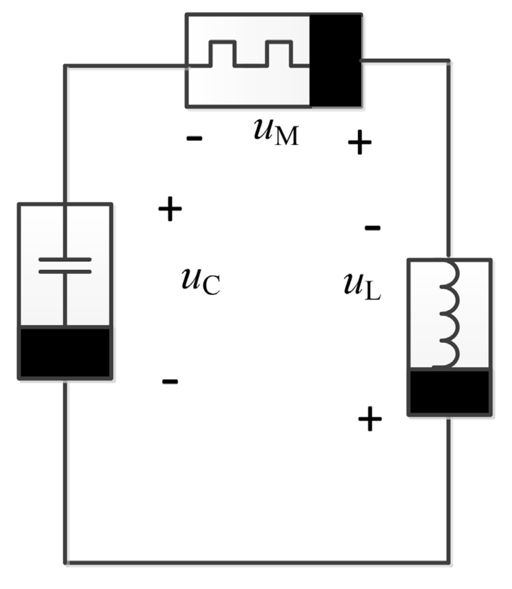


FIGURE 4 | Circuit of the chaotic system.

controlled memristor, and a meminductor. The circuit can be expressed as **Figure 4**:

The circuit includes three kinds of nonlinear elements, meminductor, memristor, and memcapacitor. Suppose the charge flowing through the memcapacitor is q, φ denotes the magnetic flux, the internal state variable of the memristor is y, the time integral of the charge quantity q is denoted as σ, and the time integral of the magnetic flux φ is denoted as ρ. Then, according to the characteristic equations of each nonlinear element and Kirchhoff's voltage–current law, the equation of the circuit was obtained as follows:

$$\begin{cases} \dot{\varphi} = -u_c - u_M \\ \dot{y} = i - dy - iy \\ \dot{q} = i \\ \dot{\sigma} = q \\ \dot{\rho} = \varphi \end{cases}, \tag{4}$$

Five nonlinear equations are included in system (4), which contains five variables φ, y, q, σ, and ρ. Set $\sigma = u$, $\varphi = x$, $q = z$, $y = y$, and $\rho = w$. Moreover, dimensionless processing is performed on **Eq. 4**, and then **Eq. 4** becomes

$$\begin{cases} \dot{x} = -(e + gu)z - a\left(y^2 - 1\right)(c + d\cos w)x \\ \dot{y} = (c + d\cos w)x - by - (c + d\cos w)xy \\ \dot{z} = (c + d\cos w)x \\ \dot{u} = z \\ \dot{w} = x \end{cases}, \tag{5}$$

When the initial conditions are set as (1, 40, 1.5, 20, -15) and the parameters are set as $b = 20$, $a = 49$, $g = 2$, $e = 100$, $c = 0.2$, and $d = -0.02$, the phase diagram simulated by MATLAB is expressed in **Figure 5**.

Equilibrium Point Set and Stability

The calculation method of divergence of the chaotic system is shown as

$$\begin{aligned} \nabla V &= \frac{\partial \dot{x}}{\partial x} + \frac{\partial \dot{y}}{\partial y} + \frac{\partial \dot{z}}{\partial z} + \frac{\partial \dot{u}}{\partial u} + \frac{\partial \dot{w}}{\partial w} \\ , \quad &= -a\left(y^2 - 1\right)(c + d\cos w) - b - (c + d\cos w)x. \end{aligned} \tag{6}$$

For the parameters set as $a = 49$, $b = 20$, $g = 2$, $e = 100$, $c = 0.2$, and $d = -0.02$. The divergence of the system is calculated to be -16,881. This means that the system can produce chaos.

Let the system state variable $\dot{x} = \dot{y} = \dot{z} = \dot{u} = \dot{w} = 0$, we can obtain the calculation result as follows:

$$\begin{cases} -(e + gu)z - a\left(y^2 - 1\right)(c + d\cos w)x = 0 \\ (c + d\cos w)x - by - (c + d\cos w)xy = 0 \\ (c + d\cos w)x = 0 \\ z = 0 \\ x = 0 \end{cases}, \tag{7}$$

Through calculation, the equilibrium point of the system is obtained as $O(0, 0, 0, n, m)$. It shows that the system had an infinite number of equilibrium points. According to **Eq. 7**, the Jacobian matrix J_E of the system can be obtained

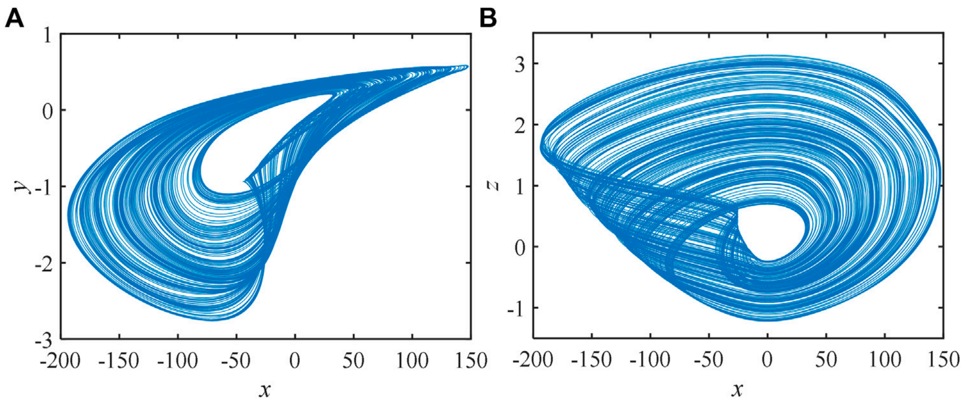


FIGURE 5 | Phase diagram **(A)** x–y plane **(B)** x–z plane.

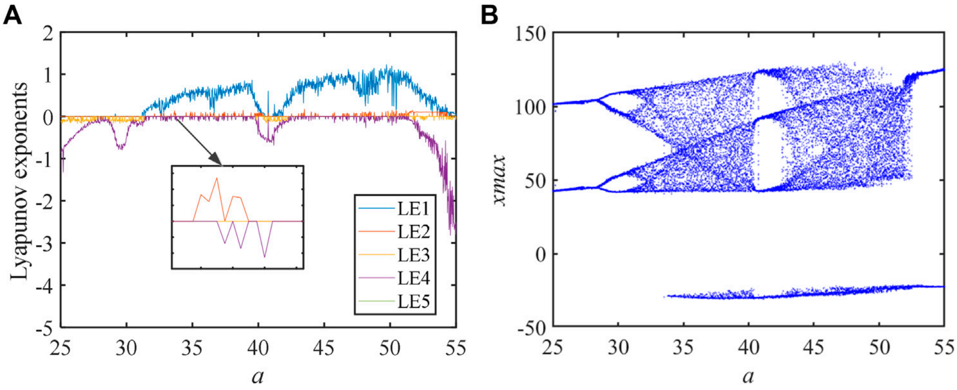


FIGURE 6 | Bifurcation diagram and LEs with parameter *a*, **(A)** LEs, and **(B)** bifurcation diagram.

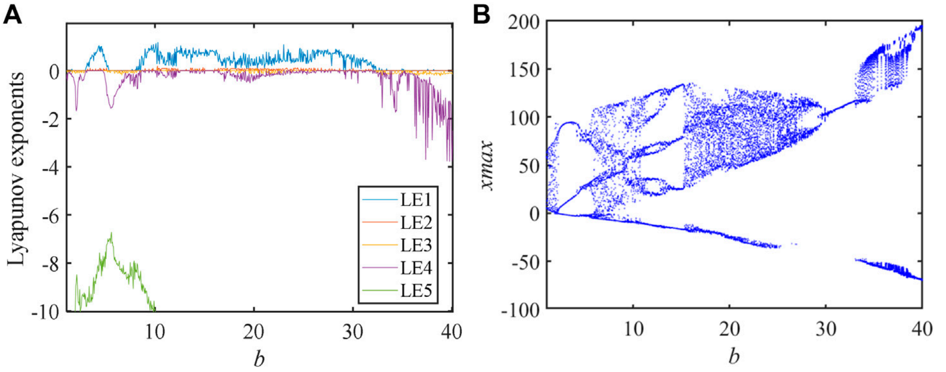


FIGURE 7 | Bifurcation diagram and LEs with parameter *b*, **(A)** LEs, and **(B)** bifurcation diagram.

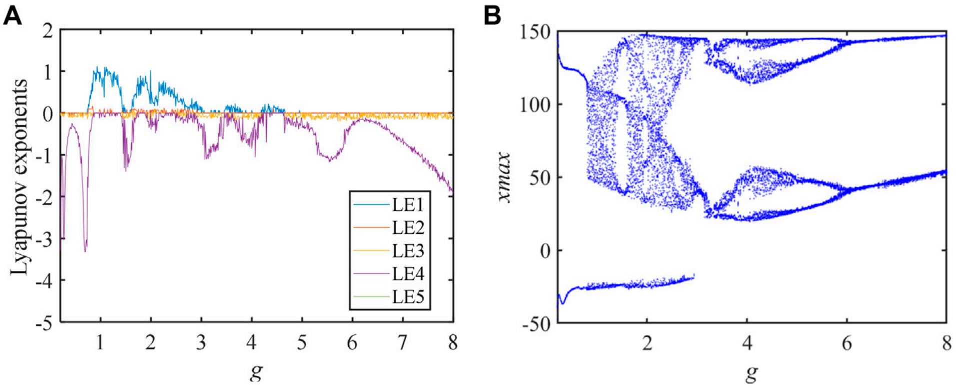


FIGURE 8 | Bifurcation diagram and LEs with parameter *g*, **(A)** LEs, and **(B)** bifurcation diagram.

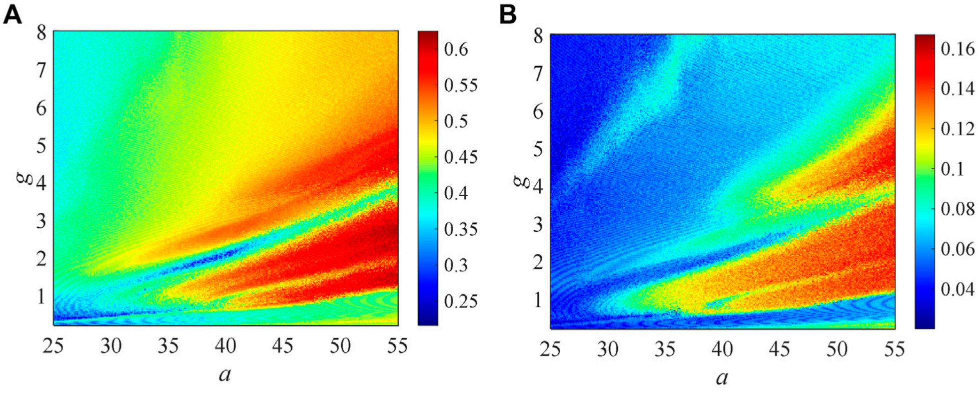


FIGURE 9 | Complexity of the system. **(A)** SE complexity; **(B)** C0 complexity.

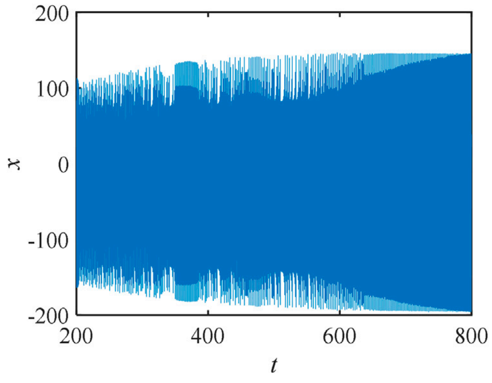


FIGURE 10 | Timing diagram with time *t*=(200, 800).

$$J_{\mathrm{E}} = \begin{bmatrix} -a\left(y^2-1\right)(c+d\cos w) & -2ay(c+d\cos w)x & -(e+gu) & -gz & ad\left(y^2-1\right)\sin(w)x \\ (c+d\cos w)-(c+d\cos w)y & -b-(c+d\cos w)x & 0 & 0 & -d\sin(w)x+d\sin(w)xy \\ (c+d\cos w) & 0 & 0 & 0 & gx \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix},$$
$$= \begin{bmatrix} a(c+d\cos m) & 0 & -(e+gn) & 0 & 0 \\ (c+d\cos m) & -b & 0 & 0 & 0 \\ (c+d\cos m) & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{8}$$

Therefore, the characteristic polynomial of the system can be obtained through the Jacobian matrix of the system.

$$\lambda^2\left(\lambda^3 + m_1\lambda^2 + m_2\lambda + m_3\right) = 0, \tag{9}$$

where $m_1 = b\text{-}a(c + d\cos m)$, $m_2 = [(c + d\cos m)\ (e + gn)\text{-}ab(c + d\cos m)]$, and $m_3 = b((c + d\cos m)\ (c + dn))$.

Equation 9 means system (5) has three non-zero eigenvalues and one zero eigenvalue, where m_1, m_2, and m_3 are coefficients. By the Routh–Hurwitz criterion, system (5) is stable if $m_1 > 0$, $m_1 m_2\text{-}m_3 > 0$, and $m_3(m_1 m_2\text{-}m_3) > 0$. At least one of the eigenvalues of the chaotic systems is positive, so the m_1, $m_1 m_2 m_3$, and m_3 should be not all positive. So, we could set O such as O_1 (0, 0, 0, 20, -15). Let the parameters be $a = 49$, $b = 20$, $g = 2$, $e = 100$, $c = 0.2$, and $d = -0.02$, and we can get $m_1 = 9.4552 > 0$, $m_2 = -180.768 > 0$, $m_3 = 602.56 > 0$, $m_1 m_2\text{-}m_3 < 0$, $m_3(m_1 m_2\text{-}m_3) < 0$ and $\lambda_1 = \lambda_2 = 0$, $\lambda_3 = -20$, $\lambda_4 = 5.2722 + 1.5266i$, $\lambda_5 = 5.2722–1.5266i$. This means that the system is unstable, and it is possible for such a system to produce chaos.

The Impact of the Parameters

To display the dynamical behaviors of the system designed by us more directly, we use the bifurcation diagram and LEs to study the dynamic behaviors of the system when parameter a changes. Other parameters of the chaotic system are set as $d = -0.02$, $g = 2$, $e = 100$, $b = 20$, and $c = 0.2$, and the initial condition is 1, 40, 1.5, 20, and -15. The calculation results are as follows **Figure 6**.

According to the LEs and bifurcation diagram, we can understand that the bifurcation diagram and LEs can correspond to each other. According to the definition of LEs, the system has two positive Lyapunov exponents in a certain parameter range, which shows that the system is a hyperchaotic system. According to the bifurcation diagram, with the change of parameter a, the system enters into a chaotic state by means of period-doubling bifurcation.

To study the dynamic behaviors of the simple system parameter b is treated as a variable. Other parameters are set as $a = 49$, $g = 2$, $e = 100$, $c = 0.2$, and $d = -0.02$, and the initial values are 1, 40, 1.5, 20, and −15. The results of numerical simulation are expressed in **Figure 7**.

By observing the LEs, we can see that the bifurcation diagram and LEs agree perfectly. From LEs, we can find simply that the system not only has a chaotic state but also has a hyperchaotic state. According to the bifurcation diagram, when the parameter b increases, the system enters into chaos by period-doubling bifurcation.

When the system parameters $a = 49$, $b = 20$, $e = 100$, $c = 0.2$, and $d = -0.02$ and the initial values are 1, 40, 1.5, 20, −15, just the parameter g is changed. The bifurcation diagram and LEs are expressed in **Figure 8**.

By the analysis of LEs, we can know that when the parameter g increases, the system changes frequently between periodic state, chaotic state, and hyperchaotic state. Also, the system is produced by a counter-periodic doubling of the chaotic state. The bifurcation diagram and LEs agree well.

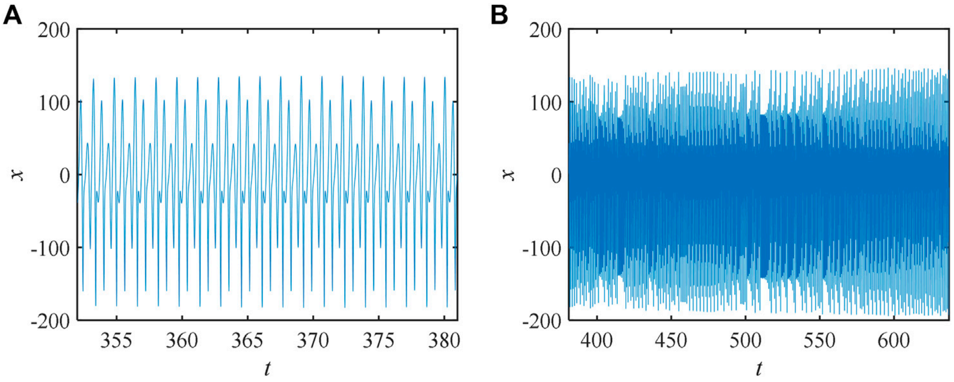


FIGURE 11 | Timing diagram, **(A)** $t\in(352, 381)$; **(B)** $t\in(381, 637)$.

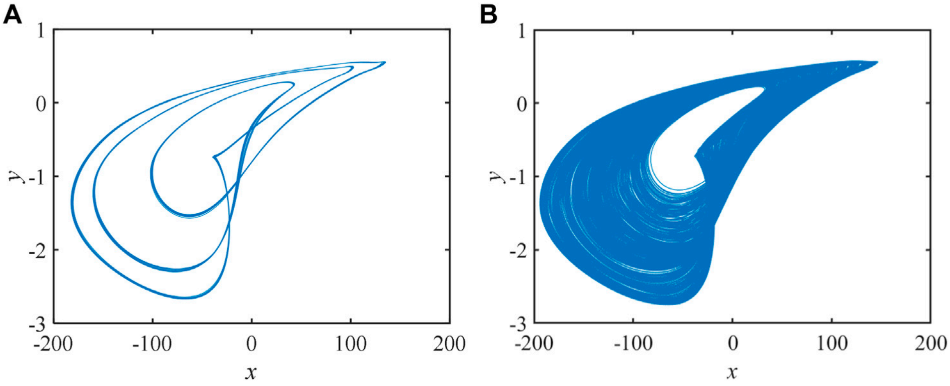


FIGURE 12 | Phase diagram, **(A)** $t\in(352, 381)$; **(B)** $t\in(381, 637)$.

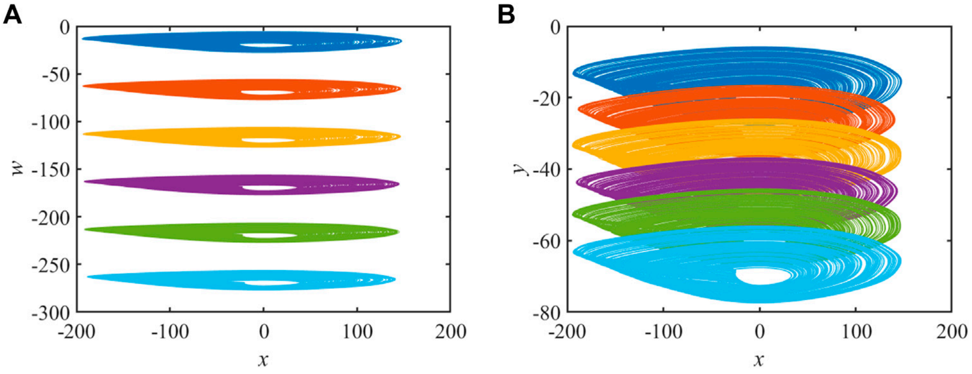


FIGURE 13 | Coexisting attractors with different z_0 and w_0.

Complexity Analysis

The chaotic state and hyperchaotic state are the most valuable intervals of a chaotic system. When the system is chaotic or hyperchaotic, the sequence generated by the system has high pseudo-randomness, which makes the system have good anti-decoding performance. When the system shows higher complexity, its dynamic performance is also superior. So, in image encryption, people mostly choose the parameter range with higher complexity. To further understand the dynamic behaviors, two complexity algorithms C0 and SE(spectral entropy) are used to analyze the system complexity. The results show that the SE and C0 algorithms reflect the

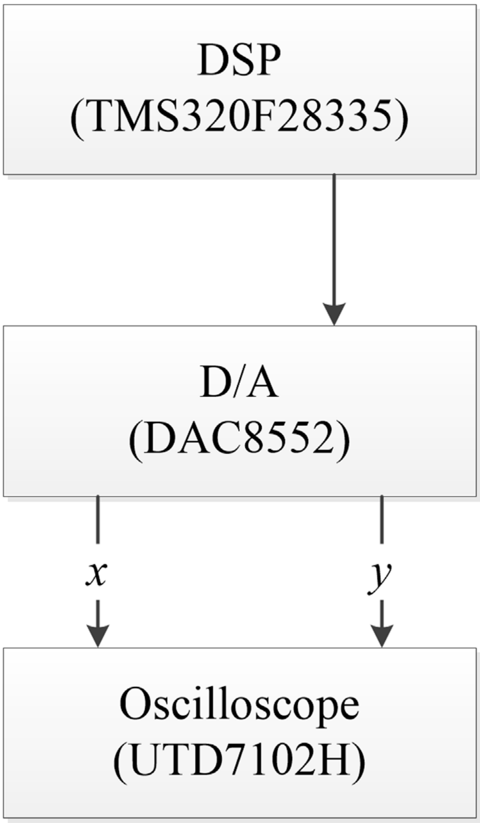


FIGURE 14 | Flow chart.

dynamic behaviors. The initial values are set as 1, 40, 1.5, 20, and -15 and the system parameters $b = 20$, $e = 100$, $d = -0.02$, $c = 0.2$ are kept unchanged. Only the system parameters a and g are changed. As system parameters a and g change, SE and C0 complexity is shown in **Figures 9A,B**. When the system parameters $a = [24, 54]$ and $g = (0.2, 8)$, the complexity value of the chaotic system changes with parameters a and g, as shown in **Figure 9**, and the periodic state and chaotic state of the system can be clearly displayed. By analyzing the complexity of the system, we can know the changing rules of the dynamical characteristics of the system. Also, because these two complexity algorithms are calculated based on the FFT complexity algorithm, their variation trend is roughly the same.

State Transition

Leaving the initial values and parameters unchanged, when the state changes with time, this phenomenon is called state transition. The special phenomenon of state transition exists in the system constructed in the study. Let the system parameters $a = 49$, $g = 2$, $e = 100$, $c = 0.2$, $d = -0.02$, and $b = 20$ and the initial condition be 1, 40, 1.5, 20, and −15. The timing diagram of the system is expressed in **Figure 10**:

The detailed sequence diagram of periodic and chaotic states is expressed in **Figure 11**:

When $t = [352, 381]$ and $t = [381, 637]$, the simulation results are shown as follows **Figure 12**:

Coexistence of Attractors

Coexistence of attractors is a special phenomenon of chaotic systems [55–57]. For the chaotic circuit system with the memristor, meminductor, and memcapacitor, let $a = 49$, $b = 20$, $e = 100$, $c = 0.2$, $d = -0.02$, $g = 2$, and simulation step $t = 0.001$s. In addition, let the initial condition be 1, 20, z_0, 20, and w_0. When the initial value is changed, the results are shown in **Figure 13**:

The initial value is set to (1, 40, 1.5, 20, −15), (1, 40, 11.5, 20, −15), (1, 40, 21.5, 20, −15), (1, 40, 31.5, 20, −15), (1, 40, 41.5, 20, −15) and (1, 40, 51.5, 20, −15). The simulation results of the system are shown in **Figure 13A**:

Let the initial values be (1, 40, 1.5, 20, −15), (1, 40, 1.5, 20, −25), (1, 40, 1.5, 20, −35), (1, 40, 1.5, 20, −45), (1, 40, 1.5, 20, −55) and (1, 40, 1.5, 20, −65), and the numerical simulation results are expressed in **Figure 13B**.

DSP IMPLEMENTATION

Because DSP implementation has strong anti-interference ability, the parameters in the system are also better controlled. The DSP is called

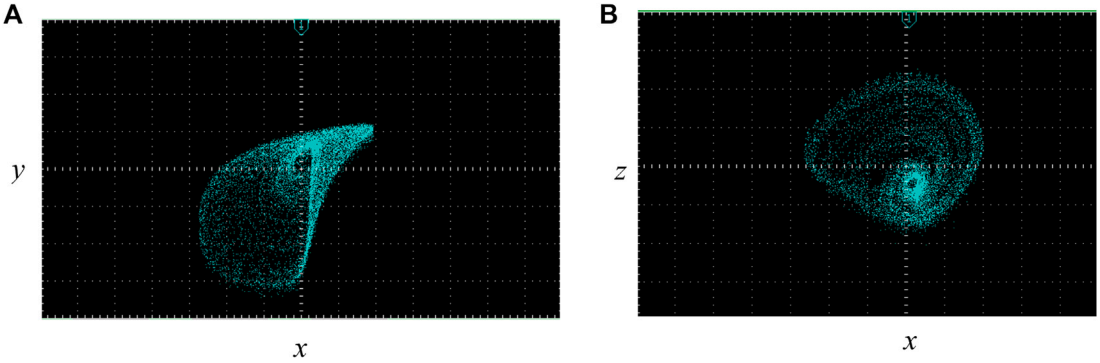


FIGURE 15 | Experimental results of DSP implementation. **(A)** x–y plane; **(B)** x–w plane.

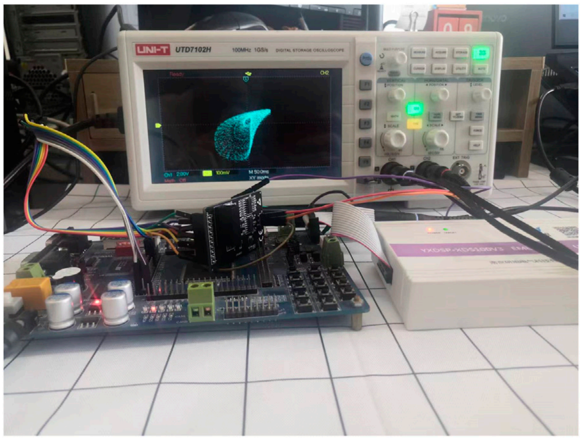


FIGURE 16 | Experimental platform for DSP implementation.

digital signal processor. Digital signal processing is the use of computers or special processing equipment in the digital form of signal acquisition, transformation, filtering, valuation, enhancement, compression, recognition, and other processing in order to get the signal form that meets people's needs. We choose to use the DSP to achieve the chaotic system. However, because of the limited precision of DSP hardware implementation, we choose the TMS320F28335 chip so as to meet the requirements of system precision. To make the oscilloscope capture the image more easily, we use a D/A converter to convert the DSP-generated sequence into the analog sequence and then transmit the signal to the oscilloscope (UTD7102H) through the D/A converter (8552). The flow chart of DSP implementation is shown in **Figure 14**.

Let its parameters be $a = 49$, $b = 20$, $e = 100$, $c = 0.2$, $d = -0.02$, and $g = 2$. The initial value is (1, 40, 1.5, 20, −15). First, the parameters and initial value are set and then iterative calculation is started. The data generated by each iteration are stored using stack techniques to avert corruption. The results of DSP implementation are shown in **Figure 15**.

Hardware such as oscilloscope, D/A converter, and F28335 chip used in DSP implementation are expressed in **Figure 16**.

CONCLUSION

In this study, the circuit with the memcapacitor, memristor, and meminductor is constructed. The study not only introduces the model of the meminductor, memcapacitor, and memristor but also obtains the dynamics equation of the system by Kirchhoff's voltage–current law. The stability of the equilibrium point is analyzed by means of traditional dynamics. Through the analysis of LEs, complexity, and bifurcation diagram, it is found that when the initial conditions and parameters change, the chaotic system shows complex dynamical behavior. Also, the system has not only a chaotic state but also a hyperchaotic state. In addition, some special phenomena such as state transition and coexistence of attractors are found in the analysis of the systems. Finally, the DSP is used to realize the chaotic circuit, and the simulation diagram in the DSP is highly consistent with the results of numerical simulation. The rich dynamic characteristics show that it can be used in chaotic image encryption.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Materials, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

XL designed and carried out experiments, analyzed data, and wrote the manuscript. JW made the theoretical guidance for this study. All authors reviewed the manuscript.

FUNDING

This work was supported by the National Natural Science Foundation of China (Grant Nos. 62061014 and 61773010); the Natural Science Foundation of Liaoning province (2020-MS-274); the Basic Scientific Research Projects of Colleges and Universities of Liaoning Province (Grant No. J202148).

REFERENCES

1. Zhang D, Zhao A, Yang X, Sun Y, Xiao J. Generalized Synchronization between Chen System and Rucklidge System. *IEEE Access* (2019) 7: 8519–26. doi:10.1109/access.2018.2890311
2. Liu X, Bi X, Yan H. A Chaotic Oscillator Based on Meminductor, Memcapacitor, and Memristor[J]. *Complexity* (2021) 1. doi:10.1155/2021/7223557
3. Liu T, Yan H, Banerjee S, Mou J. A Fractional-Order Chaotic System with Hidden Attractor and Self-Excited Attractor and its DSP Implementation. *Chaos, Solit Fractals* (2021) 145:110791. doi:10.1016/j.chaos.2021.110791
4. Liu T, Banerjee S, Yan H, Mou J. Dynamical Analysis of the Improper Fractional-Order 2D-SCLMM and its DSP Implementation. *Eur. Phys. J. Plus* (2021) 136(5):506. doi:10.1140/epjp/s13360-021-01503-y
5. Ma C, Mou J, Li P, Liu T. Dynamic Analysis of a New Two-Dimensional Map in Three Forms: Integer-Order, Fractional-Order and Improper Fractional-Order. *Eur Phys J Special Top* (2021) 230(7):1945–57. doi:10.1140/epjs/s11734-021-00133-w
6. Peng Y, He S, Sun K. Parameter Identification for Discrete Memristive Chaotic Map Using Adaptive Differential Evolution Algorithm[J]. *Nonlinear Dyn*(2021) 1–13.
7. Ma C, Mou* J, Xiong L, Banerjee S, Liu T, Han X. Dynamical Analysis of a New Chaotic System: Asymmetric Multistability, Offset Boosting Control and Circuit Realization. *Nonlinear Dyn*(2021) 103(3):2867–80. doi:10.1007/s11071-021-06276-8
8. Prodromakis T, Peh BP, Papavassiliou C, Toumazou C. A Versatile Memristor Model with Nonlinear Dopant Kinetics. *IEEE Trans. Electron Devices* (2011) 58(9):3099–105. doi:10.1109/ted.2011.2158004
9. Chua L. Memristor-the Missing Circuit Element. *IEEE Trans. Circuit Theory* (1971) 18(5):507–19. doi:10.1109/tct.1971.1083337
10. Strukov DB, Snider GS, Stewart DR, Williams RS. The Missing Memristor Found. *Nature* (2008) 453(7191):80–3. doi:10.1038/nature06932
11. Tour JM, He T. The Fourth Element. *Nature* (2008) 453(7191):42–3. doi:10.1038/453042a

