

**UNIVERSITY OF DERBY**

**RESEARCH ON DIGITAL IMAGE  
WATERMARK ENCRYPTION BASED  
ON HYPERCHAOS**

**Pianhui Wu**

**Doctor of Philosophy**

**2013**





# **RESEARCH ON DIGITAL IMAGE WATERMARK ENCRYPTION BASED ON HYPERCHAOS**

A thesis submitted in partial fulfillment of the requirements for  
the degree of Doctor of Philosophy

**By**

**Pianhui Wu BSc. MSc.**

**Faculty of Business, Computing and Law**

**University of Derby**

**May 2013**

To my parents

## **Acknowledgements**

I would like to thank sincerely Professor Zhengxu Zhao for his guidance, understanding, patience and most importantly, his friendship during my graduate studies at the University of Derby. His mentorship was paramount in providing a well-round experience consistent with my long-term career goals.

I am grateful to many people in Faculty of Business, Computing and Law at the University of Derby for their support and help.

I would also like to thank my parents, who have given me huge support and encouragement. Their advice is invaluable. An extra special recognition to my sister whose love and aid have made this thesis possible, and my time in Derby a colorful and wonderful experience.

## Glossary

<b>AC</b>	Alternating Current
<b>AES</b>	Advanced Encryption Standard
<b>CCS</b>	Combination Coordinate Space
<b>CWT</b>	Continue Wavelet Transform
<b>BMP</b>	Bit Map
<b>DC</b>	Direct Current
<b>DCT</b>	Discrete Cosine Transform
<b>DWT</b>	Discrete Wavelet Transform
<b>DFT</b>	Discrete Fourier Transform
<b>DES</b>	Data Encryption Standard
<b>FFT</b>	Fast Fourier Transform
<b>GIF</b>	Graphic Interchange Format
<b>HVS</b>	Human Visual System
<b>HAS</b>	Human Auditory System
<b>HH</b>	High, High
<b>HL</b>	High, Low
<b>IPP</b>	Index in Primitive Production Process
<b>ISBN</b>	International Standard Book Number
<b>ISRC</b>	International Standard Recording Code
<b>IDCT</b>	Inverse Discrete Cosines Transform
<b>IF</b>	Intermediated Frequency
<b>IDEA</b>	International Data Encryption Algorithm
<b>IWT</b>	Inversely Wavelet Transformed
<b>JPEG</b>	Joint Photographic Experts Group
<b>LL</b>	Low, Low
<b>LH</b>	Low, High
<b>LSB</b>	Least Significant Bit
<b>MRA</b>	Multiresolution Analysis
<b>MWT</b>	Multiresolution-based Wavelet Transform
<b>MPEG</b>	Moving Picture Experts Group
<b>MPSNR</b>	Multiscale Peak Signal to Noise Ratio
<b>MSA</b>	Multiple Sensor Annunciator
<b>MSE</b>	Mean Square Error
<b>NC</b>	Normalized Cross Correlation
<b>NIST</b>	National Institute of Standards and Technology
<b>RGB</b>	Red Green Blue
<b>RLE</b>	Run Length Encoding
<b>RSA</b>	Rivest, Shamir, Adleman
<b>2D</b>	2-dimensional

<b>3D</b>	3-dimensional
<b>3DACM</b>	3-dimensional Discrete Chaotic Map
<b>SNR</b>	Signal Noise Ratio
<b>TIFF</b>	Tagged Image File Format
<b>PRBS</b>	Pseudo Random Binary Sequence
<b>PSNR</b>	Peak Signal to Noise Ratio
<b>PCX</b>	PC Paintbrush Graphic
<b>VLSI</b>	Very-large-scale Integration



## Nomenclature

$\forall$	The universal quantifier
$\lambda$	The Lyapunov exponent
$\exists$	The existential quantifier.
$\Sigma$	The continued summation.
$v_j$ ,	Pixel of the carried (host) image
$v_j^w$	Pixel of the watermarked image
$\psi(t)$	A basis wavelet or mother wavelet function
$a$	Factor or weight factor
$S_{x,y,z}$	Three-dimensional coordinate boundary control point on the plane of through $x, y, z$ axis
$L^2(\mathbb{R}^2)$	The vector space of measurable
$\bigcup_{j=-\infty}^{\infty}$	Dense $\{\infty, -\infty\}$
$X_p$	The $X$ Lyapunov index of hyperchaotic system
$Y_p$	The $Y$ Lyapunov index of hyperchaotic system
$A_{ij}$	Bit decomposition operator
$C_k$	Wavelet coefficient value
$V_k$	The $No.k$ component weight
$x_i(n+1)$	The random chaotic sequence is generated by various kind of chaotic iteration system at $x$ direction.
$y_j(n+1)$	The random chaotic sequence is generated by various kind of chaotic iteration system at $y$ direction.
$z_k(n+1)$	The random chaotic sequence is generated by various kind of chaotic iteration system at $z$ direction

## **Abstract**

The digital watermarking technique embeds meaningful information into one or more watermark images hidden in one image, in which it is known as a secret carrier. It is difficult for a hacker to extract or remove any hidden watermark from an image, and especially to crack so called digital watermark. The combination of digital watermarking technique and traditional image encryption technique is able to greatly improve anti-hacking capability, which suggests it is a good method for keeping the integrity of the original image. The research works contained in this thesis include:

- (1) A literature review the hyperchaotic watermarking technique is relatively more advantageous, and becomes the main subject in this programme.
- (2) The theoretical foundation of watermarking technologies, including the human visual system (HVS), the color space transform, discrete wavelet transform (DWT), the main watermark embedding algorithms, and the mainstream methods for improving watermark robustness and for evaluating watermark embedding performance.
- (3) The devised hyperchaotic scrambling technique it has been applied to color image watermark that helps to improve the image encryption and anti-cracking capabilities. The experiments in this research prove the robustness and some other advantages of the invented technique.

This thesis focuses on combining the chaotic scrambling and wavelet watermark embedding to achieve a hyperchaotic digital watermark to encrypt digital products, with the human visual system (HVS) and other factors taken into account. This research is of significant importance and has industrial application value.

# Contexts

Acknowledgements	I
Glossary	II
Nomenclature	IV
Abstract	V
Contexts	VI
List of Figures	X
List of Tables	XIV
Chapter 1 Introduction	1
1.1 Background	2
1.2 Research Motivations	3
1.3 Research Problem	11
1.4 Research Objectives	12
1.5 Structure of the Thesis	13
Chapter 2 Literature Review	15
2.1 Chaotic Theory	16
2.1.1 Study of the Origin of Chaos	17
2.1.2 Features of Classically Chaotic Motion	20
2.2 Performance Evaluation of Chaotic Encryption Algorithm	21
2.3 Overview on Cryptography	23
2.3.1 Research of Chaotic Cipher Algorithm	24
2.3.2 Problem with Chaotic Encryption Algorithm	27
2.4 The Embedding Techniques within Spatial and Frequency Domains	32
2.4.1 Basic Encryption of Digital Watermarking	32
2.4.2 Classification of Digital Watermarking	35

2.4.3	Digital Watermarking System	38
2.5	Summary	44
Chapter 3	Methods of Hyperchaotic Scrambling and DWT Embedding	45
3.1	Analysis and Design of Digital Image Scrambling Methods	46
3.1.1	The Components of Image Encryption System	47
3.1.2	System Encryption Flow Chart	50
3.1.3	Framework of Hyperchaotic Image Encryption	51
3.2	Image Encryption Research Based on Frequency Transform	55
3.2.1	Discrete Cosine Transform	56
3.2.2	The Processing Steps for DCT	58
3.2.3	Examination of DCT Watermark Algorithm	62
3.2.4	Algorithms for Watermark Extraction and Detection	65
3.3	Discrete Wavelet Transform	66
3.3.1	Continuous Wavelet Transforms	67
3.3.2	Discrete Wavelet Transform	69
3.3.3	Wavelets in Multi-resolution Analysis	70
3.3.4	Applications of DWT in Image Processing	74
3.4	Detection of Watermarking Encryption	80
3.4.1	Using the Lyapunov Exponent to Identify Chaotic System	81
3.4.2	Evaluation Criterion for Image Encryption	82
3.4.3	Histogram	84
3.4.4	Types of Watermark Attack	85
3.5	Summary	86
Chapter 4	An Encryption Algorithm Based on Improved Henon Map	87
4.1	Analysis and Comparison of Several Chaotic Systems	88
4.1.1	Logistic Map Analysis	89
4.1.2	Arnold Map Analysis	91
4.1.3	Henon Map Analysis	95

4.2	Confirmation of Hyperchaos through Numerical Simulation	100
4.2.1	3D Henon Map and Logistic Map	100
4.2.2	3D Henon Map and 2D Logistic Map	103
4.2.3	3D Henon Map and 2D Arnold Map	105
4.2.4	3D Henon Map and 2D Henon Map	107
4.3	Experimental Analysis	108
4.4	Summary	110
Chapter 5	Experiment of the Hyperchaotic Encryption System	112
5.1	Chaotic Attractor of Overflow Phenomena and Correlation Coefficient Analysis	113
5.1.1	Henon Map of Overflow Phenomena	113
5.1.2	3D Henon mapping Correlation Coefficient Analysis	115
5.2	Hyperchaotic Application and Adjusting Iterations Position	120
5.3	Multigroup Hyperchaotic Sequences in RGB Sub-image	127
5.4	Multigroup Hyperchaotic Sequences in Block-image	132
5.5	Hyper Large Size Image Encryption Algorithm	139
5.6	Summary	140
Chapter 6	DWT-based Information Hiding Algorithm	143
6.1	Watermark Image Implementation Processes	144
6.2	The Implement of Wavelet Algorithm	149
6.3	Evaluation and Improve Implement of Watermarked Image	154
6.4	Experimental Research of Watermark Image Embed and Extract	155
6.5	Summary	163
Chapter 7	Hyperchaotic Encryption Based on Combined Wavelet Transform Algorithm	165
7.1	Combination of Hyperchaotic and DWT Image Encryption	166
7.2	Improved Implementation of Method	169
7.3	Parameters Setting and Attack Test	171

7.3.1	General Watermarking Testing	172
7.3.2	Chaotic Watermark Testing	177
7.4	Summary	180
Chapter 8	Conclusions and Future Work	182
8.1	Summary of the Thesis	183
8.2	Research Contributions	184
8.3	Discussion and Future Work	186
References		187
Appendix A	Input Data for Entropy of the Proposed with the Different Combination of Hyperchaotic Maps	A-1
Appendix B	Large-scale Image Scrambled by Different Hyperchaos	B-1
Appendix C	Combining DWT with Hyperchaotic Encryption Method	C-1

## List of Figures

Figure 1.1	Arnold transform	7
Figure 1.2	Scan and partition patterns	10
Figure 2.1	The pixel value transform scrambling algorithm	28
Figure 2.2	Encryption effects of picture before and after chaos based pixel position scrambling	30
Figure 2.3	The process of watermark embedding	33
Figure 2.4	Watermark extraction processes	34
Figure 2.5	Watermark detection processes	34
Figure 3.1	Schematic flowchart of the encryption system	48
Figure 3.2	System architecture of a hyperchaotic algorithm	49
Figure 3.3	Flowchart of a hyperchaotic encryption and decryption system	50
Figure 3.4	Lorenz system	52
Figure 3.5	Chen system	54
Figure 3.6	JPEG compression standards	57
Figure 3.7	Watermark DCT embedding process	62
Figure 3.8	Logistic map sequences to scramble binary watermark pixels	64
Figure 3.9	Sequence of watermark after $M1 \times M2$ IDCT	64
Figure 3.10	Scrambling of the lena's image and derby logo	65
Figure 3.11	The extraction algorithm	65
Figure 3.12	Low (High) frequency band signal $X(n)$	71
Figure 3.13	The impulse response	71
Figure 3.14	Frequency domain representation of the DWT	72
Figure 3.15	Decomposition schematic diagram of 2D wavelet transforms	76
Figure 3.16	Schematic diagram of 3-layer wavelet decomposition	77
Figure 3.17	Decomposition schematic of 3-Layer wavelet transforms	77
Figure 3.18	1-layer (level 1) wavelet decomposition of hall.jpg	78

Figure 3.19	2-layer (level 2) wavelet decomposition of hall.jpg	78
Figure 3.20	The Daubechies wavelet decomposition	79
Figure 3.21	Some orbits with their Lyapunov exponents	82
Figure 4.1	Logistic map bifurcation diagram	90
Figure 4.2	Phase portrait of the 2D Logistic sequence	90
Figure 4.3	Arnold map	91
Figure 4.4	Iteration property of Arnold map $x(n+1)$	93
Figure 4.5	Iteration property of Arnold map $y(n+1)$	94
Figure 4.6	50 iterations of 3DAM	95
Figure 4.7	Henon map image	96
Figure 4.8	The Lyapunov exponents versus for Henon map	97
Figure 4.9	The mean and variance of 2D Henon map	98
Figure 4.10	3D Henon chaotic attractor	99
Figure 4.11	The mean and variance of 3D Henon map	100
Figure 4.12	Dynamical behaviors of 3DHL system	101
Figure 4.13	The mean and variance of 3DHL system	102
Figure 4.14	Dynamical behaviors of 3DHL system (mean value=0)	103
Figure 4.15	The mean and variance of 3DHL system (mean value =0)	103
Figure 4.16	Dynamical behaviors of 3DH2DL system	104
Figure 4.17	The mean and variance of 3DH2DL system	105
Figure 4.18	Dynamical behaviors of 3DH2DA system	106
Figure 4.19	The mean and variance of 3DH2DA system	106
Figure 4.20	Dynamical behaviors of 3DH2DH system	107
Figure 4.21	The mean and variance of 3DH2DH system	108
Figure 4.22	Key sensitive test in the 3DHLL map	109
Figure 5.1	Initial values of the bifurcate variables 'a' and 'b' for Henon map	114
Figure 5.2	The variables 'a' and 'b' initial value of Henon map	117
Figure 5.3	Image encryption and decryption experimental result	119
Figure 5.4	Combination of three digital matrices	121
Figure 5.5	One type composite hyperchaos of image encryption system	124



Figure 5.6	Encryption of castel.bmp and R, G, B pixel rearrangement	125
Figure 5.7	Castle.bmp recover process	125
Figure 5.8	Image encryption and decryption experimental result	126
Figure 5.9	The three color of coordinate system	128
Figure 5.10	Image encryption and decryption experimental result	130
Figure 5.11	An image divided into two subimages	133
Figure 5.12	Hyperchaotic pixel rearrangement and successful restoration	134
Figure 5.13	Image encryption comparison of 5 blocks matrix	135
Figure 5.14	Result of encryption by using 5 blocks	136
Figure 5.15	Encryption and decryption time comparison of block scrambling algorithm	138
Figure 5.16	The coordinates space in large size image	139
Figure 6.1	The watermark embedding procedure	145
Figure 6.2	An example of the embedding system	146
Figure 6.3	The watermark extraction procedure	147
Figure 6.4	An example of the extraction system	148
Figure 6.5	The image wavelet representations	150
Figure 6.6	Decomposition of 3-layer DWT	156
Figure 6.7	The 3-layer DWT of the watermark embedded into host image pool	158
Figure 6.8	Watermark extracted from image pool	159
Figure 6.9	The 2-layer of DWT embeded into image campus	160
Figure 6.10	IDWT to extract from image gate	160
Figure 6.11	The 2-layer of DWT embedded into image pool	162
Figure 6.12	IDWT to extract from image library	162
Figure 7.1	The encryption process on basis of hyperchaos and DWT	166
Figure 7.2	The scrambling image embed into the host image process	168
Figure 7.3	Result after embedding the scrambling image	170
Figure 7.4	Result after extraction by IDWT	171
Figure 7.5	Watermark embedded image ( $T=0.06$ )	172
Figure 7.6	Result after adding noise	174

Figure 7.7	Detection values and extraction for watermarked image result after rotation 10 °	175
Figure 7.8	Detection values and extraction for watermarked image result after rotation 30 °	175
Figure 7.9	Detection value for watermark image after cropping	176
Figure 7.10	Chaotic watermarking result	177
Figure A-1	Different behaviors in 2D Henon	A-1
Figure A-2	Different behaviors in 3D Henon	A-2
Figure A-3	Different behaviors in 3DHL	A-3
Figure A-4	Different behaviors in 3DHL (mean value=0)	A-4
Figure A-5	Different behaviors in 3DH2DL	A-5
Figure A-6	Different behaviors in 3DH2DA	A-6
Figure A-7	Different behaviors in 3DH2DH	A-7
Figure A-8	Different behaviors in 2DH3DH	A-8
Figure B-1	The two block of library image encryption and decryption	B-1
Figure B-2	Two kinds of chaotic system used in the library image	B-2
Figure B-3	Three kinds of chaotic system used in the castle image	B-3
Figure B-4	The three of kinds chaotic system used in the library image	B-4
Figure B-5	The Four kinds of chaotic system used in castle image	B-5
Figure C-1	Combining DWT with hyperchaotic encryption method	C-1
Figure C-2	Hyperchaotic encryption method	C-2
Figure C-3	DWT to recovery watermark image	C-2

## List of Tables

Table 2.1	Type of chaotic encryption algorithm	25
Table 4.1	Iteration experiments of the Arnold map $x(n+1)$	93
Table 4.2	Iteration experiments of the Arnold map $y(n+1)$	93
Table 4.3	Generalized 3DAM	95
Table 5.1	The 3D Henon sequence generated	115
Table 5.2	Parameter values in chaotic range for the R G,B sub-image at the initial position	122
Table 5.3	Information of the proposed algorithm with different compound of chaotic map	136
Table 7.1	Detection values of watermarked image after Jpeg lossy compression	173
Table 7.2	Detection values results for watermarked images after adding noise	174
Table 7.3	The detection result after crop watermark embedded image	176
Table 7.4	Detection values of chaotic watermarked image after Jpeg lossy compression	178
Table 7.5	Detection values results for chaotic watermarked images after adding noise	178
Table 7.6	Detection values results for chaotic watermarked images after rotation and cropping	179
Table A-1	2D Henon chaotic sequence	A-1
Table A-2	3D Henon chaotic sequence	A-2
Table A-3	3D Henon and 1D Logistic chaotic sequence	A-3
Table A-4	3D Henon and 1D Logistic chaotic sequence (mean value=0)	A-4
Table A-5	3D Henon and 2D Logistic chaotic sequence	A-5
Table A-6	3D Henon and 2D Arnold chaotic sequence	A-6
Table A-7	3D Henon and 2D Henon chaotic sequence	A-7
Table A-8	2D Henon and 3D Henon chaotic sequence	A-8

## **Chapter 1**

---

### **Introduction**

This chapter describes the status of information hiding technology. It will serve as an introduction to the background of this research. The motivation of the research, the research problems of adaptive hyperchaos using the digital image scrambling technologies, and the DWT embedding technology will also be discussed. The chapter finished with in brief introduction of the overall structure of this thesis.

## 1.1 Background

In recent years, with the development of digital communication technology, computer network technology and information compression technology, information dissemination and access have become increasingly convenient and fast. Every user can easily download digital multimedia (such as images, audio, and video) files from the Internet. So piracy and copyright disputes have also become an increasingly serious problem.

Therefore, how to protect copyright and information security effectively has become a pressing practical problem. Traditional encryption techniques can simply guarantee the security of digital multimedia information during transmission; however, there are still some limitations to the protection of the integrity of digital multimedia content and the prevention of unauthorized copying. In this context, a supplementary encryption technology, digital watermarking technology has been proven an effective tool for copyright protection. The copyright protections of digital multimedia information through certification have been rapidly developed based on the watermarking technology. (Graham, 2002)

Digital watermarking, or digital tagging, could be data, serial numbers, characters, image symbols embedded with logo or copyright information, which contains the tag or code of the copyright holder, and can confirm legal ownership of data and other information. Digital watermark can be secretly embedded into digital products to help identify the copyright, digital rights, and the integrity of the contents of these products.

Information hiding is the core for digital watermarking technology. Sometime, watermarking technology is combined with other information encryption method as double secret to enhance security. This method has intangibility and robustness which extract the feature of faults.

The encryption process encrypts meaningful information into randomly garbled, eavesdropping or unlawfully intercepted cipher text, which is very difficult to decipher, because watermarking technology hides meaningful information in the common picture (that can be called the vector information) via a carrier; illegal users are not aware that the common information contains other information; furthermore, it is difficult to extract or remove the hidden information even if they are aware of it.

The basic idea of digital watermark is embedding the secret information into the host, such as image, video, audio, text and software. In the context of protecting the copyright of a digital product, the technology has quickly become a hot concept for addressing the copyright protection problem in the emerging global digital network. A digital watermark needs to be properly changed and embedded into products. Therefore, the study of multimedia security algorithms is an imminent concern.

## **1.2 Research Motivations**

With the growing popularity of network and the development information processing, a high percentage of digital image information transmission in the network are related to personal privacy and confidential secrets. It is undoubtedly important to ensure information security. Data encryption is an important measure to ensure data security in a network environment; however, at this stage, most of the encryption and authentication password system structures are text messages, because digital images involve large amount of data and high redundancy; current encrypting methods have more or less shortcomings. Thus, to ensure secure transmission of images in the network, the new study of digital image encryption method, has become an urgent need. This thesis introduces a new technology that combines the image processing technologies with the modern chaotic cryptography technology.

Through computer networks, it is very convenient for people to work and to have access to study resources; resource sharing has become much easier. But, people have

to face various security problems in the meantime. According to statistics worldwide, a hacking incident takes place nearly every 20 seconds. A variety of security incidents result in the loss of over \$17 billion in the U.S every year. (Chia, 2003) Information hacking incident has increased 250% in the past 5 years, nearly 99% of big companies suffer a variety of hacking incidents, and even RAS securities websites were attacked. (Mark & Horony, 1999)

The complexity and the flexibility of network environment determine the existence of network security threats. Digital images in network transmission are prone to a variety of man-made attacks, including information theft, data tampering, data deletion, and virus attacks, which cause huge losses to the information owners. Therefore, how to ensure information security of the image has become a hot issue.

Cryptographic techniques can be used primarily to protect information security. A sound password system can not only guarantee the confidentiality of information, but is useful to ensure the integrity and authenticity of information. The chaotic encryption technology is an emerging encryption technology that uses chaotic signals generated by chaotic system for encryption, which has fine features of the password such as being a class of random chaos, and extremely sensitive to initial conditions. Thus, the use of chaotic signals can help build a good encryption system. Besides, the chaotic encryption system can be simply implemented and run fast; it is suitable for image encryption with large amount of data. (Lin, 2004)

To ensure the security of digital image information, there are two effective protective measures. The first one is digital watermarking technology, which embeds a watermark in an entire image to effectively protect digital copyright. But this method is not able to change the appearance of an image, which is not suitable for the confidential needs of image. The other method is image encryption. Through encrypting operations of this method, the original image is transformed to information similar to channel random noise, and such random noise is not recognizable to people

who do not have the encryption key. With the fast growth of networking technology, image encryption techniques have good application prospects.

For image encryption, people used to encrypt image by traditional encryption algorithms. Based on the secret keys used in the processes of encryption and decryption, traditional encryption method can be divided into asymmetric-key cryptography (also known as secret key and public-key) and symmetric-key cryptography (Lin *et al.*, 2004). Symmetric-key cryptography algorithm uses the same key for encryption and decryption, and only communicating parties know the secret key, which can encrypt or decrypt message, such as DES (Data Encryption Standard), and IDEA (International Data Encryption Algorithm) (Network Sorcery, 2009). In an asymmetric key encryption, the encryption and the decryption keys are different. Encryption key is known to everyone and only the intended recipient has the key to decrypt the message. The typical algorithms such as RSA (invented by Rivest, Shamir and Adleman) and ElGamal are typical samples of this type. In theory, the image can be encrypted through traditional encryption techniques, but most traditional encryption technologies are based on text design without considering the inherent characteristics of images, thus, using the above approach is not only inefficient but less secure. To solve these problems, many scholars turn to image encryption algorithms. Some of their typical approaches will be introduced in the following:

(1) Arnold transforms

The Arnold mapping is used to produce the chaotic sequence of watermark embedded. The algorithm introduces the correct binary values of computing to embed watermark into deep image of wavelet domain in a low-frequency graph, and watermark detection does not require the original image. (Gunjal, 2010)

The digital scrambling algorithm is based on the Arnold transform, set up the coordinate pixel  $x, y \in S = \{0, 1, 2, \dots, N - 1\}$ , Arnold transformed is:



$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \pmod{1} \quad (1.1)$$

The iterative program can be gained through repeated transformation.

It is a classical application of cryptography—the Caesar Cipher; which is based on the scrambling of position space and color space in digital image scrambling. The resulting image will be encrypted. (Chen, 2006)

Arnold transform can be considered as stretching, compressing, folding and matching processes. Through these processes,  $S$  point in discrete digital image matrix can be rearranged. Discrete digital image is a kind of finite set. The result of such transforms can cause chaotic of position changes of pixels in  $S$  set. But the repeated iterations of such finite set is periodical; therefore, reversion phenomena will occur after finite iterations, which is called Poincare recurrence theorem. (Huang & Xiao, 2009) Thus, if encryption algorithm is obtained, plain text can be recovered by any status of cipher text performing iterations. The type of attack often last very short time, which means the breaches of security and confidentiality exist. On the other hand, there are some shortcomings similar to Arnold transform existing in Hilbert pixel replacement encryption. Although these replacement technologies do not have much cipher features, they can effectively scramble input sequence of plain text, which can effectively cover up statistic characteristics of the plain text in order to resist statistic attacks. Hence, even though the Arnold transform can structure one of replacement parts of space; it is not suitable to be an independent cryptosystem. The figure 1.1 shows the process of the image as it is subjected to the Arnold map. The image as an  $n \times n$  matrix that is made up of the pixels values of the particular image, so each block will be a numerical representation of the corresponding pixel color. The figure 1.1 can see that a unit square is stretched by the linear transform and then folded by the modulo operation, mod. (Chen, 2004)



Figure 1.1 Arnold transform

(Chen, 2004)

## (2) Magic square matrix

Based on the magic square matrix of image encryption is commonly used for matrix transformation or pixel replacement encryption. The following is a magic square matrix:

For one of  $N$  order square matrix  $M = \{m_{ij}\}$ ,  $i, j \cdots N$ , if it satisfies the equations:

$$a. \quad \forall k \in \{1, 2, \dots, n\} \text{ has } \sum_{i=1}^n m_{ik} = \sum_{j=1}^n m_{jk} = C \quad (1.2)$$

(C is constant, C determined by n)

$$b. \quad \sum_{i=j} m_{ij} = \sum_{i+j-1=n} m_{ij} = C \quad (1.3)$$

$m_{ij} \in \{1, 2, \dots, n \times n\}$ ,  $M$  is  $n$ -order magic square, or the magic squares, where  $c$  is  $c = \frac{n(n^2 + 1)}{2}$ .

From digital image scrambling based on magic square, we let digital image be

the  $n$ th-order matrix  $B$ . For an  $n$ th-order magic square  $A$ , we construct matrix  $B$  in correlation to matrix  $A$  by each column and row. It moves element 1 in  $A$  to the position of element 2, and element 2 to the position of element 3. After this regularity, element  $n^2$  moves element  $m+1$ , and it will generate  $k$  times of iteration scrambling transforms. As the magic square matrix is a finite-dimensional matrix, after  $n^2$  times replacement, the scrambled picture will return to its original states. Therefore, we can also use the above-mentioned methods to decrypt.

Like the Arnold transform, Magic square transform has closed encryption algorithm, because its encryption key and algorithm are not actually separated. Once an attacker learns of the encryption algorithm, deciphering encrypted information is not complex. The scrambling of one single method is sometimes unable to achieve ideal scrambling effects. To achieve satisfied scrambling effects in practice, scrambling transform is normally processed in multiple steps, i.e. scrambling image by modules first and then scrambling rows in each image module. Dividing  $256 \times 256$  image into  $32 \times 32$  image module results in  $8 \times 8 = 64$  image modules, which will be further scrambled by using  $8 \times 8$  magic square matrix to get rid of the correlation of image pixels in neighboring domain space, and then the elements between all the way of rows and columns are transferred. Such a method is better than the single encryption method. However, it is far from ideal in practice. (Kachris, 2003)

### (3) Pseudo random sequence of encryption technology

Schwartz C (Schwartz, 1991) defined a pseudo-random sequence based encryption technology. Its working process is that a pseudo-sequence generator creates a set of pseudo sequence points in an image value range, and then connects the point generated by the sequence, and reversed out image pixels that the connection link has been through, so that white pixels become black pixels

or black pixels becomes white pixels in contrast. Such an operation is performed for binary images; but for gradation images, it can perform bit-plane decomposition first. In this perspective, this algorithm is suitable for gradation image; encryption key is the seed for generating pseudo-sequence. This encryption runs simple and fast, but the security is not high because the original image is not compressed.

#### (4) SCAN language encryption

SCAN language is an efficient two-dimensional spatial data access technology. (Kachris, 2003) It can conveniently generate massive scanning paths or space-filling curves, which can transform two-dimensional image data to one-dimensional data sequence, and then use different scanning word to represent different image cipher text. Commercial encryption algorithms such as DES and IDEA are two example algorithms. This type of encryption algorithm does not compress data for original image data, but scanned text words can be compressed without losses. In general, this algorithm is normally used to process massive data, although lossless compression is applied for scanning words. Under normal circumstances, compression ratio is not high; it can only utilize the convenience that SCAN language transforms 2D data to 1D data. Data security depends on common commercial cipher. During the encryption process, 2D data should be transformed to 1D data first; thus, pretreatment window time is needed, decrypted data needs to be rearranged, and the efficiency is not high. Like RSA, DES and AES algorithms, the SCAN algorithm is also one of block encryption. Compared with streaming encryption, block encryption is a direct method, but the biggest problem is that it requires as small a secret key as possible, so that memory space and operation time can be minimized. (Kachris, 2003) SCAN language encryption is classified as three types; they are scan patterns, partition patterns and mixed patterns. Let's have a look at an example below (figure 1.2).



Figure 1.2 Scan and partition patterns

(Kachris, 2003)

(5) The ‘cipher-image’ of the encryption technology

The ‘cipher-image’ encryption is suitable in the two-dimensional signal, its characteristic have two merits amplitude and phase. They are very important in the encryption technology. Therefore, the encryption operation is to choose another same size image (as a ‘cipher-images’) phase spectrum combined with the original image phase spectrum, then get a new phase spectrum of cipher-image. Because of the ‘cipher-image’ is totally random, it can be ‘one-time one-key’, so it is higher security. But the encryption does not require data compress that not only the processing time is long, but also it will be burden to image transform. Therefore, the cipher-image is not very convenient to use. (Refregier, 1995) (Yamazaki, 2001)

(6) Chaotic encryption techniques

Since 1992, chaotic secure communication has evolved for generations. Chaotic masking and chaotic shift keying belongs to the first generation of chaotic secure communication technology; the security is not strong, and its practicability is greatly retarded. Chaotic modulation is the second generation of

chaotic secure communication technology. Although the safety of the second generation is better than the first, it still falls short of customer satisfaction. The third generation is chaotic encryption technology. This method combines the benefits of chaos and cryptography, and it has a very high safety performance. It is followed by synchronization pulse of chaotic communication, which is the fourth generation of chaotic secure communication. (Salleh, 2002)

### **1.3 Research Problem**

In order to design hyperchaotic scrambling system for digital images, the following problems will be addressed in this research work.

- (1) What's algorithm suitable to scrambling.

The lack of robustness and security are important problem for secret key, but the development of communication technology requires, these problems will not solve. We found single chaotic system is simple than hyperchaotic system. A single chaotic system is easy to implement; because of this, it can easily be cracked, as well as limited by the computer word length. The scrambling image will return to its original state that after a certain number of iterations from the single chaotic system. The hyperchaotic system has complex characteristics and unpredictable.

- (2) How to improve algorithm for digital image encryption.

The hyperchaotic encryption speed is slower than simple chaotic. For now, the development of technology depends more and more on the digital image and high resolution photo are required for sharp reproduction in our life. The encryption process can take a long time or outage. Therefore, building a highly agile control system that hyperchaotic algorithm is used to improve secret key

space, randomly and encryption speed. Chapter 5 will illustrate this problem and make specific improvements.

- (3) What's better technology to hidden image.

Watermark technology is to require the hidden image is not destroyed original image (host image) by the presence of watermark. The hidden image is also maintained after fusion (that is robust), in normal the carrier image should be lossy. In order to prevent these thing happen, we used wavelet transforms. The embedded algorithm should be as high as possible to improve the anti-attack capability of image.

- (4) How to test the performance evaluation of proposed algorithms

Based on these results, performance evaluation of proposed encryption algorithms are the chaotic scrambling and the watermark encryption. By comparing the analysis result with the bit error rate and normalized correlation for the scrambled image and original image, and the extracted image and host image.

## **1.4 Research Objectives**

A wide range of applied multimedia delivery in the future network is becoming increasingly prevalent. Information security problems have also become increasingly prominent especially image cryptography for secure transmission. However, traditional encryption methods for digital image are limited. Therefore, it is necessary to explore new features for image encryption. Chaotic encryption is a new technology that develops rapidly in recent years. The technology, with its features of simplicity, fast encryption, and high security, is very suitable for image encryption. However, due to short development time, it is still not perfect and needs to be further studied. This

major work is divided into studying the origin of chaos and conducting simulation algorithm. Through analyzing and summarizing, the pros and cons for various algorithms can be obtained. Based on the modern cryptography, a more multi-dimensional chaotic image encryption system is designed; in-depth exposition of some key issues on the system will be made, such as achieving multi-dimensional chaotic system and specific steps of the encryption process, which confirms that it is combined with wavelet embedded image.

## **1.5 Structure of the Thesis**

This thesis is divided into eight chapters, briefly stated as follows.

Chapter 1 introduces the background of the information hiding technologies, the motivations of the research, the research problems, objectives, methodology and the structure of the thesis.

Chapter 2 presents a literature review of chaotic theories, chaotic cipher algorithms, chaotic encryption algorithm problems and the technical characteristics of digital watermarking technology (based on the space domain and frequency domain).

Chapter 3 discusses a more accurate method, reflects every phase of a complex chaotic sequence, and compares the randomness and complexity among several common chaotic sequences. The hyperchaotic system is used and evaluated in this scrambling system. Another embedding technique is Discrete Wavelet Transform (DWT).

Chapter 4 proposes to encrypt images based on a hyperchaotic system. Firstly, it reviews Lorenz map, Logistic map, Arnold map and Henon map. Secondly, it adopts a two-dimensional or three-dimensional Henon map to generate data series, and disturbs them by Lorenz map, Logistic map or Arnold map to generate different kinds



of stream ciphers. Moreover, it analyses their relations and parameters by means of the Lyapunov exponents and histograms.

Chapter 5 elaborates on chaotic attractor overflow in chaotic encryption and the image encryption influenced by the research into multi-dimensional Cartesian coordinate system; it overcomes the overflow phenomenon in chaotic image encryption and is applied in hyper large-size image encryption algorithm.

Chapter 6 focuses on application of the DWT information hiding algorithm. The algorithm designs image transform domain, and is a wavelet transform algorithm for image embedding. Firstly, it transforms the image format; secondly, it uses DWT and HVS to achieve good performance. The approach can overcome existing limitations of security and robustness. Theoretical analysis and experimental results reveal that it has strong immunity against the common attacks.

Chapter 7 deals with the full technical aspects to get a more versatile self-adaptive image encryption algorithm. Based on hyperchaotic Henon map and wavelet transform, the algorithm is proposed in order to improve the watermark security and attack resistance. Experiment results demonstrate that this scheme is robust against many common attacks such as noise, *JPEG* compression, and cropping.

Chapter 8 summarizes the studies on the DWT embedding and discusses its potential in aiding or controlling the hyperchaotic scrambling technology. The thesis ends with a suggestion on some fields of further work.

## **Chapter 2**

---

### **Literature Review**

The chapter presents a chaotic scrambling system and watermark embedding techniques. The chaotic scrambling system provides a knowledge base for developing a hyperchaotic system of 2D image. The hyperchaotic system is used and evaluated in this scrambling system. Another embedding technique is DWT (Discrete Wavelet Transform).

## 2.1 Chaotic Theory

Chaos is a complex non-linear non-equilibrium dynamics process, characterized as follows: (1) the chaotic system behavior is a collection of many orderly acts and each component does not play a leading role under normal conditions; (2) Chaos reflects randomness, and hence unpredictability; (3) it is sensitive to dependence on initial condition. Even with two identical chaotic systems, if they are in two slightly different initial states, they will rapidly evolve toward totally different states. (Batterman, 1993)

The close relationship between chaotic systems and cryptography is widely used in the field of information security. Such technologies as information encrypted, secure communications, smart card encryption, and digital watermarking all have shadow of the chaotic encryption technology. In traditional method of confidential communications, information signal is modulated and demodulated by sine carrier signal. The other one is new secure chaotic communication system, where the transmitter sends carrier information signal and the receiver demodulates it to resume the signal. The effectiveness of chaotic confidential communication comes from its own random nature of broadband carrier. The information signal will be similar to channel noise signal; crackers would think the channel noise signal was noise. So the chaos theory and cryptography can effectively guarantee the information security. By now, the chaos theory has been effectively applied in the information security field; many chaotic watermarking technologies have achieved satisfactory performance. (Bianco, 1991)

There are two major research directions: the first one is synchronization technology of chaotic system for secure communication (Frank & Shih, 2002), mainly based on chaotic circuit simulation system; the second one uses chaotic systems to generate chaotic stream cipher and block cipher (Uccheddu, 2004) (Gilani *et al.*, 2002) (Jang, 2001), mainly based on computer's limited accuracy to

achieve the digital chaotic system (Wankim, 2007).

As people studied chaotic systems in depth, they began to make the new password chaotic, which has much greater impact: Based on the research into chaotic passwords were proposed such as *S*-boxes in block cipher, chaotic block ciphers, and public key encryption technique based on multiple chaotic system, etc. (Wang *et al.*, 2005). Although performance of these encryption methods has to withstand the test of time; these new methods will provide a new way for design of chaotic encryption password system.

### **2.1.1 Study of the Origin of Chaos**

Because a chaotic system is sensitive to initial conditions, small errors in some initial values errors can be enlarged by system. Therefore, the system development in anticipating is unpredictable; also because chaotic sequence has good statistical properties, it can generate random numbers. These characteristics are well suited to sequence encryption technology. According to the information theory of mathematician Shannon's when some methods are used to produce a random sequence, the sequence can only be identified by the key. Any input of a slight change may lead to a significant effect, thus the sequence can be encrypted. (Shannon, 1948) A chaotic system can precisely meet such requirement.

Owing to the characteristics of a chaotic system, its numerical distribution does not conform to the principle of probability statistics and no stable probability distribution characteristic can be obtained. In addition, a chaotic number set is real valued, and can be extended to the complex value domain. Therefore, employing the principle of the chaos theory to encrypt the data can guard against attacks like frequency and exhaustive attack, so that it is difficult to analyze and decipher the password.

There are two kinds of ideas in the image encryption technology: space domain encryption and frequency domain encryption. Whether with space domain encryption or frequency domain encryption algorithm, they all entail three methods of thinking: alternation of gray value, transformation of the location of pixels, and a combination of both. The alteration of gray value requires a key to change the gray value of the original image; you can change it point by point. Also the original image can be divided into a few blocks, each block goes through replacement. The transformation of the location of pixels has changed the order of the pixels of the original image. (Fridrich, 1998) Fridrich used a chaotic map to change the pixel location, a combination of methods which has not only alternation of gray value, but also the transformation of the location of pixels in the encryption process. (Chuang, 1999) In the entire encryption algorithm, we need a random number generator. As discrete chaotic system is easy to achieve, it can be used for neural network that has good statistical properties and serve as random number generator. The parameters of a chaotic system are extremely sensitive to initial conditions; they can be used as a key in a good password system. The chaotic system in two-dimensional plane of the irregular nature makes it more suitable for image encryption. (Li, 2003) (Fridrich, 1998)

In 1963 the U.S. meteorologist Edward N. Lorenz announced a stunning discovery—the chaos theory. He discovered that weather is unpredictable. Small changes in temperature at the poles or the equator will result in dramatic changes of the climate. This is called ‘the butterfly effect’. The chaos theory has been used in various fields. During the 20<sup>th</sup> century, the chaotic dynamics has been widely used and developed. (Yen, 1998) (Fridrich, 1998)

(1) Li-Yorker’s chaotic

Mathematicians Kloeden concluded the definition of chaotic on the basis of the Li-Yorker’s theorem; Presume  $f(p)$  is continuous map from  $I$  interval to

itself. If it meets the following conditions (Li & Yorke, 1975):

- a.  $f(p)$  of  $s$  (cycle point) is unbounded.
- b. There exists an interval  $I$  which is uncountable to subset  $S$ , to satisfy:  
For every  $p_s, q_s \in s$ , with  $p_s \neq q_s$ ,

$$\limsup_{n \rightarrow \infty} |f^n(p_s) - f^n(q_s)| > 0 \quad (2.1)$$

and

$$\liminf_{n \rightarrow \infty} |f^n(p_s) - f^n(q_s)| = 0 \quad (2.2)$$

Form  $f(p)$  can be used to describe the system as a chaotic system.

In this definition of two initial points forms two tracks that are sometimes unlimited and close, sometimes far away for each other. The system is not regularity, but a random phenomenon. (Michael, 2007)

## (2) Melnikov's definition of chaotic

Melnikov's definition of chaos has been to high dimensional dynamical system. The Smale horseshoe map  $f$  is defined in subset  $D$ , and it is compared with the systems under an external periodic excitation, under a linear periodic parametric excitation, or under a nonlinear periodic excitation. Henon map is one example of Smale horseshoe. And it has been proved that a horseshoe map is the intersection of two Cantor invariant sets in chaotic state. Holmes cited the Melnikov method and found a computable formula for the distance between the stable and unstable manifolds of the hyperbolic periodic orbit, and then there must be chaos. Melnikov presented the method allowing an explicit determination of transverse intersection of these manifolds, but this method is only suitable for Hamilton system. (Melnikov, 1963) (Guckenheimer & Holmes, 1990)

## (3) Devaney's definition of chaotic

Devaney's definition of chaotic (Banks, 1992) defines: if  $V$  is a set of metric space, that is, for all non-empty open subsets  $x$  and  $y$  of  $V$ , there is a continuous map  $f : V \rightarrow V$ , if satisfied the Devaney's three conditions,  $f$  can be treated as  $V$  chaotic.

- a. ' $f$ ' has sensitive dependence on initial conditions, if  $\delta > 0$ , there is an every neighborhood  $\varepsilon > 0$  and  $x \in V, x$  in  $I, y$  and  $n$ , there is

$$d(f^n(x) - f^n(y)) > \delta \quad (2.3)$$

- b. ' $f$ ' is transitive; for  $X, Y \subset V$ , there exists a natural number  $k > 0$ , that is

$$f^k(X) \cap Y \neq \emptyset \quad (2.4)$$

- c. The periodic points of ' $f$ ' are dense in  $V$ , there is  $\forall x \in V, \forall \varepsilon > 0$ , then  $y \in T$  will be get

$$y - x < \varepsilon \quad (2.5)$$

(Strogatz, 1994) (Kanhng, 2009)

### 2.1.2 Features of Classically Chaotic Motion

Compared with linear systems and other nonlinear systems, the chaotic system has its own characteristics. (Salleh, 2002) The main features are:

- (1) Properties of being bounded: Chaotic is bounded; its movement is always been limited to a definite region, which is known as chaotic domain;
- (2) Periodicity: The chaotic motion is ergodic in its chaotic attractor domain, or

chaotic in a finite time, where the chaotic orbit passes every state point in the chaotic region, but will never stay at a certain state point;

- (3) Randomness: Under certain conditions, the situation may change in a qualitative way or make the system chaotic but a status may arise or may not appear;
- (4) Initial value of the sensitivity: Chaos is easy to recognize using initial-value of the sensitivity. With the passage of time, any two adjacent initial conditions will show itself independent of the time. There is sensitive dependence on initial conditions everywhere, e.g. in the famous ‘butterfly effect’ theory;
- (5) Long-term unpredictability (Jakimoski & Kocarev, 2001): Because initial conditions are confirmed to a limited precision, the initial conditions of small differences could have enormous consequences, so it is not possible to predict long-term future of a chaotic system at the dynamics;
- (6) Universal application: The so-called universal application refers to demonstrate certain common characteristics for different systems when they approach the chaotic state. They would not be affected by specific parameters of the system or the equation to change;
- (7) Fractal dimension: It is a geometric feature of a point set in n-dimensional space. It has the infinitely divisible structure.