12. Li C, Li H, Xie W, Du J. A S-type Bistable Locally Active Memristor Model and its Analog Implementation in an Oscillator Circuit. *Nonlinear Dyn*(2021) 106(1):1041–58. doi:10.1007/s11071-021-06814-4
13. Liu J, Chen G, Zhao X. Generalized Synchronization and Parameters Identification of Different-Dimensional Chaotic Systems in the Complex Field[J]. *Fractals* (2021) 29(4):2150081–824. doi:10.1142/s0218348x2150081x
14. Li Y, Li C, Zhang S. A Self-Reproduction Hyperchaotic Map with Compound Lattice Dynamics[J]. *IEEE Trans Industrial Electron* (2022):1. doi:10.1109/tie.2022.3144592
15. Li Y, Li C, Zhao Y, Liu S. Memristor-type Chaotic Mapping. *Chaos* (2022) 32(2):021104. doi:10.1063/5.0082983
16. Zhang X, Li C, Dong E. A Conservative Memristive System with Amplitude Control and Offset Boosting[J]. *Int J Bifurcation Chaos* (2022) 32(04):2250057. doi:10.1142/s0218127422500572
17. ZhenYu Yin ZY, Heng Tian H, GuanHua Chen GH, Chua LO. What Are Memristor, Memcapacitor, and Meminductor? *IEEE Trans. Circuits Syst. II* (2015) 62(4):402–6. doi:10.1109/tcsii.2014.2387653
18. Ma X, Mou J, Liu J, Ma C, Yang F, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor-Memcapacitor. *Nonlinear Dyn*(2020) 100(3):2859–76. doi:10.1007/s11071-020-05601-x
19. Hu Z, Li Y, Li J, Yu J. Chaotic Oscillator Based on Voltage-Controlled Memcapacitor. In: *International Conference on Communications* (2010). doi:10.1109/icccas.2010.5581863
20. Feali MS, Ahmadi A, Hayati M. Implementation of Adaptive Neuron Based on Memristor and Memcapacitor Emulators. *Neurocomputing* (2018) 309(OCT.2):157–67. doi:10.1016/j.neucom.2018.05.006
21. Itoh M, Chua LO. Memristor Oscillators. *Int. J. Bifurc Chaos* (2008) 18(11): 3183–206. doi:10.1142/s0218127408022354
22. Bharathwaj M, Chua LO. Simplest Chaotic Circuit[J]. *Int J Bifurcation Chaos* (2012) 20(5):1567.
23. Xu B, Wang G, Shen Y. A Simple Meminductor-Based Chaotic System with Complicated Dynamics. *Nonlinear Dyn*(2017) 88(3):2071–89. doi:10.1007/s11071-017-3363-y
24. Yuan F, Li Y. A Chaotic Circuit Constructed by a Memristor, a Memcapacitor and a Meminductor. *Chaos* (2019) 29(10):101101. doi:10.1063/1.5125673
25. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Du S. FPGA Implementation and Image Encryption Application of a New PRNG Based on a Memristive Hopfield Neural Network with a Special Activation Gradient. *Chin Phys. B* (2022) 31(2):020505. doi:10.1088/1674-1056/ac3cb2
26. Li C, Yang Y, Yang X. A Tristable Locally Active Memristor and its Application in Hopfield Neural Network[J]. *Nonlinear Dyn*(2022) 108:1697–717. doi:10.1007/s11071-022-07268-y
27. Zhou Y, Li C, Li W, Li H, Feng W, Qian K. Image Encryption Algorithm with Circle Index Table Scrambling and Partition Diffusion. *Nonlinear Dyn*(2021) 103(2):2043–61. doi:10.1007/s11071-021-06206-8
28. Li C-L, Zhou Y, Li H-M, Feng W, Du J-R. Image Encryption Scheme with Bit-Level Scrambling and Multiplication Diffusion. *Multimed Tools Appl* (2021) 80(12):18479–501. doi:10.1007/s11042-021-10631-7
29. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S. A New Multi-Scroll Chua's Circuit with Composite Hyperbolic Tangent-Cubic Nonlinearity: Complex Dynamics, Hardware Implementation and Image Encryption Application. *Integration* (2021) 81:71–83. doi:10.1016/j.vlsi.2021.05.011
30. Yu F, Zhang Z, Shen H. Design and FPGA Implementation of a Pseudo-random Number Generator Based on a Hopfield Neural Network under Electromagnetic Radiation[J]. *Front Phys* (2021) 9:302. doi:10.3389/fphy.2021.690651
31. Yang F, Mou J, Ma C, Cao Y. Dynamic Analysis of an Improper Fractional-Order Laser Chaotic System and its Image Encryption Application. *Opt Lasers Eng* (2020) 129(2020):106031. doi:10.1016/j.optlaseng.2020.106031
32. Li X, Mou J, Xiong L, Wang Z, Xu J. Fractional-order Double-Ring Erbium-Doped Fiber Laser Chaotic System and its Application on Image Encryption. *Opt Laser Technol* (2021) 140(2021):107074. doi:10.1016/j.optlastec.2021.107074
33. Xian Y, Wang X. Fractal Sorting Matrix and its Application on Chaotic Image Encryption. *Inf Sci* (2021) 547:1154–69. doi:10.1016/j.ins.2020.09.055
34. Yu F, Kong X, Chen H. A 6D Fractional-Order Memristive Hopfield Neural Network and its Application in Image Encryption[J]. *Front Phys* (2022) 10:109. doi:10.3389/fphy.2022.847385
35. Gao X. *An Effective Multiple-Image Encryption Algorithm Based on 3D Cube and Hyperchaotic Map*. Journal of King Saud University-Computer and Information Sciences (2022).
36. Li X. An Optical Image Encryption Algorithm Based on a Fractional-Order Laser Hyperchaotic System. *Int J Bifurcation Chaos* (2022) 3203:2250035. doi:10.1142/s0218127422500353
37. Gao X. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn*(2022) 108:613. doi:10.1007/s11071-021-07192-7
38. Zhou S, Wang X, Zhang Y, Ge B, Wang M, Gao S. A Novel Image Encryption Cryptosystem Based on True Random Numbers and Chaotic Systems. *Multimed Syst* (2022) 28(1):95–112. doi:10.1007/s00530-021-00803-8
39. Zhou S, Wang X, Wang M, Zhang Y. Simple Colour Image Cryptosystem with Very High Level of Security. *Chaos, Solit Fractals* (2020) 141:110225. doi:10.1016/j.chaos.2020.110225
40. Wang H, Wang X, Li C. SPICE Mutator Model for Transforming Memristor into Meminductor[J. *J. Abstr Appl Analysis* (2013) 168–84. doi:10.1155/2013/281675
41. Zhao Q, Wang C, Zhang X. A Universal Emulator for Memristor, Memcapacitor, and Meminductor and its Chaotic Circuit. *Chaos* (2019) 29: 013141. doi:10.1063/1.5081076
42. Yuan F, Deng Y, Li Y, Wang G. The Amplitude, Frequency and Parameter Space Boosting in a Memristor-Meminductor-Based Circuit. *Nonlinear Dyn*(2019) 96:389–405. doi:10.1007/s11071-019-04795-z
43. Han B, Liu W, Hu A, Bao H, Liu W, Hu A. Coexisting Multiple Firing Patterns in Two Adjacent Neurons Coupled by Memristive Electromagnetic Induction. *Nonlinear Dyn*(2019) 95:43–56.
44. Chen A, Lu J, Lü J, Yu S. Generating Hyperchaotic Lü Attractor via State Feedback Control. *Phys A Stat Mech its Appl* (2006) 364:103–10. doi:10.1016/j.physa.2005.09.039
45. Mahmoud GM, Mahmoud EE, Ahmed ME. On the Hyperchaotic Complex Lü System. *Nonlinear Dyn*(2009) 58(4):725–38. doi:10.1007/s11071-009-9513-0
46. Wang Z, Cang S, Ochola EO. A Hyperchaotic System without Equilibrium[J]. *Nonlinear Dyn*(2012) 69(1):531–7. doi:10.1007/s11071-011-0284-z
47. Yang F, Mou J, Liu J, Ma C, Yan H. Characteristic Analysis of the Fractional-Order Hyperchaotic Complex System and its Image Encryption Application. *Signal Process*(2020) 169:107373. doi:10.1016/j.sigpro.2019.107373
48. Zhang X, Wang C. Multiscroll Hyperchaotic System with Hidden Attractors and its Circuit Implementation. *Int. J. Bifurc Chaos* (2019) 29(09):1950117. doi:10.1142/s0218127419501177
49. Yu W, Wang J, Wang J, Zhu H, Li M, Li Y, et al. Design of a New Seven-Dimensional Hyperchaotic Circuit and its Application in Secure Communication. *IEEE Access* (2019) 7:125586–608. doi:10.1109/access.2019.2935751
50. He S, Sun K, Wang H, Mei X, Sun Y. Generalized Synchronization of Fractional-Order Hyperchaotic Systems and its DSP Implementation. *Nonlinear Dyn*(2018) 92(1):85–96. doi:10.1007/s11071-017-3907-1
51. He S, Sun K, Wang H. Complexity Analysis and DSP Implementation of the Fractional-Order Lorenz Hyperchaotic System. *Entropy* (2015) 17(12): 8299–311. doi:10.3390/e17127882
52. Dai S, Sun K, Ai W, Peng Y. Novel Discrete Chaotic System via Fractal Transformation and its DSP Implementation. *Mod. Phys. Lett. B* (2020) 34(Suppl. 01):2050429. doi:10.1142/s0217984920504291
53. Muthuswamy B, Chua LO. Simplest Chaotic Circuit[J]. *Int J Bifurcation Chaos* (2010) 20(05):1002707. doi:10.1142/s0218127410027076
54. Di Ventra M, Pershin YV, Chua LO. Circuit Elements with Memory: Memristors, Memcapacitors, and Meminductors. *Proc. IEEE* (2009) 97(10): 1717–24. doi:10.1109/jproc.2009.2021077
55. Lai Q, Wan Z, Kamdem Kuate PD. Modelling and Circuit Realisation of a New No-equilibrium Chaotic System with Hidden Attractor and

Coexisting Attractors. *Electron. Lett.* (2020) 56(20):1044–6. doi:10.1049/el.2020.1630

56. Lai Q. A Unified Chaotic System with Various Coexisting Attractors. *Int. J. Bifurc Chaos* (2021) 31(01):2150013. doi:10.1142/s0218127421500139
57. Lai Q, Wan Z, Kengne LK. Two-memristor-based Chaotic System with Infinite Coexisting Attractors[J]. *IEEE Trans Circuits Syst II Express Briefs* (2020) 68(6):2197–201.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Frontiers in Physics

ORIGINAL RESEARCH
published: 20 June 2022
doi: 10.3389/fphy.2022.907644

Classifying Beers With Memristor Neural Network Algorithm in a Portable Electronic Nose System

Imran Zafar[1], Yuanhui Cui[1]*, Qinghao Bai[1] and Yanqing Yang[2]

[1]School of Information Science and Engineering, Dalian Polytechnic University, Dalian, China, [2]School of International Education, Dalian Polytechnic University, Dalian, China

Quality control and counterfeit product detection have become exceedingly important due to the vertical market of beers in the global economy. China is the largest producer of beer globally and has a massive problem with counterfeit alcoholic beverages. In this research, a modular electronic nose system with 4 MOS gas sensors was designed for collecting the models from four different brands of Chinese beers. A sample delivery subsystem was fabricated to inject and clean the samples. A software-based data acquisition subsystem was programmed to record the time-dependent chemical responses in 28 different models. A back-propagation neural network based on a memristor was proposed to classify the quality of the beers. Data collected from the electronic nose system were then used to train, validate, and test the created memristor back-propagation neural network model. Over 70 tests with changes in the setup parameters, feature extraction methods, and neural network parameters were performed to analyze the classification performance of the electronic nose hardware and neural network. Samples collected from 28 experiments showed a deviation of 9% from the mean value. The memristor back-propagation network was able to classify four brands of Chinese beers, with 88.3% of classification accuracy. Because the memristor neural network algorithm is easy to fabricate in hardware, it is reasonable to design an instrument with low cost and high accuracy in the near future.

Keywords: electronic nose, memristor back-propagation neural network, classifying beers, feature extraction method, MOS sensors

OPEN ACCESS

Edited by:
Chunlai Li,
Hunan Institute of Science and Technology, China

Reviewed by:
Weiping Wu,
Shanghai Institute of Optics and Fine Mechanics (CAS), China
Jun Yu,
Dalian University of Technology, China

***Correspondence:**
Yuanhui Cui
cuiyh@dlpu.edu.cn

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *30 March 2022*
Accepted: *19 April 2022*
Published: *20 June 2022*

Citation:
Zafar I, Cui Y, Bai Q and Yang Y (2022) Classifying Beers With Memristor Neural Network Algorithm in a Portable Electronic Nose System. Front. Phys. 10:907644. doi: 10.3389/fphy.2022.907644

1 INTRODUCTION

Food quality control is becoming exceedingly popular due to the massive quantities of products in today's world. But the monitoring of food quality has become more critical with the governments' regulations and improvements in sensor technology. In the research field, monitoring system design, classification methods, or algorithms have been proved to be effective ways to enhance the accuracy of food detection.

Using an electronic nose to detect the odor of food or beverage, instead of a human nose, was first proposed in 1982. At present, the electronic nose is still an essential way of detecting food quality for different applications. In the monitoring system design, M. Falasconi [1] gave in detail about using electronic nose microbiology for detecting food product quality control in 2012. Gliszczyńska-Świgło [2] reviewed the electronic nose as a tool for monitoring the authenticity of food in 2017. Wojnowski [3] proposed a portable electronic nose based on electrochemical sensors for food quality assessment

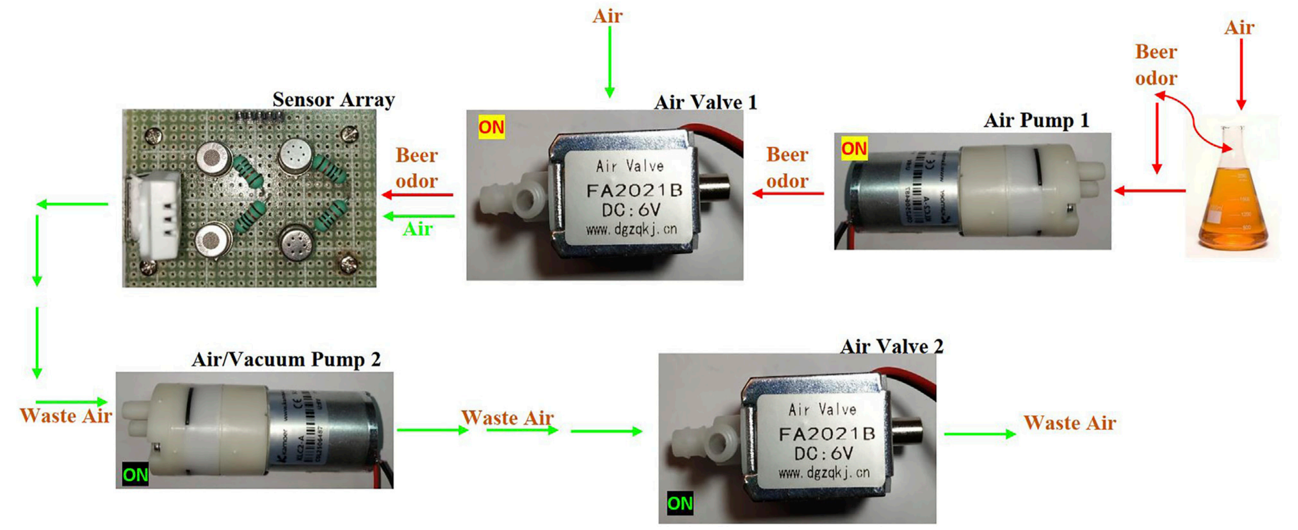


FIGURE 1 | Working modes of sample delivery system.

in 2017. Eusebio [4] published the definition of minimum performance requirements for environmental odor monitoring in electronic nose testing procedures in 2017. For improving the accuracy, Liu [5] researched a drift compensation for electronic noses by semi-supervised domain adaption. Vergara [6] put a chemical gas sensor drift compensation using classifier ensembles. Zhang [7] explained domain adaptation extreme learning machines for drift compensation in e-nose system in 2015. Another Zhang [8] used a chaotic time series prediction of the e-nose sensor drift in the embedded phase space in 2013. In the research objects, tea [9] and wine [10] have been used to monitor their quality. For algorithms, Zhang [11] listed all the methods: SVM, BPA, neural network, and so on, in his book *e-Nose Algorithms and Challenges* in 2018. Qi [12]designed a portable electronic nose to classify the real–fake detection of Chinese liquors in 2017. Pádua [13] proposed a new design of a scalable and easy-to-use EN system to detect volatile organic compounds from exhaled breath in 2018. The aforementioned articles only proposed monitoring the quality of one type of product as it is difficult to design a universal electronic nose to monitor the quality of multiple products due to inherent limitations of sensor technology.

The memristor is a new emerging technology, which has been developed in the past decade. Li, C.L [14], and Ma [15] focused on how to design and fabricate the memristor circuit. Li, C.L. [16], Li, X.J [17], and Gao [18, 19] used the memristor algorithm to encrypt images. In the past few years, hybrid algorithms based on the memristor and different neural networks were proposed in [20–23].

In this study, a new portable modular design of an electronic nose was proposed to classify four different brands of Chinese beers based on the MOS sensor array and the memristor back-propagation neural network algorithm. After simulation in MATLAB, the system and the algorithm were efficient with higher accuracy in classifying the assigned beers.

2 SYSTEM SETUP AND EXPERIMENTAL PROCEDURE

2.1 Sample Delivery Subsystem

The section consisted of air pumps and air valves with a sophisticated control setup, which diluted the aroma of the beer sample with air in an accurate ratio and delivered it to the gas detection module. The sample delivery system consisted of a sensor array, two solenoid air valves, and two air/vacuum pumps, as shown in **Figure 1**. During the experiment, in the injection mode (red), solenoid air valve 1 and air pump 1 were turned on. The beer odor sample was then drawn into the sensor chamber using air pump 1 connected to the solenoid air valve 1. In cleaning mode (green), air pump 2 and solenoid air valve 2 were turned on to clean the sensor chamber for the next sample. Air pump 2, working as a vacuum pump, is connected to the sample chamber with pipes to remove odor from it.

The sample delivery system was kept inside the EN housing, separated from the gas detection module and the sample chamber. It was connected to the gas detection module and sample chamber through pipes and routings.

2.2 Data Acquisition Subsystem

The gas detection module consists of 4 cross-sensitive MOS sensors (tgs2600, tgs2602, tgs2611, and tgs2620), which detect the mixture of gases present in the aroma. The data acquisition system then collected the response of the gas detection module in real time and stored the time-stamped data. Finally, the memristor back-propagation neural network model was trained to perform pattern recognition based on the response from the gas detection module.

After the hardware design was completed, a working prototype of the electronic nose was able to switch the sample delivery system and convert the sensor response into digital signals. Now, these data were stored for further analysis.

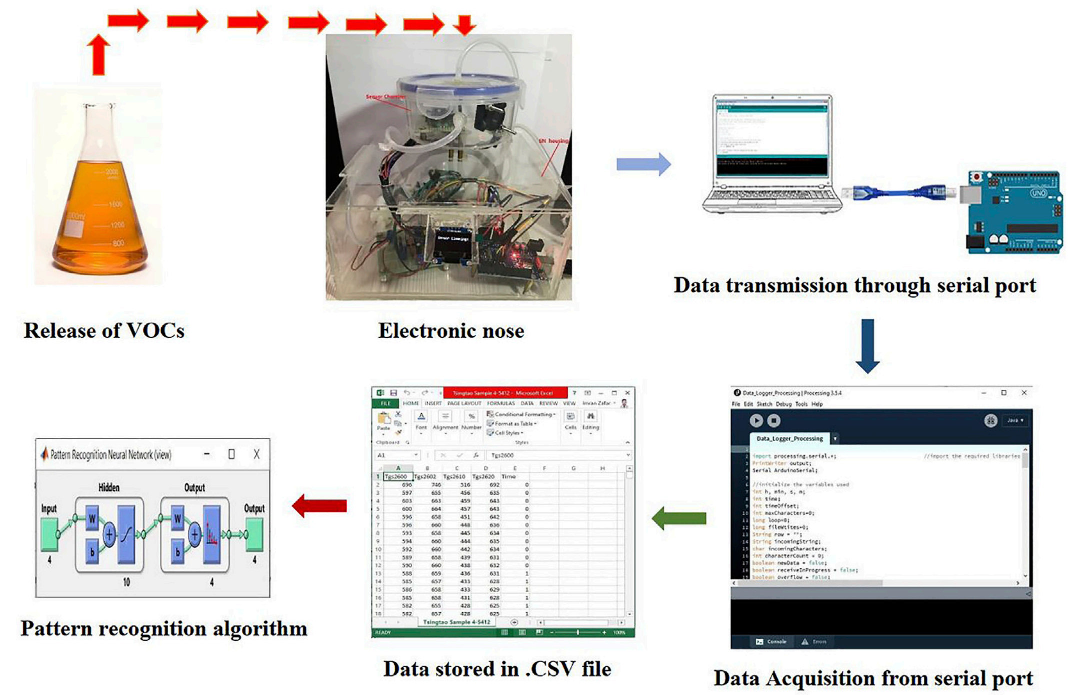


FIGURE 2 | Block diagram of data acquisition subsystem.

TABLE 1 | Set experiment parameters.

Parameter	Values
Sensors	Tgs2620, Tgs2611, Tgs2600, Tags 2602
Number of samples	28 (7 samples × 4 beers)
Cleaning 1 stage time	300 s
Cleaning 2 stage time	600 s
Injection stage time	300 s
Air pump one voltage	100% duty cycle (12 V)
Air pump one flow rate	800 ml/min
Vacuum pump two voltage	100% duty cycle (12 V)
Vacuum pump two flow rate	800 ml/min
Measurement interval	100 milliseconds
Air pump one activation interval	300–600s
Vacuum pump two activation interval	0–300 s; 600–1200 s
Air valve one activation interval	300–600 s
Air valve two activation interval	0–300 s; 600–1200 s
Time duration for one experiment (without manual steps)	1,200 s
Total time (without manual steps)	33600 s–9.34 h

During the testing stage, sensors in the gas detection module were measuring the odor and sending the analog signals to an Arduino microcontroller. A microcontroller ADC converted these analog signals into digital signals and displayed them on its serial port. Processing software was then used to process and store the data in a CSV file. Serial communication was used to deliver data between the microcontroller and PC. The block diagram of data acquisition subsystem is shown in **Figure 2**.

2.3 Set Experiment Parameters

After numerous tests, measurements were taken with the following values of adjustable parameters of the electronic nose (**Table 1**). Four brands of Chinese beer were bought from different supermarkets. Overall, seven sample bottles of each brand of beer were collected this way for the diversity of samples. Each beer sample was then taken to collect the sensor response curve. The total number of pieces was 28. To ensure uniformity, 50 ml of beer was taken from each sample bottle and poured into the sample chamber. All the experiments for data collection of beer samples were performed within one week, so the sensor array was preheated only once before starting the experiments. All sensors were heated for 48 h to ensure stable sensor characteristics.

2.4 Experiment Procedure

All steps in the system were automated. Only sample preparation was manual. The steps to record one sample were as follows: After defining the experiment parameters, wait until a steady state, then prepare the sample in the sample chamber, turn on the electronic nose, and the system will start. Next, in cleaning stage 1, the sample delivery system activated the air pumps and solenoid air valves to clean the sensor chamber for 300 s. Third, injection stage: the sample delivery system started the air pumps and solenoid air valves to deliver the beer sample odor from sample chamber to sensor chamber for 300 s. Fourth, cleaning stage 2: sample delivery system activated the air pumps and solenoid air valves to clean the sensor chamber for 600 s. During this whole procedure, the data acquisition system records the data and saves it as a CSV file. Finally, clean and wash the sample chamber for subsequent measurement. The flowchart is shown in **Figure 3**.

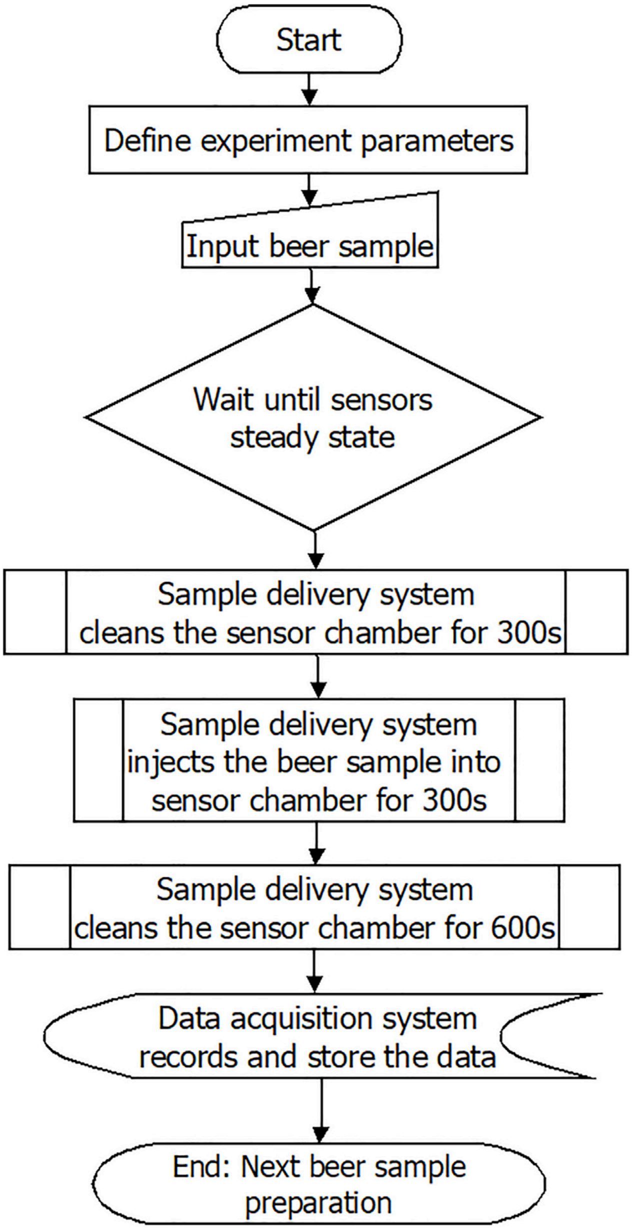


FIGURE 3 | Flowchart of the experiment procedure.

2.3 Memristor Neural Network Design

To classify the different odors of Chinese beers, MATLAB was used to create and ultimately organize the data. Following steps were followed to develop a neural network model, and this neural network was then used to test the input data and classify it according to the recognized patterns in it.

First, four sensors (tgs2600, tgs2602, tgs2611, and tgs2620) were used to record the odor in the sample, so the number of input was defined as 4. The number of outputs was also 4 as four brands of Chinese beers were being used for the classification.

Then, values were initialized. Starting weight values were set for synapses connecting each node. Here random discounts were applied. Next, the middle hidden layer was calculated, four weights from each of the nodes in the input layer were indexed, and four weights from each remote layer node to the output layer were set. Summing up the input values *I[n]* multiplied by their consequences from the synapses and adding a node threshold value (T) resulted in the consequences from the input layer to the middle hidden layer, which was denoted by *I_to_M_weight[i]* [24].

$$M[i] = \left(\sum_{1}^{n} I[n] * I\_to\_M\_weight[i]\right) + M[i] + T \tag{1}$$

M[i] was limited between 0 and 1 when adjusted with a sigmoid logistic function.

$$M[i] = \frac{1}{1 + e^{-M[i]}} \tag{2}$$

After calculating the hidden layer nodes, constrained values were used to calculate the output layer. Here, the inputs were the middle layer node values, and the outputs were the nodes in the output layer.

$$O[i] = \left(\sum_{1}^{n} M[n] * M\_to\_O\_weight[i]\right) + O[i] + T \tag{3}$$

O[i] was also limited between 0 and 1 when it was adjusted with a sigmoid logistic function.

$$O[i] = \frac{1}{1 + e^{-O[i]}} \tag{4}$$

When training began, due to the inherent randomness created during the initialization, the output error $err[i]$ equaled subtracting the correct value $actual[i]$ from the output value $O[i]$ and taking the absolute value.

$$err[i] = abs(actual[i] - O[i]) \tag{5}$$

The hidden layer weight values were updated by adding the error between the output and the middle to output weight.

$$M\_weight[i] = \sum_{1}^{n} M\_weight[i] + (err[n] * M\_to\_O\_weight[i, n]) \tag{6}$$

Here, $M\_weight[i]$ was used to calculate the error at the middle hidden layer. $M\_err[i]$ was used to reevaluate the input to middleweights.

$$M\_err[i] = M[i] * (i - M[i]) * M\_weight[i] \tag{7}$$

Then, the weights between the middle and output layers could be updated. Here, a learning rate between 0 and 1 was used to balance the degree for each epoch. Due to oscillating the output accuracy, a momentum rate was added to change from previous epochs. The learning rate was set as the following equation:

$$M\_to\_O\_weight[a,b] = M\_O\_weight[a,b] + \left(learning\ rate * M[a] * err[b]\right) \tag{8}$$

The momentum was in the following equation:

$$M\_to\_O\_weight[a,b] = M\_O\_weight[a,b] + \left(momentum\ rate * last\ M\_to\_O\_weight[a,b]\right) \tag{9}$$

Now, the weights between the input layer and middle layer were updated based on the error in the middle layer:

$$I\_to\_M\_weight[a,b] = I\_to\_M\_weight[a,b] + \left(learning\ rate * I[a] * M\_err[b]\right) \tag{10}$$

where a was from 1 to the number of input nodes and b was from 1 to the number of hidden nodes.

Finally, the threshold constants were updated from the error in each layer and the learning rate.

$$M\_threshold[i] = M\_threshold[i] + \left(learning\ rate * M\_err[i]\right) \tag{11}$$

$$O\_threshold[i] = M\_threshold[i] + \left(learning\ rate * err[i]\right) \tag{12}$$

3 SIMULATION AND RESULTS ANALYSIS

3.1 Set Learning Rate

Collected data were fed into our created neural network model. A new memristor scaled conjugate gradient back-propagation was designed to classify collected samples. The memristor back-propagating neural network used a momentum rate of 0.9 and a floating learning rate that began training at 0.4. As the average training error of the network decreased, the learning rate decreased. The network started with a higher learning and momentum rate to avoid the system becoming stuck in a local minimum or local maximum. As the average training error lowered, the learning rate decreased to try to fine-tune the parameters of the system. The change in the learning rate to the average training error is listed in **Table 2**.

TABLE 2 | Learning rate adjustment.

Average learning rate	Learning rate
ATE > 0.30	0.4
0.30 ≥ ATE > 0.25	0.2
0.25 ≥ ATE > 0.15	0.1
0.15 ≥ ATE > 0.10	0.05
0.10 ≥ ATE > 0.05	0.01

3.2 Set Numbers of Hidden Neurons

Multiple sets of hidden neurons were tested, and their effects on classification were analyzed. In this analysis, the number of HNs was changed in the neural network model, and the network was trained and tested each time with a different number of HNs. At the end of each training and testing, confusion matrixes were stored. Overall, 10, 1,000, 2000, 3,000, 4,000, and 5,000 HNs were changed in the NN model, and the results are shown in **Table 3**. Totally, 10 HNs reported the lowest classification accuracy in the confusion matrix, and 5,000 HNs reported the highest overall accuracy. The effect of the number of HNs was also directly related to the individual classification accuracy of each beer. Each beer classification rate was calculated as average. The highest overall accuracy of 88.3% was achieved with 5,000 hidden neurons. During this analysis, all other parameters in this neural network model remained the same.

3.3 Type of Feature Extraction Method

Two types of feature extraction were done on data and then fed into our neural network model. The first type of feature extraction was based on the original response curve. In this type of feature extraction, features were extracted from the whole curve. The whole curve included the baseline response and sensor response to the beer sample. Rather than taking full value features of the response curve, data from the entire curve were stored and given as input to the NN model. In the second type, feature extraction was based on the rising curve. In this type of feature extraction, the only rising curve was taken as the feature curve where features were extracted from only the rising curve data.

Changes in the feature extraction method drastically affected the overall classification accuracy of beer samples. When the neural network was trained based on the features from the original response curve, it reported a maximum accuracy of 87.8% with 4,000 hidden neurons. But the NN model reported only 79.6% classification accuracy with 4,000 hidden neurons when the NN model was trained using features extracted from the rising curve. Accuracy comparison of rising and whole curves with 1,000 and 4,000 hidden neurons is shown in **Figure 4**. Average accuracies for individual beers were also reported to be decreased.

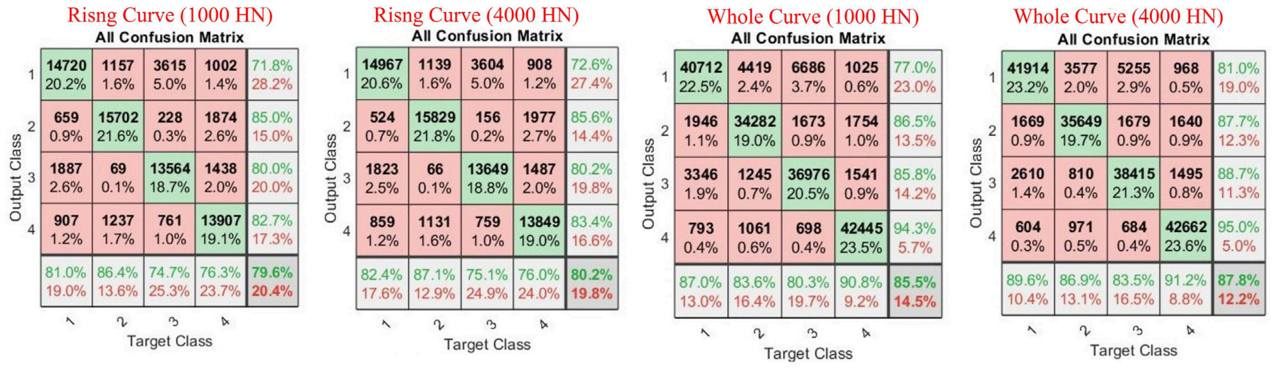
3.4 Validation and Test Data

To perform classification with a neural network, it has to be trained, validated, and then tested to give accurate predictions. To check the effect of the amount of validation data and test data on the classification accuracy of our trained model, these parameters were changed and analyzed. Training data of created neural network were kept at 70% of total data during two variations, and it was reduced to 50% of total data in the third variation. Remaining 30% data were then divided into validation and test data of multiple variations (15% validation data, 15% test data), (5% validation data, 25% test data), and (25% validation data, 25% test data).

During this analysis, all the other parameters in the system were kept constant. Hidden neurons during this analysis were maintained at 5,000. Features extracted from the whole response curve were used as feature extraction types. Study (**Figure 5**) showed that the maximum classification accuracy of 88.3% was achieved with (15% validation data and 15% test data), while (25% validation data, 25% test data) attained an accuracy of 87.8%. Parameters of

TABLE 3 | Comparison of classification accuracy with change in number of hidden neurons.

Hidden neuron	Harbin beer (%)	Laoshan beer (%)	Snow beer (%)	Tsingtao beer (%)	Total classification (%)
10	72.25	74.65	70.9	89.45	76.6
100	79.5	82.8	81.5	91.7	83.5
1,000	82	85.05	83.05	92.55	85.5
2000	82.7	85.55	83.55	92.65	86
4,000	85.3	87.3	86.1	93.1	87.8
5,000	85.85	88	86.1	93.1	88.3



Risng Curve (1000 HN)

All Confusion Matrix

Output Class	1	2	3	4	
1	14720 20.2%	1157 1.6%	3615 5.0%	1002 1.4%	71.8% 28.2%
2	659 0.9%	15702 21.6%	228 0.3%	1874 2.6%	85.0% 15.0%
3	1887 2.6%	69 0.1%	13564 18.7%	1438 2.0%	80.0% 20.0%
4	907 1.2%	1237 1.7%	761 1.0%	13907 19.1%	82.7% 17.3%
	81.0% 19.0%	86.4% 13.6%	74.7% 25.3%	76.3% 23.7%	**79.6%** **20.4%**

Target Class

Risng Curve (4000 HN)

All Confusion Matrix

Output Class	1	2	3	4	
1	14967 20.6%	1139 1.6%	3604 5.0%	908 1.2%	72.6% 27.4%
2	524 0.7%	15829 21.8%	156 0.2%	1977 2.7%	85.6% 14.4%
3	1823 2.5%	66 0.1%	13649 18.8%	1487 2.0%	80.2% 19.8%
4	859 1.2%	1131 1.6%	759 1.0%	13849 19.0%	83.4% 16.6%
	82.4% 17.6%	87.1% 12.9%	75.1% 24.9%	76.0% 24.0%	**80.2%** **19.8%**

Target Class

Whole Curve (1000 HN)

All Confusion Matrix

Output Class	1	2	3	4	
1	40712 22.5%	4419 2.4%	6686 3.7%	1025 0.6%	77.0% 23.0%
2	1946 1.1%	34282 19.0%	1673 0.9%	1754 1.0%	86.5% 13.5%
3	3346 1.9%	1245 0.7%	36976 20.5%	1541 0.9%	85.8% 14.2%
4	793 0.4%	1061 0.6%	698 0.4%	42445 23.5%	94.3% 5.7%
	87.0% 13.0%	83.6% 16.4%	80.3% 19.7%	90.8% 9.2%	**85.5%** **14.5%**

Target Class

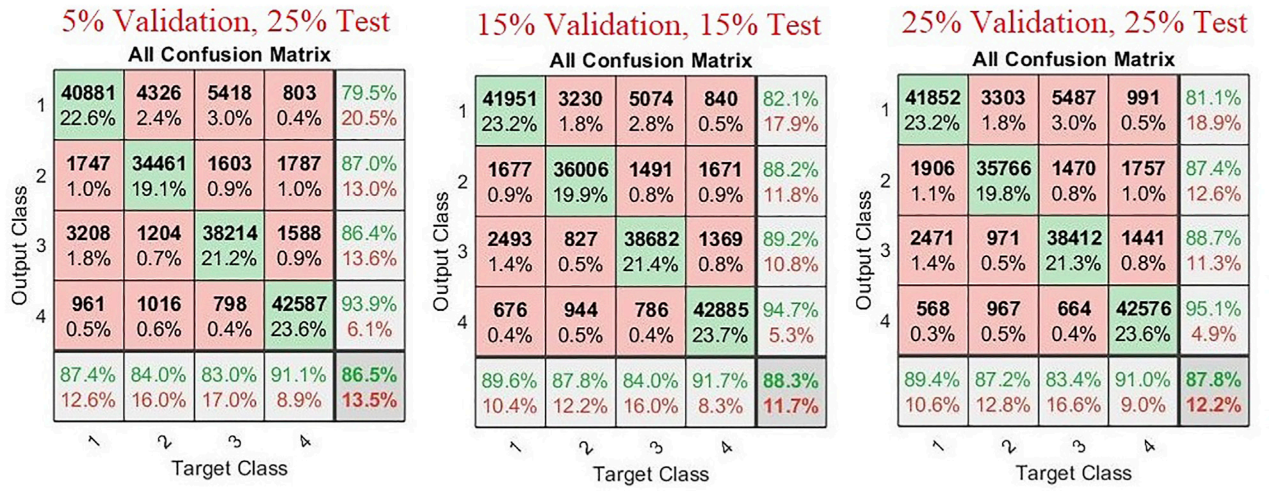
Whole Curve (4000 HN)

All Confusion Matrix

Output Class	1	2	3	4	
1	41914 23.2%	3577 2.0%	5255 2.9%	968 0.5%	81.0% 19.0%
2	1669 0.9%	35649 19.7%	1679 0.9%	1640 0.9%	87.7% 12.3%
3	2610 1.4%	810 0.4%	38415 21.3%	1495 0.8%	88.7% 11.3%
4	604 0.3%	971 0.5%	684 0.4%	42662 23.6%	95.0% 5.0%
	89.6% 10.4%	86.9% 13.1%	83.5% 16.5%	91.2% 8.8%	**87.8%** **12.2%**

Target Class

FIGURE 4 | Confusion matrices with rising and whole curve along with 1,000 and 4,000 hidden neurons.