## **2.2 Performance Evaluation of Chaotic Encryption Algorithm**

According to some studies of pseudorandom bit generator using the statistical test suite of the U.S. National Institute of Standards and Technology (NIST), a chaotic encryption algorithm should be divided into three parts: security, costs and applicability. (Andrew *et al.*, 1997) They introduce a Lorenz chaotic encryption



algorithm, and set a standard of encryption algorithm, to compare the current encryption algorithms.

(1) Security analysis

First of all, the chaotic system is very sensitive to the parameters and the initial values; it can provide a large key collection to fully satisfy the meet for the encryption. Through examining the binary sequence generated by the chaotic system, the distribution of 0 and 1, when the run-length meets the requirement of random sequence, it can be treated as random sequence. Secondly, chaotic encryption is a kind of stream encryption; so it invulnerable to attacking methods that aim at grouped encryption. Besides, for attacking methods aiming to decipher text, it is not impossible to guess the key through XOR operation because of the unidirectional and iterative nature of chaotic signals.

(2) Cost analysis

Algorithm cost includes time cost and spacing cost. Time cost includes time of preparation and time of encryption. Usually, preparation time prior to the encryption is used to create sub-key encryption; encryption time is mainly used for the exchange of deciphered text data under encryption control. As a kind of stream encryption, chaotic encryption has short preparation time. Since XOR operation is carried out each bit of data, time is mainly spent on the generation of key stream. Compared with current popular grouped encryption algorithms, it takes much less time. Run-state space refers to the temporary space needed by the algorithm during the encryption process. Because chaotic encryption algorithm has no *S*-box spaces, few temporary variables, and it is able to generate key stream by cycling, which requires less space to store variables, therefore, it occupies less space during run-time,

which suggests it is outstanding on spacing costs.

### (3) Applicability

The encryption and decryption processes of chaotic encryption algorithm can be reused; thus, the occupied space is compressed. It has good applicability to both hardware and software of information security. The algorithm has been implemented by using C++ and JAVA. DSP is designed and developed based on such an algorithm.

## 2.3 Overview on Cryptography

The development of cryptography can be divided into two phase: traditional cryptography and modern cryptography:

Traditional cryptography was founded before the 1950s. Generally, most of such encryption methods are simple because they used individual letters in the encrypted text. It is easy to encrypt and decrypt manually or mechanically. Because of this, it is seldom used now. In 1949, Shannon published ‘communication theory of secrecy’ that has laid a theoretical foundation for modern cryptography. Modern cryptography mainly consists of two kinds of cryptosystems: the symmetric cryptosystem and asymmetric cryptosystem. An encryption system is composed of five parts including plain text, cipher-text, encryption algorithm, secret key and decryption algorithm. Symmetric cryptography can be divided into two types: stream ciphers and block ciphers. The former includes LFSR, SEAL, and RC4. The latter includes IDEA, AES, DES and RAS.

In cryptanalysis, the generally accepted Kerckhoff criteria assume that the attacker knows all the other details of the cryptosystem except the secret ciphers. Therefore,

the cipher attack methods can be divided into five categories:

- (1) Cipher-text attack: The attacker has known the cipher-text;
- (2) Known plaintext attack: The cryptanalyst has known the cipher-text, using the same cipher to decrypt the one or more cipher-texts;
- (3) Selective plaintext attack: The cryptanalyst has known the cipher-text, arbitrarily chosen some plaintext as the cipher and used it to encrypt corresponding cipher-text;
- (4) Selective cipher-text attack: The cryptanalyst has known the cipher-text, purposefully selects of some cipher-text and uses it to decrypt the corresponding plaintext;
- (5) Selected text attack: The cryptanalyst has known as the cipher-text, arbitrarily chooses some plain-text, and uses it to decrypt the cipher-text, then purposefully selects the decrypted cipher-text and uses it to decrypt the corresponding plain-text.

With the rapid development of computing technology, cryptanalysis techniques become more sophisticated, some classic cryptographic algorithms have been cracked.

### **2.3.1 Research of Chaotic Cipher Algorithm**

There are some basic properties in chaotic systems, such as certainty, ergodicity, and randomness. Combined with the concepts of confusion and spread of cryptography, these properties can be used to design cryptography algorithm, which forms chaotic cryptography. Chaotic cryptography is a multidisciplinary

science that spans various fields, including chaotic theory, conventional cryptography, communication engineering, and information processing. Chaotic cryptography has two basic forms: the one is the simulation of chaotic security system that uses analog circuits and synchronization technology; the other one is achieved by digital circuits or computer digital chaotic cryptography, but it is not linked with chaotic synchronization. Studies of cryptographic systems show that most chaotic synchronizations of the communication systems have many flaws and breaches. Most of the proposed algorithms are single chaotic system; currently, there are some combinations of chaotic cryptography, called hyperchaotic system.

Table 2.1 Type of chaotic map

Map	Time domain	Space domain	Number of space dimensions	Also known as
Arnold	discrete	real	2	
Baker	discrete	real	2	
Bogdanov map				
Chossat-Golubitsky symmetry				
Circle				
Complex quadratic				
Complex squaring				
Complex Cubic				
Degenerate Double Rotor map				
Double Rotor map				
Duffing map	discrete	real	2	
Duffing equation	continuous	real	1	
Dyadic transformation	discrete	real	1	2x mod 1 map, Bernoulli map, doubling map, sawtooth map
Exponential map	discrete	complex	2	
Gauss map	discrete	real	1	mouse map, Gaussian map
Generalized Baker map				
Gingerbreadman map	discrete	real	2	

Gumowski/Mira map				
Henon map	discrete	real	3	
Hitzl-Zele map				
Horseshoe map	discrete	real	2	
Ikeda map	discrete	real	2	
Interval exchange	discrete	real	1	
Kaplan-Yorke map	discrete	real	2	
Linear map on unit square				
Logistic map	discrete	real	1	
Lorenz attractor	continuous	real	3	
Lorenz system's				
Poincare Return map				
Lozi map	discrete	real	2	
Nordmark truncated				
Pomeau-Manneville maps for intermittent chaotic	discrete	real	1 and 2	Normal-form maps for intermittency (Types I, II and III)
Rabinovich-Fabrikant equations	continuous	real	3	
Random Rotate map				
Rössler map	continuous	real	3	
Shobu-Ose-Mori piecewise-linear map	discrete	real	1	piecewise-linear approximation for Pomeau-Manneville Type I map
Standard map, Kicked rotor	discrete	real	2	Chirikov standard map, Chirikov-Taylor map
Tent map	discrete	real	1	
Tinkerbell map	discrete	real	2	
Van der Pol oscillator	continuous	real	1	
Zaslavskii map	discrete	real	2	
Zaslavskii rotation				

The table 2.1 shows the relevant portion of chaos map, as well as list time domain, space domain and number of space dimensions. (Argyris *et al.*, 1994)(Arnold, 1998)(Patrick *et al.*, 2003)

### 2.3.2 Problem with Chaotic Encryption Algorithm

There are three major problems with chaotic encryption algorithms.

(1) Cycling term response

The evaluation of cyclicity, pseudo-randomness, complexity of and correlation between the generated sequences through existing chaotic studies is based on statistical analyses or derived from experimental tests. It hardly ensures big enough cycling term for each realization of sequence and high enough complexity for safe use in cryptography. For example, in self-control mode, when the input signal is zero, the presence of the encryption is limited to cyclic term of response. Different initial states correspond to different cyclic terms, which are much shorter; this shortness somehow decreases the security of chaotic encryption systems.

(2) Limited precision effect

The chaotic sequence is always generated by devices with limited precision; a chaotic sequence generator can be defined as a limited automated device. Therefore, it is quite worthwhile to study whether or not the chaotic generator can override existing massive research outcomes yielded by limited automation and Boolean logical theory warrants careful study. (Sikorski, 1960) In most cases, the characteristic of a chaotic system implemented with finite precision are much different from its theoretical results, which can render many chaos-based applications unfeasible. Some researchers even believe that limited precision effects have become one of the difficulties in industrial application.

(3) The conflict between implementing precision and confidentiality

For the piecewise chaotic mapping encryption system, two neighboring states

may fall onto one same linear segment. In the case of high precision digital implementation, a decryptor may use this characteristic to easily recover enough a precise enough encryption key after matching a relative amount of plaintext and cipher text. In other words, it is vulnerable to selected plain-express text attack, which suggests it has no confidentiality in this case. With the development of and research into chaotic encryption technology, the difficulties will be resolved after, and chaotic encryption technology will produce valuable and useful applications.

Currently, chaotic image encryption algorithm can be divided into two types: pixel value transform and pixel position transform. In image pixel transform the chaotic system is treated as a pseudo sequence generator. The sequence generator performs certain operation with plain text, which results in cipher text. It achieves encryption by modifying the image's pixel value. In pixel coordinate transform, however, image pixel coordinate is changed by random chaos. The chaotic sequence structures a chaotic scrambling matrix, which is used to scramble the image's pixel positions. (Dittmann *et al.*, 1998)

By chaotic transform dimensions, chaotic image encryption algorithm can be divided into one-dimensional or multiple dimensional chaotic encryption algorithms. Besides, chaotic based encryption algorithms can be categorized as spatial space encryption algorithm and frequency encryption algorithm.

a. Pixel value transform



Figure 2.1 Pixel value transform scrambling algorithm

Suppose  $f$  represents an image with size  $M \times N$  and with  $L$  gradation layers. Selecting an one-dimensional chaotic system, and presume the initial value

$x(0)$  is encryption key, chaotic sequence  $\{x(i)\}$  is created by this chaotic system. The original image pixel point  $f(x,y)$  ( $0 \leq x \leq M-1$ ,  $0 \leq y \leq N-1$ ) corresponds to pixel value  $f'(x,y)$  of the encrypted image. By performing related operations with pixel values, because chaotic sequence has pseudo randomness, which makes the transformed image to appear pseudo-random, encryption is achieved. For decryption, sequence  $\{b(i)\}$  is needed to perform EOR operation with pixel values of encrypted image. Figure 2.1 shows two images before and after encryption. It turns out that the encrypted image cannot be recognized. (Zhu, 2006)

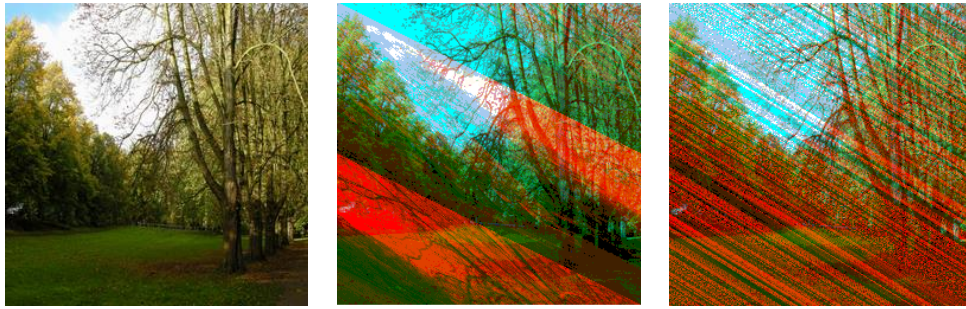
b. Pixel position transform

Compared with chaotic pixel scrambling algorithm, the implementation of pixel position scrambling algorithm is more complex. It uses two dimensional chaotic mapping to map original coordinate  $(x, y)$  of each pixel to new coordinate  $(x', y')$  in order to generate an encrypted image.

In pixel position scrambling algorithm, suppose the set  $f$  represents an image with the size of  $M \times N$ . Select two chaotic system and generate two chaotic sequences with presume with initial values  $x(0)$  and  $y(0)$ . Set the values of the chaotic sequence  $\{x(i)\}$  as integers at the range between  $[0, M-1]$  and  $[0, N-1]$ , respectively. Further process the two sequences to yield the ergodic  $[0, M-1]$  sequence  $\{x(i)\}_{i=0}^{M-1}$  and ergodic  $[0, N-1]$  sequence  $\{y(j)\}_{j=0}^{N-1}$ , and use these two sequences as rows and columns of scrambling matrix. The relationship between the scrambled image matrix and that of the original image is:

$$f'(i, j) = f(x(i), y(j)) \quad (1 \leq i \leq M, 1 \leq j \leq N) \quad (2.6)$$





(a)Original image (b) Effect after scrambling once (c) Effect after scrambling 9 times

Figure 2.2 Encryption effects of picture before and after chaos based pixel position scrambling

Figure 2.2 shows the encryption effects of pictures before and after chaos based pixel position scrambling, where (a) is the original image before encryption, (b) is scrambled image after the 1st scramble, and (c) is the resulted picture after scrambling of 9 times, the original image can be obscurely seen in (b) but it is completely unrecognizable in (c). For encryption effects, this algorithm is much better than the previous ones.

Because this algorithm employs chaos for encryption, it is simple and fast, and easy to implement. But chaotic sequence required by the encryption needs to be generated on the PC, and computer calculation has the limitation of accuracy, which makes the generated chaotic sequence periodical and some chaotic sequence period is very short. Therefore, for preventing cipher attacks against plain text and cipher text, this algorithm is not reliable.

Theoretically, chaotic sequence has good encryption features, such as class randomness, relativity and sensitivity to initial values, but in fact, due to the accuracy limitation of computer operation, chaotic sequence is normally periodical, which may greatly affect the security of chaotic encryption, and even cause chaotic encryption to fail. To solve this problem, it is more important to generate better numerical values with typical chaotic feature in limited word length of computers.

Today, most chaotic encryption algorithms are simple image pixel scrambling or image pixel position scrambling, but simply adopting either of them cannot ensure high security needs. Image pixel scrambling has good resistance against attacks on cipher text only but it is weak to resist known plain text attacks; image pixel position scrambling can only change the position of the image pixel, but cannot modify the image pixel values; thus, an attacker cannot possibly breach this encryption algorithm by pixel comparison.

Currently, most studies of chaotic encryption technology are based on one dimension and two dimensions. Some studies suggest that the confidentiality of low dimension is not good enough. Many chaotic encryption solutions with low dimension have been proposed in the past few years, but they all have more or less shortcomings of confidentiality. It is generally believed that many of current chaotic encryption solutions have security breaches. To enhance the security, the following methods can be considered:

- (1) Design quasi-chaotic sequences with long period. Thus, developing a multi-dimensional chaotic encryption system is vital to designing a robust chaotic encryption system, and designing high quality chaotic sequences is crucial for developing a high quality chaotic encryption system. (Chen, 1997)
  
- (2) Combine pixel scrambling encryption method with pixel position encryption method to encrypt image step by step. Because both image pixel positions and pixel values are scrambled in the encryption process, it greatly increases the difficulty of decrypting a cipher, which suggests the security is enhanced. Because of the better security performance of this encryption algorithm, it is suitable for adoption in situations where security requirements are higher. Besides, based on further studies of chaotic theory and cryptography, chaotic encryption algorithm combined with traditional encryption algorithms can work together and overcome major shortcomings of each other, which makes

an improved algorithm. But since an encryption process runs step by step, this combined operation load may be slightly increased, which can slow down the speeds of encryption and decryption.

- (3) Combine encryption manner and data hiding manner. In recent years, image encryption algorithms have developed rapidly; in the meantime, a variety of attacks against encrypted images are being evolved. Scrambling an image and transferring the unrecognizable image over the internet may contrarily expose the importance of the image, which can greatly increase the possibility of attracting malicious attacks. When combined with data hiding technology, transfer of the image over the Internet will be much safer.

Through the studies of a variety of chaotic systems, a set of experiments of variety chaotic systems will be introduced in next chapter, and image encryption designs of new mult-dimensional chaotic systems will be shown in the next chapter.

## **2.4 The Embedding Techniques within Spatial and Frequency Domains**

In this section, two major aspects of the digital watermarking technology will be introduced: basic model and classification.

### **2.4.1 Basic Encryption of Digital Watermarking**

From image processing perspective, the embedded watermark signal can be treated as a weak signal superimposed in a strong background, as long as such superimposition of the watermark signal strength is below the human visual system (HVS) of the contrast threshold, that is, the existence of the signal is undetectable for humans. The contrast threshold is affected by the visual system of the space, time and frequency characteristics. Therefore, by adjusting the original image, it is possible to embed some information without changes in visual effects.

From digital communication perspective, watermark embedding can be understood as a narrow-band transmission signal transferred on a broadband channel (carrier image) by spread-spectrum communication technology. Although the watermark signal has certain energy, the energy distributed on any channel is hardly extracted to be detected. Watermark decoding (extraction of detection) is used to detect a weak signal from a noisy channel.

Suppose the original image is ' $I$ ', the watermark signal is ' $W$ ', the encryption key is ' $K$ ', and the embedding algorithm is ' $F$ ', the watermark embedding can be presented by equation (2.3) and equation(2.7):

$$I_w = F(I, W, K) \quad (2.7)$$

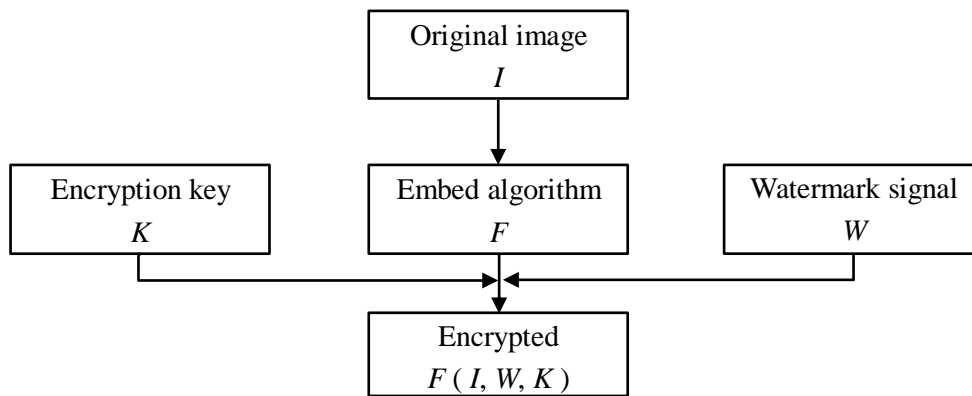


Figure 2.3 The process of Watermark embedding

Equation (2.8) shows the commonly used embedding formulas are:

$$\begin{aligned} v_j^w &= v_j + aw_j \\ v_j^w &= v_j(1 + aw_j) \end{aligned} \quad (2.8)$$

where  $v_j$  and  $v_j^w$  represents pixel of the carrier image and pixel of the embedded watermark image, respectively. And  $w_j$  is watermark signal component,  $0 \leq j \leq n$ , whereas ' $a$ ' represents the intensity factor. The selection of ' $a$ ' should consider both image character and the characters of vision system in order to increase the

intensity of embedded watermark while ensuring its non-visibility (Hernandez, 1999).

In some watermarking system, the watermark can be extracted accurately, which is called the watermark extraction process. As is show in figure 2.4, in the application of integrity confirmation, the embedded watermark must be accurately extracted in order to identify multimedia data integration by watermark integration verification. In case of partial watermark modification, it would be better to identify the location of raw data by identifying the location where the watermark is modified.

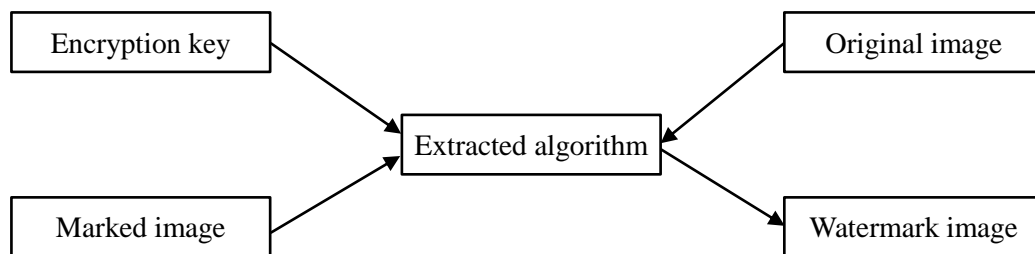


Figure 2.4 Watermark extraction processes

For robust watermarks, it's normally not allowed to exact already embedded watermark, because an application is likely to be maliciously attached if such an application is not implemented with robust watermarks. After undergoing these operations, the extracted watermark looks very different. Thus, a watermark detection process is needed. Refer to figure 2.5.

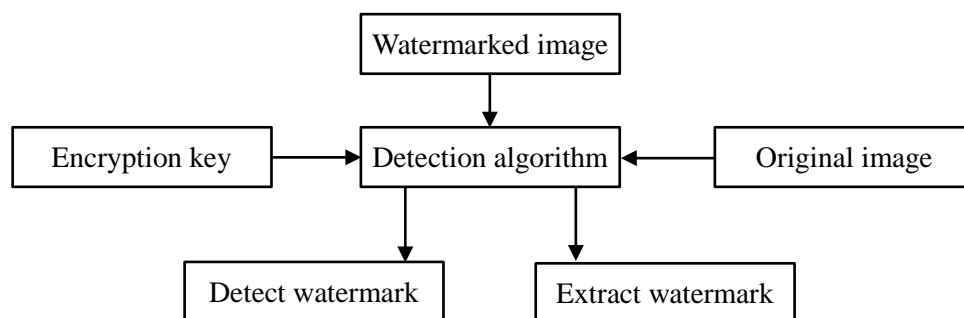


Figure 2.5 Watermark detection processes

The first step of watermark extraction is followed by watermark judgment. Watermark judgment normally refers to relativity detection. One judgment criterion is selected and the correlative value of extracted watermark and specified watermark is calculated. If the value is high enough, it will indicate that the detected data contain the specified watermark. Thus, the task of watermark extraction is to extract watermark signals from the embedded watermark.

Additionally, the result of watermark detection depends on a threshold; when detection result exceeds such a threshold, it is concluded that the specified watermark is detected. It is actually a problem of the hypothesis testing in probability studies. When the threshold of correlation is raised, the probabilities of false-positive detection decreases, while the probability of false-negative detection increases, and vice versa will be reversed. False-positive means no watermarks are contained and it is therefore not able to detect watermark signal from the image. In fact, the control of false-positive rate is more important. (Ricardo, 2009)

#### **2.4.2 Classification of Digital Watermarking**

According to some experts, methods of digital watermark processing can be divided into the following categories. (Fulton, 2000)

In them of their function, watermarks can be divided into robust watermarks and fragile watermarks. A robust watermark is mainly applied to mark copyright information in digital works; the embedded watermark is able to resist common editing processing, image processing and lossy compression. The Watermark should remain entirely identifiable after malicious attack or other attacks. A fragile watermark is usually used for the protection of integrity by judging the modification of multi-media signal and marking the modification. It requires common lossy compression, such as *JPEG* and *MP3*.