5% Validation, 25% Test

All Confusion Matrix

Output Class	1	2	3	4	
1	40881 22.6%	4326 2.4%	5418 3.0%	803 0.4%	79.5% 20.5%
2	1747 1.0%	34461 19.1%	1603 0.9%	1787 1.0%	87.0% 13.0%
3	3208 1.8%	1204 0.7%	38214 21.2%	1588 0.9%	86.4% 13.6%
4	961 0.5%	1016 0.6%	798 0.4%	42587 23.6%	93.9% 6.1%
	87.4% 12.6%	84.0% 16.0%	83.0% 17.0%	91.1% 8.9%	**86.5%** **13.5%**

Target Class

15% Validation, 15% Test

All Confusion Matrix

Output Class	1	2	3	4	
1	41951 23.2%	3230 1.8%	5074 2.8%	840 0.5%	82.1% 17.9%
2	1677 0.9%	36006 19.9%	1491 0.8%	1671 0.9%	88.2% 11.8%
3	2493 1.4%	827 0.5%	38682 21.4%	1369 0.8%	89.2% 10.8%
4	676 0.4%	944 0.5%	786 0.4%	42885 23.7%	94.7% 5.3%
	89.6% 10.4%	87.8% 12.2%	84.0% 16.0%	91.7% 8.3%	**88.3%** **11.7%**

Target Class

25% Validation, 25% Test

All Confusion Matrix

Output Class	1	2	3	4	
1	41852 23.2%	3303 1.8%	5487 3.0%	991 0.5%	81.1% 18.9%
2	1906 1.1%	35766 19.8%	1470 0.8%	1757 1.0%	87.4% 12.6%
3	2471 1.4%	971 0.5%	38412 21.3%	1441 0.8%	88.7% 11.3%
4	568 0.3%	967 0.5%	664 0.4%	42576 23.6%	95.1% 4.9%
	89.4% 10.6%	87.2% 12.8%	83.4% 16.6%	91.0% 9.0%	**87.8%** **12.2%**

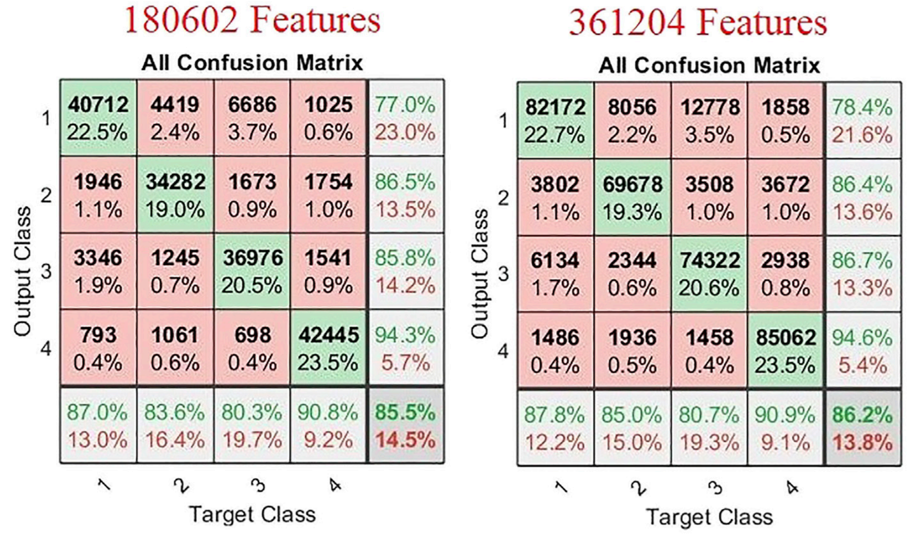
Target Class

FIGURE 5 | Confusion matrices with different percentages of validation data and test data.

(5% validation data, 25% test data) performed the worst and showed an accuracy of only 86.5%.

3.5 Doubling the Amount of Data

The number of samples recorded with our working prototype was 28, which were divided into seven samples of 4 different brands of Chinese beers. The number of features extracted from these 28 samples was 180602. These features were then used to train, validate, and test our neural network. Components were randomly selected, and the amount of data for training, validation, and testing was also randomly selected. Idea was presented to investigate if doubling the amount of data would increase its accuracy or not. Due to time constraints, data collection was not possible. We adopted the method of duplicating features. In this technique, we extracted features from our original data, which were 180602, and repeated them. The amount of features after duplication was 361204. These repeated



180602 Features

All Confusion Matrix

Output Class \ Target Class	1	2	3	4	
1	40712 22.5%	4419 2.4%	6686 3.7%	1025 0.6%	77.0% 23.0%
2	1946 1.1%	34282 19.0%	1673 0.9%	1754 1.0%	86.5% 13.5%
3	3346 1.9%	1245 0.7%	36976 20.5%	1541 0.9%	85.8% 14.2%
4	793 0.4%	1061 0.6%	698 0.4%	42445 23.5%	94.3% 5.7%
	87.0% 13.0%	83.6% 16.4%	80.3% 19.7%	90.8% 9.2%	**85.5%** **14.5%**

361204 Features

All Confusion Matrix

Output Class \ Target Class	1	2	3	4	
1	82172 22.7%	8056 2.2%	12778 3.5%	1858 0.5%	78.4% 21.6%
2	3802 1.1%	69678 19.3%	3508 1.0%	3672 1.0%	86.4% 13.6%
3	6134 1.7%	2344 0.6%	74322 20.6%	2938 0.8%	86.7% 13.3%
4	1486 0.4%	1936 0.5%	1458 0.4%	85062 23.5%	94.6% 5.4%
	87.8% 12.2%	85.0% 15.0%	80.7% 19.3%	90.9% 9.1%	**86.2%** **13.8%**

FIGURE 6 | Confusion matrices of the neural network model with different amounts of features.

features were then used for the training, validation, and testing of neural networks. In this analysis, the number of hidden neurons was taken as 1,000, and the whole response curve was used for feature extraction. In this study, 70% of all data were used as training data, and the remaining data were used for validation and training. Out of 30% remaining data, 15% were used for validation and 15% for testing the trained NN model. By doubling the number of features, a 0.7% overall increase in classification accuracy of the trained neural network was reported (**Figure 6**). But it also doubled the processing time. The time to train the neural network also significantly increased as the amount of data increased.

3.6 Classification of Chinese Beers

To perform the classification of Chinese beers, multiple analyses were done on the different parameters of the neural network. The trained neural network reported a maximum accuracy of 88.3% in **Section 3.4**. In this neural network model, 5,000 hidden layers were taken. Feature extraction was based on the original response curve, which means the features included the steady-state and the transient response of each sample. Total samples were 28, and the number of components extracted was 180602. Again 70% of the complete data were taken as training data. Training of the NN lasted 4 h, which was done on a laboratory computer. Of the remaining data, 15% were used for validation, and another 15% were used for testing the trained NN model. All the features used in training, validation, and testing were randomly selected. The confusion matrix of this NN model is shown in **Figure 7**. The receiver operating characteristic curve (ROC) of this trained neural network is established in **Figure 8**.

The ROC curve of the trained neural network also demonstrated the accuracy of the trained NN model during the classification; it stayed true to the proper positive rate curve. During training, validation, and testing, it almost followed the same pattern. The maximum number of features was recognized as a true positive for the classification accuracy of our NN.

The total error of the output nodes was summed for training, validation, and testing, as shown in **Figure 9**. The histogram compares the frequency of the output classification error over 1,000 epochs and reports the number of instances. Error histogram was the histogram of the errors between target values and predicted values after training the neural network. As these error values indicated how predicted values differed from the target values, these were negative.

The results showed that the working prototype of the electronic nose performed exceptionally well in the sample data collection. The trained neural network was successfully able to classify the samples to within 11.7% (error on average) of the actual beer samples. It successfully classified four beers with 88.3% accuracy. The neural network could compensate for a giant floating baseline and the limited amount of data with only 28 samples. The low number of models also restricted the true potential of the network to become fully developed. For this particular four-input, four-output system, the best performance was achieved with 5,000 hidden nodes based on features extracted from the whole curve with training–validation–test data of 70%-15%-15%.

4 CONCLUSION

A new portable modular design of an electronic nose was created to classify four different brands of Chinese beers based on their odor or aroma. The working prototype of this electronic nose featured an array of 4 MOS sensors that were commercially available from Figaro Engineering Inc. A fully automatic sample delivery system was designed with two air pumps and two solenoid air valves sourced from the local companies. A fully automated software-based data acquisition system was also developed, which required the electronic nose to connect to processing software through a PC. Data were then recorded in a CSV file.

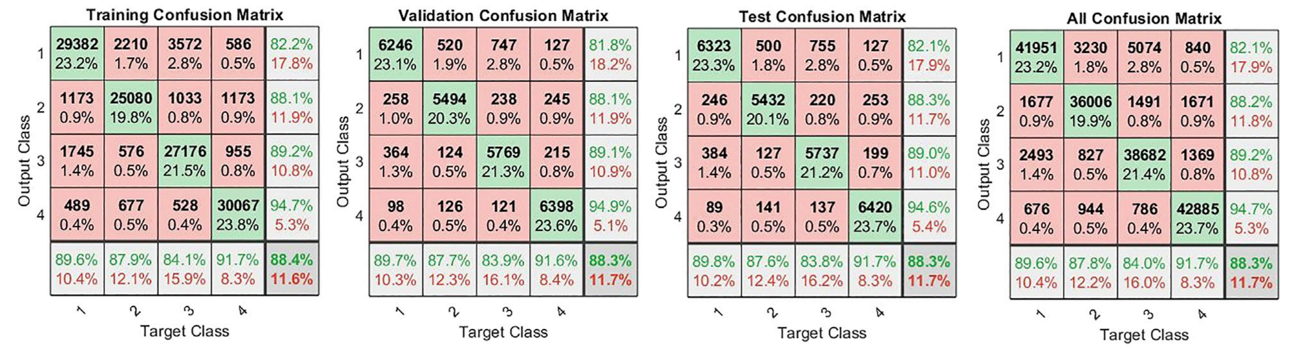


FIGURE 7 | Confusion matrix with maximum classification accuracy.

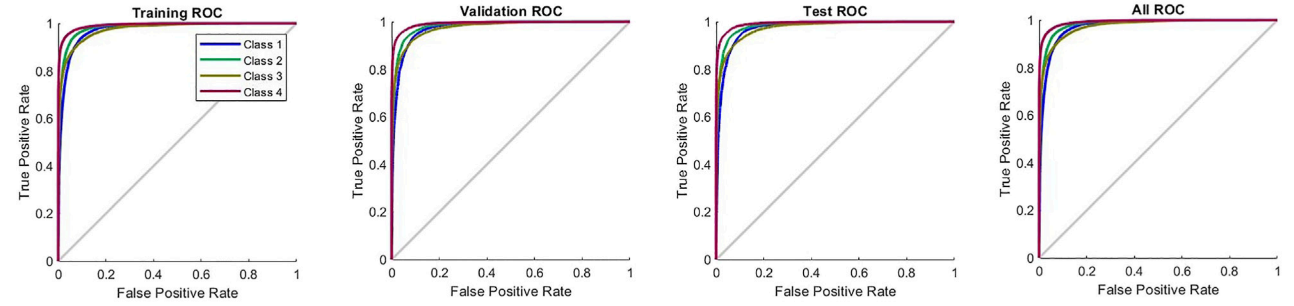


FIGURE 8 | Receiver operating characteristic curve with maximum classification accuracy.

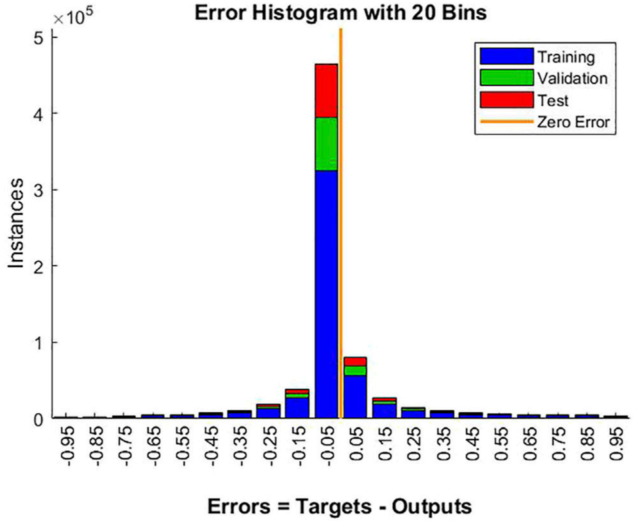


FIGURE 9 | Error histogram with maximum classification accuracy.

Totally, 28 experiments were performed to collect the data for feature extraction. Over 70 tests were performed in MATLAB with different parameters in the neural network. Classification accuracy of the trained neural network drastically improved with the increase in hidden neurons with maximum precision of 88.3% reported with 5,000 hidden neurons. The type of feature extraction also affected the overall accuracy of classification. The amount of training, validation, and testing data also negatively affected the classification performance of the neural network. The best combination for maximum performance was determined as 70% training, 15% validation, and 15% testing data. Artificially doubling the data points also improved the classification performance, but it was not significant. The neural network exhibited only a 0.7% increase in its overall classification accuracy with double data points, and it took longer to train the network model. The neural network displayed a maximum classification performance of 88.3% with 5,000 hidden neurons and features extracted from the original response curve.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

IZ designed the system, did the experiments, and wrote the manuscript. YC guided the algorithm in this paper. QB did partly the experiment. YY improved grammar and vocabulary. All authors reviewed the manuscript.

REFERENCES

1. Falasconi M, Concina I, Gobbi E, Sberveglieri V, Pulvirenti A, Sberveglieri G. Electronic Nose for Microbiological Quality Control of Food Products. *Int J Electrochem* (2012) 2012:1–12. doi:10.1155/2012/715763
2. Gliszczyńska-Świgło A, Chmielewski J. Electronic Nose as a Tool for Monitoring the Authenticity of Food. A Review. *Food Anal. Methods* (2017) 10(6):1800–16. doi:10.1007/s12161-016-0739-4
3. Wojnowski W, Majchrzak T, Dymerski T, Gębicki J, Namieśnik J. Portable Electronic Nose Based on Electrochemical Sensors for Food Quality Assessment. *Sensors* (2017) 17(12):2715. doi:10.3390/s17122715
4. Eusebio L, Capelli L, Sironi S. Electronic Nose Testing Procedure for the Definition of Minimum Performance Requirements for Environmental Odor Monitoring. *Sensors* (2016) 161548(9):1548. doi:10.3390/s16091548
5. Liu Q, Li X, Ye M, Ge SS, Du X. Drift Compensation for Electronic Nose by Semi-supervised Domain Adaption. *IEEE Sensors J.* (2014) 14(3):657–65. doi:10.1109/JSEN.2013.2285919
6. Vergara A, Vembu S, Ayhan T, Ryan MA, Homer ML, Huerta R. Chemical Gas Sensor Drift Compensation Using Classifier Ensembles. *Sensors Actuators B Chem* (2012) 166-167:320–9. doi:10.1016/j.snb.2012.01.074
7. Lei Zhang L, Zhang D. Domain Adaptation Extreme Learning Machines for Drift Compensation in E-Nose Systems. *IEEE Trans. Instrum. Meas.* (2015) 64(7):1790–801. doi:10.1109/TIM.2014.2367775
8. Zhang L, Tian F, Liu S, Dang L, Peng X, Yin X. Chaotic Time Series Prediction of E-Nose Sensor Drift in Embedded Phase Space. *Sensors Actuators B Chem* (2013) 182:71–9. doi:10.1016/j.snb.2013.03.003
9. Dutta R, Kashwan KR, Bhuyan M, Hines EL, Gardner JW. Electronic Nose Based Tea Quality Standardization. *Neural Netw*(2003) 16(5):847–53. doi:10.1016/S0893-6080(03)00092-3
10. Lozano J, Santos JP, Horrillo MC. Wine Applications with Electronic Noses. In: ML Rodríguez Méndez, editor. *Chapter 14 - Wine Applications with Electronic Noses. Electronic Noses and Tongues in Food Science*. San Diego: Academic Press (2016). p. 137–48. doi:10.1016/B978-0-12-800243-8.00014-7
11. Zhang L, Tian F, Zhang D. E-nose Algorithms and Challenges. In: *Electronic Nose: Algorithmic Challenges*. Singapore: Springer (2018). p. 11–20. doi:10.1007/978-981-13-2167-2_2
12. Qi P-F, Zeng M, Li Z-H, Sun B, Meng Q-H. Design of a Portable Electronic Nose for Real-Fake Detection of Liquors. *Rev. Sci. Instrum.* (2017) 88(9):095001. doi:10.1063/1.5001314
13. Pádua A, Osório D, Rodrigues J, Santos G, Porteira A, Palma S, et al. Scalable and Easy-To-Use System Architecture for Electronic Noses. In: Proceedings of the 11th International Joint Conference on Biomedical Engineering Systems and Technologies. 1. BIODEVICES (2018). p. 179–86. doi:10.5220/0006597101790186
14. Li C, Li H, Xie W, Du J. A S-type Bistable Locally Active Memristor Model and its Analog Implementation in an Oscillator Circuit. *Nonlinear Dyn*(2021) 106(1):1041–58. doi:10.1007/s11071-021-06814-4
15. Ma XJ, Mou J, Liu J, Ma CG, Yang FF, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor-Memcapacitor. *Nonlinear Dyn*100(3):2859. doi:10.1007/s11071-020-05601-x
16. Li C-L, Zhou Y, Li H-M, Feng W, Du J-R. Image Encryption Scheme with Bit-Level Scrambling and Multiplication Diffusion. *Multimed Tools Appl* (2021) 80:18479–501. doi:10.1007/s11042-021-10631-7
17. Li XJ, Jun Mou J, Banerjee S, Cao YH. An Optical Image Encryption Algorithm Based on Fractional-Order Laser Hyperchaotic System. *Int J Bifurcation Chaos* (2022) 32(03):2250035. doi:10.1142/s0218127422500353
18. Gao X, Mou J, Xiong L, Sha Y, Yan H, Cao Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn*(2022) 108:613–36. doi:10.1007/s11071-021-07192-7
19. Gao X, Mou J, Banerjee S, Cao Y, Xiong L, Chen X. An Effective Multiple-Image Encryption Algorithm Based on 3D Cube and Hyperchaotic Map. *J King Saud Univ - Comput Inf Sci* (2022) 34(Issue 4):1535–51. doi:10.1016/j.jksuci.2022.01.017
20. Li C, Yang Y, Yang X, Zi X, Xiao F. A Tristable Locally Active Memristor and its Application in Hopfield Neural Network. *Nonlinear Dyn*(2022) 108:1697–717. doi:10.1007/s11071-022-07268-y
21. Lin H, Wang C, Hong Q, Sun Y. A Multi-Stable Memristor and its Application in a Neural Network. *IEEE Trans. Circuits Syst. II* (2020) 67(12):3472–6. doi:10.1109/TCSII.2020.3000492
22. Kim M-H, Park H-L, Kim M-H, Jang J, Bae J-H, Kang IM, et al. Fluoropolymer-based Organic Memristor with Multifunctionality for Flexible Neural Network System. *npj Flex Electron* (2021) 5:34. doi:10.1038/s41528-021-00132-w
23. Luo YX, She K, Yu K, Yang L. Research on Vehicle Logo Recognition Technology Based on Memristive Neural Network. *J Inf Secur Res* (2021) 7(No 8):715
24. Little RL. *Neural Network Compound Predictor for Spirits in an Electronic Nose*. Ontario, Canada: The University of Guelph (2014). p. 22

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Physics

ORIGINAL RESEARCH
published: 24 June 2022
doi: 10.3389/fphy.2022.927991

Design of Grid Multi-Wing Chaotic Attractors Based on Fractional-Order Differential Systems

Yuan Lin[1]*, Xifeng Zhou[1], Junhui Gong[1], Fei Yu[2]* and Yuanyuan Huang[2]

[1]College of Electrical and Information Engineering, Hunan Institute of Engineering, Xiangtan, China, [2]School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, China

In this article, a new method for generating grid multi-wing chaotic attractors from fractional-order linear differential systems is proposed. In order to generate grid multi-wing attractors, we extend the method of constructing heteroclinic loops from classical differential equations to fractional-order differential equations. Firstly, two basic fractional-order linear systems are obtained by linearization at two symmetric equilibrium points of the fractional-order Rucklidge system. Then a heteroclinic loop is constructed and all equilibrium points of the two basic fractional-order linear systems are connected by saturation function switching control. Secondly, the theoretical methods of switching control and construction of heteromorphic rings of fractal-order two-wing and multi-wing chaotic attractors are studied. Finally, the feasibility of the proposed method is verified by numerical simulation.

Keywords: fractional differential system, saturated functions switching control, heteroclinic loops, grid multi-wing, chaotic attractors

Edited by:
Jun Mou,
Dalian Polytechnic University, China

Reviewed by:
Ciyan Zheng,
Guangdong Polytechnic Normal University, China
Yunzhen Zhang,
Xuchang University, China

***Correspondence:**
Yuan Lin
153190675@qq.com
Fei Yu
yufeiyfyf@csust.edu.cn

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: 25 April 2022
Accepted: 16 May 2022
Published: 24 June 2022

Citation:
Lin Y, Zhou X, Gong J, Yu F and Huang Y (2022) Design of Grid Multi-Wing Chaotic Attractors Based on Fractional-Order Differential Systems. Front. Phys. 10:927991. doi: 10.3389/fphy.2022.927991

1 INTRODUCTION

At present, chaotic dynamics is gradually transitioning from the basic theoretical research of mathematics and physics to the practical engineering application field. For example, chaotic theory has been greatly developed in the fields of memristor [1–6], secure communication [7–11], image encryption [12–17], neural network [18–29], so chaotic dynamics has a wide application prospect. A key factor in the application of chaos in engineering is to improve the complex dynamic characteristics of chaos. In recent years, many scholars have deeply analysed and studied the complex dynamic characteristics of chaos, and found many chaotic attractors with complex dynamics. Some research results show that chaotic systems with multi-wing or multi-scroll attractors can show richer and more complex dynamic characteristics [30–35].

Fractional calculus has a history of more than 300 years, but its applications in engineering and physics have only aroused interest in recent decades [36–38]. With the deepening of scientific research, some researchers were surprised to find that these systems have complex chaos and bifurcation phenomena when studying fractional-order nonlinear differential systems [39–45]. In [42], the author designed a method to eliminate chaos in the system trajectory through state feedback controller. In order to form multi-scroll attractors, the potential nonlinearity of fractional-order chaotic systems is changed. In [43], it is proved that the fractional-order system coupled by two fractional Lorentz systems can produce four-wing chaotic attractors. In [44], a series method of saturation function is proposed, which can enable fractional-order differential systems to generate multi-spiral chaotic attractors, including multi-scroll chaotic attractors in three directions. In [45], a suitable nonlinear state feedback controller is designed by employing the four construction criteria of the basic fractional-order differential nominal linear system to generate multi-wing chaotic attractors for the controlled fractional-order differential system. However, these multi-wing chaotic systems all contain product terms, which makes their circuit implementation complicated. In [46], Petras proposed a fractional-order Chua's model based on memristor. Through digital simulation, it is found that the fractional-order Chua's circuit can also produce two-scroll chaotic attractors. However, it is still a very challenging problem to find how to generate grid multi-wing attractors in fractal-order chaotic systems.

In this article, a new design method of generating grid multi-wing chaotic attractors from fractional-order differential system is proposed by switching control of saturation function and constructing heteroclinic loops. Because the fractional-order derivative is a nonlocal operator with weak singular kernel, the multi-wing attractors generated in fractional-order differential system are very different from the multi-wing attractors generated in the classical differential system. In addition, it can be seen from [47–52] that shil'nikov theorem can be used to construct two-wing and multi-wing chaotic attractors in classical differential systems. In this paper, the classical differential system construction method is extended to the fractional-order differential system construction method based on shil'nikov theorem [47]. Firstly, the heteroclinic loops are constructed from the fractional-order piecewise linear differential system, and then a method to generate various grid multi-wing attractors through switching control is proposed. Two basic fractional-order linear systems are constructed by linearization at two symmetrical equilibrium points of fractional-order Rucklidge system [53]. After switching the control, in order to connect all the equilibrium points of the two basic fractional-order linear systems [54], we design a heteroclinic loop. Under appropriate conditions, according to shil'nikov theorem, a variety of grid multi-wing attractors can be obtained. We use a predictor-corrector numerical simulation algorithm to confirm the effectiveness of the proposed method [55].

The other parts of this paper are organized as follows. In **Section 2**, we first introduce some preliminary knowledge about fractional-order differential systems. Two fundamental fractional differential linear systems are deduced from Rucklidge system in **Section 3**. In **Section 4**, we study the theoretical method of designing fractional-order two-wing and multi-wing chaotic attractors by switching control and constructing heteroclinic loops. Finally, the conclusions of this paper are given in **Section 5**.

2 FRACTIONAL-ORDER DIFFERENTIAL SYSTEM

Unlike ordinary differential equations, due to the lack of appropriate mathematical methods, the research on the theoretical analysis and numerical solution of fractional-order calculation is still a difficult topic. In recent years, Caputo type fractional-order differential equations have aroused great interest. Under the promotion of Adams [56], we choose Caputo version of Adams prediction correction algorithm. Next, we will give a brief introduction to the fractional-order algorithm.

Fractional-order differential equation is generally expressed by the following formula:

$$\begin{cases} D_t^{\alpha_1} x(t) = \dfrac{d^{\alpha_1} x(t)}{dt^{\alpha_1}} = f_1(x, y, z) \\ D_t^{\alpha_2} y(t) = \dfrac{d^{\alpha_2} y(t)}{dt^{\alpha_2}} = f_2(x, y, z)\, 0 \leq t \geq T \\ D_t^{\alpha_3} z(t) = \dfrac{d^{\alpha_3} y(t)}{dt^{\alpha_3}} = f_3(x, y, z) \end{cases} \tag{1}$$

When the initial values are chosen as $x(0) = x_0, y(0) = y_0, z(0) = z_0, \quad \alpha_i \in (0, 1), \quad i = 1, 2, 3,$ **Eq. 1** forms the following Volterra integral equation:

$$\begin{cases} x(t) = x(0) + \dfrac{1}{\Gamma(\alpha_1)} \displaystyle\int_0^t (t - \tau)^{\alpha_1 - 1} f_1(x(\tau), y(\tau), z(\tau)) d\tau \\ y(t) = y(0) + \dfrac{1}{\Gamma(\alpha_2)} \displaystyle\int_0^t (t - \tau)^{\alpha_2 - 1} f_2(x(\tau), y(\tau), z(\tau)) d\tau \\ z(t) = z(0) + \dfrac{1}{\Gamma(\alpha_3)} \displaystyle\int_0^t (t - \tau)^{\alpha_3 - 1} f_3(x(\tau), y(\tau), z(\tau)) d\tau \end{cases} \tag{2}$$

where $\Gamma(\alpha_i)$ is the Gamma function, which can be defined as $\Gamma(\alpha_i) = \int_0^\infty e^{-t} t^{\alpha_i - 1} dt$. Set $h = \frac{T}{N}$, $t_n = nh(n = 0, 1, 2, \ldots N)$, then **Eq. 2** can take discretization as:

$$\begin{cases} x_h(t_{n+1}) = x(0) + \dfrac{h^{\alpha_1}}{\Gamma(\alpha_1 + 2)} f_1(x_h^p(t_{n+1}), y_h^p(t_{n+1}), z_h^p(t_{n+1})) + \dfrac{h^{\alpha_1}}{\Gamma(\alpha_1 + 2)} \sum a_{1,j,n+1} f_1(x(t_j), y(t_j), z(t_j)) \\ y_h(t_{n+1}) = y(0) + \dfrac{h^{\alpha_2}}{\Gamma(\alpha_2 + 2)} f_2(x_h^p(t_{n+1}), y_h^p(t_{n+1}), z_h^p(t_{n+1})) + \dfrac{h^{\alpha_2}}{\Gamma(\alpha_2 + 2)} \sum a_{2,j,n+1} f_2(x(t_j), y(t_j), z(t_j)) \\ z_h(t_{n+1}) = z(0) + \dfrac{h^{\alpha_3}}{\Gamma(\alpha_3 + 2)} f_3(x_h^p(t_{n+1}), y_h^p(t_{n+1}), z_h^p(t_{n+1})) + \dfrac{h^{\alpha_3}}{\Gamma(\alpha_3 + 2)} \sum a_{3,j,n+1} f_3(x(t_j), y(t_j), z(t_j)) \end{cases} \tag{3}$$

where $\alpha_{i,j,n+1}$ is given by:

$$\alpha_{i,j,n+1} = \begin{cases} n^{\alpha_i + 1} - (n - \alpha_i)(n + 1)^{\alpha_i}, j = 0, \\ (n - j - 2)^{a_i + 1} + (n - j)^{a_i + 1} - 2(n - j + 1)^{a_i + 1}, 1 \leq j \leq n, (i = 1, 2, 3) \\ 1, j = n + 1 \end{cases}$$

, and the predicted value $x_h^p(t_{n+1})$ is determined by:

$$\begin{cases} x_h^p(t_{n+1}) = x(0) + \dfrac{1}{\Gamma(\alpha_1)} \displaystyle\sum_{j=0}^{n} b_{1,j,n+1} f_1\big(x_h(t_j)\big) \\ y_h^p(t_{n+1}) = y(0) + \dfrac{1}{\Gamma(\alpha_2)} \displaystyle\sum_{j=0}^{n} b_{2,j,n+1} f_2\big(x_h(t_j)\big) \\ z_h^p(t_{n+1}) = z(0) + \dfrac{1}{\Gamma(\alpha_3)} \displaystyle\sum_{j=0}^{n} b_{3,j,n+1} f_3\big(x_h(t_j)\big) \end{cases} \tag{4}$$

In which $b_{i,j,n+1} = \frac{h\alpha_i}{\alpha_i}((n - j + 1)\alpha_i - (n - j)\alpha_i)(i = 1, 2, 3)$, $0 \leq j \leq n$. The estimation error in this method is $e = \max\{\max|x(t_j) - x_h(t_j)|, \max|y(t_j) - y_h(t_j)|, \max|z(t_j) - z_h(t_j)|\} = o(h^p)$, where $j = (0, 1, 2, \ldots, N), \rho = \min\{1 + \alpha_1, 1 + \alpha_2, 1 + \alpha_3\}$. By using this method, we can determine the numerical solution of the fractional-order difference equation system.

3 DESIGN OF TWO FUNDAMENTAL FRACTIONAL-ORDER LINEAR SYSTEMS

Considering the fractional-order version of Rucklidge system, it can be expressed by the following formula:

$$\begin{cases} D_t^{\alpha_1} x(t) = \dfrac{d^{\alpha_1} x(t)}{dt^{\alpha_1}} = -2\text{x} + 6.7y - yz \\ D_t^{\alpha_2} y(t) = \dfrac{d^{\alpha_2} y(t)}{dt^{\alpha_2}}(t) = x \\ D_t^{\alpha_3} z(t) = \dfrac{d^{\alpha_3} y(t)}{dt^{\alpha_3}} = y^2 - z \end{cases} \tag{5}$$

where $\alpha_{\text{i}}\,(i = 1, 2, 3)$ is the fractional-order satisfying $0 < \alpha_i \leq 1$. Clearly, system (**Eq. 5**) has three equilibria: $E_0 = (0, 0, 0)$, $E_1 = (0, \sqrt{6.7}, 6.7), E_2 = (0, -\sqrt{6.7}, 6.7)$. Linearizing system (**Eq. 5**) at equilibrium point E_1, one gets the following fractional-order linear system:

$$\begin{pmatrix} D_t^{\alpha_1} x_1(t) \\ D_t^{\alpha_2} \text{y}_1(t) \\ D_t^{\alpha_3} z_1(t) \end{pmatrix} = \begin{pmatrix} -2 & 0 & -\sqrt{6.7} \\ 1 & 0 & 0 \\ 0 & 2\sqrt{6.7} & -1 \end{pmatrix} \begin{pmatrix} \text{x}_1 \\ y_1 \\ z_1 \end{pmatrix} = J_1 X_1 \tag{6}$$

In the same way, we linearize system (**Eq. 5**) at equilibrium point E_2, and the following fractional-order linear system can be obtained:

$$\begin{pmatrix} D_t^{\alpha_1} x_2(t) \\ D_t^{\alpha_2} \text{y}_2(t) \\ D_t^{\alpha_3} z_2(t) \end{pmatrix} = \begin{pmatrix} -2 & 0 & \sqrt{6.7} \\ 1 & 0 & 0 \\ 0 & 2\sqrt{6.7} & -1 \end{pmatrix} \begin{pmatrix} \text{x}_2 \\ y_2 \\ z_2 \end{pmatrix} = J_2 X_2 \tag{7}$$

Through the processing of the above method, systems (**Eqs 6**, **7**) can be called basic fractional-order linear systems. Obviously, the only equilibrium point of systems **Eqs 6**, **7** is $O_0 = O_1 = (0, 0, 0)$, and the corresponding eigenvalues are $\lambda_1 = \gamma = -3.5145$ and $\lambda_{2,3} = \sigma \pm j\omega = 0.2577 \pm j1.9353$. Therefore, equilibrium points O_1 and O_2 become saddle focus with index 2. Moreover, $\lambda_1 < 0$, $R_e(\lambda_{2,3}) > 0$ and $|\lambda_1| > R_e(\lambda_{2,3})$, satisfy the conditions of Shil'nikov Theorem [47]. If all eigenvalues of Jacobian matrix $A = \partial f / \partial x$ meet the following condition:

$$|\arg(eig(A))| > \alpha\pi/2 \tag{8}$$

According to the analysis in [56], the equilibrium points in systems **Eqs 6**, **7** are locally asymptotically stable. If a system has more memory, the system is usually more stable than those systems with less memory [57]. It can be seen from inequality (**Eq. 8**) that due to the large memory of fractional-order differential equation systems, they are more stable than integer order equation systems. Through analysis, we conclude that the unstable regions are shown in **Figure 1**. It can be seen from the figure that except for the unstable regions, other regions are stable regions.