In terms of their vectors, watermarks can be divided into text watermark image watermark, video watermark, audio watermark and graphic watermark. Image watermark refers to the watermark embedded into a static image, which is mainly used in image database, online image presentation and so on. Video watermark is used to embed digital watermark into video stream in order to control the propagation of video application, i.e. DVD control and access. Image watermark is used to embed watermark into 2D and 3D images created by computers, which has the purpose of informing copyright.

In terms of their detection methods, watermarks can be divided into non-blind watermarks and blind watermarks (Yeo & Holliman, 1998). Prescribed watermark requires raw data and has better robustness, but its limitations affect its general application. In contrast, blind watermark does not require raw data and has extensive usage; especially for those network applications in which the original image are unavailable, they require more complex watermarking technology.

In terms of their contents, watermarks can be categorized as meaningful watermarks and meaningless watermarks. A meaningful watermark is a visual image ranging from a binary image, a gray-scale image or a color image. It has strength that people still can judge the integrity of the watermark by visual observation. Meaningless watermark is a series of random code. Its existence can only be identified by statistical methods.

In terms of their usage, watermark can be divided into as copyright protection watermark, tampering-detection watermark, hidden identification watermark, and watermark for the security of negotiable instruments. A copyright protection watermark is one that the owner of an image hopes to embed into the image for identifying copyright information and for protection of commercial interest. After malicious or unsurpassed attacks, the watermark still can be obtained from the image to identify the ownership. Failure to detect the watermark suggests the

compromise of copyright protection. A tempering-detection watermark is used to protect the integrity of image contents, and tempering mark modified contents and resist from common lossy compression. Watermarks for the security of negotiable instrument exists in paper bills or notes and are usually embedded during digital processing; the watermark can be detected during printing, publishing and scanning processes, but it disappears after copying for security protection. A hidden identification watermark is used to add comment information to an image in order to illustrate the image, to introduce the creator, etc.

Watermarks can be also categorized by their hiding location; they can be divided into time (space) domain watermarks and transform domain watermarks (Berghel, 1998). Time (space) domain watermarks directly add watermark information in signal space. With the development of watermarking technology, various watermark algorithms have been created and the hidden location of watermarks are no longer limited to those mentioned above, which suggests it is possible to hide watermark in the exchange space if there is a signal switch.

In terms of their visibility, watermarks can be categorized as visible watermarks and invisible watermarks. For visible watermarks, the owners can be aware of any modification, but it does not affect the display of the original information. This is an existing form in the early period of digital watermark creation; its use is limited because it is attacked easily. Because of the weakness of visible watermark, invisible watermarks have become the main subject of research. An invisible watermark is one embedded into the host information and one that can hardly be seen by common users, hence the test indicator of being 'hidden'. Whether it is 'hidden' well or not can have significantly different effects on watermarking solution. In certain circumstance, watermark hiding requires a very high standard, and it also requires extremely high quality of multimedia products. This brings a big challenge to digital watermarking design. Invisible watermark is part of research in this paper (Fulton, 2000).



### 2.4.3 Digital Watermarking System

Traditionally, many people believe that communication security can be achieved by the process of both encryptions and decryption; and that a hacker is hardly able to decrypt the confidential information from cipher text transmission over the network communication. There are information security breaches existing in the traditional cryptographic techniques. They have 3 major shortcomings: (1) Sensitive and important data transmitted over the networks are prone to attack by potential hackers; (2) If the encryption shell were destroyed, the encrypted contents can be easily cracked; (3) The attackers are capable of destroying the encrypted contents so that the authorized receiver failed to decrypt them. So cryptography has become a main information security technique for the communication applications. In recent years, people try to hide secret information in the protected documents, and then track violations so as to obtain legal proofs. This is called information hiding or steganography. (Sarah *et al.*, 2006)

In recent years, digital watermarking technology research has made great progress; some of the watermarking algorithms with their typical analyses are introduced as follows (Patrick & Michael, 1998):

#### (1) Space domain algorithms (SDA)

Space domain of digital watermarking technology is a way to directly superimpose digital watermarks on the image space domain (Cox *et al.*, 1997). Generally, the watermark signal is embedded into the brightness or color signal strength value of the image. The SDA has little impact on the host image; the arithmetic is simple; the operation is fast and large amount of information can be embedded in view of the characteristics of the human visual system. The shortcomings of SDA include resisting geometry

deformation, noise and poor image compression's capability. The typical applications of such algorithm are LSB and Patchwork.

Scientist Schyndel *et al.* elaborated an important concept of watermarking and the general detection method of robust watermark. (Schyndel, 1994) (Stefan, 2000) Firstly,  $m$ -sequence form of watermarks is created; secondly, image pixel digital bits are compressed; finally, pixels of a direct image are replaced by lowest (LSB) watermark or several half watermarks are embedding into LSB. The Least Significant Bit (LSB) is a simple and widely used method of inserting a watermark. In this method, the watermarking is a robust. The watermark can be easily removed from the images because it is embedded in the image of LSB.

## (2) Transform domain and image processing

In 1995, Cox and colleagues developed the DCT (Discrete Cosine Transform) domain technology. (Watson, 1993) They were the pioneers in the watermarking transform domain. This algorithm used to be one of the most frequently used algorithms. Since then, other transform domain algorithms appeared one after another. There are a variety of schemes for embedding the watermark, including DFT (Discrete Fourier Transform), DCT, DWT (Discrete Wavelet Transform), and Mellin-Fourier Transform Algorithm.

### a. DCT

DCT is widely used for digital watermarking. The digital image is divided into  $8 \times 8$  non-overlapping block of pixels. In block DCT transform, some randomly selected pseudo-frequency blocks can be obtained by noise sequence of DCT coefficients. In certain circumstances, the watermarking signal of carrier is embedded in some DCT coefficients controlled by the

pirate key. Based on the expansion of their train of thoughts, Cox and his colleagues found the algorithm to embed the watermark embedded in the DCT domain. (Uccheddu *et al.*, 2006)

The watermark is represented as:

$$X = x_1, x_2, \dots, x_n \quad (2.9)$$

Each  $x_i$  is chosen according to  $n(0, 1)$ , where  $n$  denotes a normal distribution. The watermark is inserted into the DCT coefficients of the image by the frequency components  $v_i$  in the original image  $v'_i$  by using the following equation.

$$v'_i = v_1, v_2 \wedge v_i \quad (2.10)$$

Practically, Barni replaced the Cox's algorithm with blinded watermarking algorithm, which is built based on wavelet transform and Human Visual System (HVS). (Barni *et al.*, 1998)

A watermark can be easily embedded in the significant middle frequency band. The human visual system uses the DCT extensively. When a JPEG image is modified as a result of steganography, certain colors will convert to another color according to the image color table. If a given color A appears less frequently than B, the chance of A converting to B is greater than that of B converting to A. Therefore the difference in color frequencies will be decreased and an analysis of color frequency would not yield much information. Instead, an analysis of DCT coefficients should be proven more fruitful. An  $n_2$  test on the image should show distortion from the embedded data. An image with hidden data should have a similar frequency for adjacent

DCT coefficients. Therefore, the DCT formula is presented as below: (Gilani & Skodras, 2001)

$$y_i' = \frac{(n_{2i} + n_{2i+1})}{2} \quad (2.11)$$

b. DFT

DFT is an algorithm by which watermarks are embedded in the DFT- domain. Most researchers believe that phase modulation may be more suitable for robust watermark. First, the phase component of the DFT is more important to the intelligibility of an image. It has great impact on the human visual system than the magnitude component. Therefore, if there is a higher degree of redundancy of the watermark in order to resist malicious attacks, attempts by the attackers to remove the watermark will result in unacceptable damage to the quality of the image. Second, according to communication theories, phase modulation has more robust immunity to signal noise than amplitude modulation. (Ruanaidh *et al.*, 1996)

c. DWT

Kunder and Hatzinakos proposed the DWT-based watermarking algorithm. They utilized multi-resolution wavelet to decompose an image (to separate several sub-bands, very similar to the way in which the retina divides an image into several parts). Therefore, wavelet transform is a good match of the visual system for the decomposition of space and frequency. The image watermarking algorithm uses a digital watermark processing system to transform the host image and watermark by wavelet transform to obtain the wavelet coefficients and then adds the reduced sub-bands of the watermark to the corresponding sub-bands of the host image. The wavelets of the combined sub-bands are inversely wavelet transformed (IWT) to generate a watermarked image. (Kunder *et al.*, 1999)

d. Mellin-Fourier

Ruanaidh is the first researcher to propose a watermark algorithms using Mellin-Fourier transforms. Digital watermarks are embedded in the rotation, zoom and pan in the same region after DFT transformation and coordinate transformation (Log Polar Map: LMP). This algorithm followed by many scholars who improved it and proposed a series of watermarking algorithms to resist anti-geometric deformation attacks. (Lu *et al.*, 2004)

(3) Other Algorithms

a. Fractal watermark.

Puate and Jordan first developed the technique of fractal image compression. This method can be used for both color images and gray images; they can change the search scope of the defined domain blocks and geometric transforms. Different fractal codes can be embedded into the same watermark in the two-tier process. They also can be embedded into different types of double-watermark and no original image is needed when extracting the watermark. The simulation results show that fractal watermark embedded in a single image has robust resistance to various attacks. For particular cutting attacks, the reliability of watermarking detection is much higher. Watermarks embedded using two different methods can expand the scope of application with fractal watermarking algorithm, which can be used to protect copyright and track the illegal re-distribution. (Puate, 1996)

The information to be embedded is specified as  $b$ ,  $b \in (0, 1)$ . A block is randomly selected from in the image and then divided into two equal-sized sub-blocks, and each sub-block is assigned to one bit. A search is then conducted. Each block in the sub-block that contains the corresponding value of the bit is encoded. In the recovery process, fractal compression is

performed on the watermarked image and then the entire image is searched. The location of the marked blocks contains the embedded information. Experiments show that this kind of watermarks can effectively resist JPEG compression. But such watermarking algorithm still has the disadvantages of being slow and computationally expensive, which mainly result from fractal compression. (Puata, 1996)

b. Spread spectrum watermarking

Tirke firstly realized that the spread spectrum technology can be used for digital watermarks (Aura, 1995). Subsequently, a large number of watermarking techniques using this technology have been invented. A watermarking algorithm using the spread spectrums technology based on wavelet transform is proposed by Wolfgang. This method has many advantages including transparency, reliability, robustness and anti-jamming. (Ingemar, 1998)

c. The algorithm based on the watermarking characteristics

In 1999, Kutter *et al.* introduced the concept of the second generation watermarking technology and a specific algorithm based on this concept. They proposed an algorithm based on point features in an image using a sale integration technique based on 2D continuous wavelets. Bas *et al.* (1999) described an algorithm based on the detection of point of interest in an image and the insertion of marks into blocks with similar features to these points (Bas *et al.*, 1999). Wang *et al.* proposed a content-based localized watermarking algorithm based on DWT, which adaptively embeds corresponding watermarks into the local regions of the image instead of the entire one, so that the watermarks can still be extracted by those points of interest although the image been cropped (Wang *et al.*, 2001).

## 2.5 Summary

Chaotic dynamical systems have become a hot research topic in multimedia information security. However, with the development of hacking techniques, more and more dynamic chaotic encryption systems have been cracked. Therefore, the safety and practicality of these systems have been questioned. The main factors that affect the development of such systems include:

- (1) The lack of rigorous mathematical theory or the lack of other systematic approaches. This problem can be solved by the combining of chaotic dynamic encryption and traditional dynamic encryption so that can be complementary and improve the algorithm's security. Chaotic cryptographic algorithms take more advantages (e.g. extreme sensitivity, randomness and periodicity) of the sensitivity to the initial values. Therefore, the combination of chaotic cipher and regular cipher can not only help expand the cipher space, but also improve the complexity of the algorithm and enhance its robustness.
- (2) There are no adequate tests for the chaotic dynamical cipher systems.
- (3) Limited computing precision effects the degradation of chaotic dynamics cipher in the computer systems.
- (4) The chaotic system has an initial threshold; once the system exceeded this threshold, the algorithm is invalid.
- (5) The watermarking encryption has different model; as a good watermarking system must possess three properties (robustness, imperceptibility and capacity). (Zheng *et al.*, 2007)

## **Chapter 3**

---

### **Methods of Hyperchaotic Scrambling and DWT Embedding**

In this chapter, the current research on image watermarking techniques will be briefly reviewed, and two major methods of hyperchaotic scrambling and DWT embedding will be introduced. The analysis of existing chaotic scrambling systems provides a knowledge base for the development of a hyperchaotic scrambling system of 2D image. Watermark embedding technique using DWT (Discrete Wavelet Transform) has the advantages of greater compatibility with the HVS and larger capacity of embedded information.



### 3.1 Analysis and Design of Digital Image Scrambling Methods

Many researchers proposed a variety of image encryption solutions based on chaotic theories, using chaotic systems that can generate massive, non-related, noise-like, and recyclable chaotic sequences. Such sequences are hard to be reversing remodeled and forecasted, and are therefore much more difficult to crack. An hyperchaotic system is one that contains two or more positive Lyapunov index numbers. Compared with single chaotic systems, it is more complex, and the randomness of data sequence it generates is more arbitrary. Therefore, its confidentiality is better and it is more suitable for data encryption.

One of the rational presumptions for any image encryption system is that an absolutely secure status does not exist. Commonly, if the costs spent on cracking certain encryption algorithm exceed the costs of data encryption, such a data encryption system is believed 'secure'. If the time spent on cracking certain algorithm is longer than the data confidentiality period, such an algorithm can be treated as a secure one, too. Moreover if the amount of data for cracking an algorithm is more than that for secret key encryption, such an algorithm is also considered 'secure'.

When designing an encryption, the following factors should be considered:

- (1) The security of a sound cipher system should be related with the secret key only; it should not be concerned with the algorithm. Even though all the details of the algorithm are known, the expected confidentiality should not be compromised.
- (2) The number of secret key should be large enough in order to meet confidential requirements;
- (3) Weak secret keys should not exist; all the selected secret keys should have equal confidentiality;

- (4) An encryption algorithm can resist known plain text attack; encrypted data and the secret key should not be decrypted even if a piece of plain text with corresponding cipher text is leak;
- (5) Exhaustive attack method is an effective type; the designed encryption system should be tested by exhaustive attack to evaluate the robustness of the system.

For image encryption, a few features of the image should be considered such as massive amount of data, high redundancy, and strong relativity among pixels. Thus, algorithm design should meet the following requirements.

Encryption algorithm should use as simple operation type as possible. Because an image contains large amount of data, if a complex algorithm is used, the encryption will be computationally expensive. Besides, considering encryption time, a designer should adopt a parallel processing algorithm. To achieve better encryption effects, the pixels of an image should be scrambled. To achieve higher image redundancy, it is better to use an encryption system combining with compression and encryption.

### **3.1.1 The Components of Image Encryption System**

The components of an image encryption system are shown in figure 3.1:

Plain text space: the original space of the message itself. For image encryption, it refers to the image to be encrypted; it could be spatial or frequency information of the image.

Cipher text space: the image to be transformed to illegible and chaotic information space after encryption. For computing binary data, the encrypted cipher-text is equal to or slightly bigger than the plain-text. Secret key space: the specific information space occupied by the control algorithm. (Fridrich, 1997)

The cipher algorithm defines a complex function transformation mode between plain text and cipher text, which is also used to encrypt and decrypt data. There are normally two related functions, one for encryption and one for decryption. (Stephen, *et al.*, 2005)

The cipher algorithm and the secret key are two fundamental elements of an encryption system.

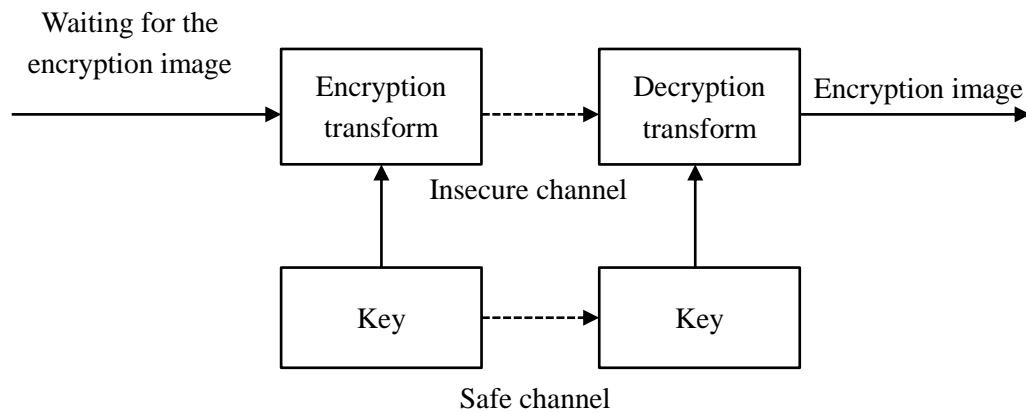


Figure 3.1 Schematic flowchart of the encryption system

There are commonly three methods for image encryption: gray-scale value replacement, pixel position transform and the combination of both. The combination of scrambling and replacement can greatly increase the robustness of the encryption. Thus, it is more effective to use two chaotic systems to separately implement scrambling and replacement, but this method is time consuming because it takes a lot of time to seek the no-repeat values for structure scrambling matrix, especially for an extremely large image. An improved method is to extend the dereferencing scope of image position value and to find a new transform algorithm to convert original position and extension position, which can be computationally economical. A new algorithm will be introduced below. It combines two chaotic systems to scramble images and the combination is optimized according to the characteristics of each chaotic system. (Stephen, *et al.*, 2005)

Considering the features of image encryption system, this chapter introduces chaotic method for encryption. Because chaotic sequence encryption is simple and fast, and since a chaotic algorithm is independent from compression algorithm, it will achieve better effects both for encryption and decryption. Moreover with this algorithm, one-time pair of keys can be generated, which has the merit of better security. Its system architecture is shown in figure 3.2 below:

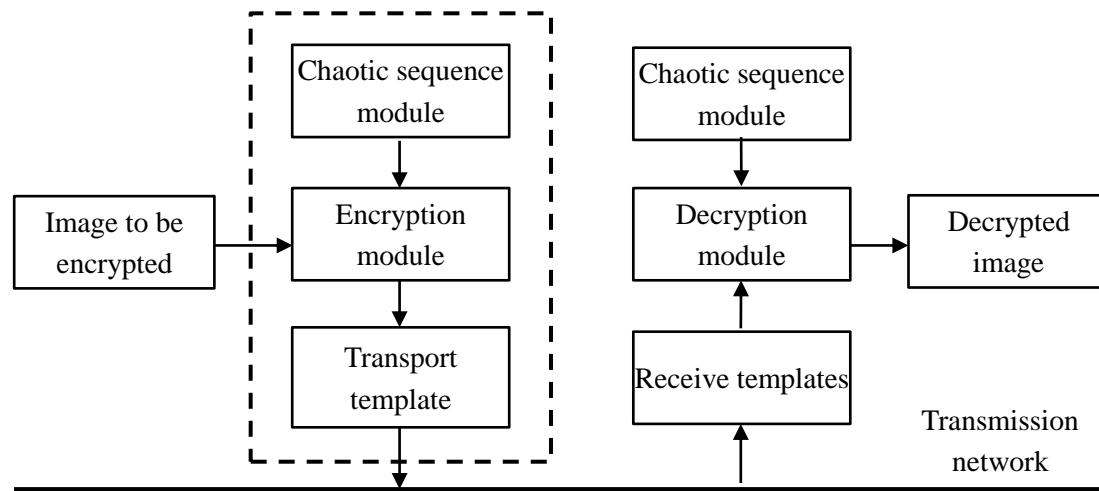


Figure 3.2 System architecture of a hyperchaotic algorithm

A chaotic sequence has some features that can benefit image encryption, such as certainty, pseudo-randomness, aperiodicity and non-convergence, and being extremely sensitively dependent on initial values, utilizing those features can benefit image encryption. The chaotic sequence of the real number can be transformed to randomly distribute binary sequence by quantizing. Due to limited accuracy, the chaotic sequence of real numbers can be transformed to pseudo-sequence or periodic sequence by computing operation. However, such transformation has some shortcomings. For example, the fixed chaotic parameter  $p$  is prone to be cracked; it is also likely to fall into a periodic circle that is not large enough.

A hyperchaotic encryption system is composed of a chaotic sequence module, an encryption module and a transfer module. The chaotic sequence generation module is the core of the whole system, which generates the random sequence to implements

the encryption algorithm; the encryption module is to process the image; and the transfer module deals with transferring the encrypted image and secret key. Generally, a conventional chaotic system is used to generate a pseudo random binary sequence. A real number sequences generated by such a chaotic system are not homogeneous; if it is used as a secret key stream, encryption effects will be affected. The phase space partitioning method can be used to generate pseudo random binary sequence (PRBS) in order to achieve satisfactory encryption effects.