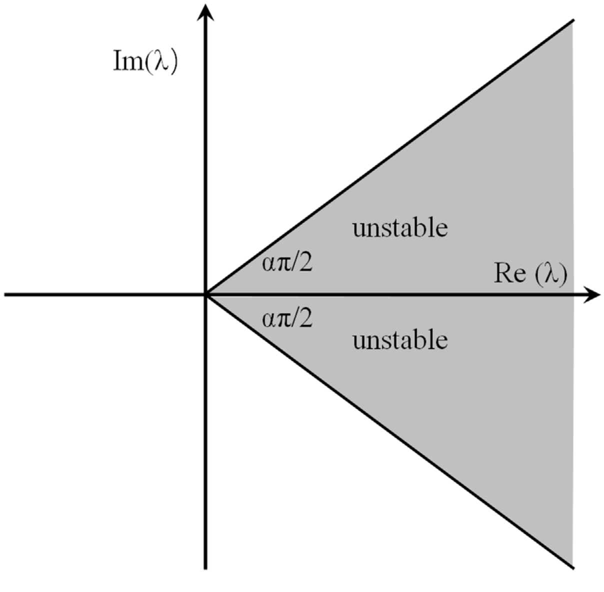


FIGURE 1 | Stability region of fractional-order linear system when fractional-order is equal to $\alpha\pi/2$.

$\alpha\pi/2$When the values of $\alpha_i < 1$ (i = 1, 2, 3) change, system **Eqs 6**, **7** do not always remain chaotic. According to inequality (**Eq. 8**), if the system (**Eq. 6**) wants to maintain a chaotic state, in the unstable regions, at each non original equilibrium point of the system (**Eq. 6**), the Jacobian matrix must have two conjugate eigenvalues [58]. According to this description, we have $\alpha_i > \frac{2}{\pi}\arctan\left(\frac{1.9353}{0.2577}\right) \approx 0.916$, for i = 1, 2, 3. Moreover, their corresponding eigenvectors are given as follows:

$$\begin{cases} \eta_1 = \begin{pmatrix} -0.4725 \\ 0.1737 \\ 0.7149 \end{pmatrix} \pm \text{j} \begin{pmatrix} 0.4050 \\ 0.2673 \\ 0 \end{pmatrix} \\ \mu_1 = \begin{pmatrix} -0.8381 \\ 0.2384 \\ -0.4907 \end{pmatrix} \end{cases} ; \begin{cases} \eta_2 = \begin{pmatrix} 0.4725 \\ -0.1737 \\ 0.7149 \end{pmatrix} \pm \text{j} \begin{pmatrix} -0.4050 \\ -0.2673 \\ 0 \end{pmatrix} \\ \mu_2 = \begin{pmatrix} -0.8381 \\ 0.2384 \\ 0.4907 \end{pmatrix} \end{cases}$$

Through analysis and calculation, one-dimensional stable eigenline $E^S(O_1)$ and two-dimensional unstable eigenplane $E^U(O_1)$ of system (**Eq. 6**) at O_1 can be obtained:

$$\begin{cases} E^S(O_1)\colon \dfrac{x}{l_1} = \dfrac{y}{m_1} = \dfrac{z}{n_1} \\ E^U(O_1)\colon A_1 x + B_1 y + C_1 z = 0 \end{cases} \tag{9}$$

Here $l_1 = -0.8381, m_1 = 0.2384, n_1 = -0.4907$, $A_1 = -0.1911$, $B_1 = 0.2895$, $C_1 = -0.1966$. Similarly, one-dimensional stable eigenline $E^S(O_2)$ and two-dimensional unstable eigenplane $E^U(O_2)$ of system (**Eq. 7**) at O_2 can be obtained:

$$\begin{cases} E^S(O_2)\colon \dfrac{x}{l_2} = \dfrac{y}{m_2} = \dfrac{z}{n_2} \\ E^U(O_2)\colon A_2 x + B_2 y + C_2 z = 0 \end{cases} \tag{10}$$

Here $l_2 = -0.8381, m_2 = 0.2384, n_2 = 0.4907$, $A_2 = 0.1911, B_2 = -0.2895, C_2 = -0.1966$. Clearly, the stable manifolds $E^S(O_1)$ and $E^S(O_2)$ are symmetric to a certain extent, respectively, as are unstable manifolds $E^U(O_1)$ and $E^U(O_2)$. Based on this symmetry, we can construct a heteroclinic loop, and heteroclinic chaos can be generated from fractional-order multi piecewise Rucklidge system, just

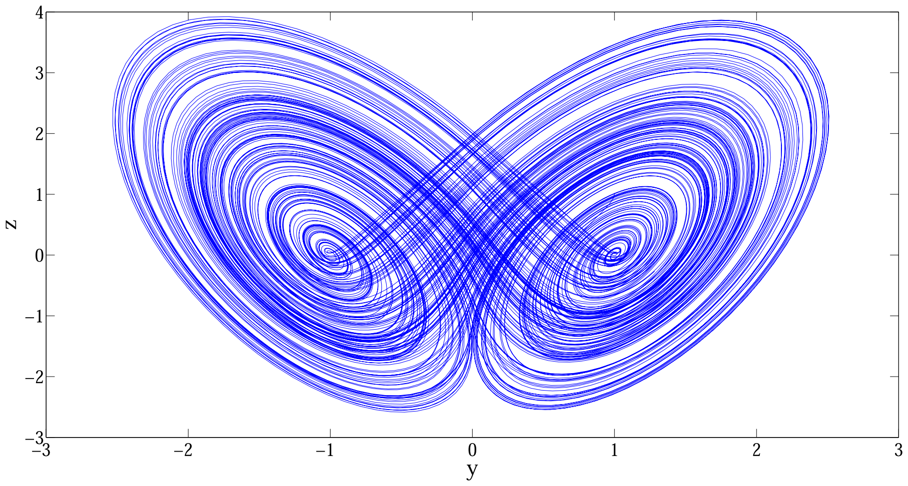


FIGURE 2 | Double-wing buttery chaotic attractor of system (**Eq. 12**) When $\alpha_i = 0.92$.

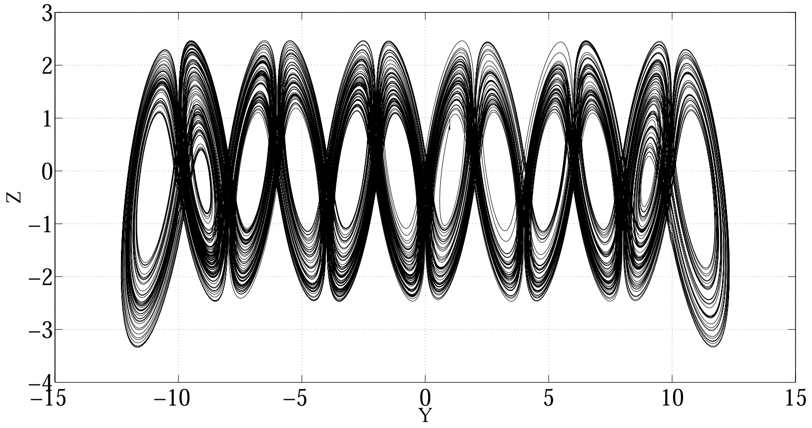


FIGURE 3 | 12-wing chaotic attractor of system (**Eq. 13**) when $\alpha_i = 0.92$.

as heteroclinic chaos can be generated from integer multi piecewise linear system [53].

4 DESIGN OF TWO-WING AND MULTI-WING CHAOTIC ATTRACTORS

In this section, we construct heteroclinic loops, and then use the switching control method to design two-wing and multi-wing chaotic attractors in the two basic fractional-order linear systems introduced earlier. Based on the fundamental fractional linear systems **Eqs 6**, **7**, extend the heteroclinic Shil'nikov theorem [47], we can design a switch controller and then connect the heteroclinic track of systems **Eqs 6**, **7** to make the track form a heteroclinic loop. Set the switching controller to $w = F(x, y, z)$ and the switching plane to $S = \{(x, y, z)|y = 0\}$. From systems **Eqs 6**, **7**, the following switching systems can be constructed:

$$\begin{cases} \begin{pmatrix} D_t^{\alpha_1} x(t) \\ D_t^{\alpha_2} y(t) \\ D_t^{\alpha_3} z(t) \end{pmatrix} = \begin{pmatrix} -2 & 0 & -\sqrt{6.7} \\ 1 & 0 & 0 \\ 0 & 2\sqrt{6.7} & -1 \end{pmatrix} \left(\begin{pmatrix} \mathrm{x} \\ y \\ z \end{pmatrix} - w \right) = J_1 (X - w) V \in V_1 = \{(x, y, z)|y > 0\} \\ \begin{pmatrix} D_t^{\alpha_1} x(t) \\ D_t^{\alpha_2} y(t) \\ D_t^{\alpha_3} z(t) \end{pmatrix} = \begin{pmatrix} -2 & 0 & \sqrt{6.7} \\ 1 & 0 & 0 \\ 0 & -2\sqrt{6.7} & -1 \end{pmatrix} \left(\begin{pmatrix} \mathrm{x} \\ y \\ z \end{pmatrix} - w \right) = J_2 (X - w) V \in V_2 = \{(x, y, z)|y > 0\} \end{cases} \tag{11}$$

It should be noted here that according to Ref. [54], the existence condition of heteroclinic loop in system (**Eq. 11**) can determine the detailed mathematical expression of switching controller $w = F(x, y, z) = [f_1(x, y, z), f_2(x, y, z), f_3(x, y, z)]^T$. It is obvious that the equilibrium points $P_2(x_2, y_2, z_2) \in V_2$ and $P_1(x_1, y_1, z_1) \in V_1$ of system **Eq. 11** are located on either side of the switching plane $S = \{(x, y, z)|y = 0\}$.

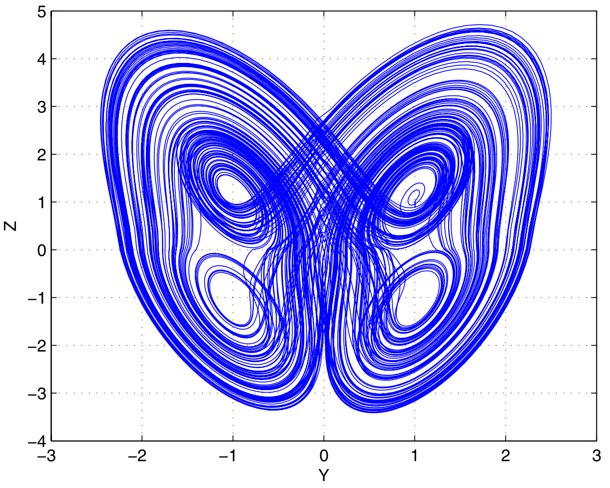


FIGURE 4 | 2 × 2-wing chaotic attractor of system (**Eq. 13**) when $\alpha_i = 0.94$

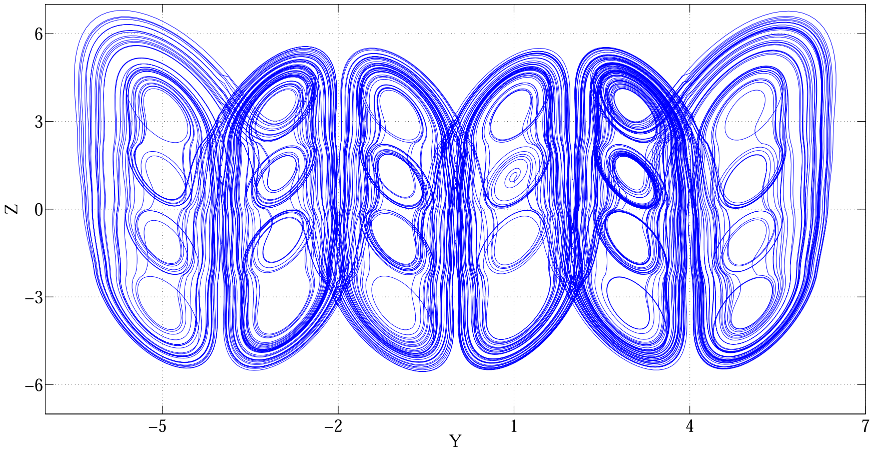


FIGURE 5 | 6 × 4-wing chaotic attractor of system (**Eq. 13**) When $\alpha_i = 0.93$

Here, $E^S(P_1)$ and $E^S(P_2)$ are the eigenlines systems **Eqs 6, 7** at P_1 and P_2, $E^U(P_1)$ and $E^U(P_2)$ are the eigenplanes of systems **Eqs 6, 7** at P_1 and P_2, respectively. Let:

$$Q_1 = E^S(P_1) \cap S = \left(x_1 - \frac{l_1}{m_1} y_1, 0, z_1 - \frac{n_1}{m_1} y_1\right),$$

$$Q_2 = E^S(P_2) \cap S = \left(x_2 - \frac{l_2}{m_2} y_2, 0, z_2 - \frac{n_2}{m_2} y_2\right),$$

$$L_1 = E^U(P_1) \cap S = A_1(x - x_1) + B_1(0 - y_1) + C_1(z - z_1) = 0,$$

$$L_2 = E^U(P_2) \cap S = A_2(x - x_2) + B_2(0 - y_2) + C_2(z - z_2) = 0.$$

If Q_1 is at L_2, there is heteroclinic orbital $H_1 = E^U(P_2) \cup Q_1 \cup E^S(P_1)$ between P_2 and P_1. Similarly, if Q_2 is at L_1, there is heteroclinic orbital $H_2 = E^U(P_1) \cup Q_2 \cup E^S(P_2)$ between P_1 and P_2. Thus, if Q_1 is located at L_2, Q_2 is located at L_1, then a heterotopic loop is formed by two heteroclinic orbitals H_1 and H_2, which join P_1 and P_2 together. According to heteroclinic Shil'nikov theory [47], if this situation exists, the system (6) has chaotic state in the sense of Smale's horseshoe.

From the transformation $(x, y, z) \to (-x, -y, z)$, it can be seen that there is invariance of the system, so the switching plane $S = \{(x, y, z) | y = 0\}$ satisfying $P_1(x_1, y_1, z_1) \in V_1$ and $P_2(x_2, y_2, z_2) \in V_2$ with $x_1 = -x_2 = x_0$, $y_1 = -y_2 = y_0 > 0$, and $z_1 = z_2 = z_0$ can be selected. Thus, one can deduce the necessary conditions of $Q_1 \in L_2$ and $Q_2 \in L_1$ as follows: $x_0 = \frac{A_1 l_2 - B_1 m_2 + C_1 n_2}{2A_1 m_2} y_0 = \frac{A_2 l_1 - B_2 m_1 + C_2 n_1}{2A_2 m_1} y_0$.

Which indicates that y_0 depending on x_0, and z_0 can be any values. In this case, let $y_0 = y_1 = -y_2 = 1$ and $z_1 = z_2 = z_0 = 0$, then one gets $x_0 = x_1 = -x_2 = 0.0585$, $P_1(x_1, y_1, z_1) = P_1(0.0585, 1, 0)$, and $P_2(x_2, y_2, z_2) = P_2(-0.0585, -1, 0)$. Thus, the switching controller is $F(x, y, z) = (x_0 s(y), y_0 s(y), z_0 s(z))^T$ with $x_0 = 0.0585, y_0 = 1$, and $z_0 = 0$, where s(y) (or s(z))is the saturated function, is described by: $s(y) = \frac{1}{2\alpha}(|y + \alpha| - |y - \alpha|)$.Where α decides the slope of the saturated function, here we set $\alpha = 0.01$.

According to the above theoretical analysis, switch controller $s_0 = s(y)$, $w_0 = F(x, y, z) = (x_0 s(y), y_0 s(y), z_0 s(z))^T$ can be designed, where $x_0 = 0.0585, y_0 = 1$, and $z_0 = 0$. From system (**Eq. 11**), one gets

$$\begin{pmatrix} D_t^{\alpha_1} x(t) \\ D_t^{\alpha_2} y(t) \\ D_t^{\alpha_3} z(t) \end{pmatrix} = \begin{pmatrix} -2 & 0 & -s(y)\sqrt{6.7} \\ 1 & 0 & 0 \\ 0 & s(y) 2\sqrt{6.7} & -1 \end{pmatrix} \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix} - F(x, y, z) \right) \tag{12}$$

The simulation of various fractional-order double-wing buttery chaotic attractor can be obtained when $\alpha_i > 0.916$, as shown in **Figure 2** when $\alpha_i = 0.92$.

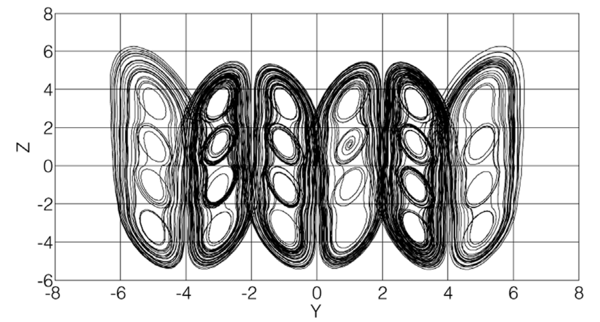


FIGURE 6 | 6 × 4 -wing chaotic attractor of system (**Eq. 13**) When $\alpha_i = 0.95$

$\alpha_i = 0.92$In the same way, we can design the following systems according to systems (**Eqs 6**, **11**):

$$\begin{pmatrix} D_t^{\alpha_1} x(t) \\ D_t^{\alpha_2} y(t) \\ D_t^{\alpha_3} z(t) \end{pmatrix} = \begin{pmatrix} -2 & 0 & -T\sqrt{6.7} \\ 1 & 0 & 0 \\ 0 & 2T\sqrt{6.7} & -1 \end{pmatrix} \left(\begin{pmatrix} x \\ y \\ z \end{pmatrix} - F(x, y, z) \right) \tag{13}$$

where $F(x, y, z)$ is the equilibrium switching controller, and $T = T(x, y, z)$ is the parameter switching controller. $F(x, y, z)$ and $T(x, y, z)$ are the sequence of saturation functions here, as shown below:

$$T(x, y, z) = s(y) + \sum_{m=1}^{M} (-1)^m \left[s(y + 2my_0) + s(y - 2my_0) \right] \tag{14}$$

$$F(x, y, z) = \begin{pmatrix} f_1(x, y, z) \\ f_2(x, y, z) \\ f_3(x, y, z) \end{pmatrix} = \begin{pmatrix} x_0 s(y) + \sum\limits_{m=1}^{M} x_0 \left[s(y + 2my_0) + s(y - 2my_0) \right] \\ y_0 s(y) + \sum\limits_{m=1}^{M} y_0 \left[s(y + 2my_0) + s(y - 2my_0) \right] \\ z_0 s(y) + \sum\limits_{n=1}^{N} z_0 \left[s(z + 2mz_0) + s(z - 2mz_0) \right] \end{pmatrix} \tag{15}$$

where $s(y + 2\mathrm{my}_0) = \frac{1}{2\alpha} \left(|y + 2my_0 + \alpha| - |y + 2my_0 - \alpha| \right)$.

The prediction correction algorithm is used to solve the fractional-order differential system (**Eq. 13**), and the simulation results of the fractional-order 12-wing butterfly chaotic attractor are obtained when $\alpha_i > 0.916$, $x_0 = 0.0585, y_0 = 1$, $z_0 = 0$ $N = 0$ and $M = 5$, the multi-wing attractors are shown in **Figure 3** when $\alpha_i = 0.92$.

A grid multi-wing butterfly chaotic attractor with a grid of 2 × 2 is obtained when $x_0 = 0.0585$, $y_0 = 1$, $z_0 = 1$, $M = 0$, $N = 0$ and $\alpha_i = 0.94$, as shown in **Figure 4**.

$\alpha_i = 0.94$A grid multi-wing butterfly chaotic attractor with a grid of 6 × 4 is obtained when $x_0 = 0.0585$, $y_0 = 1$, $z_0 = 1.125$, $M = 2$, $N = 1$ and $\alpha_i = 0.93$, as shown in **Figure 5**.

$\alpha_i = 0.93$Keeping other parameters unchanged, change the order to $\alpha_i = 0.95$. The simulation results are shown in **Figure 6**, it can be seen that the system (**Eq. 13**) can also generate grid 6 × 4-wing chaotic attractors.

From the above simulation results, it can be seen that if the appropriate parameters are set, when $\alpha_i > 0.916$, the system can generate multi-wing and grid multi-wing chaotic attractors.

5 CONCLUSION

In this paper, based on fractional-order linear differential system, a novel grid multi-wing chaotic attractor is proposed by switching saturation function control and constructing heteroclinic loops. Firstly, the two symmetric equilibrium points of the fractional Rucklidge system are linearized to obtain two basic fractional-order linear systems. Then all the equilibrium points of the two basic fractional-order linear systems are connected by a saturation function switching control and a heteroclinic loop. Finally, the effectiveness of the proposed design method is verified by numerical simulation. Since the proposed fractional-order chaotic system can generate multi-wing chaotic attractors with complex dynamic characteristics, however, it does not contain product terms and is easy to implement in circuits, so the chaotic system proposed in this paper has abundant potential engineering applications. In the future, we will further design the circuit realization of the fractional-order multi-wing chaotic system.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

YL, XZ, FY and YH contributed to conception and design of the study. JG and YH organized the database. YL and FY performed the statistical analysis. YL and FY wrote the first draft of the manuscript. YL, XZ, JG, and FY wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

FUNDING

This work is supported by the Hunan Provincial Natural Science Foundation of China under Grant 2019JJ60034, and the Scientific Research Fund of Hunan Provincial Education Department under Grants 19A106, 19B131, 21B0345 and 20k306, and by the Industry University Research Innovation Fund of Chinese Universities—a new generation of information technology innovation project under Grant 2020ITA07029.

REFERENCES

1. Yu F, Kong X, Chen H. A 6D Fractional-Order Memristive Hopfield Neural Network and its Application in Image Encryption. *Front Phys* (2022) 10: 847385. doi:10.3389/fphy.2022.847385
2. Xu Q, Cheng S, Ju Z, Chen M, Wu H. Asymmetric Coexisting Bifurcations and Multi-Stability in an Asymmetric Memristive Diode-Bridge-Based Jerk Circuit. *Chin J Phys* (2021) 70:69–81. doi:10.1016/j.cjph.2020.11.007
3. Xu C, Wang C, Jiang J, Sun J, Lin H. Memristive Circuit Implementation of Context-dependent Emotional Learning Network and its Application in Multi-Task. *IEEE Trans Comput.-Aided Des Integr Circuits Syst* (2021) 1. doi:10.1109/TCAD.2021.3116463
4. Yao W, Wang C, Sun Y, Zhou C. Robust Multimode Function Synchronization of Memristive Neural Networks with Parameter Perturbations and Time-Varying Delays. *IEEE Trans Syst Man Cybern: Syst* (2022) 52(1):260–274. doi:10.1109/TSMC.2020.2997930
5. Wan Q, Yan Z, Li F, Liu J, Chen S. Multistable Dynamics in a Hopfield Neural Network Under Electromagnetic Radiation and Dual Bias Currents. *Nonlinear Dyn* (2022). doi:10.1007/s11071-022-07544-x
6. Ma M, Yang Y, Qiu Z, Peng Y, Sun Y, Li Z, et al. A Locally Active Discrete Memristor Model and its Application in a Hyperchaotic Map. *Nonlinear Dyn* (2022) 107:2935–49. doi:10.1007/s11071-021-07132-5
7. Yu F, Qian S, Chen X, Huang Y, Liu L, Shi C, et al. A New 4D Four-Wing Memristive Hyperchaotic System: Dynamical Analysis, Electronic Circuit Design, Shape Synchronization and Secure Communication. *Int J Bifurcation Chaos* (2020) 30(10):2050147. doi:10.1142/S0218127420501473
8. Zhou L, Tan F, Yu F. A Robust Synchronization-Based Chaotic Secure Communication Scheme with Double-Layered and Multiple Hybrid Networks. *IEEE Syst J* (2019) 14(2):2508–19. doi:10.1109/JSYST.2019.2927495
9. Yu F, Zhang Z, Liu L, Shen H, Huang Y, Shi C, et al. Secure Communication Scheme Based on a New 5D Multistable Four-Wing Memristive Hyperchaotic System with Disturbance Inputs. *Complexity* (2020) 2020, 5859273. doi:10.1155/2020/5859273
10. Li Y, Li Z, Ma M, Wang M. Generation of Grid Multi-Wing Chaotic Attractors and its Application in Video Secure Communication System. *Multimed Tools Appl* (2020) 79:29161–77. doi:10.1007/s11042-020-09448-7
11. Yu F, Liu L, He B, Huang Y, Shi C, Cai S, et al. Analysis and FPGA Realization of a Novel 5D Hyperchaotic Four-Wing Memristive System, Active Control Synchronization, and Secure Communication Application. *Complexity* (2019) 2019, 4047957. doi:10.1155/2019/4047957
12. Li X, Mou J, Banerjee S, Cao Y. An Optical Image Encryption Algorithm Based on Fractional-Order Laser Hyperchaotic System. *Int J Bifurcation Chaos* (2022) 32(03):2250035. doi:10.1142/s0218127422500353
13. Fei Y, Zhang Z, Shen H. FPGA Implementation and Image Encryption Application of a New PRNG Based on a Memristive Hopfield Neural Network with a Special Activation Gradient. *Chin Phys B* (2022) 31(2): 020505. doi:10.1088/1674-1056/ac3cb2
14. Deng J, Zhou M, Wang C, Wang S, Xu C. Image Segmentation Encryption Algorithm with Chaotic Sequence Generation Participated by Cipher and Multi-Feedback Loops. *Multimed Tools Appl* (2021) 80:13821–40. doi:10.1007/s11042-020-10429-z
15. Gao X, Mou J, Xiong L, Sha Y, Yan H, Cao Y. A Fast and Efficient Multiple Images Encryption Based on Single-Channel Encryption and Chaotic System. *Nonlinear Dyn* (2022) 108:613–36. doi:10.1007/s11071-021-07192-7
16. Zeng J, Wang C. A Novel Hyperchaotic Image Encryption System Based on Particle Swarm Optimization Algorithm and Cellular Automata. *Secur. Commun. Netw.* (2021) 2021:6675565. doi:10.1155/2021/6675565
17. Gao X, Mou J, Banerjee S, Cao Y, Xiong L, Chen X. An Effective Multiple-Image Encryption Algorithm Based on 3D Cube and Hyperchaotic Map. *J King Saud Univ - Computer Inf Sci* (2022) 34:1535–51. doi:10.1016/j.jksuci.2022.01.017
18. Yu F, Liu L, Li K, Cai S. A Robust and Fixed-Time Zeroing Neural Dynamics for Computing Time-Variant Nonlinear Equation Using a Novel Nonlinear Activation Function. *Neurocomputing* (2019) 350:108–16. doi:10.1016/j.neucom.2019.03.053
19. Yang L, Wang C. Emotion Model Of Associative Memory Possessing Variable Learning Rates With Time Delay. *Neurocomputing* (2021) 460(14):117–125. doi:10.1016/j.neucom.2021.07.011
20. Xu C, Wang C, Sun Y, Hong Q, Deng Q, Chen H. Memristor-based Neural Network Circuit with Weighted Sum Simultaneous Perturbation Training and its Applications. *Neurocomputing* (2021) 462:581–90. doi:10.1016/j.neucom.2021.08.072
21. Lin H, Wang C, Deng Q, Xu C, Deng Z, Zhou C. Review on Chaotic Dynamics of Memristive Neuron and Neural Network. *Nonlinear Dyn* (2021) 106(1): 959–73. doi:10.1007/s11071-021-06853-x
22. Xu Q, Ju Z, Ding S, Feng C, Chen M, Bao B. Electromagnetic Induction Effects on Electrical Activity within a Memristive Wilson Neuron Model. *Cogn Neurodyn* (2022). doi:10.1007/s11571-021-09764-0
23. Zhou C, Wang C, Sun Y, Yao W. Weighted Sum Synchronization of Memristive Coupled Neural Networks. *Neurocomputing* (2020) 403, 211–223. doi:10.1016/j.neucom.2020.04.087
24. Fei Y, Zhang Z, Shen H. Design and FPGA Implementation of a Pseudo-random Number Generator Based on a Hopfield Neural Network under Electromagnetic Radiation. *Front Phys* (2021) 9:690651. doi:10.3389/fphy.2021.690651
25. Yao W, Wang C, Sun Y, Zhou C, Lin H, et al. Synchronization of Inertial Memristive Neural Networks With Time-Varying Delays via Static or Dynamic Event-Triggered Control. *Neurocomputing* (2020) 404:367–80. doi:10.1016/j.neucom.2020.04.099
26. Xiong B, Yang K, Zhao J, Li K. Robust Dynamic Network Traffic Partitioning against Malicious Attacks. *J Netw Computer Appl* (2017) 87:20–31. doi:10.1016/j.jnca.2016.04.013
27. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S. Dynamics Analysis, Hardware Implementation And Engineering Applications Of Novel Multi-Style Attractors In A Neural Network Under Electromagnetic Radiation. *Chaos, Solitons & Fractals* (2021) 152:111350. doi:10.1016/j.chaos.2021.111350
28. Yao W, Wang C, Cao J, Sun Y, Zhou C. Hybrid Multisynchronization of Coupled Multistable Memristive Neural Networks With Time Delays. *Neurocomputing* (2019) 363:281–94. doi:10.1016/j.neucom.2019.07.014
29. Long M, Zeng Y. Detecting Iris Liveness with Batch Normalized Convolutional Neural Network. *CMC-Computers Mater Continua* (2019) 58(2):493–504. doi:10.32604/cmc.2019.04378
30. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S. A New Multi-Scroll Chua's Circuit With Composite Hyperbolic Tangent-Cubic Nonlinearity: Complex Dynamics, Hardware Implementation and Image Encryption Application. *Integration* (2021) 81:71–83. doi:10.1016/j.vlsi.2021.05.011
31. Zhang X, Wang C. A Novel Multi-Attractor Period Multi-Scroll Chaotic Integrated Circuit Based on CMOS Wide Adjustable CCCII. *IEEE Access* (2019) 7(1):16336–50. doi:10.1109/access.2019.2894853
32. Yu F, Chen H, Kong X. Dynamic Analysis and Application in Medical Digital Image Watermarking of a New Multi-Scroll Neural Network with Quartic Nonlinear Memristor. *Eur Phys J Plus* (2022) 137:434. doi:10.1140/epjp/s13360-022-02652-4
33. Zhou L, Wang C, Zhou L. A Novel No-Equilibrium Hyperchaotic Multi-Wing System via Introducing Memristor. *Int J Circ Theor Appl* (2018) 46(1):84–98. doi:10.1002/cta.2339
34. Zhou L, Wang C, Zhou L. Generating Hyperchaotic Multi-wing Attractor in a 4D Memristive Circuit. *Nonlinear Dyn* (2016) 85(4):2653–63. doi:10.1007/s11071-016-2852-8
35. Cui L, Lu M, Ou Q, Duan H, Luo W. Analysis and Circuit Implementation of Fractional Order Multi-wing Hidden Attractors. *Chaos, Solitons and Fractals* (2020) 138:109894. doi:10.1016/j.chaos.2020.109894
36. Podlubny I. *Fractional Differential Equations*. New York: Academic Press (1999):340–68.
37. Hifer R. Applications of Fractional Calculus in Physics. *World Scientific* (2001) 120–75.
38. Cafagna D, Grassi G. Bifurcation and Chaos in the Fractional-Order Chen System via a Time-Domain Approach. *Int J Bifurcation Chaos* (2008) 18: 1845–63. doi:10.1142/s0218127408021415
39. Yang F, Mou J, Liu J. Characteristic Analysis of the Fractional-Order Hyperchaotic Complex System and its Image Encryption Application. *Signal Process.* (2020) 169:107373. doi:10.1016/j.sigpro.2019.107373

40. Xie W, Wang C, Lin H. A Fractional-Order Multistable Locally Active Memristor and its Chaotic System with Transient Transition, State Jump. *Nonlinear Dyn* (2021) 104:4523–41. doi:10.1007/s11071-021-06476-2
41. Deng W, Li C. Synchronization of Chaotic Fractional Chen System. *J Phys Soc Jpn* (2005) 74:1645–8. doi:10.1143/jpsj.74.1645
42. Ahmad WM. Generation and Control of Multi-Scroll Chaotic Attractors in Fractional Order Systems. *Chaos, Solitons and Fractals* (2005) 25:727–35. doi:10.1016/j.chaos.2004.11.073
43. Cafagna D, Grassi G. Fractional-order Chaos: a Novel Four-wing Attractor in Coupled Lorenz Systems. *Int J Bifurcation Chaos* (2009) 19:3329–38. doi:10.1142/s0218127409024785
44. Deng W, Lü J. Design of Multidirectional Multiscroll Chaotic Attractors Based on Fractional Differential Systems via Switching Control. *Chaos* (2006) 16: 043120. doi:10.1063/1.2401061
45. Zhang C, Yu S. Generation of Multi-wing Chaotic Attractor in Fractional Order System. *Chaos, Solitons and Fractals* (2011) 44:845–50. doi:10.1016/j.chaos.2011.06.017
46. Petras I. Fractional-Order Memristor-Based Chua's Circuit. *IEEE Trans Circuits Syst* (2010) 57(12):975–9. doi:10.1109/tcsii.2010.2083150
47. Silva CP. Shil'nikov's Theorem-A Tutorial. *IEEE Trans Circuits Syst* (1993) 40: 675–82. doi:10.1109/81.246142
48. Chua L, Komuro M, Matsumoto T. The Double Scroll Family. *IEEE Trans Circuits Syst* (1986) 33:1072–118. doi:10.1109/tcs.1986.1085869
49. Shilnikov LP. Chuas Circuit: Rigorous Results and Future Problems. *IEEE Trans Circuits Syst Fundam Theor Appl* (1993) 40:784–6. doi:10.1109/81.246153
50. Mees A, Chapman P. Homoclinic and Heteroclinic Orbits in the Double Scroll Attractor. *IEEE Trans Circuits Syst* (1987) 34:1115–20. doi:10.1109/tcs.1987.1086251
51. Li Z, Chen G, Halang WA. Homoclinic and Heteroclinic Orbits in a Modified Lorenz System. *Inf Sci* (2004) 165:235–45. doi:10.1016/j.ins.2003.06.005
52. Li G, Chen X. Constructing Piecewise Linear Chaotic System Based on the Heteroclinic Shil'nikov Theorem. *Commun Nonlinear Sci Numer Simulation* (2009) 14:194–203. doi:10.1016/j.cnsns.2007.07.007
53. Rucklidge AM. Chaos in Models of Double Convection. *J Fluid Mech* (1992) 237:209–29. doi:10.1017/s0022112092003392
54. Yu S, Lu J, Chen G, Yu X. Generating Grid Multiwing Chaotic Attractors by Constructing Heteroclinic Loops into Switching Systems. *IEEE Trans Circuits Syst* (2011) 58(5):314–8. doi:10.1109/tcsii.2011.2149090
55. Vaněček A, Čelikovský S. *Control Systems: From Linear Analysis to Synthesis of Chaos*. Englewood Cliffs, NJ: Prentice-Hall (1996):433–60.
56. Diethelm K, Ford NJ, Freed AD. A Predictor Corrector Approach for the Numerical Solution of Fractional Differential Equations. *Nonlinear Dyn* (2002) 29(1):3–22. doi:10.1023/a:1016592219341
57. Ahmed E, El-Sayed AMA, El-Saka HAA. Equilibrium Points, Stability and Numerical Solutions of Fractional-Order Predator-Prey and Rabies Models. *J Math Anal Appl* (2007) 325(1):542–53. doi:10.1016/j.jmaa.2006.01.087
58. Mohammad ST, Mohammad H. A Necessary Condition for Double Scroll Attractor Existence in Fractional-Order Systems. *Phys Lett A* (2007) 367(1): 102–13. doi:10.1016/j.physleta.2007.05.081

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Physics

ORIGINAL RESEARCH
published: 28 June 2022
doi: 10.3389/fphy.2022.912606

Parameter Identification for Memristive Chaotic System Using Modified Sparrow Search Algorithm

Qi Xiong[1], Jun Shen[2]*, Bing Tong[1] and Ying Xiong[1]

[1]International College, Hunan University of Arts and Sciences, Changde, China, [2]School of Computing and Information Technology, University of Wollongong, Wollongong, NSW, Australia

A memristor is a non-linear element. The chaotic system constructed by it can improve its unpredictability and complexity. Parameter identification of a memristive chaotic system is the primary task to implement chaos control and synchronization. To identify the unknown parameters accurately and quickly, we introduce the Sine Pareto Sparrow Search Algorithm (SPSSA), a modified sparrow search algorithm (SSA). in this research. Firstly, we introduce the Pareto distribution to alter the scroungers' location in the SSA. Secondly, we use a sine-cosine strategy to improve the producers' position update. These measures can effectively accelerate the convergence speed and avoid local optimization. Thirdly, the SPSSA is used to identify the parameters of a memristive chaotic system. The proposed SPSSA exceeds the classic SSA, particle swarm optimization algorithm (PSO), and artificial bee colony algorithm (ABC) in simulations based on the five benchmark functions. The simulation results of parameter identification of a memristive chaotic system show that the method is feasible, and the algorithm has a fast convergence speed and high estimation accuracy.

Keywords: parameter identification, sparrow search algorithm, swarm intelligence, memristive chaotic system, pareto distribution, sine cosine algorithm

Edited by:
Jun Mou,
Dalian Polytechnic University, China

Reviewed by:
Chunbiao Li,
Nanjing University of Information Science and Technology, China
Jacques Kengne,
University of Dschang, Cameroon
Yuexi Peng,
Xiangtan University, China

***Correspondence:**
Jun Shen
jshen@uow.edu.au

Specialty section:
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

Received: *05 April 2022*
Accepted: *19 May 2022*
Published: *28 June 2022*

Citation:
Xiong Q, Shen J, Tong B and Xiong Y (2022) Parameter Identification for Memristive Chaotic System Using Modified Sparrow Search Algorithm. Front. Phys. 10:912606. doi: 10.3389/fphy.2022.912606

1 INTRODUCTION

A memristor is the fourth fundamental circuit element except for resistance R, capacitance C, and inductance. L. Strukov et al [1]. realized the first memristor in the world in 2008. It has set off an upsurge of memristor research. In recent years, memristors have been intensely studied in many application fields, such as memory [2], neural networks [3, 4], and image processing [5]. A memristor is a non-linear element. A chaotic system constructed by the memristor can improve the unpredictability and complexity of the system [6–10]. Applying a memristive chaotic system is also one of the research hotspots [11–13].

In practical engineering applications, the parameter identification of chaotic systems is the primary problem in realizing chaos control and synchronization. The accuracy of the parameter identification will directly affect the control effect of a chaotic system. Therefore, it is essential to accurately identify the parameters of a chaotic system, which has crucial research significance. The parameter identification of a chaotic system is essentially a complex non-linear numerical optimization problem based on multi-dimensional search space. Because swarm intelligence (SI) optimization algorithm does not need the derivative information of objective function, it has more advantages than traditional optimization algorithm in parameter identification. At present, many research results have emerged for some classical continuous memristive chaotic systems, such as

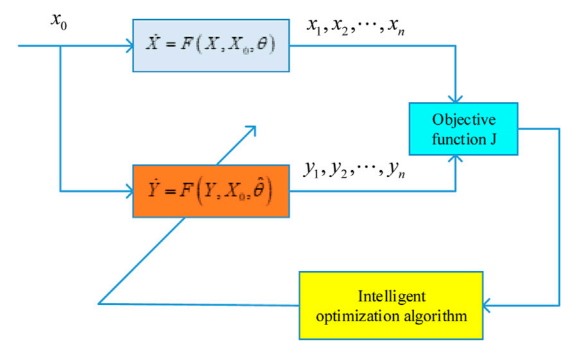


FIGURE 1 | Identification principle of a chaotic system [20].

TABLE 1 | Benchmark test functions.

Benchmark Functions	Dim	Range	F_{min}
$F_1(x) = \sum_{i=1}^{n} x_i^2$	30	[-100, 100]	0
$F_2(x) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	[-10, 10]	0
$F_3(x) = \sum_{i=1}^{n} (\sum_{j-1}^{i} x_j)^2$	30	[-100, 100]	0
$F_4(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$	30	[-100, 100]	0
$F_5(x) = \sum_{i=1}^{n} -x_i \sin(\sqrt{\lvert x_i \rvert})$	30	[-500,500]	-418.9829×n

particle swarm optimization algorithm [14, 15], differential evolution algorithm [16, 17], artificial bee colony optimization algorithm [18], bird swarm algorithm [19], Jaya algorithm [20]. For a discrete memristive chaotic system, the swarm intelligence optimization algorithms also provide many practical solutions, such as an improved PSO algorithm [21], meta-heuristic algorithm [22], and enhanced differential evolution algorithm [23].

The Sparrow search algorithm (SSA) is one of SI optimization algorithms proposed in recent years [24]. It is created by imitating the foraging behavior of a group of sparrows. Compared with other SI optimization algorithms, SSA has the advantages of fewer parameters, simple calculation, and easy implementation in dealing with multi-dimensional problems and global search. Therefore, SSA is used to solve the problem of parameter identification in a chaotic system. However, the traditional SSA is also easy to fall into local optimization. To avoid this phenomenon, many researchers proposed some improved methods. For example, Xiong et al. [25] used a fractional-order chaotic sequence to increases the diversity of the population and used Pareto mutation to escape local best. These methods achieve better results than traditional SSA on 12 benchmark functions. But it is more time-consuming than SSA. In this study, we propose a modified SSA called Sine Pareto Sparrow Search Algorithm (SPSSA). The following are the primary contributions of this paper:

(1) We use a sine-cosine strategy to improve the producers' position update.
(2) To improve the location of the scroungers, we employ Pareto distribution. This concept is useful for speeding up global convergence and avoiding local minimum points.
(3) A memristive chaotic system's parameters are determined using the SPSSA.

In the following sections, **Section 2** introduces preliminaries for parameter identification and some concepts of the SPSSA. **Section 3** is the experimental results of the paper. Finally, our conclusions presented in **Section 4**.

TABLE 2 | Experiment results of benchmark functions.

F	SI	Best	Ave	Std	Computation time (Second)
F1	PSO	7.134288e+01	1.959255e+02	60.0022	1.17
	ABC	4.574426	8.274577	2.3966	17.44
	SSA	0	1.874825e-140	1.02688e-139	4.56
	SPSSA	0	0	0	8.01
F2	PSO	8.160150	3.057447e+01	20.0995	1.26
	ABC	2.483748	3.248522e+01	23.3522	16.77
	SSA	9.537863e-299	7.955561e-68	4.34102e-67	4.56
	SPSSA	0	0	0	8.28
F3	PSO	2.908686e+03	8.612808e+03	5414.9	6.76
	ABC	3.853256e+04	6.251440e+04	10947.1	27.90
	SSA	0	4.102431e-78	2.24699e-77	12.23
	SPSSA	0	0	0	13.40
F4	PSO	3.345754	7.431952	1.91963	1.28
	ABC	45.48858	57.4692	5.74807	16.76
	SSA	0	2.347171e-70	1.01756e-69	4.55
	SPSSA	0	7.690814e-318	0	8.08
F5	PSO	-9.476918e+03	-7.601357e+03	1069.6	1.91
	ABC	-1.453578e+63	-8.408056e+61	2.94628e+62	22.96
	SSA	-9.937985e+03	-8.648037e+03	656.408	5.62
	SPSSA	-1.256949e+04	-1.252106e+04	159.05	8.545

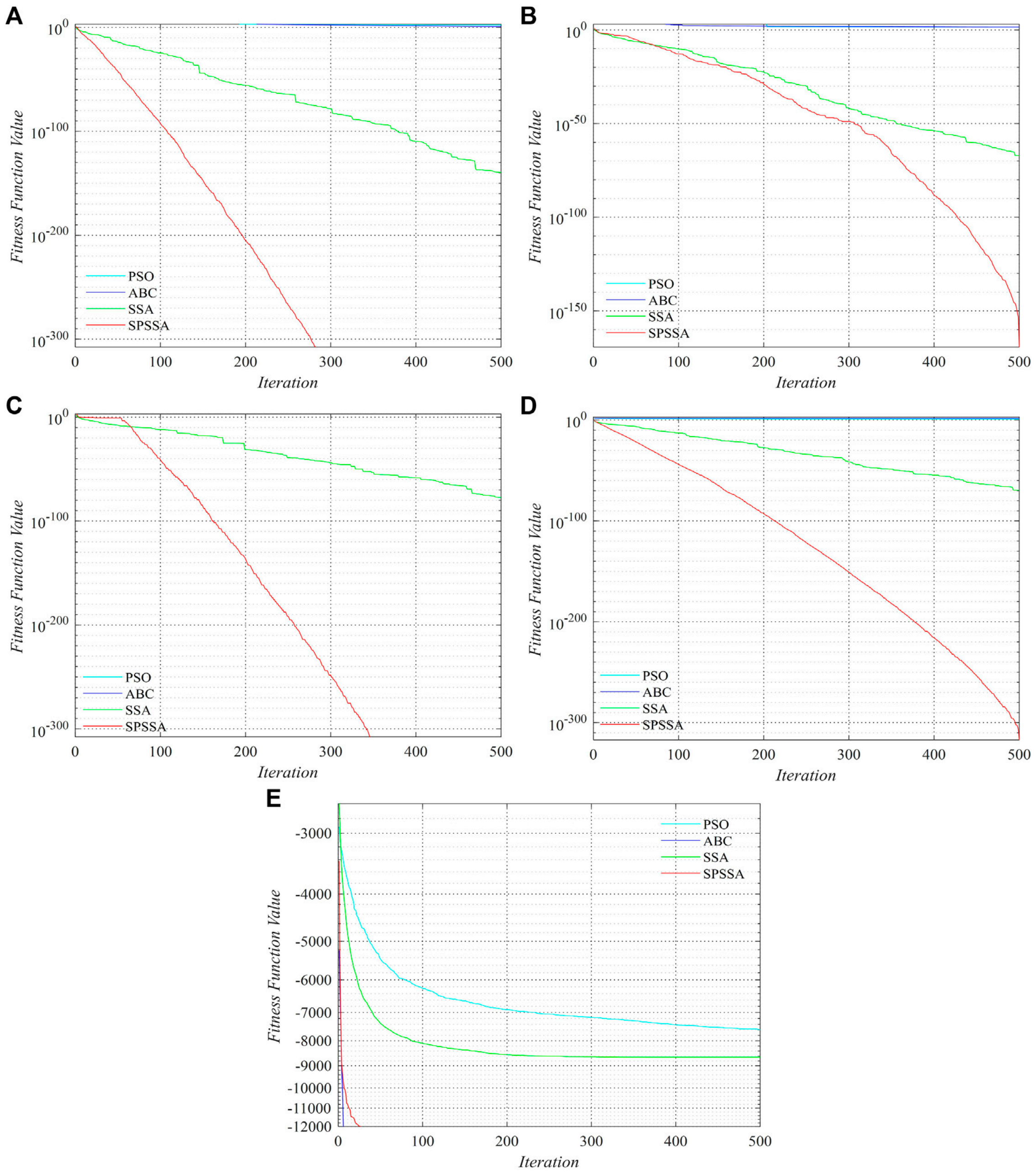


FIGURE 2 | Four algorithms' convergence curves on five benchmark functions. **(A)** F1; **(B)** F2; **(C)** F3; **(D)** F4; **(E)** F5.

2 METHODS

2.1 Parameter Identification of the Chaotic System

Consider the n-dimensional original chaotic system with m parameters

$$\dot{X} = F(X, X_0, \theta) \tag{1}$$

where, $X = (x_1, x_2, \cdots, x_n)^{\mathrm{T}} \in R^n$ is the state vector, $\theta = (\theta_1, \theta_2, \cdots, \theta_m)^{\mathrm{T}} \in R^m$ is system parameter vector, X_0 is the initial state of the system. F: $R^n \times R^m \rightarrow R^n$ is a given non-linear vector function.

Assuming that the system structure is known, the identified system can be defined as **Eq. 2**:

$$\dot{Y} = F\left(Y, X_0, \hat{\theta}\right) \tag{2}$$

where, $Y = (y_1, y_2, \cdots, y_n)^{\mathrm{T}} \in R^n$ is the state vector of the identified system. $\hat{\theta} = (\hat{\theta}_1, \hat{\theta}_2, \cdots, \hat{\theta}_m)^{\mathrm{T}} \in R^m$ is the identified system parameter.

The parameter identified problem can be transformed into a **Eq. 3**

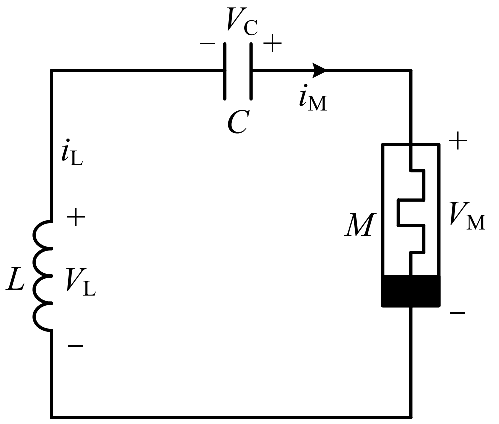


FIGURE 3 | The simplest memristor chaotic circuit.

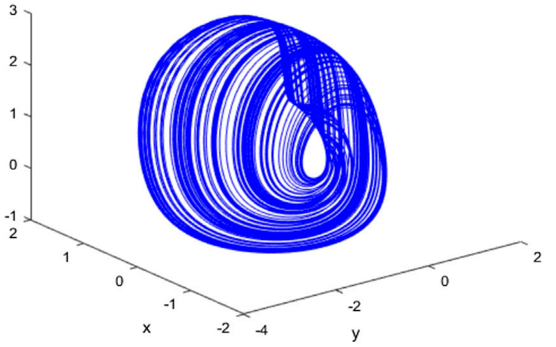


FIGURE 4 | 3D diagram of the memristive chaotic system **Eq. 11**

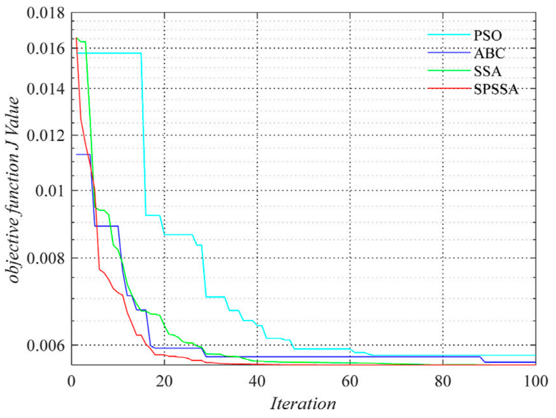


FIGURE 5 | The convergence curve of a memristive chaotic system.

$$\hat{\theta} = \arg\min_{\theta} J(\theta) = \arg\min_{\theta} \frac{1}{D_n}\sum_{i=1}^{D_n}\left\|x_i - y_i\right\|^2 \tag{3}$$

where, D_n represents the data length used for state variables, x_i and y_i represent the actual value and estimated value of the system under their state variables, respectively. Through the above analysis, the parameter identification problem of a chaotic system can be transformed into a multivariable optimization problem, and the relevant variables can be adjusted to minimize the target value *J*. The optimization principle is shown in **Figure 1**.

It is a multi-dimensional and multi-mode non-linear function, and the chaotic system is dynamically unstable and sensitive to initial parameters. Therefore, it is difficult to identify the parameters of a chaotic system effectively and accurately. The sparrow search algorithm is easy to realize in dealing with multi-dimensional problems, so this paper proposes a modified SSA and takes **Eq. 3** as the objective function.

2.2 The Sparrow Search Algorithm

When a group of sparrows is searching for food, we can divide them into three roles: producers, scroungers, and scouts. Producers, also called finders, are some sparrows searching for food. Scroungers are some sparrows following the producers' track to search for food. Scouts are sparrows watching for dangers. Producers usually account for 10%–20% of the population. Some sparrows will be selected randomly as scouts, which typically account for 10%–20% of the population.

In SSA, the behavior of sparrows searching for food can be simulated as the process of solving optimization problems. Consider that there exist *N* sparrows, the location of the *i*th sparrow is $X_i = [x_{i1}, \ldots, x_{ij}, \ldots, x_{iD}]$, where $i = 1, 2, \ldots, N$, *D* is *D*-dimensional search space.

The formula for the location update of producers is described as below:

$$x_{ij}^{t+1} = \begin{cases} x_{ij}^{t} \cdot \exp\left(\dfrac{-i}{\alpha \cdot T_{\max}}\right), R_2 < ST \\ x_{ij}^{t} + QL, R_2 \geqslant ST \end{cases} \tag{4}$$

where, t presents the current iteration, $j = 1,2, \ldots D$. $T_{\max}$ indicates the cycles number. $\alpha \in (0,1]$ and $R_2 \in [0,1]$ are all random numbers . $ST \in [0.5,1]$ is the alarm value. Q is a random number subject to standard normal distribution; L is a matrix of $1 \times D$. The initial values of all elements in the D are set to 1.

The location of scroungers is updated as follows:

$$X_{ij}^{t+1} = \begin{cases} Qexp\left(\dfrac{x_{\text{worst}}^{t} - x_{ij}^{t}}{i^2}\right) & \text{if } i > N/2 \\ X_P^{t+1} + \left|X_{i,j}^{t} - X_P^{t+1}\right| \cdot A^{+} \cdot L & \text{otherwise} \end{cases} \tag{5}$$

where, X_P^{t+1} is the producers' best location at iteration $t+1$, while x_{worst}^t denotes the current global worst location. A is a matrix of $1 \times D$. In A, all of the elements are initialized at random $A^+ = A^T (AA^T)^{-1}$.

In addition, the position of the scouts is updated by **(6)**

$$X_{i,j}^{t+1} = \begin{cases} X_{\text{best}}^t + \beta \cdot \left| X_{i,j}^t - X_{\text{best}}^t \right| & \text{if } f_i > f_g \\ X_{i,j}^t + K \cdot \left(\dfrac{\left| X_{i,j}^t - X_{\text{worst}}^{t+1} \right|}{(f_i - f_w) + \varepsilon} \right) & \text{if } f_i = f_g \end{cases} \tag{6}$$

where, X_{best}^t is the optimal place at iteration t. β is a parameter that controls the size of the steps. It is a random number with a normal distribution, with a mean of 0 and a variance of 1. The current sparrow fitness value, the best fitness value, and the worst fitness value are f_i, f_g and f_w, respectively. K is a random number that ranges from -1 to 1. ε is a small constant that prevents zero-division-error.

2.3 The Modified Sparrow Search Algorithm

2.3.1 Updating Scroungers' Locations

In the process of foraging, scroungers often forage around the best producer. During this process, the producers and the scroungers may switch roles with each other due to the competition for food. According to the optimization method proposed in [26], the Pareto distribution is introduced to improve scroungers' location so that we can avoid the algorithm falling into local optimization.

A random variable is said to obey the Pareto distribution if its cumulative distribution function has the following equation [26].

$$F(x_3) = \begin{cases} 1 - \left(\dfrac{k}{x_3} \right)^h, & x_3 \geq k \\ 0, & x_3 < k \end{cases} \tag{7}$$

where, $k > 0$ is the scale parameter, $h > 0$ is the shape parameter (Pareto's index of inequality).

To avoid the local minimum points, the Pareto distribution is utilized [25]. As a result, the scroungers' location update formula is changed by **Eq. 8**

$$x_{ij}^{t+1} = \begin{cases} Q \cdot \exp\left(\dfrac{x_{worst}^t - x_{ij}^t}{i^2} \right), & i > N/2 \\ x_{ij}^t + \alpha \otimes Pareto(k,h) \otimes \left(x_i^t - x_{best} \right), & \text{otherwise} \end{cases} \tag{8}$$

where, $Pareto(k,h)$ is a random number that follows the Pareto distribution. α is the step scale factor, and $\otimes$ is the point-to-point multiplication.

2.3.2 Updating Producers' Locations

In the **Eq. 4**, when $R_2 < ST$, as the number of iterations increases, the producers gradually approach the local best location. Scroungers will also pour into that location, and the diversity of population locations will be decreased. The algorithm may inevitably fall into local optimum. To solve this problem, the idea of the sine cosine algorithm (SCA) is integrated into the producers' location update method [27], and a learning factor is introduced. The learning factor has a bigger value in the early stage of the search, which is conducive to global exploration, and a smaller value in the later stage, which is conducive to improving local development ability and accuracy. The learning factor formula and the improved producers' location formula are described as follows:

$$\omega = \omega_{\max} - t \times (\omega_{\max}/T_{\max}) \tag{9}$$

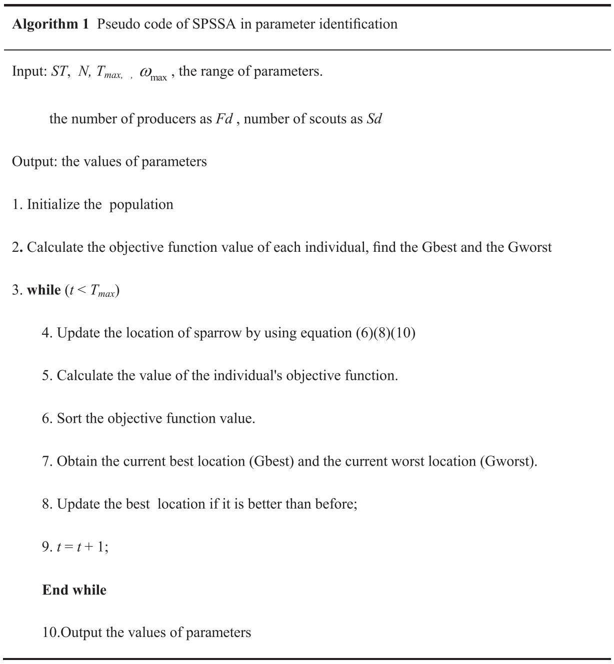
$$X_{i,j}^{t+1} = \begin{cases} (1-\omega) \cdot X_{i,j}^t + \omega \cdot \sin(r_1) \cdot \left| r_2 \cdot X_{\text{best}}^t - X_{i,j}^t \right|, R_2 < ST \\ (1-\omega) \cdot X_{i,j}^t + \omega \cdot \cos(r_1) \cdot \left| r_2 \cdot X_{\text{best}}^t - X_{i,j}^t \right|, R_2 \geqslant ST \end{cases} \tag{10}$$

where, $\omega_{\max}$ is a constant, t indicates the current iteration, r_1 is a random number in [0,2π], and r_2 is a random number in [0, 2].

Implementation of an improved sparrow search algorithm for parameter identification of a memristive chaotic system.

Algorithm 1 shows the pseudo code for determining a memristive system's parameters using the SPSSA.

Algorithm 1. Pseudo code of SPSSA in parameter identification.



```
Algorithm 1  Pseudo code of SPSSA in parameter identification
Input: $ST$, $N$, $T_{max}$, , $\omega_{\max}$ , the range of parameters.
    the number of producers as $Fd$ , number of scouts as $Sd$
Output: the values of parameters
1. Initialize the population
2. Calculate the objective function value of each individual, find the Gbest and the Gworst
3. while ($t < T_{max}$)
    4. Update the location of sparrow by using equation (6)(8)(10)
    5. Calculate the value of the individual's objective function.
    6. Sort the objective function value.
    7. Obtain the current best location (Gbest) and the current worst location (Gworst).
    8. Update the best location if it is better than before;
    9. $t = t + 1$;
    End while
    10.Output the values of parameters
```

3 EXPERIMENTS AND DISCUSSIONS

To check the performance of SPSSA, we carry out some simulation experiments. The software platform is Windows 10 and MATLAB 2021b. The hardware platform is a desktop PC which a CPU is 3.20 GHz, and a memory size is 16 GB.

3.1 Benchmark Function Comparison Experiment

To prove the performance of the proposed SPSSA, we select five benchmark functions, which have been widely used to test the effectiveness of the SI algorithm [23]. **Table 1** shows the name, range, and the minimum value of the benchmark test function.

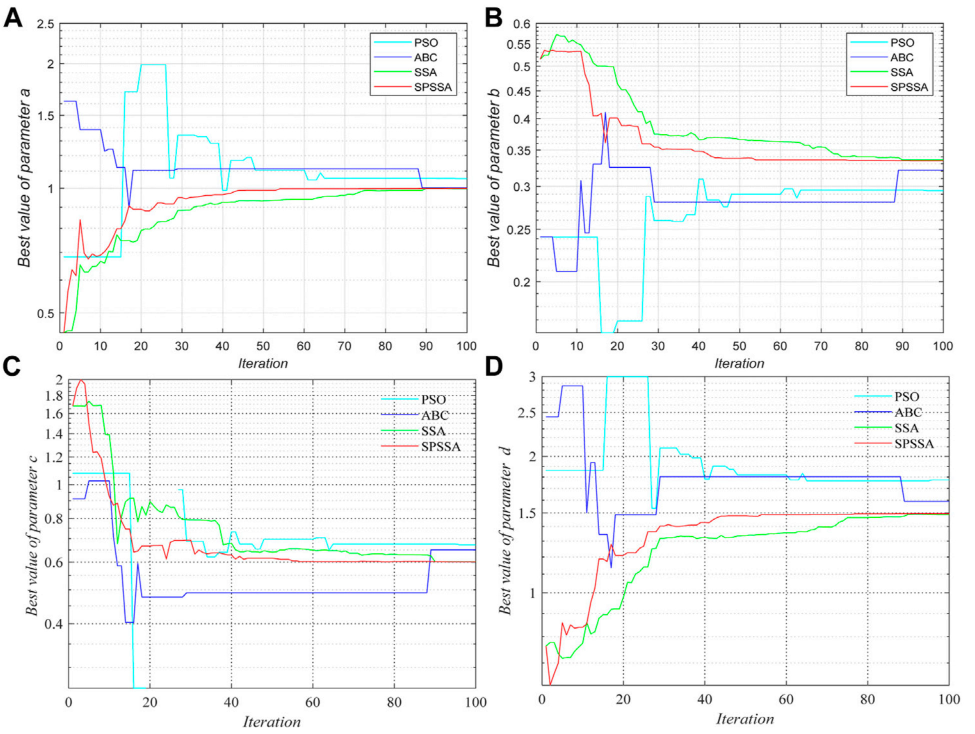


FIGURE 6 | Convergence and optimization curve of parameter identification for a memristive chaotic system. **(A)** Optimization curve of parameter a. **(B)** Optimization curve of parameter b. **(C)** Optimization curve of parameter c. **(D)** Optimization curve of parameter d.

TABLE 3 | Identified parameters by PSO,ABC,SSA, and BSA algorithms.

SI	*a*	*b*	*c*	*d*	Best *J* value
PSO	1.0551	0.2943	0.6722	1.7747	5.800800e-03
ABC	1.0016	0.3213	0.6507	1.5913	5.670232e-03
SSA	0.9960	0.3355	0.6023	1.4876	5.620233e-03
SPSSA	0.9982	0.3342	0.6014	1.4953	5.620033e-03

In our experiment, we input the population number $N = 50$, $T\max = 500$, $\omega_{\max} = 0.6$, $Fd = Sd = 0.2$. The dimension and the search range are set according to the **Table 1**. The five benchmark functions are used as fitness functions. Each benchmark test function is run 30 times using four different SI algorithms. **Table 2** contains the best value, average, standard deviation, and computation time. The best value reflects exploration ability, the average value demonstrates convergence accuracy, and the standard deviation depicts the SPSSA's stability under the same benchmark test function [25].

In terms of the best value, average value, and standard deviation, the optimization accuracy of SPSSA is superior than the other three algorithms, as shown in **Table 2**. Especially for benchmark functions F5, only SPSSA converges to the optimal value. Although SPSSA is not the fastest, its accuracy and stability are the best. It shows that the SPSSA has a strong global search ability and better equilibrium. To further prove the dynamic convergence performance, four swarm intelligence techniques are utilized to show the convergence curves of the five benchmark functions (in **Figure 2**).

From **Figure 2**, we can see that SPSSA proposed in this paper has a faster downward trend, steeper slope, and fewer iterations compared with the other three algorithms, obviously in **Figure 2A,C**, and **Figure 2E**. **Figures 2B,D** also show that SPSSA can get the best result. As a result, SPSSA outperforms PSO, ABC, and SSA in terms of convergence speed and optimization impact.This shows the feasibility and superiority of SPSSA.

3.2 Parameter Identification of a Memristive Chaotic System

Figure 3 shows the simplest memristor chaotic circuit model. A linear passive inductor, a linear passive capacitor, and a non-linear active charge driven memristor make up the circuit [28].

The dimensionless equation of that simplest memristor circuit system can be described by **Eq. 11**

$$\begin{cases} \dot{x} = ay \\ \dot{y} = -b\left(x + d\left(z^2 - 1\right)y\right) \\ \dot{z} = y - cz + yz \end{cases} \tag{11}$$

where, $\dot{x} = \frac{dx}{dt}, \dot{y} = \frac{dy}{dt}, \dot{z} = \frac{dz}{dt}$. a, b, c, d are system parameters. When the system parameters **Eq. 11** are set to $a = 1$, $b = 1/3$,

$c = 0.6, d = 1.5$ the system **Eq. 11** exhibits hyperchaotic behavior. When the initial state values are set to $x(0) = 0.1, y(0) = 0, z(0) = 0$, we can use fourth-order Runge Kutta method to solve the differential **Eq. 11**. The phase diagrams of the hyperchaotic attractor are shown in **Figure 4**, in which the step length h = 0.1 and the sampling times $Sn = 20000$.

In the following simulations, we set $a = 1$, $b = 1/3$, $c = 0.6$, $d = 1.5$. $x(0) = 0.1$, $y(0) = 0$, $z(0) = 0$ are initial value of the original system. The step length h is 0.01 and the sampling times $Sn = 500$. The parameters of the SPSSA are input as follows: $N = 20$, $T_{\max} = 100$. Other parameters are same as those in **Section 3.1**. The range of the parameters of the memristive chaotic system is set to $0 \le a \le 3, 0 \le b \le 2, 0 \le c \le 3, 0 \le d \le 2$.

Figure 5 displays the convergence curve of the chaotic system **Eq. 11**. **Figure 6** show the curve of parameters identification convergence. The final identified parameters of the PSO, ABC, SSA, and SPSSA are provided in **Table 3**.

As shown in **Figure 5**, it is obvious that the SPSSA has the fastest convergence speed and the smallest J value. Although SSA converges to the optimal value, its rate is significantly slower than SPSSA. This shows that among the four algorithms, SPSSA has the best performance. From **Figure 6** we can see among the four algorithms, SPSSA always converges to the optimal value fastest. It is clear to see from **Table 3** that the parameter values calculated by SPSSA are the closest to the actual value. All these show that the convergence and stability of SPSSA are better than the other three algorithms.

4 CONCLUSION

In this paper, a novel swarm intelligence optimization algorithm, the modified sparrow search algorithm, is used for the parameter estimation of a memristive chaotic system. The proposed algorithm, SPSSA, uses a sine cosine method to the sparrow finders' position to avoid falling into local optimization in the later search stage. Pareto distribution is used to adjust the current individual position, to improve the speed and the global optimization accuracy. Five standard test functions are used to verify the algorithm, and the results show that the SPSSA has high search accuracy. The simulation results show that the SPSSA can identify the parameters of the simplest memristive chaotic system more accurately, more rapidly, and more stable than the PSO, the ABC, and the SSA. This proves that SPSSA has good effectiveness and robustness. Other memristive chaotic systems can benefit from the SPSSA. If we know the equation of other systems, we can identify the system parameters by the methods mentioned in the article.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/Supplementary Materials, further inquiries can be directed to the corresponding author.

AUTHOR CONTRIBUTIONS

QX: system analysis, software design, and draft writing. JS: checking the whole analysis and manuscript revision. BT: numerical analysis. YX: checking the whole analysis.

FUNDING

This research was funded by the Natural Science Foundation of Hunan Province (No. 2021JJ50137), Doctoral research foundation program in Hunan University of arts and science (No.21BSQD30), the Scientific Research Project of Hunan University of arts and science (No. 21YB12).