### 3.1.2 System Encryption Flow Chart

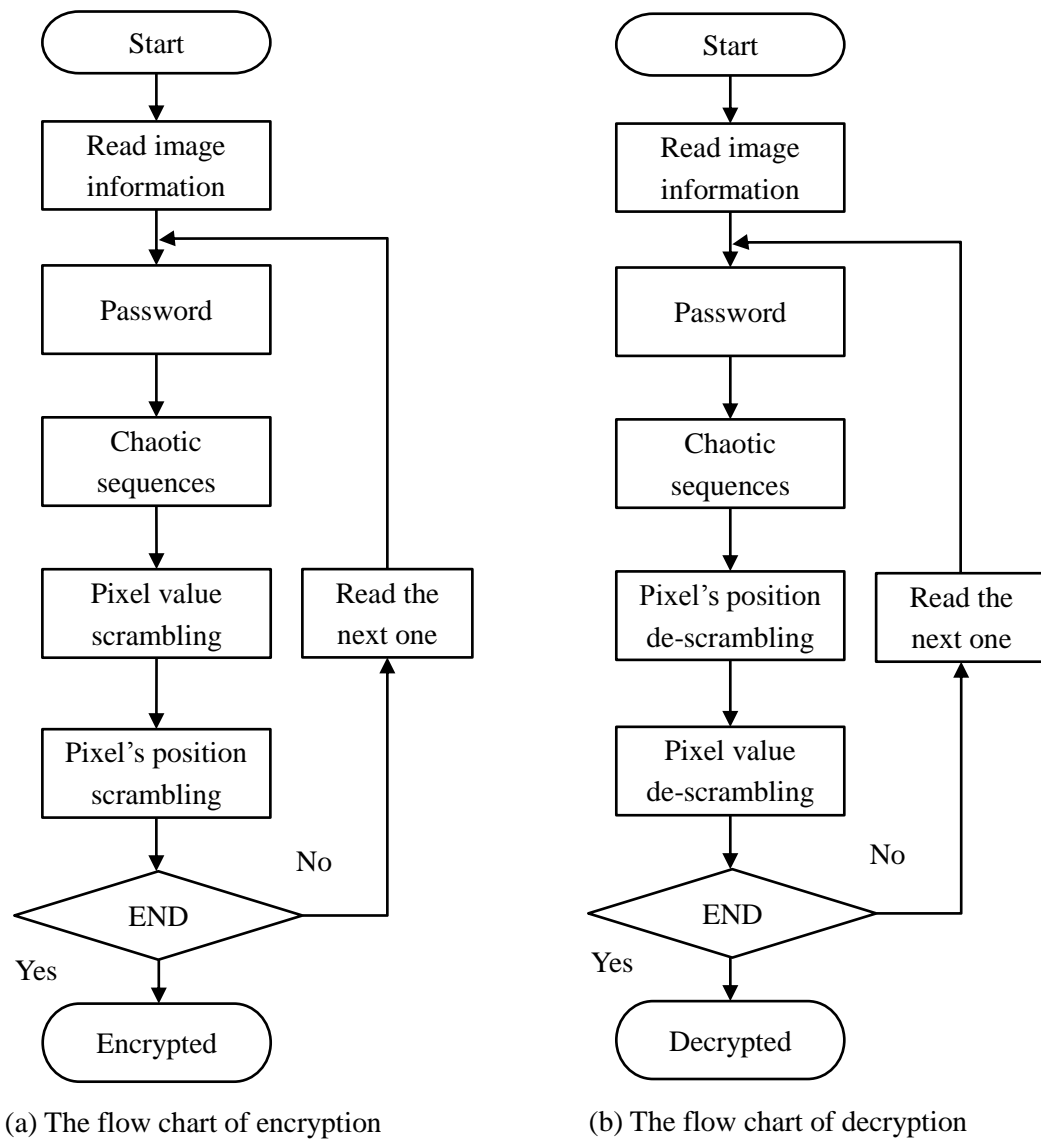


Figure 3.3 Flowchart of a hyperchaotic encryption and decryption system

The processes of encrypting and decrypting an image using a hyperchaotic system are shown in figure 3.3. In figure 3.3 (a), the image is firstly read by the computer. After inputting the initial password, the chaotic sequence is generated from the information of the image and initial password. The chaotic sequence is then utilized to scramble pixel value and pixel position for the candidate image for encryption.

Figure 3.3 (b) is a flow chart that illustrates the decryption process of the encrypted image. In the decryption process, the steps of pixel value and pixel position scrambling should be reversed in order to decrypt the image correctly.

### **3.1.3 Framework of Hyperchaotic Image Encryption**

In 1963, the Lorenz system was innovated as a mathematical model in chaotic theory for meteorological research and it was applied to other areas of research since then. In 1999, Chen discovered another chaotic system similar to Lorenz system, but with different topology. Since the Chen system was discovered, there has been a renewed interest in the Lorenz system, especially in the continuous chaotic anti-control.

Compared with the conventional chaotic system, the hyperchaotic system has two or more positive Lyapunov exponents, thus the chaotic behavior is more complex. So it can be applied to improve the security in aspects of communications, such as signal processing, chaotic synchronization, hyperchaos and other fields.

In this section, the Lorenz system and the Chen system are combined to generate the anti-control model of hyperchaotic system. Using the same model, multiple hyperchaotic systems can be generated from the Henon chaotic system, such hyperchaotic systems can be used for image encryption.

(1) Lorenz system

The Lorenz system is a 3D quadratic autonomous system, it was originally developed by Saltzman in 1962 as a ‘minimalist’ model of thermal convection in a box. (Lu *et al.*, 2003)

$$\begin{cases} \dot{x} = a(y - x) \\ \dot{y} = cx - xz - y \\ \dot{z} = xy - bz \end{cases} \quad (3.1)$$

The properties of the Lorenz map are nonlinearity, symmetry, volume contraction-dissipative and fixed points (Lorenz, 1963). Where the parameters are  $a=10$ ,  $b=8/3$ , and  $c=28$ , it has one chaotic attractor.

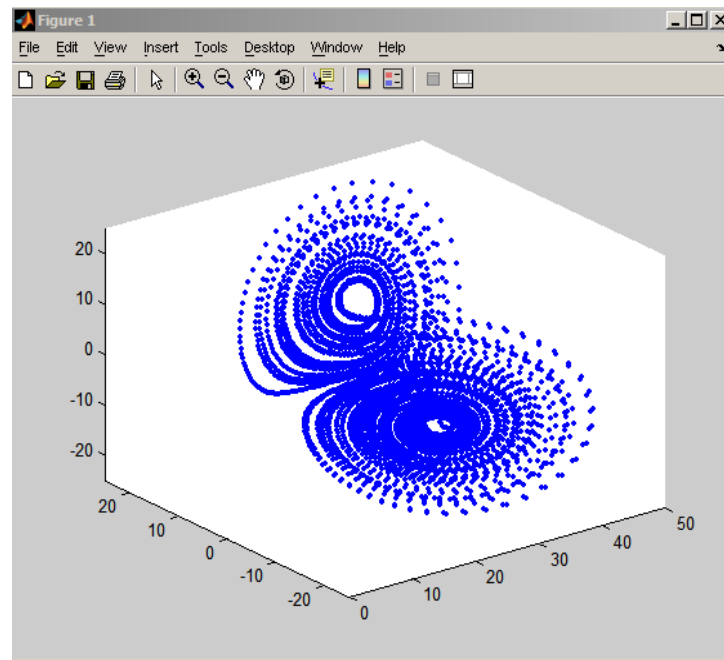


Figure 3.4 Lorenz system

The Lorenz equations have seven terms on the right-hand side, which contain two nonlinear quadratic ( $xz$  and  $xy$ ). In 1976, Rossler found a 3D quadratic autonomous chaotic system similar to the Lorenz system. The Rossler system is simpler in structure than the Lorenz system because it has only one quadratic non-linearity ( $xz$ ). The Rossler system is defined as:

$$\begin{cases} \dot{x} = -y - z \\ \dot{y} = x + ay \\ \dot{z} = zx - cz + b \end{cases} \quad (3.2)$$

When the parameters satisfy  $a=b=0.2, c=5.7$ . The system has a Rossler chaotic attractor. (Rossler, 1976).

## (2) The Chen system

In the process of extending his research on anti-control of chaos in discrete system to continuous dynamical system, Mr.Chen discovered that some parameters of the Lorenz system are not in the chaotic area (Chen, 1999). In this state, a simple linear partial state-feedback controller can drive the Lorenz system, so can the Chen system: (Ueta & Chen, 2000)

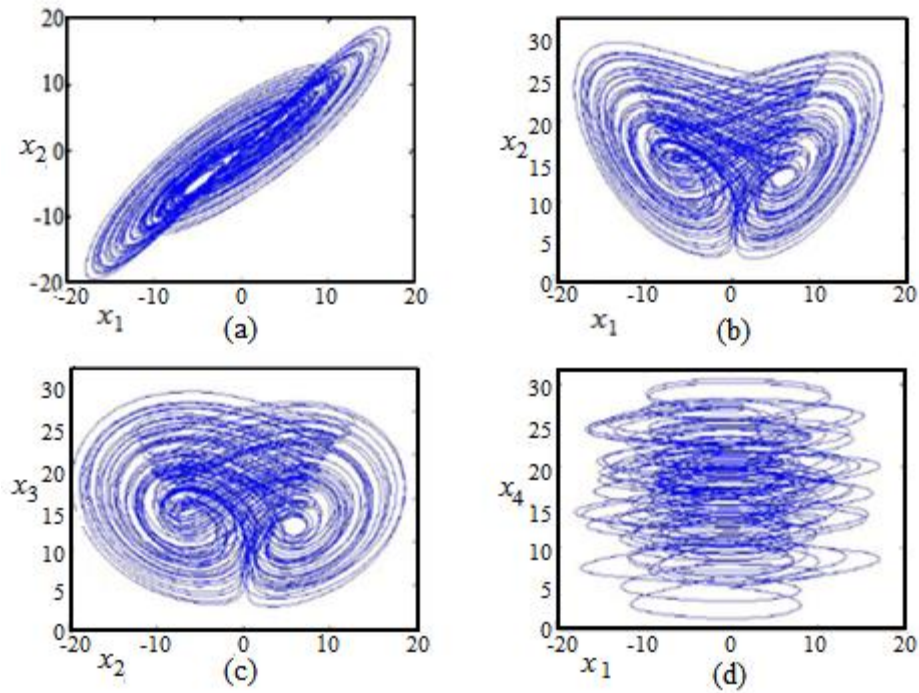
$$\begin{cases} \dot{x} = a(y - x) + u \\ \dot{y} = dx - xz + cy \\ \dot{z} = xy - bz \end{cases} \quad (3.3)$$

When  $a=35, b=3$  and  $c=28$ , the system has a chaotic attractor whose phase portraits are shown in figure 3.5. Both the Lorenz system and the Chen system possess two quadratic equations. But they cannot transform the topologically from one system to another. Compared with the Lorenz system, Chen system has more complex topological structure and dynamical behavior. For nearly 40 years, many simple chaotic have been found, some of chaotic have the framework of three-dimensional quadratic autonomous systems. The Chen system is three-dimensional quadratic autonomous equations, which can generate two 1-scroll chaotic attractors simultaneously with three equilibria, and two 2-scroll chaotic attractors simultaneously with five equilibria. Dynamical behaviors of this Chen chaotic system includes some basic dynamical properties, bifurcations, periodic windows, routes to chaos, compound structures of the new attractors and their



connections, etc. (Chen, 2004)

Figure 3.5 (Li, 2005) shows the phase portraits of the hyperchaotic system with  $u=0.2$ , (a)  $x_1$ - $x_2$  plane, (b)  $x_1$ - $x_3$  plane, (c)  $x_2$ - $x_3$  plane, (d)  $x_1$ - $x_4$  plane. There are four Lyapunov exponents as show in following:



$$\lambda_1= 1.552, \lambda_2= 0.023, \lambda_3= 0, \lambda_4= -12.573$$

Figure 3.5 Chen system

(Li, 2005)

### (3) The hyperchaotic system

In 1979, Rossler found the hyperchaotic system. The hyperchaotic system has two or more positive Lyapunov exponents, so it is more complex than other chaotic systems. Complex hyperchaotic system can improve the confidentiality of communications. In continuous autonomous systems, the minimal number of dimensions of phase space to display the non-linear properties of a chaotic attractor should be three or more. However, in a hyperchaotic system, the minimal dimensionality of the phase space that embeds the hyperchaotic attractor should be four, which requires the minimum number of coupled first-order

autonomous ordinary differential equation to be four. So hyperchaotic systems have more complicated topological structures. (Li, 2005)

The Chen's hyperchaotic system is described as follows. In the Chen chaotic system (3.3), where  $a, b, c$  and  $d$  are constants. When  $a=35$ ,  $b=3$ ,  $c=12$ ,  $d=7$ , the system has a hyperchaotic attractor, as displayed in figure 3.5. At this point, the Chen system has three Lyapunov exponents, the first is positive  $\lambda_1=0.6567$ , the second is zero, and the last one is negative  $\lambda_2=-26.6439$ . The nonlinear feedback controller is added to the right-hand side of the first equation (3.3). (Chen, 1999) Then the equation becomes:

$$\begin{cases} x = a(y - x) + u \\ y = dx - xz + cy \\ z = xy - bz \\ u = yz + ru \end{cases} \quad (3.4)$$

### 3.2 Image Encryption Research Based on Frequency Transform

Currently, application of digital watermarking mainly exists in two domains: space and frequency. These two types of watermarking have different uses for different industries. Spatial domain watermarking includes: Least Significant Bit algorithm (LSB), statistic algorithm and Delay Echo method (for audio work). These basic replacement systems are trying to encode secret information in the carriers by replacing insignificant parts of the carriers with bits containing such disguised information. Once the recipient knows the location of embedded secret information it can be extracted easily. Although the resistance to common attacks (compression, cropping and median Filter) is relatively weak, these modifications are not prone to be detected by attackers. Besides, these methods are able to embed a lot of secret information and to reduce the quantity of disguised carriers, which may help reduce network load.

Frequency transform domain watermarking methods includes: Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT). These types of digital watermarking methods are implemented by performing certain operations to transform the carrier; the secret information is embedded in a disguised transform domain. Secret information thus embedded has good resistance to compression, cropping and median filter attacks, so the robustness of the watermarks can be improved greatly. However, there are conflicts existing between embedding quantity and the robustness obtained. Relatively speaking, for one carrier, the amount of secret information embedded in the frequency domain is less than that in the spatial domain.

Fourier transform is one of the best and widely used tools for image processing, but it is just a pure analysis of the frequency domain, and it has no positioning capability in the space-time domain, which suggests that Fourier transform is unable to provide frequency information in any partial time quantum. To overcome those shortcomings, DCT and DWT are proposed to replace Fourier transform; they have various good characteristics of Fourier transform without its shortcomings. The French geophysicist J.Morlet proposed the wavelet transform theory in 1984 (Morlet & Grossman, 1984). It has become a new research project for digital watermark processing. The principles of wavelet transform include: a digital image is processed in multi-resolution; the digital image is decomposed into sub-images in different spaces and different frequencies, and the sub-images are processed afterwards according to their specific features.

### **3.2.1 Discrete Cosine Transform**

Discrete Cosine Transform (DCT) is similar to the Fourier transform in that they both operate on real data with even symmetry. But since DCT uses only real numbers. It does not require the complex calculation of Fourier transform, so it is most twice as efficient. The vectors of DCT matrix are quite similar to those of the Toeplitz matrix.

The correlative characteristics of the human visual system (HVS) and image information can be reflected by the Toeplitz matrix; thus, DCT is regarded as the quasi-optimal transform for aural or visual signals. At the same time, a DCT algorithm can be easily and quickly executed in a digital signal processor. Therefore, it plays an important role in the image coding, and has become an important ingredient of a number of international standards for image coding (*JPEG*, *MPEG*, *H261*, etc). According to the *JPEG* compression standard, an image is divided into  $8 \times 8$  blocks. After the magnitude of the DCT large transform coefficients concentrated in the upper-left corner of the DCT array, for the remaining relatively small ones, their values are shifted from a positive range to one centered around zero. And those of the high frequency part of close to 0 will be discarded.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

**Before JPEG still picture compression standard**

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

**After JPEG still picture compression standard**

Figure 3.6 *JPEG* compression standards

(Rabbani, 2002) (Christopoulos, 2000)

On an  $M \times N$  digital image  $f(x, y)$ ,  $0 \leq x \leq M, 0 \leq y \leq N$ , the 2D DCT used in digital image processing can be presented as below:

$$C(u, v) = a_u a_v \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \quad (3.5)$$

$$C(u, v) = \frac{2}{N} \sum \sum f(x, y) \cos \frac{(2x+1)u\pi}{2M} \cos \frac{(2y+1)v\pi}{2N} \quad (3.6)$$

where  $u = 0, 1, \dots, M-1$ ;  $v = 0, 1, \dots, N-1$ ;  $C(u, v)$  is the transform coefficient matrix,  $x, y = \{0, 1, 2, \dots, N-1\}$  and  $f(x, y)$  is 2D vector in spatial domain.

Inverse discrete cosine transform (IDCT) is:

$$f(x, y) = \frac{1}{N} C(0, 0) + \frac{2}{N} \sum_{u=1}^{N-1} \sum_{v=1}^{M-1} C(u, v) \cos \frac{(2x+1)u\pi}{2M} \cos \frac{(2y+1)v\pi}{2N} \quad (3.7)$$

### 3.2.2 The Processing Steps for DCT

2D DCT is the core of the JPEG system, which is one of the popular lossy digital image compression systems. Therefore, embedding digital watermark in DCT domain can effectively resist JPEG lossy compression. According to JPEG standard, an image is first divided into  $8 \times 8$  image blocks. Each block goes through DCT, and the resultant DCT coefficient is then quantified. When quantifying, all the DCT coefficients are divided by a set of quantification values (see table 3.6) to get the nearest integer. When compressing, all the  $8 \times 8$  DCT coefficients are scanned from low to high frequency in Zig-Zag sequence. The 1<sup>st</sup> value (at the upper-left corner) is a DC coefficient, and the rest are AC coefficients. Among the DCT coefficients, those in the upper-left corners are DC coefficients with low frequency, while those in the lower-right corners are high frequency AC coefficients, and the middle section is occupied by those with intermediate frequency. Low frequency indicates slow change in image pixels and high frequency stands for fast change in image pixels. Thus, high frequency section represents noise in the image, which is likely to be removed by lossy compression or wave filtering processing. Because mid-low frequency sections

contain most energy of an image, to maintain image's visibility, image compression and processing normally retain the integers of intermediate and low frequencies in an image. However, the modification of low frequency can possibly result in considerable change of an image; therefore, to bind a watermark with the most important visual part of the carrier image together, it is normally embedded into intermediate section in order to keep the visibility of image intact and the watermark unlikely to be compromised. (Syed, 2003) (Feng *et al.*, 2007) (Willie, 2003)

Compared with image watermarks in space domain, watermarks in DCT (Discrete Cosine Transform) domain maintain better robustness for image compression, wave filtering and other digital processing algorithms. Also, as DCT is compatible with the commonly used image compression standard-*JPEG*. The DCT of watermark technology is the most widely used. (Andrew, 1994) (Wang *et al.*, 2009)

The process of embedding a chaotically encrypted digital image as a watermark mainly consists of 3 steps:

(1) Image splitting and block categorizing

First of all, the original image  $I$  is divided into non-coverage of the image block  $8 \times 8$ , marked as:

$$f_k(x, y) \quad k = 0, 1, 2, \dots, N \quad (3.8)$$

Such as:

$$I = \bigcup_{k=0}^{k-1} f_k(x, y), \dots, 0 \leq x, y < \theta \quad (3.9)$$

Block categorizing: the finer is the texture of the image, the higher the visibility of the watermark. According to the complexity of the texture in a specific area of the image, the signals of the embedded watermark should be as strong as possible.

This can increase the stability of the watermark accordingly. So the image blocks are divided into two categories: (1) Image blocks with the finer texture (marked as  $S_1$ ); (2) Image blocks with the blurry texture (marked as  $S_2$ ). As the HVS is more sensitive to the modification of 2<sup>nd</sup> category of image blocks, the intension of the signals of the embedded watermark in these blocks should not be too strong.

(2) DCT and watermark embedding

After categorizing, each image block goes through DCT to obtain a coefficient:

$$F_k(u, v) = DCT\{f_x(x, y)\} \quad 0 \leq x, y < \infty, 0 \leq u, v < \infty \quad (3.10)$$

The multiplication rule is adopted for watermark embedding:

$$DCT_D(u, v) = \begin{cases} 0 \\ DCT_i(u, v) \times (1 + a w(k)) \quad \dots \text{if } \dots u \end{cases} \quad (3.11)$$

Where  $w(k)$  is the chaotic watermark signal, a Logistic chaotic sequence with the length  $k$ .  $\alpha$  is the factor for the intensity of the embedded signal. The above formula indicates: the intensity of the superimposed watermark signal should be proportional to the average brightness of the image blocks.

$$a = \begin{cases} 0.8, \text{ if } f_k(x, y) \in s_1 \\ 0.35, \text{ if } f_k(x, y) \in s_2 \end{cases} \quad (3.12)$$

(3) IDCT to recover the watermarked image

A watermarked Image  $D$  is obtained by taking IDCT of the watermarked DCT coefficients.

$$D = \bigcup_{k=0}^{k=1} iDCT \{F_K(u, v)\} \quad (3.13)$$

## (4) Watermark detection

The embedded watermark can be detected using the following formula:

$$W_k' = \frac{(DCT(D(k)) - DCT(I(k)))}{a \times DCT(I(k))} \quad (3.14)$$

First, the similarity of the watermark is calculated:

$$SIM(w, w') = \frac{\sum_{k=0}^{k-1} w(k) \times w'(k)}{\sqrt{\sum_{k=0}^{k-1} w(k)^2}} \quad (3.15)$$

with the judging criteria:  $SIM(w, w') > T$ , a watermark exists in the image.

The scrambling algorithm adopts two chaotic systems to get an iterative sequence, scramble two-dimensional images, and embed them into the DCT domain, but this algorithm is too weak to resist attack.

The second algorithm is applied to color images. The original image and the watermark are divided into the same number of DCT blocks. Then the blocks of the watermark are all scrambled using the Arnold transform, which embeds the scrambled watermark to DC coefficients and *IF* (Intermediate Frequency) coefficients in DCT domain. This algorithm has stronger resistance to cropping and *JPEG* compression than to the addition of noise, rotation and other geometric attacks.

The chaotic sequence is easy to generate, sensitive to initial conditions, and because of its statistics of white noise and other characteristics. It is widely used in digital watermarking technology. At the same time, combining of chaotic technology and



image scrambling can improve concealment and robustness of the watermark, enhancing its security.

### 3.2.3 Examination of DCT Watermark Algorithm

The watermark algorithm based on chaotic Scrambling in DCT domain, uses Logistic map to scramble a watermark image, and then embed it in the original image, generating image a watermarked image.