REFERENCES

1. Strukov DB, Snider GS, Stewart DR, Williams RS. The Missing Memristor Found. *nature* (2008) 453(7191):80–3. doi:10.1038/nature06932
2. Haj-Ali A, Ben-Hur R, Wald N, Ronen R, Kvatinsky S. Not in Name Alone: A Memristive Memory Processing Unit for Real In-Memory Processing. *IEEE Micro* (2018) 38(5):13–21. doi:10.1109/mm.2018.053631137
3. Wang Y, Wang G, Shen Y, Iu HHC. A Memristor Neural Network Using Synaptic Plasticity and its Associative Memory. *Circuits Syst Signal Process* (2020) 39(7):3496–511. doi:10.1007/s00034-019-01330-8
4. Zhang Y, Cui M, Shen L, Zeng Z. Memristive Fuzzy Deep Learning Systems[J]. *IEEE Trans Fuzzy Syst* (2020) 29(8):2224–38.
5. Hu X, Feng G, Duan S, Liu L. A Memristive Multilayer Cellular Neural Network with Applications to Image Processing. *IEEE Trans Neural Netw Learn Syst* (2017) 28(8):1889–901. doi:10.1109/TNNLS.2016.2552640
6. Mannan ZI, Yang C, Kim H. Oscillation with 4-Lobe Chua Corsage Memristor. *IEEE Circuits Syst Mag* (2018) 18(2):14–27. doi:10.1109/mcas.2018.2821724
7. Bao H, Jiang T, Chu K, Chen M, Xu Q, Bao B. Memristor-based Canonical Chua's Circuit: Extreme Multistability in Voltage-Current Domain and its Controllability in Flux-Charge Domain[J]. *Complexity* (2018) 2018. doi:10.1155/2018/5935637
8. Guo M, Yang W, Xue Y, Gao Z, Yuan F, Dou G, et al. Multistability in a Physical Memristor-Based Modified Chua's Circuit. *Chaos* (2019) 29(4):043114. doi:10.1063/1.5089293
9. Huang L, Yao W, Xiang J, Zhang Z. Heterogeneous and Homogenous Multistabilities in a Novel 4D Memristor-Based Chaotic System with Discrete Bifurcation Diagrams[J]. *Complexity* (2020) 2020:2408460. doi:10.1155/2020/2408460
10. Ma X, Mou J, Liu J, Ma C, Yang F, Zhao X. A Novel Simple Chaotic Circuit Based on Memristor-Memcapacitor. *Nonlinear Dyn* (2020) 100(3):2859–76. doi:10.1007/s11071-020-05601-x
11. Chen M, Wang Z, Nazarimehr F, Jafari S. A Novel Memristive Chaotic System without Any Equilibrium point. *Integration* (2021) 79:133–42. doi:10.1016/j.vlsi.2021.04.002
12. Chen M, Wang Z, Nazarimehr F, Jafari S. A Novel Memristive Chaotic System without Any Equilibrium point. *Integration* (2021) 79:133–42. doi:10.1016/j.vlsi.2021.04.002
13. Ostrovskii V, Fedoseev P, Bobrova Y, Butusov D. Structural and Parametric Identification of Knowm Memristors. *Nanomaterials* (2021) 12(1):63. doi:10.3390/nano12010063
14. Lazzús JA, Vega-Jorquera P, López-Caraballo CH, Palma-Chilla L, Salfate I. Parameter Estimation of a Generalized Lotka-Volterra System Using a Modified PSO Algorithm. *Appl Soft Comput* (2020) 96:106606. doi:10.1016/j.asoc.2020.106606

15. Peng Y, Sun K, He S, Yang X. Parameter Estimation of a Complex Chaotic System with Unknown Initial Values[J]. *The Eur Phys J Plus* (2018) 133(8): 1–13. doi:10.1140/epjp/i2018-12091-1
16. Du W, Miao Q, Tong L, Tang Y. Identification of Fractional-Order Systems with Unknown Initial Values and Structure. *Phys Lett A* (2017) 381(23): 1943–9. doi:10.1016/j.physleta.2017.03.048
17. Gu W, Yu Y, Hu W. Parameter Estimation of Unknown Fractional-Order Memristor-Based Chaotic Systems by a Hybrid Artificial Bee colony Algorithm Combined with Differential Evolution[J]. *Nonlinear Dyn* (2016) 84(2):779–95.
18. Hu W, Yu Y, Zhang S. A Hybrid Artificial Bee colony Algorithm for Parameter Identification of Uncertain Fractional-Order Chaotic Systems[J]. *Nonlinear Dyn* (2015) 82(3):1441–56.
19. Zhang P, Yang R, Yang R, Ren G, Yang X, Xu C, et al. Parameter Estimation for Fractional-Order Chaotic Systems by Improved Bird Swarm Optimization Algorithm. *Int J Mod Phys C* (2019) 30(11):1950086. doi:10.1142/s0129183119500864
20. Chen F, Ding Z, Lu Z, Zeng X. Parameters Identification for Chaotic Systems Based on a Modified Jaya Algorithm. *Nonlinear Dyn* (2018) 94(4):2307–26. doi:10.1007/s11071-018-4490-9
21. Peng Y, Sun K, He S, Peng D. Parameter Identification of Fractional-Order Discrete Chaotic Systems. *Entropy* (2019) 21(1):27. doi:10.3390/e21010027
22. Peng YX, Sun KH, He SB. Dynamics Analysis of Chaotic Maps: From Perspective on Parameter Estimation by Meta-Heuristic Algorithm*. *Chin Phys. B* (2020) 29(3):030502. doi:10.1088/1674-1056/ab695c
23. Peng Y, He S, Sun K. Parameter Identification for Discrete Memristive Chaotic Map Using Adaptive Differential Evolution Algorithm. *Nonlinear Dyn* (2022) 107(1):1263–75. doi:10.1007/s11071-021-06993-0
24. Xue J, Shen B. A Novel Swarm Intelligence Optimization Approach: Sparrow Search Algorithm. *Syst Sci Control Eng* (2020) 8:22–34. doi:10.1080/21642583.2019.1708830
25. Xiong Q, Zhang X, He S, Shen J. A Fractional-Order Chaotic Sparrow Search Algorithm for Enhancement of Long Distance Iris Image. *Mathematics* (2021) 9(21):2790. doi:10.3390/math9212790
26. Wei J, Chen Y, Yu Y, Chen Y. Optimal Randomness in Swarm-Based Search. *Mathematics* (2019) 7:828. doi:10.3390/math7090828
27. Mirjalili S. SCA: A Sine Cosine Algorithm for Solving Optimization Problems. *Knowledge-based Syst* (2016) 96:120–33. doi:10.1016/j.knosys.2015.12.022
28. Muthuswamy B. Implementing Memristor Based Chaotic Circuits. *Int J Bifurcation Chaos* (2010) 20(05):1335–50. doi:10.1142/s0218127410026514

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 10 August 2022
DOI 10.3389/fphy.2022.963795

OPEN ACCESS

EDITED BY
Shaobo He,
Central South University, China

REVIEWED BY
Hongjun Liu,
University of Jinan, China
Yuexi Peng,
Xiangtan University, China

*CORRESPONDENCE
Wei Feng,
fengw1981@126.com

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 07 June 2022
ACCEPTED 11 July 2022
PUBLISHED 10 August 2022

CITATION
Qian K, Feng W, Qin Z, Zhang J, Luo X and Zhu Z (2022), A novel image encryption scheme based on memristive chaotic system and combining bidirectional bit-level cyclic shift and dynamic DNA-level diffusion. *Front. Phys.* 10:963795.
doi: 10.3389/fphy.2022.963795

A novel image encryption scheme based on memristive chaotic system and combining bidirectional bit-level cyclic shift and dynamic DNA-level diffusion

Kun Qian[1,2], Wei Feng[3]*, Zhentao Qin[3], Jing Zhang[3], Xuegang Luo[3] and Zhengguo Zhu[3]

[1]Key Laboratory of Hunan Province on Information Photonics and Freespace Optical Communications, Hunan Institute of Science and Technology, Yueyang, China, [2]College of Physics and Electronics, Hunan Institute of Science and Technology, Yueyang, China, [3]School of Mathematics and Computer Science, Panzhihua University, Panzhihua, China

In recent years, many researchers have leveraged various memristors to design many novel memristive chaotic systems with complex dynamics. Compared with other chaotic systems, applying these memristive chaotic systems to image encryption is expected to solve some key problems in this field. Therefore, exploiting a recently reported memristive chaotic system, this paper proposes a novel image encryption scheme based on the memristive chaotic system and combining bidirectional bit-level cyclic shift and dynamic DNA-level diffusion (IES-M-BD). First, a discrete memristive chaotic map is employed to generate chaotic sequences. Then, the plaintext image is shifted circularly on bit-level according to chaotic sequences and the hash value of the plaintext image. After that, the shifted matrix is recombined on the bit plane and encoded dynamically by DNA encoding rules. Next, dynamic DNA-level diffusion and DNA-level permutation are carried out in two rounds. Finally, the encrypted image is obtained after dynamic DNA decoding. Simulation tests and performance analyses are also carried out in this paper. The simulation results and the security analyses demonstrate that this encryption scheme has a high security level and can resist various attacks.

KEYWORDS

image encryption, memristor, memristive chaotic system, cyclic shift, DNA sequence operation, permutation, diffusion

1 Introduction

In 1971, Professor Chua first hypothesized the memristor concept [1]. A memristor is a passive two-terminal electronic element that describes the relationship between charge and flux. Due to the nonlinearity of the memristor, it is easy to cause the chaotic phenomenon in the circuit and enhance the complexity of the chaotic system [2]. It was not until 2008 that HP Labs reported the first physical implementation of a memristor [3].

The invention of the memristor has further stimulated researchers' interest in the application of chaotic systems. Numerous memristive chaotic systems have so far been introduced [4–6]. The most notable distinction between memristive chaotic systems and conventional dynamical systems is that the former's long-term dynamical behavior heavily depends on the memristor's initial state, which results in the dynamic phenomenon of coexisting attractors [7–9]. As a result, this feature will offer sufficient pattern selectivity in encrypted secret communication.

As one of the important carriers of information, images are widely transmitted and stored over the Internet. Thus, the privacy and security of image information are receiving more and more attention [10–16]. Different from text, an image has bulk data capacity, and the adjacent pixels have a strong correlation. When an image is encrypted with the traditional encryption methods, the encryption performance is not ideal. In recent years, various encryption schemes for image information have been proposed, and the image encryption scheme based on chaos has been widely discussed due to the pseudo-randomness, highly sensitive to initial conditions, and ergodicity of chaotic systems [17–24]. A chaos-based image cryptosystem generally adopts the structure of permutation and diffusion. Recently, bit-level permutation has gradually replaced pixel-level permutation [25–29], owing to the latter disturbs the pixel position of an image, but does not change the statistical characteristics. Zhu et al. [27] proposed chaos-based image encryption using a bit-level permutation. In 2017, Li et al. [28] proposed a hyper-chaos-based image encryption algorithm, the pixel-level permutation and bit-level permutation are utilized to strengthen the security of the cryptosystem. In 2018, Teng et al. [29] employed bit-level permutation in a chaotic color image encryption. The bit-level permutation can not only change the position of the pixels but also change their value. Therefore, image encryption algorithms that adopt bit-level permutations may possess a high security level. Meanwhile, image encryption schemes based on DNA complementary rules have been proposed continuously [30–34]. Zhang et al. [30] proposed an image fusion encryption algorithm based on DNA sequence operations and combined hyper-chaotic maps. In [31], Chai et al. presented an image encryption algorithm using DNA sequence operations and a 2D Logistic chaotic system. Moreover, a new color image encryption scheme based on a four-wing hyperchaotic system and dynamic DNA encoding was proposed in [33]. In these studies, image encryption schemes based on DNA technology showed high security, and DNA technology possesses other advantages such as working in parallel and having ultra-low power consumption.

While the research on image encryption continues to advance, there are also many researchers engaged in cryptanalysis work that is connected to image encryption [26, 35–42]. After their analysis and evaluation, some image encryption schemes are confirmed to be not as secure as they claim and have some rationality and practicality problems. Specifically, the representative problems with the current image encryption schemes are as follows:

- 1. The employed chaotic systems have distinguishing characteristics, such as limited chaotic range and uneven trajectory distribution.
- 2. Some schemes' secret key designs are irrational, necessitating the replacement of the secret key each time a new image is encrypted. When there are a lot of images to be encrypted, such designs are impractical.
- 3. The hash value of the plaintext image is utilized inappropriately. When encrypting different images, such designs necessitate the generation of chaotic sequences on a constant basis, thus resulting in low encryption efficiency.
- 4. The entire encryption process of some schemes is independent of the plaintext image, making them difficult to effectively resist plaintext attacks.

In this paper, to address the problems of existing image encryption schemes, a novel image encryption scheme based on a memristive chaotic system and combining bidirectional bit-level cyclic shift and dynamic DNA-level diffusion (IES-M-BD) is proposed. The following is a summary of the novelties and contributions of our proposed scheme.

- 1. A memristive chaotic system with excellent chaotic performance and uniform trajectory distribution is adopted to generate chaotic sequences.
- 2. Our proposed IES-M-BD is designed to exploit a more reasonable binary sequence as the secret key, and does not need to change the secret key when encrypting different images.
- 3. A bidirectional bit-level cyclic shift operation is designed, which can realize permutation and confusion at the same time.
- 4. In IES-M-BD, the hash value of the plaintext image is utilized to affect the cyclic shift and DNA-level diffusion operations. Thus, the plaintext sensitivity of IES-M-BD is ensured.
- 5. A novel DNA sequence operation design, including dynamic column-level DNA encoding, dynamic DNA-level diffusion, DNA-level permutation, and dynamic column-level DNA decoding, not only improves the encryption efficiency, but also ensures the security of IES-M-BD.

The rest of this paper is organized as follows. In Section 2, the preliminaries are introduced. In Section 3, we describe IES-M-BD in detail. In Section 4, the simulation tests and security analyses are presented. Finally, the conclusions are drawn in Section 5.

2 Preliminaries

In this section, we briefly introduce the adopted discrete memristive chaotic system, the SHA-256 hash value of the plaintext image, and the DNA Sequence operation.

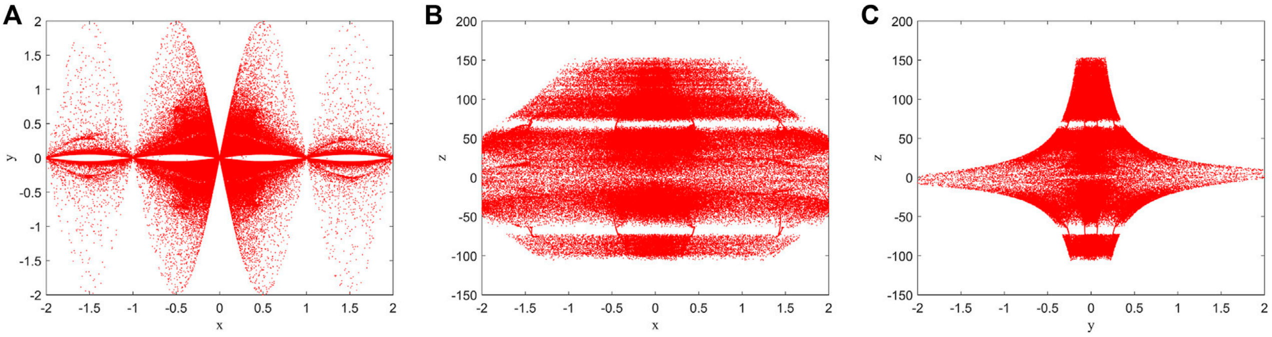


FIGURE 1
The phase diagrams with parameters *a* = 2, *b* = 3 and *c* = 3: **(A)** *x*-*y* plane; **(B)** *x*-*z* plane; **(C)** *y*-*z* plane.

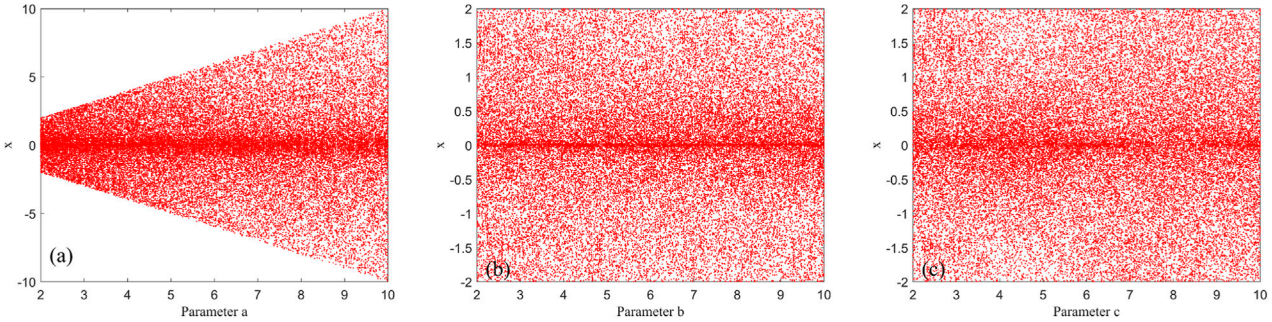


FIGURE 2
Bifurcation diagrams: **(A)** 2 < *a* < 10; **(B)** 2 < *b* < 10; **(C)** 2 < *c* < 10.

2.1 Memristive chaotic system

A 3D chaotic map with discrete memristor is adopted to generate pseudo random sequences for bit-level cyclic shift and dynamic DNA-level diffusion. Based on the discrete HP memristor model, this chaotic map is derived from the sine map and an iterative chaotic map with infinite collapse [5], which is defined as:

$$\begin{cases} x_{n+1} = \alpha \sin(\omega y_n) \sin\left(\dfrac{\gamma}{x_n}\right), \\ y_{n+1} = \alpha \sin(\omega x_{n+1}) \sin\left[\dfrac{\gamma}{y_n(\alpha - \beta \times z_n)}\right], \\ z_{n+1} = z_n + x_{n+1}, \end{cases} \tag{1}$$

where x, y, and z represent the system states variables, α, γ, ω are amplitude, internal perturbation, and angular frequency, β is a system parameter evolved from HP memristor model. Here ω is set to π. Set $\alpha = 2$, $\beta = 3$, $\gamma = 3$, and the initial state values $(x_0, y_0, z_0) = (0.3, 0.5, 1)$, one can get the attractors presented in Figure 1. Addtionally, the bifurcation diagrams and Lyapunov exponential spectrums are shown in Figure 2 and Figure 3, respectively.

When $a \in [2, 10], b \in [2, 10], c \in [2, 10]$, this system is in chaotic state. There is at least one Lyapunov exponent always positive in Figure 3, which can verify that the system is chaotic. Figure 2 shows the bifurcation diagrams of this system, where the ranges are $(a, b, c) \in [2, 10]$ with an increment step of $\Delta a = 0.1$, $\Delta b = 0.1$, $\Delta c = 0.1$. The bifurcation diagrams demonstrate that the chaotic system has complex dynamic characteristics, and the chaotic trajectories are widely distributed and relatively uniform. These characteristics make this discrete memristive chaotic system very suitable for image encryption.

2.2 SHA-256 hash value

SHA-256 is one of the most widely used hash algorithms in the world, it can convert image data into a 256-bit hash value. The hash value will change dramatically if one make any minor change to the image. SHA-256 is often used in encryption

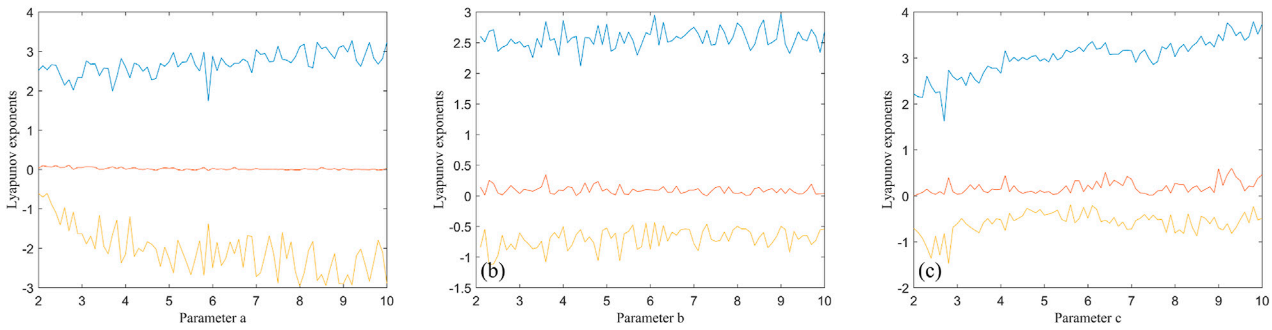
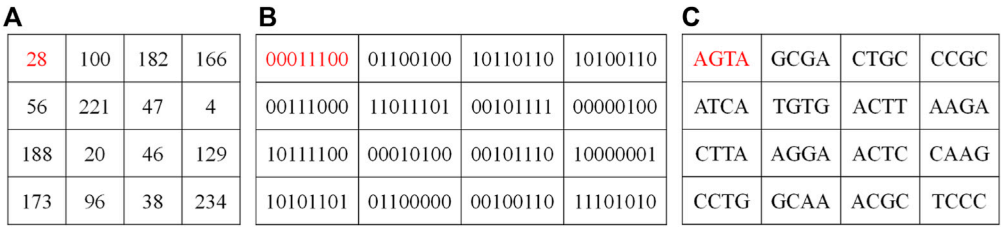


FIGURE 3
Lyapunov exponent spectrums: **(A)** 2 < *a* < 10; **(B)** 2 < *b* < 10; **(C)** 2 < *c* < 10.



A

28	100	182	166
56	221	47	4
188	20	46	129
173	96	38	234

B

00011100	01100100	10110110	10100110
00111000	11011101	00101111	00000100
10111100	00010100	00101110	10000001
10101101	01100000	00100110	11101010

C

AGTA	GCGA	CTGC	CCGC
ATCA	TGTG	ACTT	AAGA
CTTA	AGGA	ACTC	CAAG
CCTG	GCAA	ACGC	TCCC

FIGURE 4
An example of DNA encoding: **(A)** Pixel matrix; **(B)** Binary matrix; **(C)** DNA base matrix.

schemes to enhance their sensitivity to the plaintext image. In IES-M-BD, the hash value of the plaintext image is adopted to affect the cyclic shift and DNA-level diffusion operations. Thus, the plaintext sensitivity of IES-M-BD is enhanced. Even if the plaintext image only changes by one pixel bit, the encrypted image will be completely different. In this paper, we denote the hash value of the plaintext image as $H^{(1)} = b_1 b_2 \dots b_{32}$, and let

$$\begin{cases} H^{(2)} = (b_1 + b_2 + \cdots + b_{32}) \bmod 256, \\ H^{(3)} = b_1 \oplus b_2 \oplus \dots \oplus b_{32}, \end{cases} \tag{2}$$

where b_i $(i = 1, 2, \dots, 32)$ represents the *i*th byte consisting of eight consecutive bits in $H^{(1)}$, and $\oplus$ represents the bitwise XOR operation.

2.3 DNA sequence operation

Each DNA sequence has four types of nucleic acid bases, which are Adenine (A), Thymine (T), Cytosine (C), and Guanine (G). These four nucleic acid bases follow the principle of complementary, where A and T, C and G are complementary pairs. Each nucleic acid base is encoded by a 2-bit binary code. So, there are types of encoding rules, but only eight of them can satisfy the complementary rule, which are shown in Table 1.

TABLE 1 Binary coding rules for DNA sequences.

Rule	A	T	C	G
Rule 1	00 (0)	11 (3)	10 (2)	01 (1)
Rule 2	00 (0)	11 (3)	01 (1)	10 (2)
Rule 3	11 (3)	00 (0)	10 (2)	01 (1)
Rule 4	11 (3)	00 (0)	01 (1)	10 (2)
Rule 5	10 (2)	01 (1)	00 (0)	11 (3)
Rule 6	10 (2)	01 (1)	11 (3)	00 (0)
Rule 7	01 (1)	10 (2)	00 (0)	11 (3)
Rule 8	01 (1)	10 (2)	11 (3)	00 (0)

For an 8-bit gray image, each pixel can be encoded as a DNA sequence, whose length is 4. For example, a pixel value of 28 in a gray image, as shown in Figure 4, expressed as a binary number of 00011100, which can be encoded to a DNA sequence AGTA by DNA encoding Rule 1. Moreover, the addition, subtraction, and XOR operations of DNA sequences are similar to traditional binary addition, subtraction, and XOR operations. Corresponding to the eight types of DNA encoding rules, there are eight types of addition, subtraction, and XOR operations for DNA sequences.

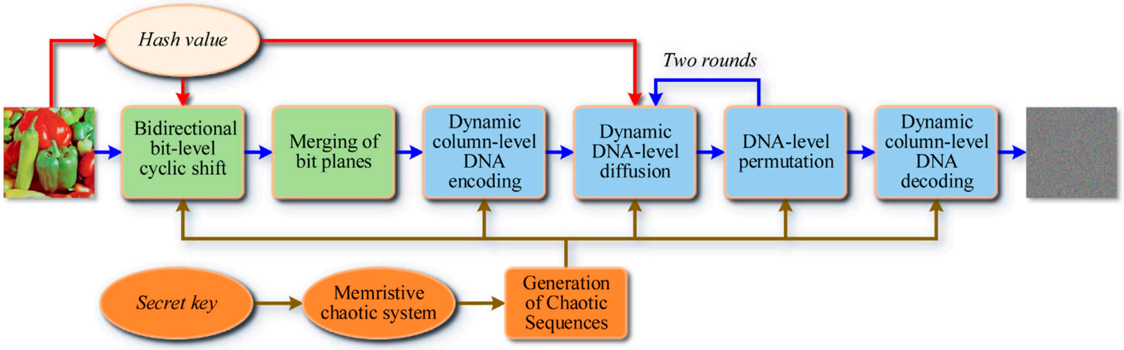


FIGURE 5
Encryption flow chart of IES-M-BD.

3 Proposed encryption scheme

As a symmetrical image encryption scheme, IES-M-BD mainly consists of seven encryption steps. These encryption steps are Generation of Chaotic Sequences, Bidirectional Bit-level Cyclic Shift, Merging of Bit Planes, Dynamic Column-level DNA Encoding, Dynamic DNA-level Diffusion, DNA-level Permutation, and Dynamic Column-level DNA Decoding, respectively, as shown in Figure 5. In this section, we will describe each encryption step one by one.

3.1 Generation of chaotic sequences

In this section, we will describe in detail the process of generating chaotic sequences using the secret key and memristive chaotic system. The process mainly consists of three steps as shown below. In this paper, unless otherwise specified, we all assume that the size of the plaintext image that needs to be encrypted is $M \times N$.

- **Step 1**: Convert the secret key K into the control parameters (α, β, γ) and initial state values (x_0, y_0, z_0) of the memristive chaotic system. In IES-M-BD, K is a binary sequence with a length of 312 bits, that is, $K = a_1a_2 \ldots a_{312}$. The specific conversion method is as follows.

$$\begin{cases} \alpha = 2 + \left(a_1a_2 \ldots a_{52} \times 2^{-52}\right), \\ \beta = 2 + \left(a_{53}a_{54} \ldots a_{104} \times 2^{-52}\right), \\ \gamma = 2 + \left(a_{105}a_{106} \ldots a_{156} \times 2^{-52}\right), \\ x_0 = a_{157}a_{158} \ldots a_{208} \times 2^{-52}, \\ y_0 = a_{209}a_{210} \ldots a_{260} \times 2^{-52}, \\ z_0 = a_{261}a_{262} \ldots a_{312} \times 2^{-52}. \end{cases} \tag{3}$$

- **Step 2**: Iterate the memristive chaotic system $M \times N + H^{(2)}$ times with the chaotic system parameters obtained in the previous step. Discard the system state values obtained in the previous $H^{(2)}$ iterations, and save the remaining system state values as chaotic sequences $X = \{x_1, x_2, \ldots x_{M\times N}\}$, $Y = \{y_1, y_2, \ldots y_{M\times N}\}$, and $Z = \{z_1, z_2, \ldots z_{M\times N}\}$.
- **Step 3**: Further process X, Y, Z to obtain six chaotic sequences $S^{(1)}$, $S^{(2)}$, $S^{(3)}$, $S^{(4)}$, $S^{(5)}$, $S^{(6)}$that needs to be used in the subsequent encryption steps. The specific processing method is as follows.

$$S_i^{(1)} = \left(\lfloor |x_i| \times 10^{15} \rfloor \bmod (N \times 8)\right) + 1, \tag{4}$$

where $i = H^{(2)} + 1, H^{(2)} + 2, \ldots, H^{(2)} + M$, $\lfloor \bullet \rfloor$ returns the integer part of an operand, $|\bullet|$ returns the absolute value of an operand.

$$S_i^{(2)} = \left(\left\lfloor |y_i| \times 10^{15} \right\rfloor \bmod M\right) + 1, \tag{5}$$

where $i = H^{(3)} + 1, H^{(3)} + 2, \ldots, H^{(3)} + N \times 8$.

$$S_i^{(3)} = \lfloor |x_i| \times 10^{15} \rfloor \bmod 256, \tag{6}$$

where $i = 1, 2, \ldots, M \times N$.

$$S_i^{(4)} = \left\lfloor |y_i| \times 10^{15} \right\rfloor \bmod 256, \tag{7}$$

where $i = 1, 2, \ldots, M \times N$.

$$S_i^{(5)} = \lfloor |z_i| \times 10^{15} \rfloor \bmod 256, \tag{8}$$

where $i = 1, 2, \ldots, M \times N$.

$$S^{(6)} = \{X + Y, Y + Z, Y + Z, X + Y + Z\}. \tag{9}$$

3.2 Bidirectional bit-level cyclic shift

For a gray image, the brightness of each pixel is an integer number ranging from 0 to 255, where 0 means completely black. Thus, the brightness could be transformed into an 8-bit binary value. So, an image of size $M \times N$ can be converted to a $M \times (N \times 8)$ binary matrix. In IES-M-BD, we adopt the bit-level permutation to scramble

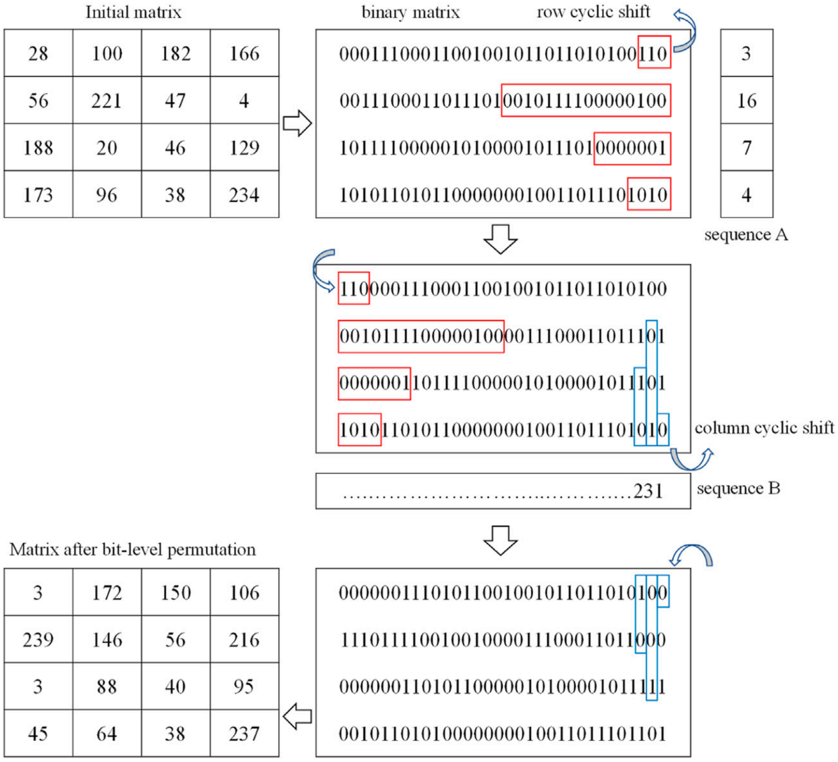


FIGURE 6
BBCS of IES-M-D.

the positions of these binary values. The bit-level permutation uses row cyclic shift and column cyclic shift, so we call it Bidirectional Bit-level Cyclic Shift (BBCS). For example, as shown in Figure 6, an initial matrix of size 4 × 4 was transformed into a 4 × 32 binary matrix, then each row of the binary matrix was operated by row cyclic shift in turn. The value of row cyclic shift was obtained by the chaotic sequence $S^{(1)}$ and the hash value $H^{(1)}$. The next step was to perform a similar column cyclic shift. Finally, a matrix after BBCS was obtained. Specifically, the BBCS of IES-M-BD can be subdivided into the following steps.

- **Step 1**: Convert the plaintext image P into the binary matrix B of size $M \times (N \times 8)$.
- **Step 2**: Initialize a column vector μ of length M, and let

$$\mu(1) = \left(\left\lfloor S^{(1)}(1) + \frac{\operatorname{mean}(S^{(1)}) - \min(S^{(1)})}{\max(S^{(1)}) - \min(S^{(1)})} \times 10^{15} \right\rfloor \bmod (N \times 8 - 1) \right) + 1, \tag{10}$$

where mean (•) returns the average value of the elements in an operand, min (•) returns the minimum value of the elements in an operand, and max (•) returns the maximum value of the elements in an operand. Then, use $\mu(1)$ to perform a cyclic shift operation on the first row of B.

- **Step 3**: Perform a cyclic shift operation on the remaining rows of B row by row. That is, for the ith row of B, let

$$\mu(i) = \left(\left(\mu(i-1) + S^{(1)}(i) + H^{(1)}\left(t^{(r)}\right) \right) \bmod (N \times 8 - 1) \right) + 1, \tag{11}$$

where $i = 2, 3, \ldots, M$, and $t^{(r)} = (S^{(1)}(i) \bmod 32) + 1$. Then, use $\mu(i)$ to perform a cyclic shift operation on the ith row of B.

- **Step 4**: Initialize a row vector v of length $N \times 8$, and let

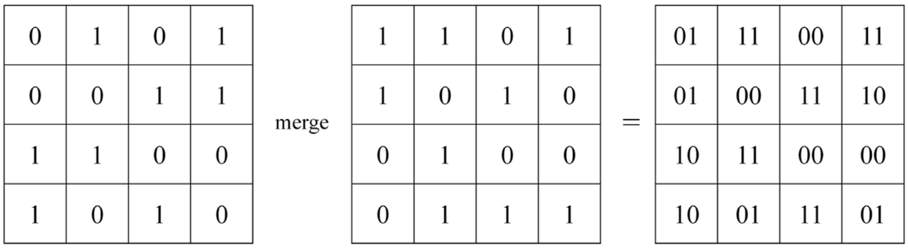
$$v(1) = \left(\left\lfloor S^{(2)}(1) + \frac{\operatorname{mean}(S^{(2)}) - \min(S^{(2)})}{\max(S^{(2)}) - \min(S^{(2)})} \times 10^{15} \right\rfloor \bmod (M - 1) \right) + 1. \tag{12}$$

Then, use $v(1)$ to perform a cyclic shift operation on the first column of B.