A one-dimensional chaotic Logistic map is used to scramble a watermark image. The Logistic map is shown below:

$$c_{n+1}' = f(c_n'{}^2) = 1 + 2c_n'{}^2, \quad c_n' \in [-1, 1] \quad (3.16)$$

This one-dimensional Logistic map has a chaotic sequence with an average of 0. This characteristic is quite similar to white noise in probability statistics. The above fundamental characteristic generates an extreme sensitivity to initial value. The Logistic map is then transformed into a binary sequence using the following formula:

$$c_n = \frac{(\sin(c_n') + 1)}{2}; \quad (n = 0, 1, 2 \dots, n - 1) \quad (3.17)$$

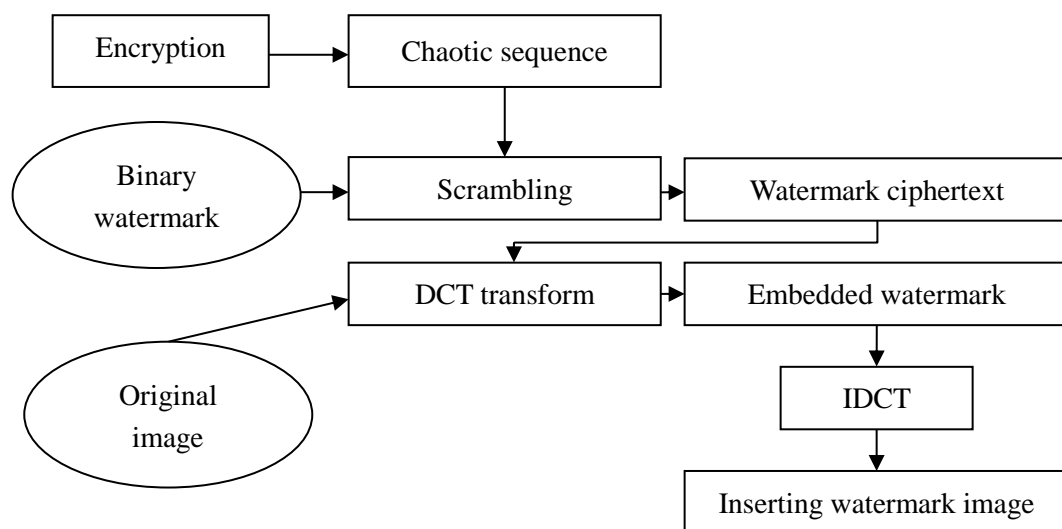


Figure 3.7 Watermark DCT embedding process

Set  $N_1 \times N_2$  size for the original gray  $I$  image and the embedded binary watermark image  $W$  of size is  $M_1 \times M_2$ . Under normal circumstances, the watermark image is smaller than the original image, that is,  $M_1 \leq N_1$ ,  $M_2 \leq N_2$ . The watermark embedding process is shown in figure 3.7.

### Step 1. 2D DCT

An  $8 \times 8$  block original image  $I$  is transformed using 2D DCT. These blocks are not overlapping.

### Step 2. Generation of keys

To increase the security, the watermark image is scrambled by the Logistic map. Logistic map is used for generating chaotic analog sequence. It is defined by:

$$x_{n+1} = \mu x_n (1 - x_n) \quad (3.18)$$

The Logistic map is used to generate a sequence of pseudo random numbers  $S$ . Its initial condition of  $X_0$ ,  $\mu$  and the chaotic sequence length  $N$  are the three keys of watermarking algorithm:  $k_1$ ,  $k_2$  and  $k_3$ . In the algorithm,  $X_0=0.8$ ,  $\mu=3.8$ ,  $N=M_1 \times M_2$  are shared keys for encryption and decryption

### Step 3. Scrambling of watermark image

First, a chaotic sequence  $S_1$  is generated, and it is sorted in ascending order to form an ordered sequence  $S_1'$ . A second sequence  $S_2$  is used to remember the original position of each chaotic element and a  $S_2'$  is used to map the sorted sequence. Then, the watermark image pixels are scrambled by transforming the watermark sequence from a 2D matrix to a one-dimensional array, and recorded as  $S_3$ . Next, the corresponding mapping relationship of  $S_2$  and  $S_3$  is established. After that, all the positions of the

watermark image pixels from  $S_3$  are sorted according to  $S_2$  to get a scrambled sequence  $S_3'$ . In the end, the one-dimensional array  $S_3'$  is transformed back to a two-dimensional matrix.

The following figure 3.8 illustrates the Logistic process of binary watermark image scrambling ( $x_0=0.8$ ,  $\mu=3.8$ ,  $N=6$ , the watermark sequence length of value is 6) :

$$\begin{array}{ccc}
 S_1=[0.59 \ 0.92 \ 0.26 \ 0.74 \ 0.68 \ 0.82] & \rightarrow & S_1'=[0.26 \ 0.59 \ 0.68 \ 0.74 \ 0.82 \ 0.92] \\
 \uparrow \downarrow \uparrow \downarrow \uparrow \downarrow & & \uparrow \downarrow \uparrow \downarrow \uparrow \downarrow \\
 S_2=[ \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 ] & \rightarrow & S_2'=[ \ 3 \ 1 \ 5 \ 4 \ 6 \ 2 ] \\
 S_3=[ \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 ] & \rightarrow & S_3'=[ \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 ]
 \end{array}$$

Figure 3.8 Logistic map sequences to scramble binary watermark pixels

Step 4. The original image takes the Discrete Cosine Transform method (DCT) for each sub-block, and the intermediate frequency coefficient is selected to embed the watermark by using the formula 3.19 below:

$$F'_k(u, v) = F_k(u, v) \times (1 + a x_i) \tag{3.19}$$

$F_k(u, v)$  is the host image of DCT coefficient and  $F'_k(u, v)$  is the embedded watermark data.  $x_i$  ( $i = 1, 2, \dots, L; M_1 \times M_2$ ) is the sequence of watermark. And 'a' is the scale factor, whose the 'a' value is determined by the specific image.

	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63

Figure 3.9 Sequence of watermark after  $M_1 \times M_2$  IDCT

Step 5. Finally, the inverse discrete cosines transform (IDCT) is taken to get the watermarked image.

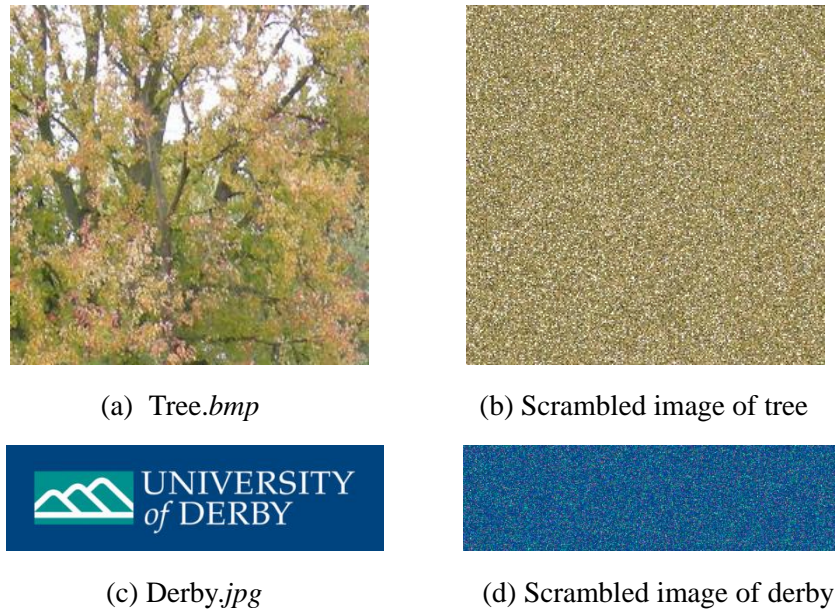


Figure 3.10 Scrambling of the tree's image and derby logo

### 3.2.4 Algorithms for Watermark Extraction and Detection

Watermark extraction is the inverse process to watermark embedding. The extraction algorithm is shown in figure 3.12

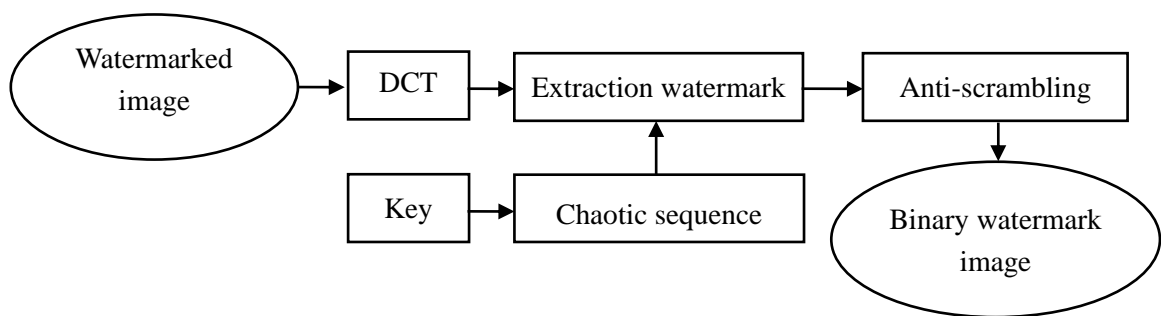


Figure 3.11 The extraction algorithm

The main steps are:

Step 1. An  $8 \times 8$  block original image  $I$  and a watermarked image  $I'$  are transformed by two-dimensional DCT to sub-blocks. These sub-blocks do not overlap.

Step 2. The difference between the original image and the watermarked image to be detected can be determined by using the formula (3.20) below:

$$p_i = F_k'(u, v) - \frac{F_k(u, v)}{a F_k(u, v)}, \quad i = 1, 2, \dots; M_1 \times M_2 \quad (3.20)$$

Hence, the embedded watermark sequence  $p_i$  can be obtained.

Step 3. The resulting sequence is converted into binary bit  $p_i$  using a threshold 0.5 in the following formula (3.21). The data value  $x_i$  is the watermark sequence that is extracted from the scrambling watermark in the DCT domain.

$$x_i' = \begin{cases} 0, & p_i \leq 0.5 \\ 1, & p_i > 0.5 \end{cases} \quad i = 1, 2, \dots; M_1 \times M_2 \quad (3.21)$$

Step 4. The scrambled watermark  $x_i'$  is extracted and decrypted. Anti-Scrambling can pick up the binary watermark image  $W^*$ .

In contrast, DWT watermarking is more perceptively faithful than DCT watermarking. Since DCT only represent spatial domain using frequency domain, it makes no use of the characteristics of the space-frequency relationship of an image. The space-frequency relationship coincides with some of the visual features of the human eyes. Wavelet transform can cater HVS characteristics. The blocks of DCT reconstructed image will appear mosaic, but the DWT will not. In addition, DWT watermark fusion technology will be distributed to the carrier of multiple scales in order to improve the robustness of watermark. So the digital watermarking technology DWT will be employed in the present study.

### 3.3 Discrete Wavelet Transform

Wavelet analysis is one of time-frequency localization with fixed window size but

changeable shape and time-frequency window. That is, it has higher frequency resolution ratio and lower time resolution ratio in low frequency part, whereas it has lower frequency resolution ratio and higher time resolution ratio in high frequency part. Hence, wavelet analysis is also called a ‘mathematical microscope’. Because of those characteristics, wavelet transform is self-adaptive to signals.

### 3.3.1 Continuous Wavelet Transforms

‘In mathematics, a wavelet series is a representation of a square-integrable (real-or-complex-valued) function by a certain orthonormal series generated by a wavelet. This article provides a formal, mathematical definition of an orthonormal wavelet and the integral wavelet transform.’ (Chui & Charles, 1992)

Wavelet is a small area of the wave, also called sub-wavelet; it is a kind of oscillation waveform with limited length and zero average value. ‘Small’ refers to the attenuation property (local non-zero nature), in which the number of non-zero coefficient reflects texture complexity and abundant layer of image’s high frequency components.

The wavelet transform can also be divided into CWT (Continue Wavelet Transform), DWT (Discrete Wavelet Transform) and MWT (Multiresolution-based Wavelet Transform) (Daubechies, 1992).

Definition of wavelets: Let  $\psi(t)$  be any square-integrable function, that is  $\psi(t) \in L^2(R)$ . If the Fourier transform  $\psi(t)$  can be allowed to satisfy the condition. (Addison, 2002)

$$C_\psi = \int_R \frac{|\psi(t)|^2}{t} d\omega < \infty \quad (3.22)$$

Then  $\psi(t)$  is a basis wavelet or mother wavelet function, which has two characteristics:

- (1) A ‘small’  $\psi(t)$  has compact support or similar compact support in the time domain.
- (2) The wavelet has alternating positive and negative turn of the fluctuating property with a zero DC component.

The continue wavelet functions derives from dilation and translation of mother wavelet  $\psi(t)$  which satisfies the following equation (3.23):

$$\psi_{a,\tau}(t) = \frac{1}{\sqrt{|a|}} \psi\left(\frac{t-\tau}{a}\right) \quad a, \tau \in R \quad (3.23)$$

Where  $a$  is the dilation coefficient, and  $\tau$  is the translation coefficient, with  $\psi(t)_{\omega=0}=0$ ,  $\psi_{a\tau}(t)$  is continuous basic wavelet function which is dependent on the parameters  $a$  and  $\tau$ .

For any  $L^2(R)$  space, if the function  $f(t)$  is convoluted by a set of functions generated by the basis wavelet, that convolution can be referred to as Continue Wavelet Transform (CWT), which can be represented by the following equation (3.24):

$$WT_f = \frac{1}{c_\psi} \int_0^\infty \frac{da}{a^2} \int_{-\infty}^\infty WT_f(a, \tau) \psi\left(\frac{t-\tau}{a}\right) dt \quad (3.24)$$

To recover the original signal  $f(t) \in L^2(R)$ , the following reconstruction formulas (inverse) are used:

Time Domain  $f(t) = \frac{1}{c_\psi} \int_0^\infty \frac{da}{a^2} \int_{-\infty}^\infty WT_f(a, \tau) \psi\left(\frac{t-\tau}{a}\right) dt \quad (3.25)$

Frequency Domain  $CWT(\alpha, \tau) = \sqrt{\alpha} F(\tau) \mathcal{O}(\alpha, \tau) \quad (3.26)$

(Calderon 1964), (Grossmann & Morlet, 1984), (Mallat, 1998)

When  $C_\psi = \int \frac{|\psi(a\omega)|^2}{a} da < \infty$ ,  $\psi(t)$  can be the proposed permitting conditions. From

the above analysis, the features and function of wavelet transform can be summarized as:

- (1) The wavelet transform has multi-resolution (multi-scale) characteristics that can be used to observe the signal step by step. Coarse (low resolution) version and details (high resolution) of the original image are contained in the domain form  $LL$  sub-bands to the  $HL$ ,  $LH$  and  $HH$  sub-bands, respectively.
- (2) In wavelet transform, the signal is filtered by the band-pass filter as the fundamental frequency, using the  $\psi(\omega)$  and different scales factor 'a'. We select the appropriate wavelet basic function on condition that it requires  $\psi(\omega)$  to be in the time domain with compact support.

### 3.3.2 Discrete Wavelet Transform

Due to the redundant operation of continuous wavelet transform, there are a lot of redundancies existing in the information after one dimensional signal  $x(t)$  is transformed to two-dimensional  $WT_x(a, \tau)$  by wavelet transform. For data compression and time saving, wavelet transformation can be calculated under some discrete scale and displacement. Different wavelets are obtained through scattering displacement parameter  $r$  and scale parameter  $a$ . Binary wavelet transformation is the most common method in use for image processing.

To reduce the redundancy of wavelet coefficients, the displacement parameters  $r$  and scale parameter  $a$  for the wavelet basis functions are set at values of discrete points. The scale parameter  $a$  is usually discretized using power series discretization. Even if  $a=a_0^m$ ,  $a_0>0$ ,  $m$  is an integer. For the discretization of the displacement parameter  $\tau$ , it is usually set evenly at discrete values to cover the entire timeline. In order to prevent



information loss, sample interval  $r_0$  is required to meet the sampling theorem that the sampling frequency should be bigger than frequency of scale or equal twice the pass-band scale frequency. When  $m$  increases by 1, the scale parameter would be doubled, but the corresponding frequency of the displacement  $\tau$  would reduce by half. Thus, the sampling can also be reduced by half without data loss. (Meyer, 1992)

The wavelet function  $\psi_{\alpha\tau}(t)$  is presented as below equation (3.27) (Meyer, 1992):

$$\psi_{\alpha,\tau}(t) = \alpha_0^{-\frac{1}{2}} \psi \left[ \alpha_0^{-j} (t - k \alpha_0^j \tau_0) \right] = \alpha_0^{-\frac{1}{2}} \psi \left[ \alpha_0^{-j} (t - k \tau_0) \right] \quad (3.27)$$

In practical application, the binary discrete wavelet is often used. If  $a_0$  is set at 2, the above equation can be referred to as a dyadic wavelet transform. The 2D discrete wavelet transform function is given by equation (3.28) (Meyer, 1992):

$$WT_f(j, k) = \int f(t) \psi_{j,k}(t) dt \quad j = 0, 1, 2 \dots \quad (3.28)$$

For the wavelet transform, there is also inverse wavelet transform. Wavelet transform is a type of reversible transformation of information. All of the information of the original image or signal is retained in the wavelet transforming coefficients. Wavelet transform only redistributes the energy of original image. This is the main reason that wavelet transform is widely used. (Meyer, 1992)

### 3.3.3 Wavelets in Multi-resolution Analysis

A multi-resolution analysis (MRA) or multiscale approximation (MSA) was proposed by S.G. Mallat (Mallat, 1989). A Multi-resolution analysis is also known as multi-scale analysis, which is in  $L^2(R)$  function space, the function  $f(x)$  is described as the limit of a series of similar functions, and each function is an approximation of  $f(x)$  at different resolutions on the subspace projection. These approximations are obtained at different scales. The process is multi-resolution analysis of wavelet transforming.

When the sampling frequency of the signal  $X(n)$  satisfies the condition of sampling theorem, the digital frequency bands will be limited between  $-\pi \sim \pi$ . An ideal low-pass filter  $L(\omega)$  and the high-pass filter  $H(\omega)$  can be used to decompose the frequency bands into the low-frequency part  $0 \sim \pi/2$  and the high-frequency part  $\pi/2 \sim \pi$ , which represent the approximation signals and details of signals. (Shown in figure 3.12)

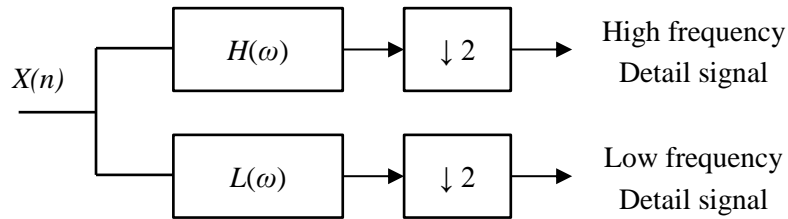


Figure 3.12 Low (High) frequency band signal  $X(n)$

The two decomposed outputs must be orthogonal (the bands cannot overlap), and because outputs of both frequency band are reduced by half, the sampling rate can be reduced by half accordingly without causing the loss of information (see figure 3.13)

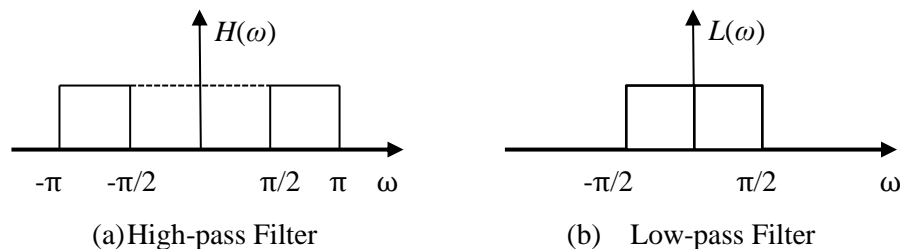


Figure 3.13 The impulse response

A similar process can be repeated over and over again, which means that each layer of the input signal is further decomposed to roughly approximate part of a low frequency and detail part of a high frequency, and the output sampling rate of each layer can be reduced by half. Thus, the original signal  $X(n)$  is processed by multi-resolution decomposition, where  $n$  is an integer. (Taubman, 1999) Figure 3.12 and figure 3.13 show low (high) frequency band signal and the impulse response respectively.

The above is just a rough description of multi-resolution analysis; the conclusion is not comprehensive enough. Based on function space decomposition, mathematician Mallet discovered the relationship between wavelet transform and multi-resolution analysis. In Mallet's theory, the approximation of each layer of a function is the result of a low pass smoothing function to  $X(n)$  smoothing. Multi-resolution analysis just uses different resolutions to analyze the  $X(n)$  function. (Vaidyanathan, 1993)

The  $X(n)$  function is first expressed as a one-dimensional signal and then extended to 2D for the space of the functions, and set 'j' to denote integers. One-dimensional functions (original signal):

$$f(x) \in L^2(R)$$

$$V_j = W_{j-1} \oplus V_{j-1} = W_{j-1} \oplus W_{j-2} \oplus V_{j-2} = \dots W_0 \oplus W_1 \oplus \dots W_{j-1} \oplus V_0 \quad (3.29)$$

The space is broken down into subspaces.

'j' is  $-\infty \sim \infty$ , so

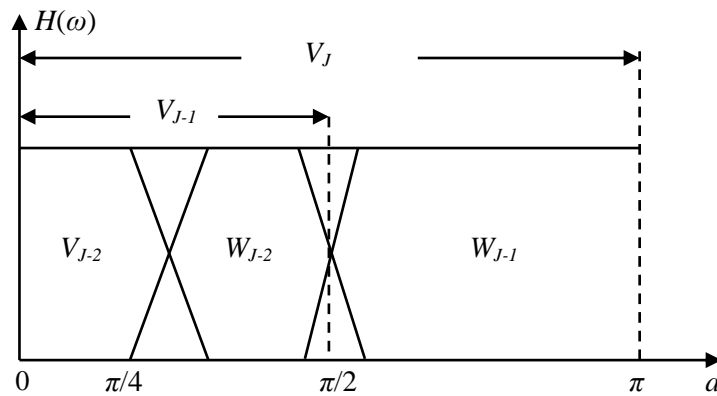


Figure 3.14 Frequency domain representation of the DWT

The scaling function and wavelet function subspaces are presented as:

$$V_j = V_{j-1} + W_{j-1} \quad (3.30)$$

$$V_{j-1} = V_{j-2} + W_{j-2} \quad (3.31)$$

The nature of the function space:

An MSA is refers to the analysis of sets of subspace  $\{V_1\}$  in space  $L^2(R)$ . It has the following properties (Mallat, 1989):

- (1) Monotonousness (inclusiveness)

This is a causality property:

$$V_{j-1} \supset V_j, \quad \forall j \in Z \quad (3.32)$$

- (2) Approximation:

$\bigcup_{j=-\infty}^{+\infty} V_{2^j}$  is dense in  $L^2(R)$ . Clearly,  $\bigcup_{j=-\infty}^{+\infty} V_{2^j} = \{0\}$ , thus sequence of vector spaces  $(V_{2^j})$ ,  $j \in Z$  may be considered as a multi-resolution approximation of  $L^2(R)$ . The approximated signal contains less and less information and converges to zero. So the equation is:

$$close\{\bigcup_{j=-\infty}^{\infty}\} = L^2(R), \quad \bigcap_{j=-\infty}^{\infty} = \{0\} \quad (3.33)$$

- (3) Scalability:

An approximation operation is similar at all resolutions.

$$\varphi(t) \in V_j \Leftrightarrow \varphi(2t) \in V_{j-1} \quad (3.34)$$

- (4) Translation invariance

$$\varphi(t) \in V_j \Leftrightarrow \varphi(2^j t - k) \in V_j, \quad \forall k \in Z \quad (3.35)$$

- (5) Existence of Riesz (orthogonal) basis:

$$\varphi(t) \in V_0, \quad \{\varphi(2^{-j} t - k)\}_{k \in Z} \quad (3.36)$$

### 3.3.4 Applications of DWT in Image Processing

When applying DWT to image processing, a one-dimensional resolution  $\{V_k^1\}$  and  $\{V_k^2\}$ , should be designed first. Then the tensor product space  $\{V_k\}$  is expended for 2D resolution. (Mallat, 1989)

$$V_{k+1}^1 = V_k^1 + W_k^1 \quad (3.37)$$

$$V_{k+1}^2 = V_k^2 + W_k^2 \quad (3.38)$$

Since  $V_{k+1} = V_k^1 \otimes V_k^2$ , so it can be iterating on 'k' to get  $V_{k+1} = V_{k+1}^1 \otimes V_{k+1}^2$ . Thus

$$V_{k+1} = V_{k+1}^1 \otimes V_{k+1}^2 = (V_k^1 + W_k^1) \otimes (V_k^2 + W_k^2) = V_k^1 \otimes V_k^2 + V_k^1 \otimes W_k^2 + W_k^1 \otimes V_k^2 + W_k^1 \otimes W_k^2 \quad (3.39)$$

'+' indicates the sum of the two spaces.

There is:

$$W_k = W_k^{(1)} + W_k^{(2)} + W_k^{(3)} \\ W_k^{(1)} = V_k^1 \otimes W_k^2, W_k^{(2)} = W_k^1 \otimes V_k^2, W_k^{(3)} = W_k^1 \otimes W_k^2 \quad (3.40)$$

When  $f_k(x, y) \in V_k, g_k(x, y) \in W_k$ , there is

$$f_{k+1}(x, y) = f_k(x, y) + g_k(x, y) \quad (3.41)$$

$$g_k = g_k^{(1)} + g_k^{(2)} + g_k^{(3)} \quad (3.42)$$

From (3.35), we can get

$$g_k^{(i)} \in W_k^{(i)} (i=1, 2, 3) \quad (3.43)$$

Analysis of the multi-section of 2D  $L^2(\mathbb{R}^2)$  is the vector space of measurable, square-integrals two dimensional functions  $f(x_1, x_2) \in L^2(\mathbb{R}^2)$ , where  $x_1, x_2$  are

referenced abscissa and ordinate. The function  $\psi(x_1, x_2)$  is based on a two-dimensional wavelet, defined as 2D continuous wavelet. In 2D displacement and scaling  $\psi(x_1, x_2)$  can be expressed as  $\psi_{a_1, b_1, b_2}(x_1, x_2)$ . (Mallat, 1989)

$$\psi_{a, b_1, b_2}(x_1, x_2) = \frac{1}{a} \psi\left(\frac{x_1 - b_1}{a}, \frac{x_2 - b_2}{a}\right) dx_1 dx_2 \quad (3.44)$$

One of the factors  $\frac{1}{a}$  is to ensure the same energy before and after the wavelet expansion to the introduction of normalization factor, the 2D continuous wavelet transform:

$$WT_f(a, b_1, b_2) = \left\langle f(x_1, x_2), \psi_{a, b_1, b_2}(x_1, x_2) \right\rangle = \frac{1}{a} \iint f(x_1, x_2) \psi\left(\frac{x_1 - b_1}{a}, \frac{x_2 - b_2}{a}\right) dx_1 dx_2 \quad (3.45)$$

The corresponding inverse wavelet transform is:

$$f(x_1, x_2) = \frac{1}{C_\psi} \int_0^\infty \frac{da}{a^3} \iint WT_f(a, b_1, b_2) \psi\left(\frac{x_1 - b_1}{a}, \frac{x_2 - b_2}{a}\right) db_1 db_2 \quad (3.46)$$

Where:

$$C_\psi = \frac{1}{4\pi^2} \iint \frac{|\psi(\omega_1, \omega_2)|}{|\omega_1^2 + \omega_2^2|} d\omega_1 d\omega_2 \quad (3.47)$$

(Kaiser 1994)

The above is the 2D continuous wavelet transforms evolved from the one-dimensional wavelet transform. In terms of scale expansion and also the stretching of coordinate rotation, 2D wavelet transform is more complex than the one-dimensional wavelet transform. So the scale factor can be rewritten as the matrix below. (Mallat, 1989)

$$A = a r_0$$

$$r_0 = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \quad (3.48)$$

2D multi-resolution analysis includes 3 steps: first,  $\phi(x_1)$  and  $\psi(x_1)$  are analyzed by

$x_1$  direction; second,  $f(x_1, x_2)$  is decomposed into smooth approximation and details;  $\phi(x_2)$  and  $\psi(x_2)$ , which are similarly analyzed in  $x_2$  direction; third, obtained 4 outputs are shown in figure 3.15. Let the layer of wavelet decomposition be  $j > 0$ :

$$A_j f(x_1, x_2) = \langle f(x_1, x_2), \varphi_{jk_1}(x_1) \varphi_{jk_2}(x_2) \rangle \quad (3.49)$$

$$D_j^{(1)} f(x_1, x_2) = \langle f(x_1, x_2), \varphi_{jk_1}(x_1) \psi_{jk_2}(x_2) \rangle \quad (3.50)$$

$$D_j^{(2)} f(x_1, x_2) = \langle f(x_1, x_2), \psi_{jk_1}(x_1) \varphi_{jk_2}(x_2) \rangle \quad (3.51)$$

$$D_j^{(3)} f(x_1, x_2) = \langle f(x_1, x_2), \psi_{jk_1}(x_1) \psi_{jk_2}(x_2) \rangle \quad (3.52)$$

$A_1 f(x_1, x_2)$  is the low frequency component (smooth variation) *LL*.

$D_1^{(1)} f(x_1, x_2)$  reflects the horizontal low-frequency and vertical high-frequency, which refers to the horizontal Component *LH*.

$D_1^{(2)} f(x_1, x_2)$  reflects the horizontal high-frequency and vertical low-frequency, which refers to the vertical Component *HL*.

$D_1^{(3)} f(x_1, x_2)$  reflects the horizontal high-frequency and vertical high-frequency, which refers to the diagonal component *HH*.

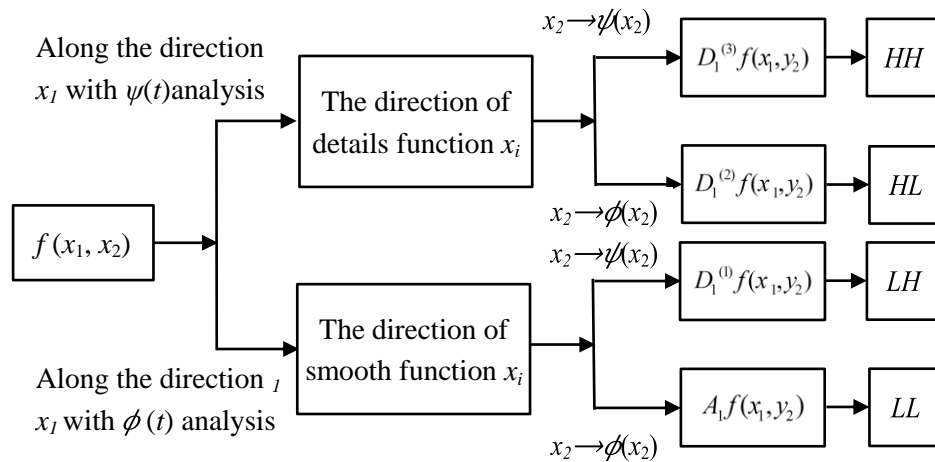
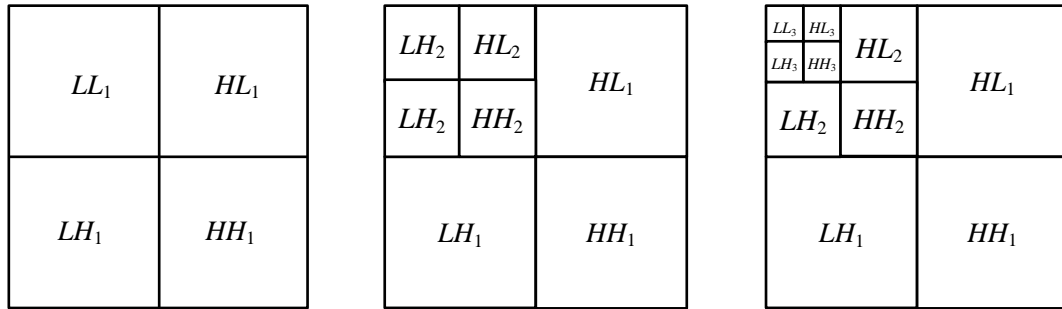


Figure 3.15 Decomposition schematic diagram of 2D wavelet transforms

Figure 3.16 is the diagram of image after wavelet decomposition. (Mallat, 1989) The image (a) represents the single level decomposition of host image, the image (b) shows two level decomposition and the image (c) shows three level decomposition.  $L$  means the low-frequency;  $H$  means the high-frequency.



(a) Single level decomposition (b) Two level decomposition (c) Three level decomposition

Figure 3.16 Schematic of 3-layer wavelet decomposition

The Mallat algorithm or Mallat-tree decomposition was based on an algorithm for 2D DWT, the image was decomposed into one low frequency approximation subgraph  $LL$  and three high frequency subgraphs:  $LH$ ,  $HL$  and  $HH$ . Then the decomposition with the Daubechies basis (or Haar) is repeated three times. So we can get as many as ten subgraphs:  $LL_3$  and  $LH_i, HL_i, HH_i$  ( $i=1, 2$ , and  $3$ ).

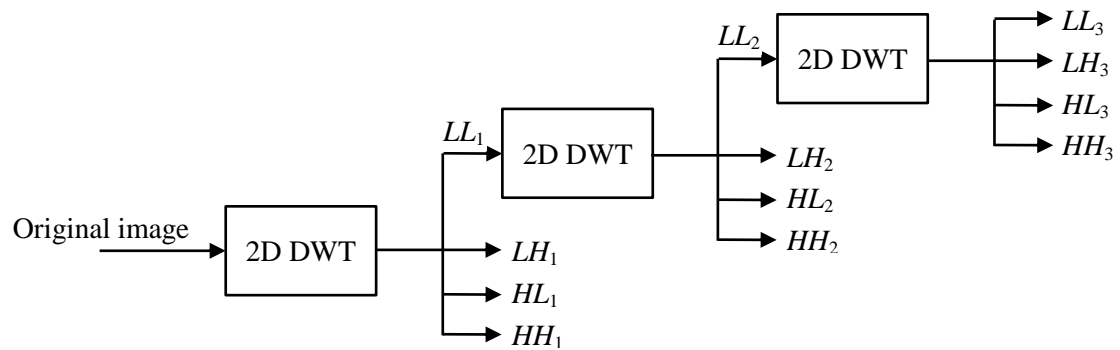


Figure 3.17 Decomposition schematic of 3-Layer wavelet transforms

The following is the MATLAB simulation for experimental image  $512 \times 512$  color-scale image of *hall.jpg*. The coefficient  $k=8$  is used to scrambles the matrix sequence. The sequence  $f(k)$  ( $k=1, 2, \dots, n$ ) has 32 incompatible matrices. *Hall.jpg* is decomposed into eight sub-images of different layers. Each sub-image is used in 1 layer of wavelet transform. Figure 3.18 shows 1-layer (level 1) wavelet



decomposition of *hall.jpg*. Figure 3.19 shows 2-layer (level 2) wavelet decomposition of *hall.jpg*



Figure 3.18 1-layer (level 1) wavelet decomposition of *hall.jpg*

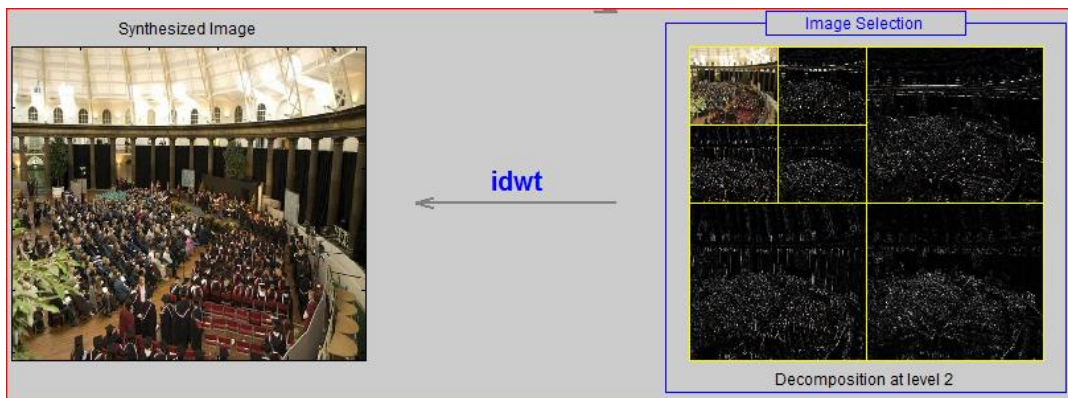


Figure 3.19 2-layer (level 2) wavelet decomposition of *hall.jpg*

(1) The Daubechies Wavelet Transform

An analysis filter bank decomposes 1-D signals into low-pass and high-pass components. A similar decomposition can be performed on images by first applying 1-D filtering along rows of image and then along columns, or the other way round.

If we set the function  $c_n=0$  for  $n < N^-$  and  $n > N^+$ ,  $N$  is filter length, where ‘length’ is the length of the D-filters, which must be a positive even integer. Also note that, ‘ $N^-$ ’ is a narrow filtering and ‘ $N^+$ ’ is a width filtering.

We get  $\psi$ ,

$$\text{sup}(\psi) \subset \left[ \frac{1}{2}(1 - N^+ + N^-), \frac{1}{2}(1 + N^+ - N^-) \right] \quad (3.53)$$

and 
$$\psi(x) = \sum_n (-1)^n \bar{c}_1 - n \phi(2x - n) \quad (3.54)$$

This operation is illustrated figure 3.19. Using the same filters  $H_0(e^{j\omega})$  and  $H_1(e^{j\omega})$  for horizontal and vertical filtering, we can get a set of four  $N/2 \times M/2$  subimages: the so-called *LL* (low, low), *LH* (low, high), *HL* (high, low) and *HH* (high, high) subbands. (Daubechies, 1992) (Daubechies, *et al.*, 1998)

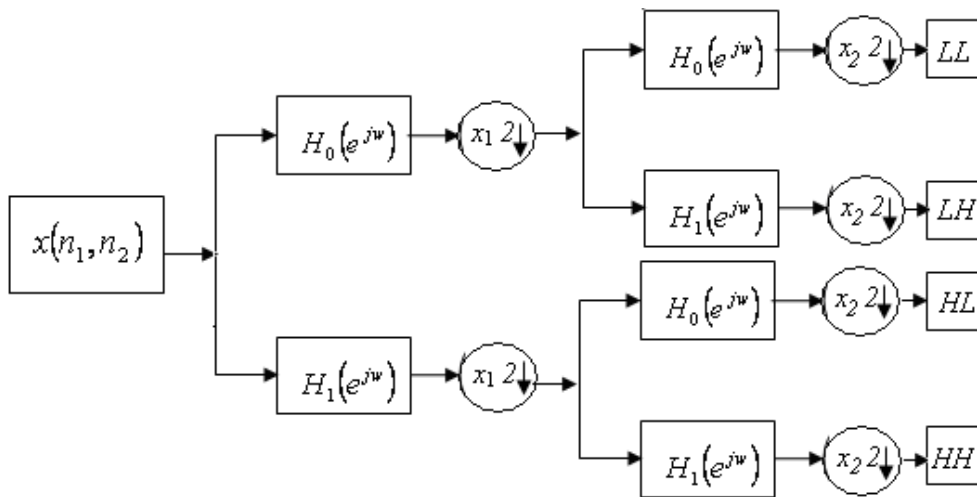


Figure 3.20 The Daubechies wavelet decomposition

(Mallat, 1989)

(2) The Haar wavelet transform

The discrete wavelet transform requires a convolution operation on the signals of the image by the filters. The convolution operation depends upon the low pass and the high-pass Haar filters that are involved by the filter of low pass  $h_0 = \left\{ \frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2} \right\}$  for lower sampling and the filter of high pass

$h_1 = \left\{ -\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2} \right\}$  for upper sampling. If we set the function ( $c_0 = c_1 = 1$ ) (Haar, 1910).

We get

$$\psi(x) = \begin{cases} 1, & 0 \leq x < \frac{1}{2} \\ -1, & \frac{1}{2} \leq x < 1 \end{cases} \quad (3.55)$$

For the Haar wavelet filter, there are two kinds of data: the sparse data and the detailed data. The coefficients for these data are extracted from the original data by using two kinds of filters, high pass (details) and low pass (average). After applying these two filters, the filtered data from the high pass filter is stored as coefficients for later reconstruction of the signal (details). The filtered data from the low pass filter is then treated as the original data, to which the low and high pass filters are applied again. The operation is repeated until only one number is yielded by the filtered data from the low pass filter. (Germano, 1999)

In the same circumstances, the Haar wavelet is more suitable for application in digital watermarking. The choice of this type of DWT for watermarking is very important, because the Haar wavelet can be used for perfect reconstruction of the signal, and it complies with the law of conservation of energy. So it is a more effective wavelet technique. In particular, the algorithm is symmetry wavelets. Otherwise, the reduction coefficient will be dislocation.

### 3.4 Detection of Watermarking Encryption

Reasonable assessment of watermarking is important. Performance evaluation of watermarking includes two aspects: the one is the robustness of the watermark; and the other is the assessment of image quality and distortion. Generally, we need to compromise between invisibility and robustness. Therefore, for a fair and reasonable

assessment, it is necessary that watermarking system be tested under comparable conditions. So a technology for the test of image scrambling is required-the Lyapunov exponent.

### 3.4.1 Using the Lyapunov Exponent to Identify Chaotic System

The chaotic system depends upon initial condition. The trajectories will be separated continuously as time goes on. The Lyapunov exponent is a quantitative description of the volume of this phenomenon.

The Lyapunov exponent is an important quantitative indicator of the dynamics properties of chaotic systems; it is the indicator used to measure the mean exponential rate of divergence. The existence of chaotic behavior is determined using the maximum Lyapunov exponent which must be greater than zero: in the chaos, a positive Lyapunov exponent will evolve over time so that the rate will be constant. Lyapunov exponents are considered as the evidence of hyperchaotic behavior.

Currently, Lyapunov exponents are used to evaluate chaotic cryptography. The Lyapunov exponent is shown below:

$$\lambda = \lim_{x \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n \ln \left| \frac{d f(x)}{d(x)} \right|_{x=x_i} \quad (3.56)$$

where  $\lambda$  is the Lyapunov exponent. The Lyapunov exponent and uniform distribution of chaos are determined by some quantitative measures.

The Lyapunov exponent is the constant separation for each average iteration point. Therefore, it can be used as a criterion of the chaotic behavior. While  $\lambda < 0$ , it indicates that the adjacent points are merged into a fixed point or stable periodic orbit. These are non-conservative (dissipative) and exhibit asymptotic stability. While  $\lambda > 0$ , it

it indicates that the adjacent points will be diverged irrespective of each other and the corresponding orbit appears chaotic and partially unstable. While  $\lambda=0$ , the system is neutrally stable and conservative. They exhibit Lyapunov stability. (Glenn, 2007)

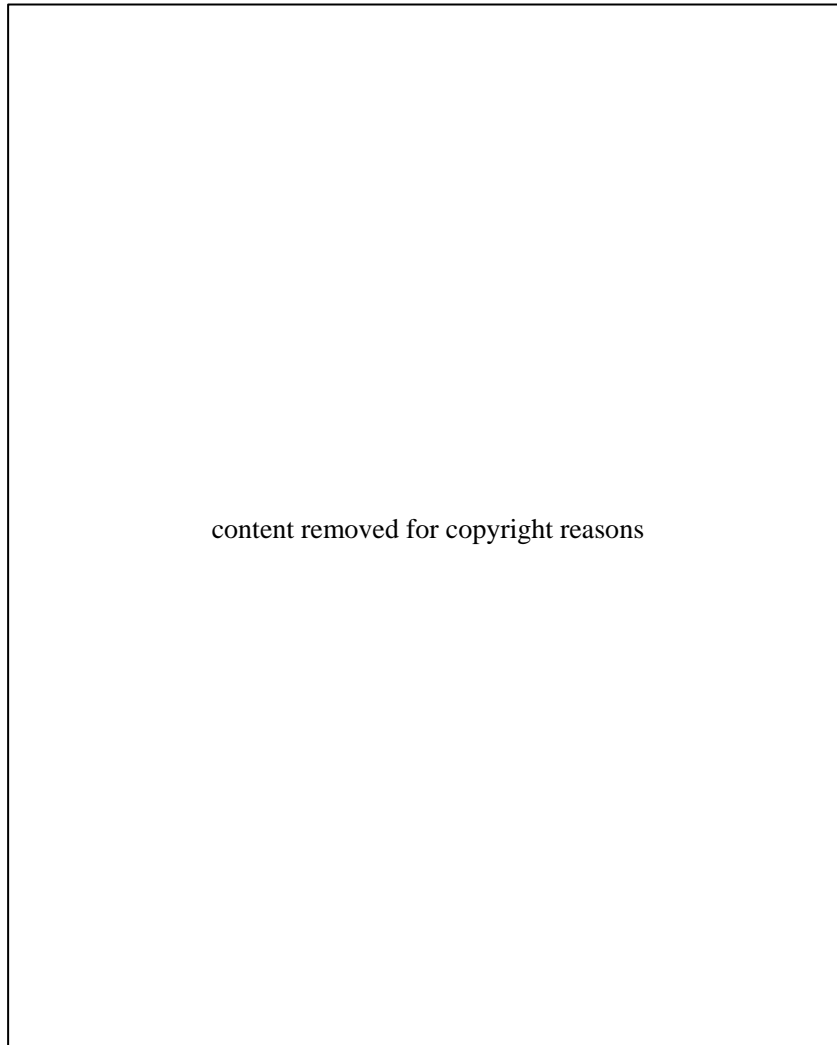


Figure 3. 21 Some orbits with their Lyapunov exponents

(Glenn, 2007)

### 3.4.2 Evaluation Criterion for Image Encryption

The experiment has proven that the signal-to-noise-ratio (*SNR*), peak signal to noise ratio (*PSNR*) and normalized cross correlation (*NC*) can be used to measure the similarity between the watermarked image and the original host image, as well as the invisible and robustness of the embedded watermark. *PSNR* and *NC* are defined as

follows:

The *PSNR* is used to evaluate the quality between an attacked image and the host image. (Ghanbari, 2008) It is defined as

$$PSNR(MSE) = 10 \log_{10} \frac{255^2}{MSE} \quad (3.57)$$

The *NC* was first defined by Hsu and Wu *et al.* (Hsu, 1999) (Wu, 2005).

$$NC = \frac{\sum_{x=1}^M \sum_{y=1}^N f(x, y) f'(x, y)}{\sum_{x=1}^M \sum_{y=1}^N f(x, y)^2} \quad (3.58)$$

where  $f(x, y)$  and  $f'(x, y)$  are represented pixel values for a host image and a watermarked image.