- **Step 5**: Perform a cyclic shift operation on the remaining columns of B column by column. That is, for the ith column of B, let

$$v(i) = \left(\left(v(i-1) + S^{(2)}(i) + H^{(1)}\left(t^{(c)}\right) \right) \bmod (M - 1) \right) + 1, \tag{13}$$

where $i = 2, 3, \ldots, N \times 8$, and $t^{(c)} = (S^{(2)}(i) \bmod 32) + 1$. Then, use $v(i)$ to perform a cyclic shift operation on the ith column of B.



$$\begin{array}{|c|c|c|c|}\hline 0&1&0&1\\\hline 0&0&1&1\\\hline 1&1&0&0\\\hline 1&0&1&0\\\hline\end{array}\ \text{merge}\ \begin{array}{|c|c|c|c|}\hline 1&1&0&1\\\hline 1&0&1&0\\\hline 0&1&0&0\\\hline 0&1&1&1\\\hline\end{array} = \begin{array}{|c|c|c|c|}\hline 01&11&00&11\\\hline 01&00&11&10\\\hline 10&11&00&00\\\hline 10&01&11&01\\\hline\end{array}$$

FIGURE 7
An example of merging bit planes.

As can be seen from Figure 6, BBCS not only disorganizes the position of pixels in one image, but also changes the value of each pixel. As a result, this bit-level permutation has the function of both permutation and confusion.

3.3 Merging of bit planes

As mentioned in Section 3.2, after the processing of BBCS, the plaintext image P will become the binary matrix B of size $M \times (N \times 8)$. Logically, B consists of 8 bit planes $B^{(1)}$, $B^{(2)}$, $B^{(3)}$, $B^{(4)}$, $B^{(5)}$, $B^{(6)}$, $B^{(7)}$, *and* $B^{(8)}$. Next, merge the 8 bit planes in pairs to obtain the four quaternary matrices $Q^{(1)}$, $Q^{(2)}$, $Q^{(3)}$, *and* $Q^{(4)}$, as shown in Figure 7.

Specifically, $Q^{(1)}$ is obtained by merging $B^{(1)}$ and $B^{(2)}$, $Q^{(2)}$ is obtained by merging $B^{(3)}$ and $B^{(4)}$, $Q^{(3)}$ is obtained by merging $B^{(5)}$ and $B^{(6)}$, and $Q^{(4)}$ is obtained by merging $B^{(7)}$ and $B^{(8)}$. In this way, after the above processing, a quaternary matrix Q with the size of $M \times (N \times 4)$ can actually be obtained.

3.4 Dynamic column-level DNA encoding

Compared with the DNA encoding method in which the encoding rule remains unchanged, the dynamic DNA encoding that constantly changes the encoding rule during the encoding process can achieve better encryption effects [32, 34]. In addition, different from the previous element-by-element encoding method, in order to improve the encryption efficiency, IES-M-BD adopts the column-level dynamic encoding method. Specifically, this encryption step can be further subdivided into the following steps.

- **Step 1**: Reshape the chaotic sequences $S^{(3)}$ and $S^{(4)}$ obtained in Section 3.1 into the matrices of size $M \times N$. Then, the DNA encoding rules are generated in groups of four columns, as shown below.

$$\begin{cases} R^{(E)}(:,(j-1)\times 4+1) = \left(S^{(3)}(:,j) \bmod 8\right)+1, \\ R^{(E)}(:,(j-1)\times 4+2) = \left(S^{(4)}(:,j) \bmod 8\right)+1, \\ R^{(E)}(:,(j-1)\times 4+3) = \left(\left(S^{(3)}(:,j)-S^{(4)}(:,j)\right) \bmod 8\right)+1, \\ R^{(E)}(:,(j-1)\times 4+4) = \left(\left(S^{(3)}(:,j)+S^{(4)}(:,j)\right) \bmod 8\right)+1, \end{cases} \tag{14}$$

where $R^{(E)}$ is the obtained DNA encoding rule matrix whose size is $M \times (N \times 4)$, and $j = 1, 2, \ldots, N$.

- **Step 2**: According to $R^{(E)}$ obtained in the previous step, the dynamic DNA encoding of Q is performed in column units, so as to get the DNA base matrix $D^{(C)}$ of size $M \times (N \times 4)$.
- **Step 3**: Reshape the chaotic sequence $S^{(5)}$ obtained in Section 3.1 into the matrix of size $M \times N$, and then utilize $R^{(E)}$ to encode it into the chaotic DNA base matrix $D^{(S)}$ of size $M \times (N \times 4)$ in the same way as in **Step 2.**
- **Step 4**: Using the Nth column of the DNA base matrix $D^{(C)}$, the first column of the chaotic matrices $S^{(3)}$, $S^{(4)}$, $S^{(5)}$, and the hash value parameters $H^{(2)}$ and $H^{(3)}$, generate a column vector

$$V = \left(D^{(C)}(:,N) + \left(S^{(3)}(:,1)+S^{(4)}(:,1)+S^{(5)}(:,1)\right)\times\left(H^{(2)}+H^{(3)}\right)\right) \bmod 4 \tag{15}$$

of size $M \times 1$. Then, encode V as a column vector $D^{(V)}$of DNA bases with the first column of $R^{(E)}$.

3.5 Dynamic DNA-level diffusion

According to Shannon's suggestion, a secure cryptosystem should not only meet confusion requirements, but also diffusion requirements. Therefore, IES-M-BD further performs DNA-level diffusion operations on D^C to ensure that the diffusion requirements can be met. Likewise, in order to improve encryption efficiency, the diffusion operation here is performed in units of columns. Besides, to ensure extremely high plaintext sensitivity, we introduce $H^{(1)}$, $H^{(2)}$ and an iterative structure during the encryption process. Specifically, the dynamic

DNA-level diffusion of IES-M-BD can be further subdivided into two steps.

- **Step 1**: Diffuse the first four columns of $D^{(C)}$, so as to get the first four columns of the diffusion result matrix $C^{(1)}$, as shown below.

$$\begin{cases} C^{(1)}(:,1) = D^{(C)}(:,1) \oplus D^{(S)}(:,1) \oplus D^{(V)}, \\ C^{(1)}(:,2) = D^{(C)}(:,2) \odot D^{(S)}(:,1) \oplus C^{(1)}(:,1), \\ C^{(1)}(:,3) = D^{(S)}(:,3) \odot D^{(C)}(:,3) \oplus C^{(1)}(:,2), \\ C^{(1)}(:,4) = D^{(C)}(:,4) \otimes D^{(S)}(:,4) \oplus C^{(1)}(:,3), \end{cases} \tag{16}$$

where ⊕, ⊙, and ⊗ represent DNA addtion, DNA substraction and DNA XOR operations, respectively.

- **Step 2**: Sequentially diffuse the fifth to Nth column of $D^{(C)}$ in order to get the entire $C^{(1)}$. The specific diffusion method is as follows.

$$\begin{cases} C^{(1)}(:,(j-1)\times 4+1) = D^{(C)}(:,(j-1)\times 4+1) \oplus D^{(S)}(:,(j-1)\times 4+1) \oplus C^{(1)}(:,(j-1)\times 4), \\ C^{(1)}(:,(j-1)\times 4+2) = D^{(C)}(:,(j-1)\times 4+2) \odot D^{(S)}(:,(j-1)\times 4+2) \oplus C^{(1)}(:,(j-1)\times 4+1), \\ C^{(1)}(:,(j-1)\times 4+3) = D^{(S)}(:,(j-1)\times 4+3) \odot D^{(C)}(:,(j-1)\times 4+3) \oplus C^{(1)}(:,(j-1)\times 4+2), \\ C^{(1)}(:,(j-1)\times 4+4) = D^{(C)}(:,(j-1)\times 4+4) \otimes D^{(S)}(:,(j-1)\times 4+4) \oplus C^{(1)}(:,(j-1)\times 4+3), \end{cases} \tag{17}$$

where $j = 2, 3, \ldots, 4 \times M \times N$.

3.6 DNA-level permutation

As mentioned in Section 3.5, in order to improve encryption efficiency, the dynamic DNA-level diffusion of IES-M-BD is a one-way diffusion operation in column units. Obviously, this diffusion method cannot ensure the sufficiency of diffusion. Therefore, IES-M-BD further introduces a DNA-level permutation operation, which together with the dynamic DNA-level diffusion forms a two-round iterative structure, thus ensuring the sufficiency of diffusion. Specifically, the DNA-level permutation of IES-M-BD includes the following steps.

- **Step 1**: Sort the chaotic sequence $S^{(6)}$ obtained in Section 3.1, and save the index obtained by sorting as a 1D index row vector I of length $4 \times M \times N$.
- **Step 2**: Stretching $C^{(1)}$ from a DNA matrix into a DNA sequence of length $4 \times M \times N$.
- **Step 3**: Initialize a DNA sequence $C^{(2)}$ of length $4 \times M \times N$, and let $C^{(2)}(i) = C^{(1)}(I(i))$, where $i = 1, 2, \ldots, 4 \times M \times N$.
- **Step 4**: Reshape the DNA sequence $C^{(2)}$ obtained in the previous step into a matrix of size $M \times (N \times 4)$.

As shown in Figure 5, after the first round of DNA-level diffusion and permutation operations, IES-M-BD performs another round of such operations. That is, $C^{(2)}$ is diffused again to get $C^{(3)}$, and then $C^{(3)}$ is permutated again to obtain fully diffused $C^{(4)}$.

3.7 Dynamic column-level DNA decoding

Similar to Section 3.4, the DNA decoding operation of IES-M-BD is also dynamic and column-wise, as shown below.

- **Step 1**: Taking the matrices $S^{(3)}$ and $S^{(4)}$ obtained in Section 3.4, generate the DNA decoding rules in groups of four columns, as shown below.

$$\begin{cases} R^{(D)}(:,(j-1)\times 4+1) = (T^{(1)} \bmod 8) + 1, \\ R^{(D)}(:,(j-1)\times 4+2) = (T^{(2)} \bmod 8) + 1, \\ R^{(D)}(:,(j-1)\times 4+3) = (T^{(1)} - T^{(2)}) \bmod 8 + 1, \\ R^{(D)}(:,(j-1)\times 4+4) = (T^{(1)} + T^{(2)}) \bmod 8 + 1, \end{cases} \tag{18}$$

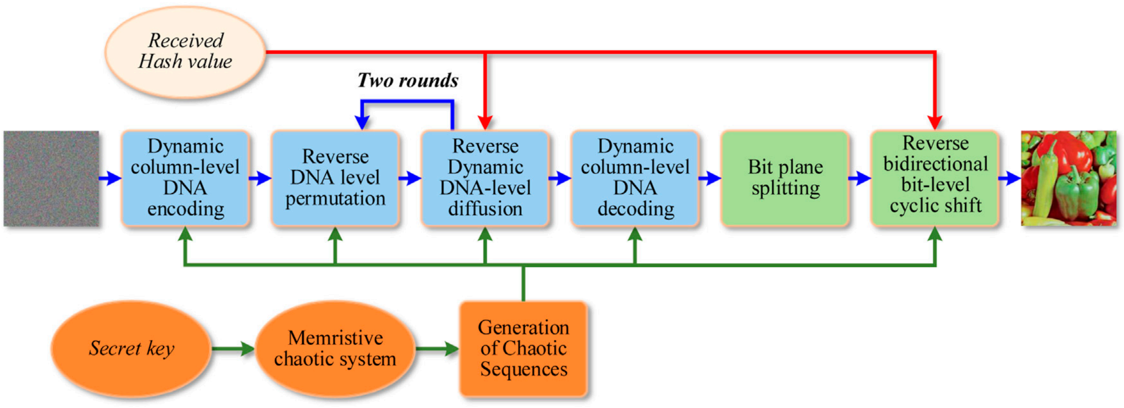


FIGURE 8
Decryption flow chart of IES-M-BD.

TABLE 2 Software and hardware configurations.

Configuration	Description
CPU	Intel Xeon E3-1,231 3.40 GHz
Memory	8 GB RAM
Operating system	Windows 7 (64 bit)
Simulation software	MATLAB R2017a (9.2.0538062)

where $R^{(D)}$ is the obtained DNA encoding rule matrix whose size is $M \times (N \times 4)$, $j = 1, 2, \ldots, N$, $T^{(1)} = \lfloor S^{(3)}(:, j)/4 \rfloor$, and $T^{(2)} = \lfloor S^{(4)}(:, j)/4 \rfloor$.

- **Step 2**: According to $R^{(D)}$ obtained in the previous step, the dynamic DNA decoding of $C^{(4)}$ is performed in column units, so as to obtain the quaternary matrix $Q^{(D)}$ of size $M \times (N \times 4)$.
- **Step 3**: Merge the four 2-bit planes of $Q^{(D)}$ to obtain the final ciphertext image $C^{(F)}$ of size $M \times N$.

Since IES-M-BD is an image encryption scheme with a symmetric structure, its decryption process is the reverse process of the encryption process, as shown in Figure 8.

4 Simulation tests and performance analysis

In this section, we will perform simulation tests and performance analysis on IES-M-BD. These tests and analyses include encryption and decryption effect test, key space analysis, key sensitivity analysis, differential attack analysis, pixel value distribution test, correlation analysis, information entropy analysis, noise and data loss attack tests, and time analysis. Without loss of generality, when testing and analyzing IES-M-BD, we all use randomly generated secret keys. And the images used in this section are from The USC-SIPI Image Database (http://sipi.usc.edu/database/). In addition, the hardware and software configurations used in this paper is shown in Table 2.

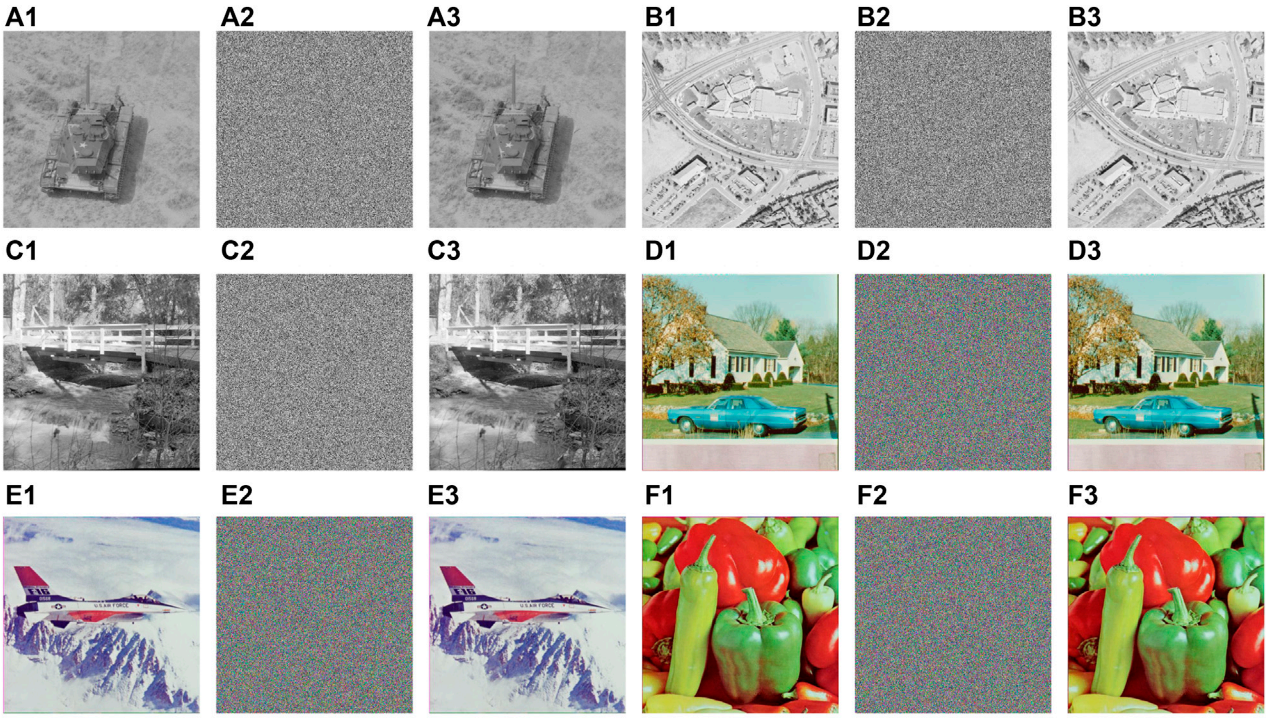


FIGURE 9
Encryption and decryption results of IES-M-BD **(A1)** 7.1.03; **(A2)** ciphertext of **(A1)**; **(A3)** decrypted image of **(A2)**; **(B1)** 5.2.09; **(B2)** ciphertext of **(B1)**; **(B3)** decrypted image of **(B2)**; **(C1)** 5.2.10; **(C2)** ciphertext of **(C1)**; **(C3)** decrypted image of **(C2)**; **(D1)** house; **(D2)** ciphertext of **(D1)**; **(D3)** decrypted image of **(D2)**; **(E1)** 4.2.05; **(E2)** ciphertext of **(E1)**; **(E3)** decrypted image of **(E2)**; **(F1)** 4.2.07; **(F2)** ciphertext of **(F1)**; **(F3)** decrypted image of **(F2)**.

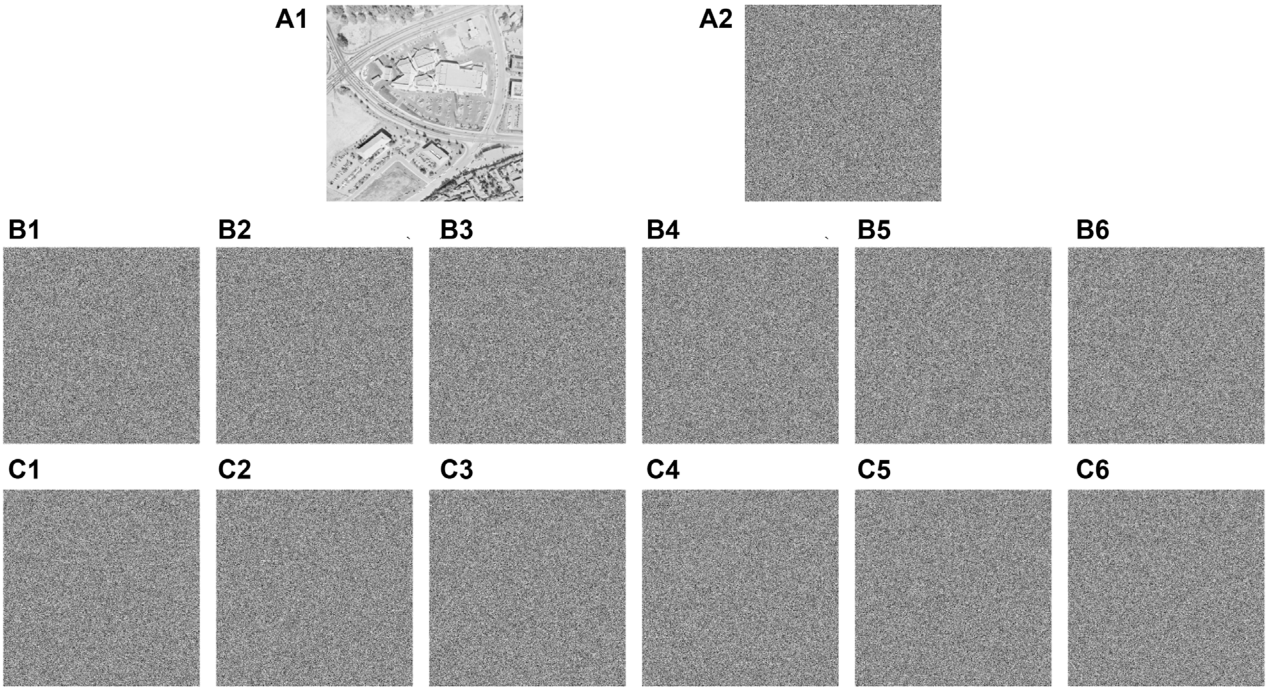


FIGURE 10
Key sensitivity test results of IES-M-BD **(A1)** 5.2.09; **(A2)** ciphertext image $\breve{C}_R$ obtained by K_R; **(B1)** ciphertext image $\breve{C}_C^{(1)}$ obtained by $K_C^{(1)}$; **(B2)** ciphertext image $\breve{C}_C^{(2)}$ obtained by $K_C^{(2)}$; **(B3)** ciphertext image $\breve{C}_C^{(3)}$ obtained by $K_C^{(3)}$; **(B4)** ciphertext image $\breve{C}_C^{(4)}$ obtained by $K_C^{(4)}$; **(B5)** ciphertext image $\breve{C}_C^{(5)}$ obtained by $K_C^{(5)}$; **(B6)** ciphertext image $\breve{C}_C^{(6)}$ obtained by $K_C^{(6)}$; **(C1)** difference image between $\breve{C}_R$ and $\breve{C}_C^{(1)}$; **(C2)** difference image between $\breve{C}_R$ and $\breve{C}_C^{(2)}$; **(C3)** difference image between $\breve{C}_R$ and $\breve{C}_C^{(3)}$; **(C4)** difference image between $\breve{C}_R$ and $\breve{C}_C^{(4)}$; **(C5)** difference image between $\breve{C}_R$ and $\breve{C}_C^{(5)}$; **(C6)** difference image between $\breve{C}_R$ and $\breve{C}_C^{(6)}$.

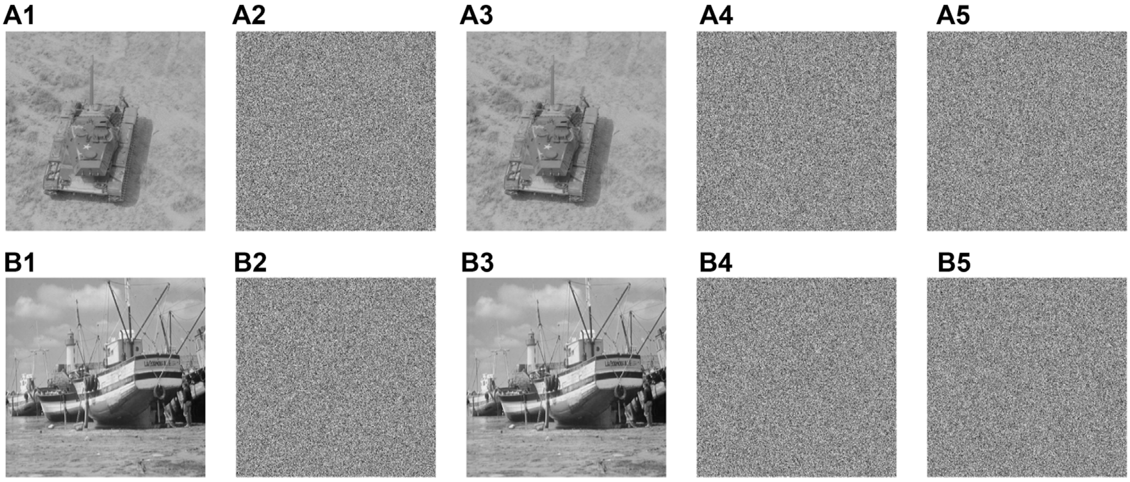


FIGURE 11
Differential attack test results of IES-M-BD **(A1)** 7.1.03; **(A2)** ciphertext of **(A1)**; **(A3)** 1 bit of the pixel at (256,256) in **(A1)** is changed; **(A4)** ciphertext of **(A3)**; **(A5)** difference image between **(A2)** and **(A4)**; **(B1)** boat.512; **(B2)** ciphertext of **(B1)**; **(B3)** 1 bit of the pixel at (512,512) in **(B1)** is changed; **(B4)** ciphertext of **(B3)**; **(B5)** difference image between **(B2)** and **(B4)**.

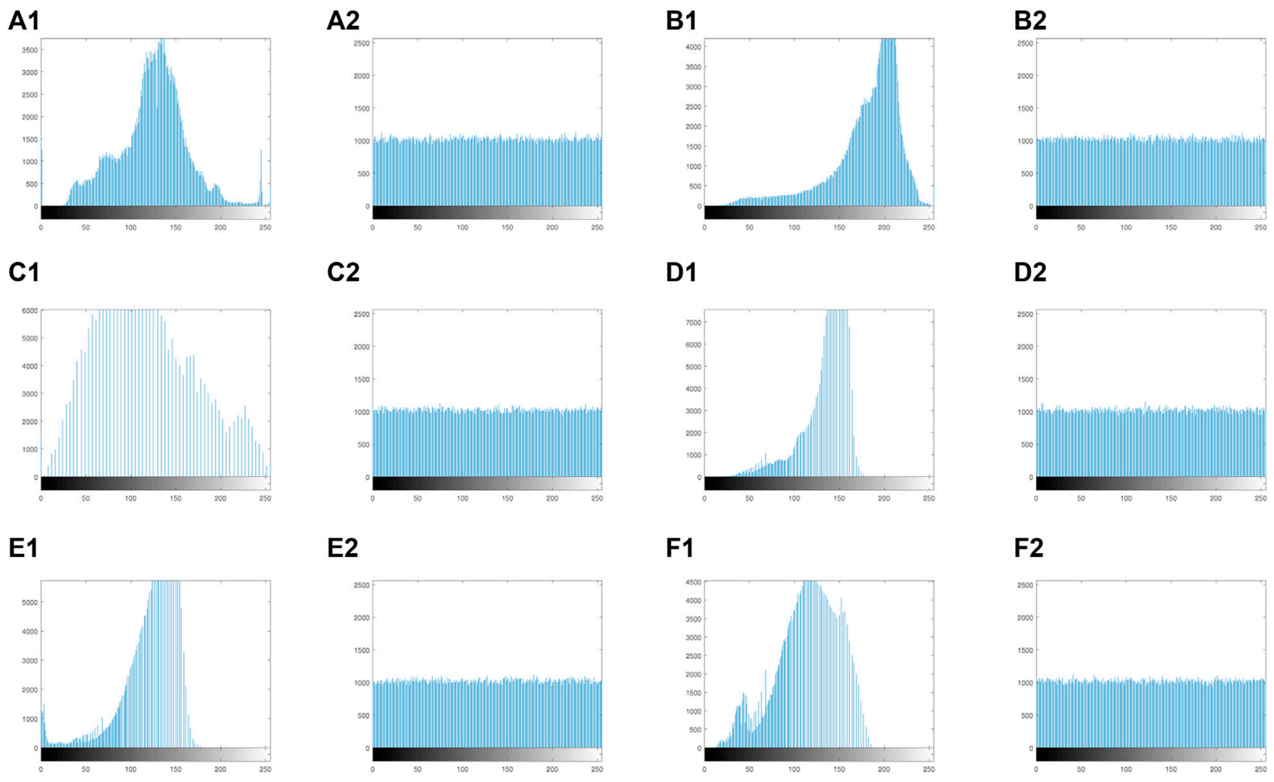


FIGURE 12
Histograms of plaintext images and corresponding ciphertext images **(A1)** 5.2.08; **(A2)** ciphertext of 5.2.08; **(B1)** 5.2.09; **(B2)** ciphertext of 5.2.09; **(C1)** 5.2.10; **(C2)** ciphertext of 5.2.08; **(D1)** 7.1.03; **(D2)** ciphertext of 7.1.03; **(E1)** 7.1.04; **(E2)** ciphertext of 7.1.04; **(F1)** 7.1.05; **(F2)** ciphertext of 7.1.05.

TABLE 3 NPCR and UACI test results of IES-M-BD and other image encryption schemes.

Filename	Reference [48]		Reference [49]		Reference [50]		IES-M-BD	
	NPCR	UACI	NPCR	UACI	NPCR	UACI	NPCR	UACI
Boat.512	99.6183	33.4997	99.6178	33.3916	99.5998	33.4519	99.6133	33.4727
Gray21.512	99.6172	33.4663	99.6032	33.4086	99.5949	33.4314	99.6173	33.4528
Ruler.512	99.6063	33.4744	99.6122	33.4917	99.6006	33.4521	99.6125	33.4825
5.2.10	99.6346	33.4315	99.6243	33.4848	99.6166	33.3925	99.6180	33.5226
7.1.02	99.6265	33.4850	99.6273	33.5207	99.6109	33.4415	99.6007	33.3661
7.1.03	99.6003	33.4638	99.5834	33.5305	99.6147	33.4455	99.6021	33.5262
Average	99.6172	33.4701	99.6114	33.4713	99.6063	33.4358	**99.6107**	**33.4623**
Std. Dev	0.0126	0.0231	0.0162	0.0580	0.0090	**0.0226**	**0.0075**	0.0586

The bolded values here emphasize better performance than other schemes.

4.1 Effect of encryption and decryption

In order to verify the effect of encryption and decryption, we encrypted some common test images with IES-M-BD, and then decrypted the generated ciphertext images. The relevant test results are shown in Figure 9. It can be seen that the ciphertext images generated by IES-M-BD are very similar to the noise image, and the encryption effect is excellent. In addition, the decrypted image are

TABLE 4 Chi-square test results of IES-M-BD.

Cipher image	Chi-square value	Result
	$\chi^2_{0.05}(255) = 293.2478$	
5.2.08	237.3184	Pass
5.2.09	249.1406	Pass
5.2.10	242.5469	Pass
7.1.03	233.7500	Pass
7.1.04	247.6367	Pass
7.1.05	265.9063	Pass
boat.512	252.4824	Pass
gray21.512	248.2910	Pass

exactly the same as the original plaintext images without any loss of information, so the decryption effect is also excellent.

4.2 Key space analysis

As we know, an encryption scheme must have a large enough key space to effectively resist brute force attacks. In general, it is believed that the key space of an encryption scheme should be at least greater than 2^{128} [34]. Considering the problems pointed out by some researchers in their cryptanalytic studies [39, 41–43], we define the secret key of IES-M-BD as a binary sequence of 312 bits, that is, $K = a_1 a_2 \ldots a_{312}$. Specifically, IES-M-BD uses six sets of bit sequences (52 × 6) in K to generate the control parameters (α, β, γ) and initial state values (x_0, y_0, z_0) of the memristive chaotic system. In this way, the key space of IES-M-BD is 2^{312}. Undoubtedly, this size is much larger than the normally required 2^{128}. Therefore, IES-M-BD can effectively resist brute force attacks.

4.3 Key sensitivity analysis

A secure image encryption scheme should not only have a large enough key space, but also be extremely sensitive to small changes in the secret key, so as to mask the statistical relationship between the secret key and the ciphertext image [44–47]. In order to test the sensitivity of IES-M-BD to the secret key, we first randomly generated a secret key K_R. The control parameters and initial state values of the memristive chaotic system generated by K_R are shown below.

$$\begin{cases} \alpha = 2.998657690759933 \\ \beta = 2.975155356302302 \\ \gamma = 2.396588179913862 \\ x_0 = 0.908100616630951 \\ y_0 = 0.750189035360335 \\ z_0 = 0.271661247330187 \end{cases}$$

Next, we sequentially made minimal changes to each set of 52-bit binary sequences in K_R, changing only one binary bit in one set of binary sequences at a time. In this way, we obtained six secret keys with the smallest difference from K_R, namely $K_C^{(1)}$, $K_C^{(2)}$, $K_C^{(3)}$, $K_C^{(4)}$, $K_C^{(5)}$, and $K_C^{(6)}$. Finally, we encrypted the same plaintext image 5.2.09. tiff with these seven secret keys, and calculated the difference image between the resulting ciphertext images. The relevant test results are shown in Figure 10.

As one can see from Figure 10, the ciphertext image will change dramatically even with only minimal changes to the secret key. Consequently, IES-M-BD has extremely high key sensitivity.

4.4 Differential attack analysis

As one know, various differential attacks including chosen-plaintext attack and known-plaintext attack are the most common attack methods for attackers. In order to effectively resist differential attacks, an encryption scheme must have extremely high sensitivity to the plaintext image. This means that even if only a very small change occurs in the plaintext image, the ciphertext image generated by this encryption scheme should change extremely significantly. In this paper, many differential attack tests are performed on IES-M-BD using randomly generated secret keys, and the relevant test results are shown in Figure 11.

According to the test results, even if the plaintext image has only a small change of 1 bit, the ciphertext image generated by IES-M-BD has changed completely. The

TABLE 5 CCs of different plain and cipher images under IES-M-BD.

Image	CC			
	Horizontal	Vertical	Diagonal	Opp. diag
5.2.09	0.8614	0.8911	0.8057	0.8043
Cipher of 5.2.09	**−0.0031**	**0.0042**	**−0.0039**	**−0.0047**
5.2.10	0.9246	0.9333	0.8926	0.8916
Cipher of 5.2.10	**0.0037**	**−0.0053**	**0.0032**	**−0.0018**
Elaine.512	0.9731	0.9719	0.9741	0.9754
Cipher of elaine.512	**−0.0039**	**−0.0027**	**−0.0031**	**0.0040**
Boat.512	0.9736	0.9326	0.9232	0.9164
Cipher of boat.512	**0.0047**	**0.0040**	**−0.0025**	**0.0038**
Gray21.512	0.9981	0.9937	0.9946	0.9934
Cipher of gray21.512	**0.0015**	**−0.0110**	**−0.0118**	**0.0045**

The bolded values here emphasize that the ciphertext images have extremely low CC values.

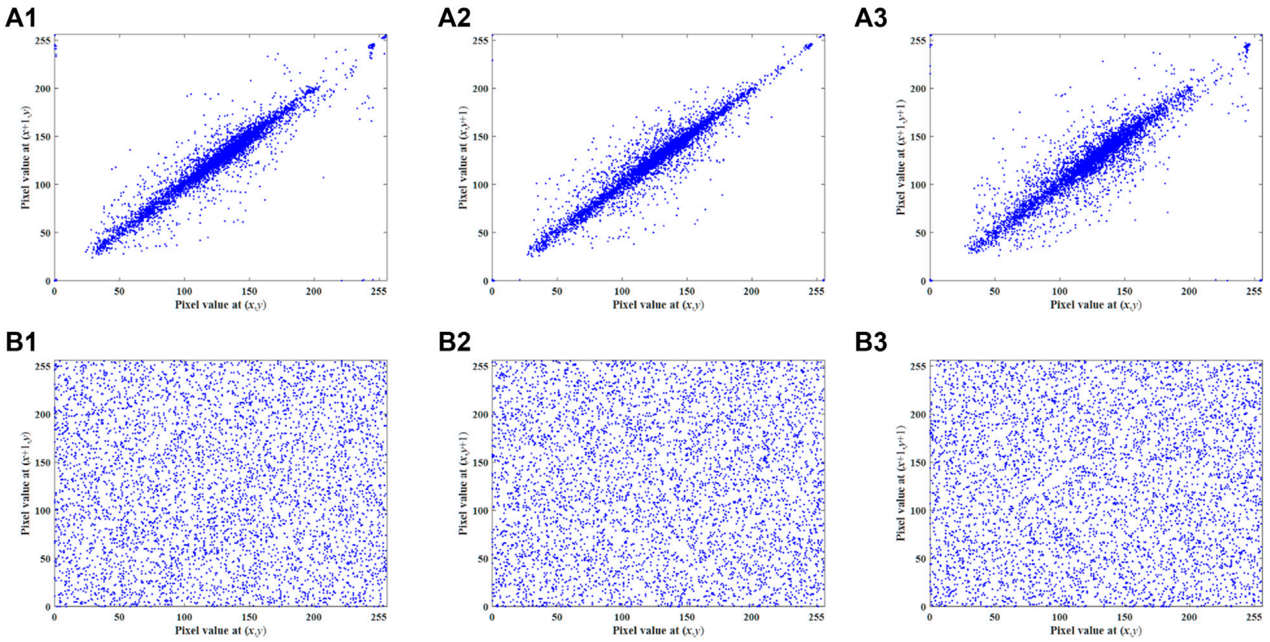


FIGURE 13
Pixel correlations of 5.2.08 and its ciphertext image generated by IES-M-BD **(A1)** horizontal direction of 5.2.08; **(A2)** vertical direction of 5.2.08; **(A3)** diagonal direction of 5.2.08; **(B1)** horizontal direction of 5.2.08 ciphertext; **(B2)** vertical direction of 5.2.08 ciphertext; **(B3)** diagonal direction of 5.2.08 ciphertext.

difference image between the ciphertext images before and after the change is similar to the ordinary ciphertext image and the noise image. Besides, this significant change is independent of the location of changed pixel bits. Consequently, IES-M-BD has excellent ability to resist differential attacks.

In addition, we also utilize the number of pixel change ratio (NPCR) and unified average change in intensity (UACI) to further evaluate the ability of IES-M-BD to resist differential attacks, as shown in Table 3. Mathematically, these two metrics can be defined as follows.

$$\begin{cases} \mathrm{NPCR} = \dfrac{1}{M \times N} \sum_{i,j} D(i,j) \times 100\%, \\ D(i,j) = \begin{cases} 0, I_1(i,j) = I_2(i,j), \\ 1, I_1(i,j) \neq I_2(i,j), \end{cases} \end{cases} \tag{19}$$

$$\mathrm{UACI} = \frac{1}{M \times N}\left[\sum_{i,j} \frac{|I_1(i,j) - I_2(i,j)|}{255}\right] \times 100\%, \tag{20}$$

where I_1, I_2 are two images, each of them has the size of $M \times N$. According to the test results in Table 3 and the comparison with the other three encryption schemes, one can know that the test results of IES-M-BD are the closest to the ideal value and have

TABLE 6 CCs of adjacent pixels in Lena cipher images.

Scheme	CC		
	Horizontal	Vertical	Diagonal
Reference [48]	0.0055	−0.0068	−0.0032
Reference [51]	−0.0158	−0.0042	−0.0039
Reference [52]	−0.0066	0.0025	0.0042
Reference [53]	−0.0065	−0.0017	0.0132
IES-M-BD	**−0.0036**	**−0.0012**	**0.0024**

The bold values here emphasize that IES-M-BD has better performance than other schemes.

TABLE 7 Information entropy test results of IES-M-BD.

Name	Plaintext image	Ciphertext image
5.2.08	7.2010	7.9994
5.2.09	6.9940	7.9993
5.2.10	5.7056	7.9991
Boat.512	7.1914	7.9993
Gray21.512	4.3923	7.9992
7.1.03	5.4957	7.9993
7.1.04	6.1074	7.9992
7.1.05	6.5632	7.9993

TABLE 8 Test results of five different encryption schemes in terms of information entropy.

Scheme	Reference [53]	Reference [54]	Reference [55]	Reference [56]	IES-M-BD
Inf. Entropy	7.9976	7.9971	7.9980	7.9909	**7.9992**

The bold value here emphasize that IES-M-BD has better performance than other schemes.

TABLE 9 LSE test results of three image encryption schemes.

Name	Reference [48]	Reference [50]	IES-M-BD
5.2.08	7.902314	7.899817	7.902442
5.2.09	7.902032	7.902151	7.902186
5.2.10	7.904711	7.900943	7.902561
boat.512	7.902762	7.902369	7.902253
gray21.512	7.903553	7.898930	7.902549
7.1.03	7.902564	7.899937	7.901930
7.1.04	7.902923	7.901742	7.902127
7.1.05	7.901372	7.902194	7.901931
Average	7.902737	7.901094	**7.902247**
Std. Dev	0.000932	0.001293	**0.000253**
Pass/All	5/8	3/8	**8/8**

The bold values here emphasize that IES-M-BD has better performance than other schemes.

very high stability. Therefore, IES-M-BD does have excellent resistance to differential attacks.

4.5 Pixel value distribution

The uniformity of the pixel value distribution of the ciphertext image is also an important aspect to measure the security of an encryption scheme, because it is related to whether it can effectively resist statistical attacks. In order to verify the ability of IES-M-BD to resist statistical attacks, we have drawn some histograms of the plaintext images and the ciphertext images generated by IES-M-BD respectively, as shown in Figure 12. It can be seen that in these ciphertext images, the pixel distribution features in the plaintext images have been completely eliminated, and one cannot perceive any relevant features.

In addition to drawing histograms, one can also quantitatively analyze the pixel distribution uniformity of ciphertext images through the chi-square test [50]. The chi-square test can be defined as follows.

$$\chi^2 = \sum_{i=1}^{k} \frac{(n_i - M \times N \times p)^2}{M \times N \times p}, \tag{21}$$

where n_i is the number of pixels whose value is $i - 1$, $M \times N$ is the size of the cipher image, k is the number of all possible pixel values ($k = 256$ for grayscale images), and $p = 1/k$. Next, one can calculate the critical value of the chi-square test at the significant level $\alpha = 0.05$. When the chi-square value of a cipher image is less than the critical value, it is considered to have passed the chi-square test. From the test results shown in Table 4, all ciphertext images generated by IES-M-BD have passed the chi-square test. This means that IES-M-BD does have excellent resistance to statistical attacks.

4.6 Correlation analysis

There is a high correlation between adjacent pixels of the plaintext images in each direction, as shown in the first row of Figure 13. Therefore, a secure image encryption scheme should be able to effectively remove such correlations. Since pixel-level cyclic shift and DNA-level permutation are introduced in the encryption process of IES-M-BD, IES-M-BD can completely eliminate the correlation between adjacent pixels, as shown in the second row of Figure 13.

In addition, in order to more accurately verify the correlation between adjacent pixels, many researchers use the correlation coefficient (CC) to perform quantitative analysis. Mathematically, we can define CC as follows.

$$\mathrm{CC} = \frac{E\left((v_x - E(v_x)) \times \left(v_y - E\left(v_y\right)\right)\right)}{\sqrt{D(v_x) \times D\left(v_y\right)}}, \tag{22}$$

where $E(v)$ and $D(v)$ are the expectation and variance of the grayscale value v, v_x and v_y are the gray values of two adjacent pixels in a certain direction. After calculating the CC values of a large number of plaintext images and some ciphertext images, we found that the CC values of the plaintext images are high, while the CC values of the ciphertext images generated by IES-M-BD are very low, as shown in Table 5. This means that IES-M-BD can indeed significantly eliminate the correlations between adjacent pixels. As shown in Table 6, compared with some recent encryption schemes, the Lena cipher image generated by IES-M-BD demonstrates certain superiority in the correlation of adjacent pixels. It can be seen that the ciphertext image generated by IES-M-BD has the lowest correlations between adjacent pixels in all directions.

4.7 Information entropy analysis

In order to measure the randomness of images generated by encryption schemes, researchers often use information entropy as an evaluation metric. Specifically, we can define information entropy as follows.

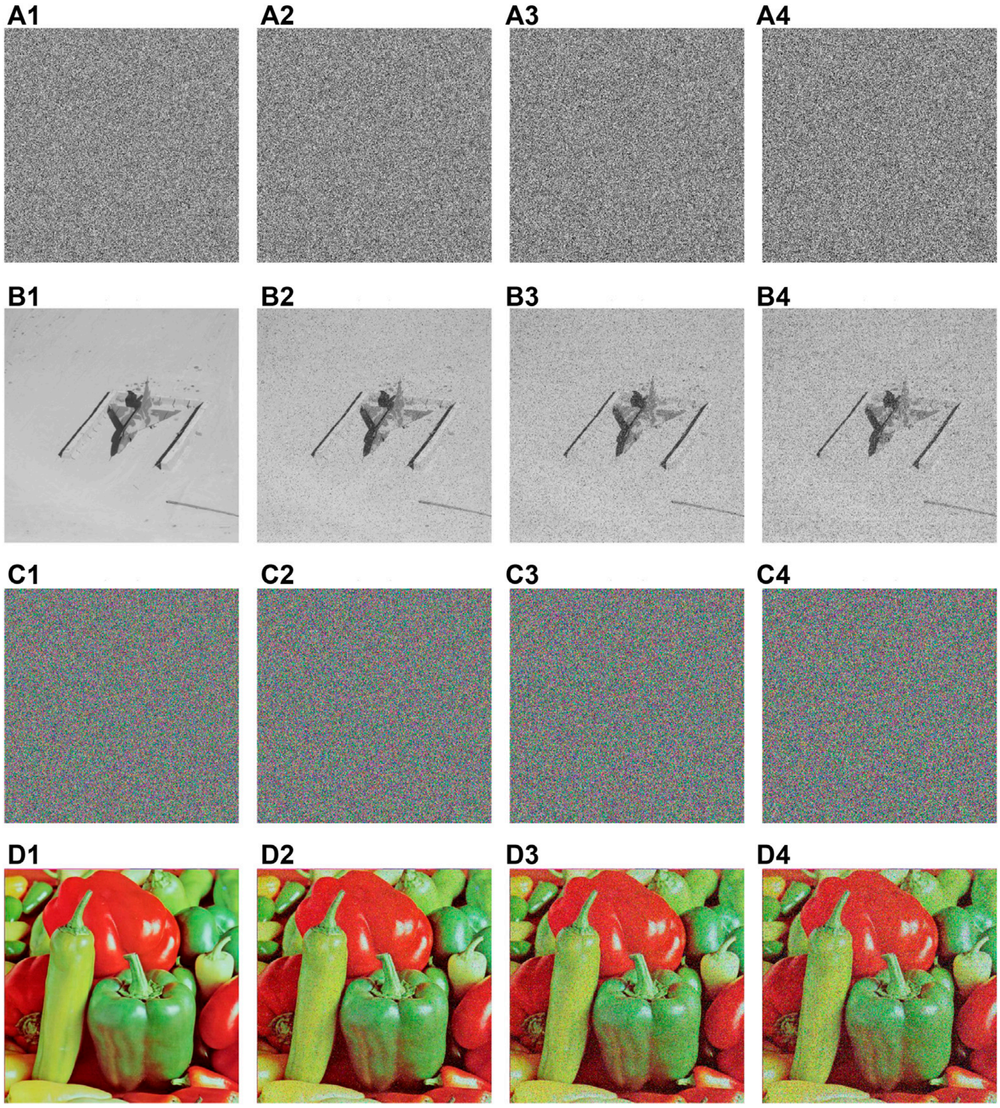


FIGURE 14
Noise attack test results of IES-M-BD **(A1)** ciphertext image of 7.1.02 **(A2)** contaminated by SPN with density = 0.05; **(A3)** contaminated by SPN with density = 0.10; **(A4)** contaminated by SPN with density = 0.15; **(B1)** decrypted image of **(A1)**; **(B2)** decrypted image of **(A2)**; **(B3)** decrypted image of **(A3)**; **(B4)** decrypted image of **(A4)**; **(C1)** ciphertext image of 4.2.07 **(C2)** contaminated by SPN with density = 0.05; **(C3)** contaminated by SPN with density = 0.10; **(C4)** contaminated by SPN with density = 0.15; **(D1)** decrypted image of **(C1)**; **(D2)** decrypted image of **(C2)**; **(D3)** decrypted image of **(C3)**; **(D4)** decrypted image of **(C4)**.

$$I(u) = \sum_{x=1}^{N} p(u_x)\log_2 \frac{1}{p(u_x)}, \tag{23}$$

where u_x is one of the symbols with a total number of N, and $p(u_x)$ represents the occurrence probability of u_x. If the number of gray levels is 256, then the ideal value of image information entropy is 8. Therefore, the information entropy of an image generated by an encryption scheme is closer to 8, which means less information leakage. Table 7 compares the information entropy values of some plaintext images and the corresponding ciphertext images. After the encryption of IES-M-BD, the information entropy of the image becomes very close to the ideal value, that is, the obtained ciphertext image has excellent randomness. Not only that, IES-M-BD also shows significant advantages compared with other image encryption schemes, as shown in Table 8.

Considering the limitation of information entropy, another improved metric, local Shannon entropy (LSE), is proposed to

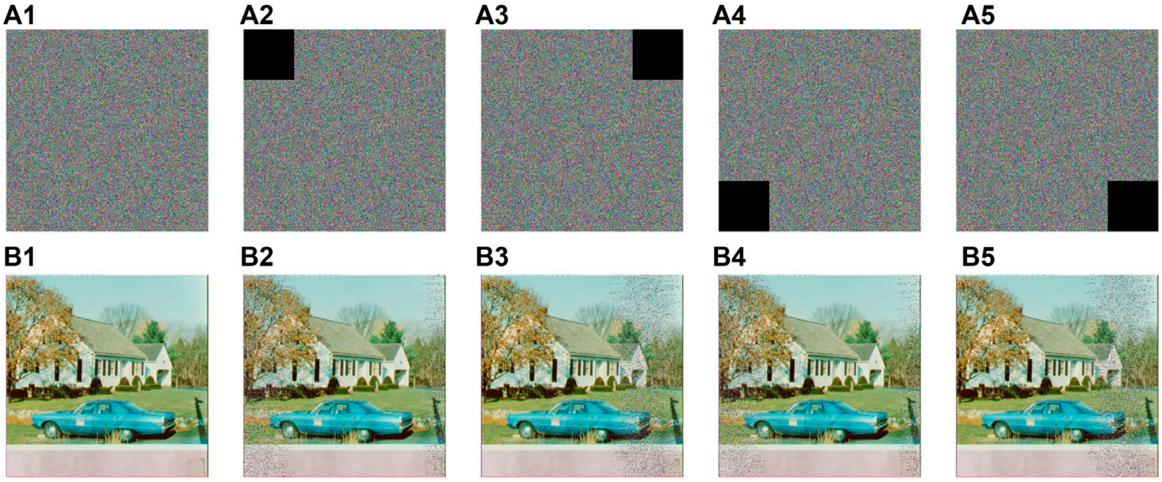


FIGURE 15
Occlusion attack test results of IES-M-BD **(A1)** ciphertext of house; **(A2)** 1/16 data loss at top-left corner; **(A3)** 1/16 data loss at top-right corner; **(A4)** 1/16 data loss at bottom-left corner; **(A5)** 1/16 data loss at bottom-right corner; **(B1)** decrypted ciphertext of **(A1)**; **(B2)** decrypted ciphertext of **(A2)**; **(B3)** decrypted ciphertext of **(A3)**; **(B4)** decrypted ciphertext of **(A4)**; **(B5)** decrypted ciphertext of **(A5)**.

TABLE 10 Average times required by different schemes to encrypt some common test images.

Size	Name	IES-M-BD	Reference [49]	Reference [50]	Reference [56]
256 × 256	5.1.09	0.4348 s	0.3351 s	1.0532 s	0.4547 s
256 × 256	5.1.10	0.4334 s	0.3841 s	1.0426 s	0.4637 s
512 × 512	5.2.08	1.7632 s	1.3741 s	4.1631 s	1.8421 s
512 × 512	5.2.09	1.7553 s	1.3548 s	4.3552 s	1.8724 s
512 × 512	5.2.10	1.7556 s	1.3827 s	4.2325 s	1.8523 s
512 × 512	7.1.03	1.7589 s	1.4728 s	4.1317 s	1.8325 s
512 × 512	7.1.04	1.7557 s	1.5748 s	4.2746 s	1.8643 s
512 × 512	7.1.05	1.7626 s	1.3748 s	4.3183 s	1.9012 s

better measure the randomness of ciphertext images [57]. Mathematically, LSE can be defined as follows.

$$L_{N,S}(o) = \sum_{i=1}^{N} \frac{H(o_i)}{N}, \tag{24}$$

where N represents the number of image blocks, S represents the number of pixels each image block has, o_i represents each image block randomly selected from the measured image, and $H(o_i)$ represents the information entropy of each image block.

According to [57], if the LSE test value of the tested image is between 7.901901305 and 7.903037329, it can be confirmed that the image is random in the sense of LSE. Consequently, we have tested the ciphertext images generated by IES-M-BD, and the specific test results are included in Table 9. According to Table 9, it can be concluded that the test values of IES-M-BD are more stable than that of other schemes, and the pass rate of the LSE test is also higher.

4.8 Resistance to noise and data loss

There are two situations that often occur during the transmission of ciphertext images, that is, the transmitted image is contaminated by noise or part of the data is lost. In fact, malicious attackers may also cause these situations to occur. Therefore, an image encryption scheme must be able to resist these attacks in order to be practical. To simulate these noise attacks, we deliberately add salt and pepper noise (SPN) to the ciphertext images generated by IES-M-BD. Figure 14 shows the relevant test results.

From the test results shown in Figure 14, although these ciphertext images are contaminated by noise, IES-M-BD can still decrypt the images normally. Although the quality of the decrypted image will decrease with the increase of the noise intensity, this does not hinder the transmission of the visual information of the plaintext image.

Similarly, to evaluate the reliability of IES-M-BD under data loss conditions, we intentionally removed some pixels from the ciphertext images. The relevant test results are shown in Figure 15. As one can see, despite the loss of data, IES-M-BD can still decrypt the images normally. That is, the transmission of the visual information of the plaintext image is not significantly affected. Specifically, the more data is lost, the lower the quality of the decrypted image, while IES-M-BD can still maintain high image quality when some pixels are missing.

4.9 Time analysis

Undoubtedly, in many scenarios of information security application, the encryption efficiency of the cryptographic system is a key factor. Therefore, when designing an image encryption scheme, we should not only consider the security, but also consider the efficiency of the encryption scheme. To this end, we tested IES-M-BD and compared the test results with other encryption schemes, as shown in Table 10. It can be seen that IES-M-BD has higher or comparable encryption efficiency compared with other image encryption schemes. Notably, according to the various analyses above, IES-M-BD is better than the remaining encryption schemes in terms of security. Definitely, in the future, we will also consider further optimization of IES-M-BD to make it more efficient.

5 Conclusion

Aiming at the key problems existing in image encryption, this paper proposes a novel encryption scheme that embraces several innovative designs. Firstly, a discrete memristive chaotic system is exploited to enhance the ergodicity of the generated chaotic sequences. Secondly, the proposed encryption scheme leverages the hash value in a more reasonable way. That is, the proposed image encryption scheme utilizes the hash value to influence the cyclic shift and DNA diffusion operations instead of using it directly as the secret key. Such ingenious design can not only solve the practicability problem brought by the one-time pad secret key design, but also significantly improve the sensitivity of the encryption scheme to the plaintext image. Thirdly, a bidirectional bit-level cyclic shift operation is designed, which can realize permutation and confusion at the same time, thereby further improving the security of the encryption scheme. Finally, the novel DNA sequence operation design, including dynamic column-level DNA encoding, dynamic DNA-level diffusion, DNA-level permutation, and dynamic column-level DNA decoding, not only improves encryption efficiency, but also ensures the security of the proposed encryption scheme once again. In the future, we will continuously optimize this encryption scheme, and extend it to the field of video encryption.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

JZ and XL provided the idea of algorithm, KQ and ZZ carried out the simulations, arranged the architecture and drafted the manuscript. WF and ZQ supervised the work and revised the manuscript. Both authors read and approved the final manuscript.

Funding

This research was funded by the Research Foundation of Education Bureau of Hunan Province of China (No. 19C0864), the Science and Technology Development Center Project of Chinese Ministry of Education (No. 2021KSA01008), the Project of the Sichuan Higher Education Society of China (No. GJXHXXH21-YB-27), the Guiding Science and Technology Plan Project of Panzhihua City (No. 2020ZD-S-40), and the Doctoral Research Startup Foundation of Panzhihua University (No. 2020DOC019).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher’s note

References

1. Chua L. Memristor-the missing circuit element. *IEEE Trans Circuit Theor* (1971) 18:507–19. doi:10.1109/TCT.1971.1083337

2. Buscarino A, Fortuna L, Frasca M, Gambuzza LV. A chaotic circuit based on hewlett-packard memristor. *Chaos* (2012) 22:023136. Art. No. doi:10.1063/1.4729135

3. Strukov DB, Snider GS, Stewart DR, Williams RS. The missing memristor found. *nature* (2008) 453:80–3. doi:10.1038/nature06932

4. Peng Y, He S, Sun K. Parameter identification for discrete memristive chaotic map using adaptive differential evolution algorithm. *Nonlinear Dyn* (2022) 107:1263–75. doi:10.1007/s11071-021-06993-0

5. Peng Y, He S, Sun K. A higher dimensional chaotic map with discrete memristor. *AEU - Int J Electronics Commun* (2021) 129:153539. Art. No. 153539. doi:10.1016/j.aeue.2020.153539

6. Li CL, Li ZY, Feng W, Tong YN, Du JR, Wei DQ, et al. Dynamical behavior and image encryption application of a memristor-based circuit system. *AEU - Int J Electronics Commun* (2019) 110:152861. Art. No. 152861. doi:10.1016/j.aeue.2019.152861

7. Bao H, Wang N, Bao B, Chen M, Jin P, Wang G, et al. Initial condition-dependent dynamics and transient period in memristor-based hypogenetic jerk system with four line equilibria. *Commun Nonlinear Sci Numer Simulation* (2018) 57:264–75. doi:10.1016/j.cnsns.2017.10.001

8. Sabarathinam S, Volos CK, Thamilmaran K. Implementation and study of the nonlinear dynamics of a memristor-based duffing oscillator. *Nonlinear Dyn* (2017) 87:37–49. doi:10.1007/s11071-016-3022-8

9. Rajagopal K, Jafari S, Karthikeyan A, Srinivasan A, Ayele B. Hyperchaotic memcapacitor oscillator with infinite equilibria and coexisting attractors. *Circuits Syst Signal Process* (2018) 37:3702–24. doi:10.1007/s00034-018-0750-7

10. Zhao R, Zhang Y, Xiao X, Ye X, Lan R. Tpe2: Three-pixel exact thumbnail-preserving image encryption. *Signal Process.* (2021) 183:108019. Art. No. 108019. doi:10.1016/j.sigpro.2021.108019

11. Hua Z, Zhou Y, Huang H. Cosine-transform-based chaotic system for image encryption. *Inf Sci* (2019) 480:403–19. doi:10.1016/j.ins.2018.12.048

12. Zhou S, He Y, Liu Y, Li C, Zhang J. Multi-channel deep networks for block-based image compressive sensing. *IEEE Trans Multimedia* (2021) 23:2627–40. doi:10.1109/TMM.2020.3014561

13. Zhou S, Deng X, Li C, Liu Y, Jiang H. Recognition-oriented image compressive sensing with deep learning. *IEEE Trans Multimedia* (2022) 1. doi:10.1109/TMM.2022.3142952

14. Zhao R, Zhang Y, Xiao X, Ye X, Lan R. TPE2: Three-pixel exact thumbnail-preserving image encryption. *Signal Process.* (2021) 183. 108019. Art. No. doi:10.1016/j.sigpro.2021.108019

15. Zhang Y, Wang P, Fang L, He X, Chen B. Secure transmission of compressed sampling data using edge clouds. *IEEE Trans Industr Inform* (2020) 16:6641–51. doi:10.1109/TII.2020.2966511

16. Feng W, Zhang J, Qin Z. A secure and efficient image transmission scheme based on two chaotic maps. *Complexity* (2021) 2021:1–19. art. no.898998. doi:10.1155/2021/1898998

17. Liu H, Kadir A, Xu C. Color image encryption with cipher feedback and coupling chaotic map. *Int J Bifurcation Chaos* (2020) 30:2050173. doi:10.1142/S0218127420501734

18. Pourasad Y, Ranjbarzadeh R, Mardani A. A new algorithm for digital image encryption based on chaos theory. *Entropy* (2021) 23:341. Art. No. 341. doi:10.3390/e23030341

19. Wang S, Wang C, Xu C. An image encryption algorithm based on a hidden attractor chaos system and the knuth–durstenfeld algorithm. *Opt Lasers Eng* (2020) 128:105995. Art. No. doi:10.1016/j.optlaseng.2019.105995

20. Si Y, Liu H, Chen Y. Constructing keyed strong s-box using an enhanced quadratic map. *Int J Bifurcation Chaos* (2021) 31:2150146. doi:10.1142/S0218127421501467

21. Li C, Tan K, Feng B, Lü J. The graph structure of the generalized discrete arnold's cat map. *IEEE Trans Comput* (2022) 71:364–77. doi:10.1109/TC.2021.3051387

22. Li C, Feng B, Li S, Kurths J, Chen G. Dynamic analysis of digital chaotic maps via state-mapping networks. *IEEE Trans Circuits Syst* (2019) 66:2322–35. doi:10.1109/TCSI.2018.2888688

23. Zhang Y, Wang P, Huang H, Zhu Y, Xiao D, Xiang Y, et al. Privacy-assured fogcs: Chaotic compressive sensing for secure industrial big image data processing in fog computing. *IEEE Trans Industr Inform* (2020) 17:3401–11. doi:10.1109/TII.2020.3008914

24. Zhang Y, Zhao R, Xiao X, Lan R, Liu Z, Zhang X, et al. Hf-tpe: High-fidelity thumbnail- preserving encryption. *IEEE Trans Circuits Syst Video Technol* (2022) 32:947–61. doi:10.1109/TCSVT.2021.3070348

25. Cao C, Sun K, Liu W. A novel bit-level image encryption algorithm based on 2d-licm hyperchaotic map. *Signal Process.* (2018) 143:122–33. doi:10.1016/j.sigpro.2017.08.020

26. Özkaynak F. Brief review on application of nonlinear dynamics in image encryption. *Nonlinear Dyn* (2018) 92:305–13. doi:10.1007/s11071-018-4056-x

27. Zhu Z, Zhang W, Wong K, Yu H. A chaos-based symmetric image encryption scheme using a bit-level permutation. *Inf Sci* (2011) 181:1171–86. doi:10.1016/j.ins.2010.11.009

28. Li Y, Wang C, Chen H. A hyper-chaos-based image encryption algorithm using pixel-level permutation and bit-level permutation. *Opt Lasers Eng* (2017) 90:238–46. doi:10.1016/j.optlaseng.2016.10.020

29. Teng L, Wang X, Meng J. A chaotic color image encryption using integrated bit-level permutation. *Multimed Tools Appl* (2018) 77:6883–96. doi:10.1007/s11042-017-4605-1

30. Zhang Q, Guo L, Wei X. A novel image fusion encryption algorithm based on dna sequence operation and hyper-chaotic system. *Optik - Int J Light Electron Opt* (2013) 124:3596–600. doi:10.1016/j.ijleo.2012.11.018

31. Chai X, Chen Y, Broyde L. A novel chaos-based image encryption algorithm using dna sequence operations. *Opt Lasers Eng* (2017) 88:197–213. doi:10.1016/j.optlaseng.2016.08.009

32. Feng W, He Y, Li H, Li C. A plain-image-related chaotic image encryption algorithm based on DNA sequence operation and discrete logarithm. *IEEE Access* (2019) 7:181589–609. doi:10.1109/ACCESS.2019.2959137

33. Chai X, Fu X, Gan Z, Lu Y, Chen Y. A color image cryptosystem based on dynamic dna encryption and chaos. *Signal Process.* (2019) 155:44–62. doi:10.1016/j.sigpro.2018.09.029

34. Li H, Li T, Feng W, Zhang J, Zhang J, Gan L, et al. A novel image encryption scheme based on non-adjacent parallelable permutation and dynamic dna-level two-way diffusion. *J Inf Security Appl* (2021) 61:102844. Art. No. 102844. doi:10.1016/j.jisa.2021.102844

35. Feng W, He Y. Cryptanalysis and improvement of the hyper-chaotic image encryption scheme based on DNA encoding and scrambling. *IEEE Photon J* (2018) 10:1–15. Art. No. 7909215. doi:10.1109/JPHOT.2018.2880590

36. Feng W, He Y, Li H, Li C. Cryptanalysis and improvement of the image encryption scheme based on 2D logistic-adjusted-sine map. *IEEE Access* (2019) 7:12584–97. doi:10.1109/ACCESS.2019.2893760

37. Liu H, Kadir A, Xu C. Cryptanalysis and constructing s-box based on chaotic map and backtracking. *Appl Mathematics Comput* (2020) 376:125153. doi:10.1016/j.amc.2020.125153

38. Feng W, Zhang J. Cryptanalzing a novel hyper-chaotic image encryption scheme based on pixel-level filtering and DNA-level diffusion. *IEEE Access* (2020) 8:209471–82. doi:10.1109/ACCESS.2020.3038006

39. Ma Y, Li C, Ou B. Cryptanalysis of an image block encryption algorithm based on chaotic maps. *J Inf Security Appl* (2020) 54:102566. Art. No. 102566. doi:10.1016/j.jisa.2020.102566

40. Feng W, Qin Z, Zhang J, Ahmad M. Cryptanalysis and improvement of the image encryption scheme based on feistel network and dynamic dna encoding. *IEEE Access* (2021) 9:145459–70. doi:10.1109/ACCESS.2021.3123571

41. Chen L, Li C, Li C. Security measurement of a medical communication scheme based on chaos and DNA coding. *J Vis Commun Image Representation* (2022) 83:103424. Art. No. 103424. doi:10.1016/j.jvcir.2021.103424

42. Liu S, Li C, Hu Q. Cryptanalyzing two image encryption algorithms based on a first-order time-delay system. *IEEE MultiMedia* (2022) 29:74–84. doi:10.1109/MMUL.2021.3114589

43. Li C, Zhang Y, Xie EY. When an attacker meets a cipher-image in 2018: A year in review. *J Inf Security Appl* (2019) 48:102361. Art. No. 102361. doi:10.1016/j.jisa.2019.102361

44. Li C, Lin D, Lü J, Hao F. Cryptanalyzing an image encryption algorithm based on autoblocking and electrocardiography. *IEEE MultiMedia* (2018) 25:46–56. doi:10.1109/MMUL.2018.2873472

45. Li C, Lin D, Lü J. Cryptanalyzing an image-scrambling encryption algorithm of pixel bits. *IEEE MultiMedia* (2017) 24:64–71. doi:10.1109/MMUL.2017.3051512

46. Zhang Y, He Q, Xiang Y, Zhang LY, Liu B, Chen J, et al. Low-cost and confidentiality-preserving data acquisition for internet of multimedia things. *IEEE Internet Things J* (2018) 5:3442–51. doi:10.1109/JIOT.2017.2781737

47. Zhang Y, He Q, Chen G, Zhang X, Xiang Y. A low-overhead, confidentiality-assured, and authenticated data acquisition framework for iot. *IEEE Trans Industr Inform* (2020) 16:7566–78. doi:10.1109/TII.2019.2957404

48. Xu M, Tian Z. A novel image cipher based on 3d bit matrix and Latin cubes. *Inf Sci* (2019) 478:1–14. doi:10.1016/j.ins.2018.11.010

49. Zhu H, Zhang X, Yu H, Zhao C, Zhu Z. An image encryption algorithm based on compound homogeneous hyper-chaotic system. *Nonlinear Dyn* (2017) 89: 61–79. doi:10.1007/s11071-017-3436-y

50. Zhu H, Zhao Y, Song Y. 2d logistic-modulated-sine-coupling-logistic chaotic map for image encryption. *IEEE Access* (2019) 7:14081–98. doi:10.1109/ACCESS.2019.2893538

51. Yin Q, Wang C. A new chaotic image encryption scheme using breadth-first search and dynamic diffusion. *Int J Bifurcation Chaos* (2018) 28:1850047. Art. No. 1850047. doi:10.1142/S0218127418500475

52. Zahmoul R, Ejbali R, Zaied M. Image encryption based on new beta chaotic maps. *Opt Lasers Eng* (2017) 96:39–49. doi:10.1016/j.optlaseng.2017.04.009

53. Zefreh EZ. An image encryption scheme based on a hybrid model of dna computing, chaotic systems and hash functions. *Multimed Tools Appl* (2020) 79: 24993–5022. doi:10.1007/s11042-020-09111-1

54. Wang X, Liu L, Zhang Y. A novel chaotic block image encryption algorithm based on dynamic random growth technique. *Opt Lasers Eng* (2015) 66:10–8. doi:10.1016/j.optlaseng.2014.08.005

55. Diaconu AV. Circular inter-intra pixels bit-level permutation and chaos-based image encryption. *Inf Sci* (2016) 355:356314–27. doi:10.1016/j.ins.2015.10.027

56. Wu X, Wang D, Kurths J, Kan H. A novel lossless color image encryption scheme using 2d dwt and 6d hyperchaotic system. *Inf Sci* (2016) 349:350137–53. doi:10.1016/j.ins.2016.02.041

57. Wu Y, Zhou Y, Saveriades G, Agaian S, Noonan JP, Natarajan P, et al. Local shannon entropy measure with statistical tests for image randomness. *Inf Sci* (2013) 222:323–42. doi:10.1016/j.ins.2012.07.049

Advantages of publishing in Frontiers

OPEN ACCESS
Articles are free to read
for greatest visibility
and readership

FAST PUBLICATION
Around 90 days
from submission
to decision

HIGH QUALITY PEER-REVIEW
Rigorous, collaborative,
and constructive
peer-review

TRANSPARENT PEER-REVIEW
Editors and reviewers
acknowledged by name
on published articles

Frontiers
Avenue du Tribunal-Fédéral 34
1005 Lausanne | Switzerland

Visit us: www.frontiersin.org
Contact us: frontiersin.org/about/contact

REPRODUCIBILITY OF RESEARCH
Support open data
and methods to enhance
research reproducibility

DIGITAL PUBLISHING
Articles designed
for optimal readership
across devices

FOLLOW US
@frontiersin

IMPACT METRICS
Advanced article metrics
track visibility across
digital media

EXTENSIVE PROMOTION
Marketing
and promotion
of impactful research

LOOP RESEARCH NETWORK
Our network
increases your
article's readership