The perception and the stability of a watermark call for watermarking redundancy because there should be a trade-off between the two. Perceived quality of the watermark can be assessed using subjective tests or quality measurement. The most widely used criterion for image and video distortion is *SNR*, with the decibel (*db*) as its unit of measurement. (Chapeau-Blondeau *et al.*, 2008)

In the following formula 3.59, it can be seen that the higher *SNR* corresponds to the degree of similarity between the images. The result can prove that the watermarked image has the relatively higher imperceptibility. Generally an *SNR* above 30db means the watermarked image could be identified by the human eye. In addition, mean square error (*MSE*) is the widely used measurement of distortion; *MSE* is smaller when the *SNR* of image is increased.

*MSE* is defined as follows,

$$MSE = \frac{I}{MN} \sum_{x=1}^M \sum_{y=1}^N [f(x, y) - f'(x, y)]^2 \quad (3.59)$$

$$SNR(MSE) = 10 \log_{10} \frac{\sum_{x=1}^M \sum_{y=1}^N f(x, y)^2}{MN, MSE} \quad (3.60)$$

Where  $M$  and  $N$  are image dimensions, while  $f(x, y)$  and  $f'(x, y)$  are pixel values of the host and watermarked images at  $(x, y)$  location.

In fact,  $SNR$  is not primarily used for the watermark visibility test. Because the robustness of the generated watermarking system is based on the HVS, the embedding is successful if the embedded watermark is invisible to the human eyes.

### 3.4.3 Histogram

The Histogram of an image is one of the important statistical characteristics. Calculating histogram is a basic statistical approach to approximating probability distributions. It can be considered as approximate value of the graylayer density function of that image. (John, 2008)

By random process theory, an image can be treated as a random field with related statistical features; graylayer density function is the most important feature. Usually, the graylayer distribution density function of an image is concerned with pixel position; i.e. if the graylayer distribution density function of image point  $(x, y)$  is  $p(x, y, z)$ , then the graylayer density function of that image is:

$$p(z) = \frac{1}{s} \iint_D p(x, y, z) dx dy \quad (3.61)$$

where  $D$  is the definition domain of an image and 's' is the area of  $D$ . Commonly, it is hard to precisely get the density function of the image gray distribution; but it can be

replaced by the gradation histogram of the digital image in practice. Gradation histogram is a graylayer function, which indicates a corresponding relationship between each graylayer of a digital image and the probability of appearance of this graylayer. If the sum of pixels of a digital image is  $N$ ,  $n_k$  is the total number of pixels that have graylayers  $r_k$ , and  $L$  is the total number of possible graylayer, then the probability of occurrence of graylayer  $r_k$  is:

$$r_k = \frac{n_k}{N}, \quad k = 0, 1, 2, \dots, L-1 \quad (3.62)$$

### 3.4.4 Types of Watermark Attack

An attack is any disrupting of the watermark detection or any unauthorized processing of the watermark information. In order to destroy the image's watermark, the attackers always try to eliminate the validity or content of the watermark, So that the corresponding watermark detection system cannot recover the correct signal. Existing attacks include four attacks:

- (1) Removal attack is the most common method of attack. It attacks the robustness of a digital watermark. Hence, it can also be referred to as robust attacks. It aims to remove the watermark information without adversely affecting the image or other data. It attempts to weaken the strength of the carrier in the watermark or damage the watermark. Such attacks can be further subdivided into: a simple attack (also known as wave attack or noise attack) and offensive removal.
- (2) Presentation attack (Synchronization Attacks) is one that aims to render watermark detection or recover impossible. There are three typical methods in this type of the attack: Geometrical Distortions Attack, Mosaic Attack and Oracle Attack.



- (3) Interpretation attack is not able to erase the watermark, but it can detect the watermark. It is possible to create false raw data and false embedded watermark to bring confusion. It is a protocol level attack based on the concept of reversible watermarks. It is originated from IBM's U.S. watermarking research team watermarking. (Craver *et al.*, 1998) There are two ways to stop attacks: one is to embed a time-stamp in the image, the other is to use a non-reversible watermark embedding algorithm.
- (4) Legal attack is not a technical one. The attackers intend to use such attacks in the courts. There is no need for the technical or scientific evidence to conduct in this kind of attack.

The classification of these common image watermarking attacks introduced above can help to recognize the shortcomings and limitations of the existing watermarking technology, but also to statistically analyze the performance of the image watermarking schemes, so as come up with better digital image watermarking.

### **3.5 Summary**

This chapter introduces the characteristics of the Lorenz system and the Chen system. Based on these characteristics, a robust chaotic watermarking system was proposed. With the combination of watermarking performance and the human visual system, the integrated technology of frequency-domain was applied. The limitation of DFT and DCT watermarking were found through in-depth research and experiments. Compared with DCT, DWT has advantages of multi-solution and fast wavelet decomposition algorithm. Although the DCT transformation is covered to multi-scale analysis of human vision, DCT watermark is of great practical value due to the extensive use of *JPEG* standard. In this chapter, it is proposed that a blind watermark be generated by a Logistic chaotic system and embedded by DCT.

## Chapter 4

---

### **An Encryption Algorithm Based on Improved Henon Map**

The chapter introduces and compares the several of chaotic maps (such as Logistic Arnold and Henon) to apply on image encryption, and the Henon map is proposed for image encrypting. Henon map is capable to generate non-uniform distribution of sequence. It adopts hyperchaos to process embedded watermark in order to get more suitable Pseudorandom Binary Sequence (PRBS), hyperchaos is generated by the combination of Henon map and other chaotic maps. Through the cascade of chaotic map and dynamically change the number of iterations, the cipher-text is more complex and unpredictability. In this chapter, all of algorithms are implemented in Matlab for computer simulations.

## 4.1 Analysis and Comparison of Several Chaotic Systems

The chaotic phenomena is a quasi-random process in a nonlinear system. Even if a set of initial conditions cannot be experimentally discriminated each other, the derived phase space orbits will exponentially separate over time in chaotic systems. As a result, these orbits will appear irrelevant, and it is incredible that they originate from almost the same initial conditions. Therefore, chaos synchronization is the core of various message encryption algorithms discussed in the chapter, and consequentially using demonstrated in computer simulations. After the chaotic encryption algorithm was proposed, there have been many cryptosystems based on widely-applied one-dimensional chaotic maps or a type of chaotic map, such as logistic map, 2D Arnold map and so on. Therefore, we try to utilize maps to achieve complex hyperchaotic system .

Not all chaotic systems are suitable for encryption system (Kocarev & Jakimovski, 1998), they must be satisfied some conditions like being sensitive to initial conditions, topologically mixing, and their periodic orbits must be dense. The chaotic cryptosystem generates an XOR operation to produce ciphertext or plaintext. The algorithm is simple, easy to implement, and fast; its security depends on chaotic encryption sequence; however, the cipher generated by adopting only one initial value and parameters of one chaotic system is not secure enough, for example, the Logistic map is a one-dimensional discrete chaotic map that is widely applied, but it is not a good robust chaotic system for an image encryption scheme, because Logistic map is characterized by small key space and weak security (Kocarev, 1998).

In order to improve security to secret key, try to compound different chaotic sequence according to new way. In the section, the Lyapunov exponents of three kinds of chaos are analyzed below, i.e., Logistic map, Arnold map and Henon map.

### 4.1.1 Logistic Map Analysis

Discrete chaotic of Logistic map is defined as follows:

$$x_{n+1} = \mu x_n (1 - x_n), \quad x \in (0,1) \quad (4.1)$$

The parameter  $\mu$  can be divided into three segments: when  $\mu=\mu_0=3.569945672$ , the system is in the periodic motion without chaotic behavior; when  $\mu_0 < \mu \leq 4$ , the Logistic map  $x_n, (n=0, 1, 2, 3\cdots)$  is in chaotic state but it is not periodic, non-convergent; and is very sensitive to initial values; when  $\mu=4$ , the map has chaotic behavior with ergodic in  $(0, 1)$ .

The Lyapunov exponent is analyzed with Logistic map as the example, as the Logistic map is one-dimensional, the system has only one Lyapunov exponent, which is shown below:

$$\begin{aligned} \text{if } & x \in (0,1), \quad f(x) = \mu x(1-x), \\ \text{then } & \frac{df(x)}{dx} = \mu(1-2x), \end{aligned}$$

The Lyapunov exponent of Logistic map is shown below:

$$\lambda = \lim_{z \rightarrow \infty} \frac{1}{n} \sum_{i=1}^z \ln |\mu(1-2x)|_{x=x_i} \quad (4.2)$$

The simulated Lyapunov exponent curve can be drawn in the Logistic map bifurcation diagram (figure 4.1), where the abscissa represents the value of  $\mu$ , and the ordinate represents the corresponding value of Lyapunov exponent. From the diagram, it is found: when  $\mu=3.57$ , Lyapunov exponent exceeds 0, which indicates there is chaotic behavior. (Argyris, *et al.*, 1994)

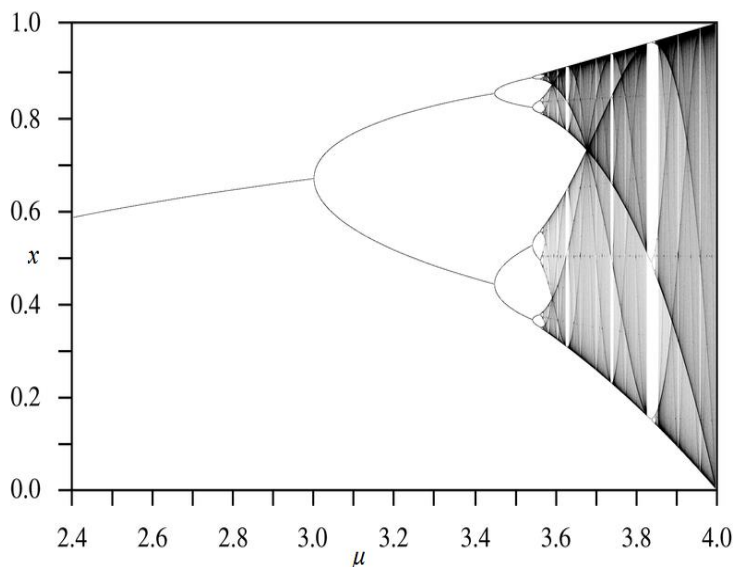


Figure 4.1 Logistic map bifurcation diagram

(Verhulst, 1845)

The parameter  $\mu$  for the Logistic map random sequence is as shown in figure 4.2, where initial values are: red line  $x_1= 3.57$ , blue line  $x_2= 3.75$ . When  $k \in [1, 0]$ , the system has two positive Lyapunov exponents. We found that these are coupled systems. The one-dimensional chaotic system is easy to be cracked, so most Logistic maps combine two or more equations to ascend to higher order dimensional, or group with other kinds of chaotic maps to comprise complex chaotic systems.

(Ponomarenko VI & Prokhorov MD, 2002)

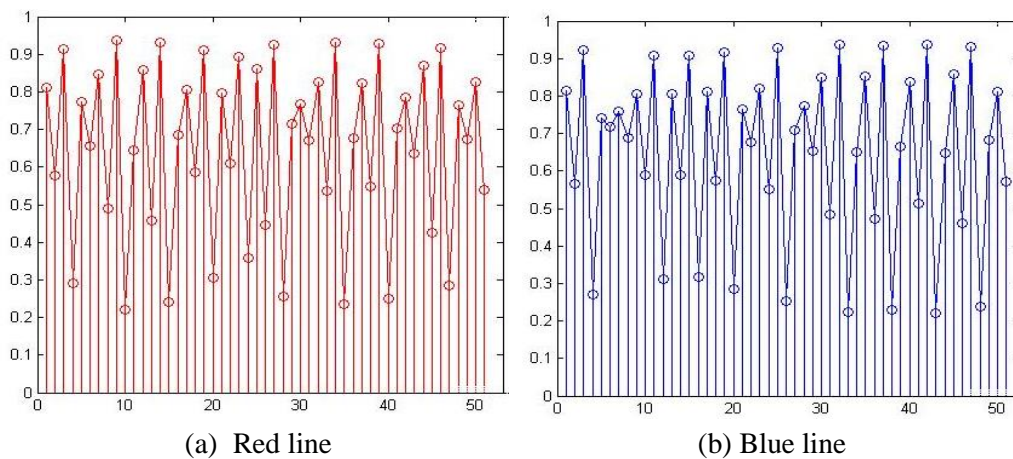


Figure 4.2 Phase portrait of the 2D Logistic sequence

### 4.1.2 Arnold Map Analysis

Arnold map was firstly proposed by Arnold in 1968, an equation of two-dimensional Arnold map is shown below (MathWorld 2007):

$$\begin{cases} x_{n+1} = (x_n + y_n) \bmod 1 \\ y_{n+1} = (x_n + k \times y_n) \bmod 1 \end{cases} \quad (4.3)$$

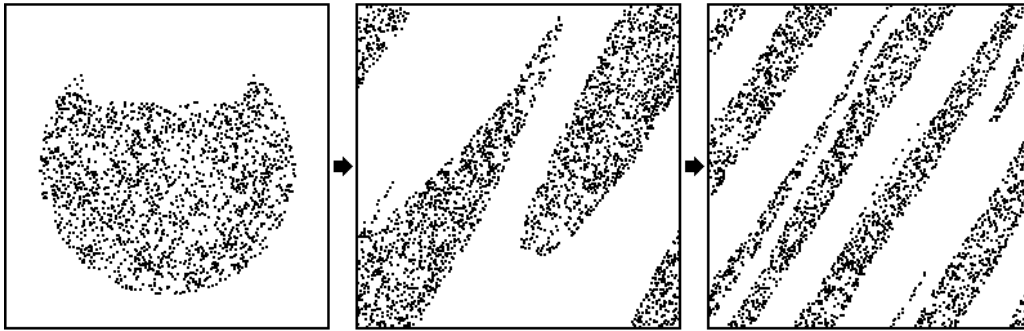


Figure 4.3 Arnold map

Where notation ' $x \bmod 1$ ' means to take the fractional part of a real number  $x$  only by adding or subtracting an appropriate integer; therefore  $(x_n, y_n)$  is confined in a unit square of  $[0,1] \times [0,1]$ . The transforming formula 4.4 in matrix form is shown as below (MathWorld 2007):

When  $k=2$

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & k \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} = C \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod 1 \quad (4.4)$$

Firstly, the determinant of matrix  $C$  is stretched by linear transformation and then folded by modulo operation, thus the Arnold map in figure 4.3 is an area-preserving map and the determinant of its linear transformation matrix, too. The map appears to

be in chaotic states; and it is an one-to-one map; hence, the watermark pixels of different positions will be moved to different embedding positions. Formula 4.4 defines matrix  $C$  where  $|C| = 1$ , the Arnold map is an area-preserving map without attractor. Meanwhile, the Arnold map is an one-to-one map, each point in unit matrix is uniquely transformed to another point in the same identity matrix. Arnold map has a very typical characteristic of chaotic motion: stretching ( $x, y$  can be larger after multiplying by matrix  $C$ ) and folding ( $x, y$  can be withdrawn back to the identity matrix by taking its modulo). (MathWorld 2007)

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} = C \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod N \quad (4.5)$$

The above Arnold map is formulized as follows. Firstly, the phase space is generalized to  $[0, 1, 2, \dots, N-1] \times [0, 1, 2, \dots, N-1]$ , that is only positive integers from 0 to  $N-1$  are taken; second, it is generalized to two-dimensional invertible area-preserving equation as in formula 4.6 (MathWorld, 2007):

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}^n \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = C^n \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \bmod N + 1 \quad (4.6)$$

Where  $a, b, c$ , and  $d$  are positive integers, and  $|C| = ad - bc = 1$  holds. Therefore, in this context, only three out of these four parameters are independent, say, we can let  $a, b$  and  $c$  be independent while  $d$  depends on the area condition  $|C|$ . To evaluate the property of the Arnold map, the following experiment has been carried out

Step 1. Set initial value  $x_1=0$ ;  $y_1 =0.15$ ;  $k=2$ , and run 50 iterations for  $x_{(n+1)}$  and  $y_{(n+1)}$  as per formula (4.3).

Step 2. Select the first 9 iterations of the experimental data of  $x_{(n+1)}$ , as shown in table 4.1 and figure 4.4.

Step 3. Repeat step 1 by adopting  $y_{(n+1)}$  and choose the first 9 significant digits. The result is shown in table 4.2 and figure 4.5.

Table 4.1 Iteration experiments of Arnold map  $x_{(n+1)}$

Times(s)	1	2	3	4	5	6	7	8	9
$x_{(n+1)}$	0	0.15	0.45	0.2	0.15	0.25	0.6	0.55	0.05

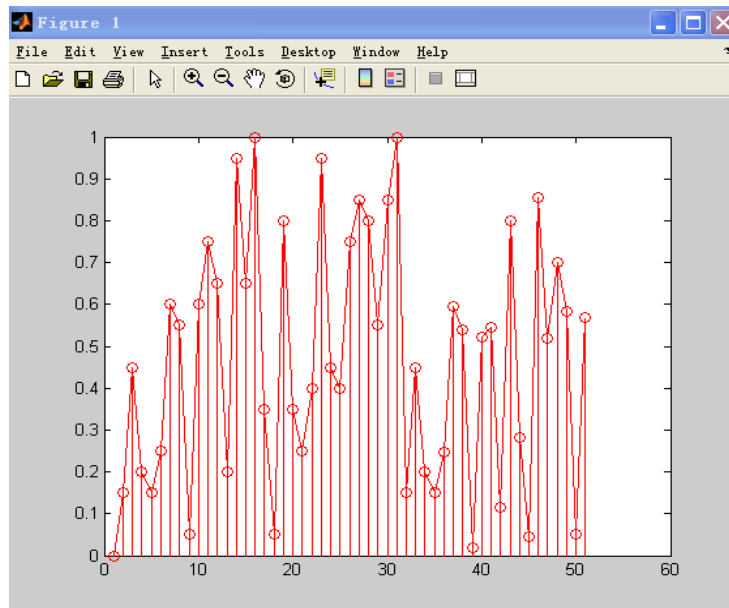
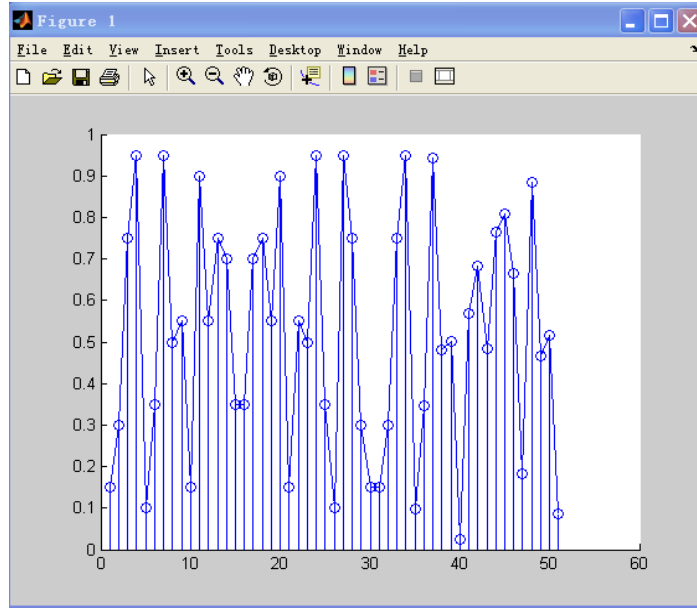


Figure 4.4 Iteration property of Arnold map  $x_{(n+1)}$

Table 4.2 Iteration experiments of Arnold map  $y_{(n+1)}$

Times(s)	1	2	3	4	5	6	7	8	9
$y_{(n+1)}$	0.15	0.3	0.75	0.95	0.1	0.35	0.95	0.5	0.55



Figure 4.5 Iteration property of Arnold map  $y_{(n+1)}$ 

The random sequence that appears in the experiment is not uniform. According to the result of the Arnold map which is a good candidate for perturbation, it can return to its original state after  $n$  iterations, though it appears to be random. It is relatively easy for trackers to get cipher, which is needed to extend from 2D Arnold to 3D Arnold; it is called as 3D Arnold Map (3DAM). The method of extending the 2D Arnold map to 3DAM is described by formula 4.7, or the 3DAM expression.

$$\begin{cases} x_1(n+1) = \text{mod}((x_1(n) + x_2(n) + x_3(n))1) \\ x_2(n+1) = \text{mod}((x_1(n) + k x_2(n))1) \\ x_3(n+1) = \text{mod}((x_1(n) + x_3(n))1) \end{cases} \quad (4.7)$$

It is known from the equations that when  $k=2$ ,  $n \geq 0$ ;  $0 \leq x_1(n) < 1$ ,  $0 \leq x_2(n) < 1$ , and  $0 \leq x_3(n) < 1$ , then the initial value is:

$$x(0) = \left( 0, \frac{1}{\sqrt{2}}, 2 \right)^T \quad (4.8)$$

The system is chaotic. The following figure 4.6 adopts 50 iterations. It shows that the point set distributes uniformly.

Table 4.3 Generalized 3DAM

Times(s)	1	2	3	4	5	6	7	8	9	10
$x_1(n+1)$	0	0.35	0.85	0.35	0.5	0.1	0.55	0.4	0.85	0.65
$x_2(n+1)$	0.15	0.3	0.95	0.75	0.85	0.2	0.5	0.55	0.5	0.85
$x_3(n+1)$	0.2	0.2	0.55	0.4	0.75	0.25	0.3	0.9	0	0.5

It is shown from the figure 4.6 that the Arnold map is a kind of chaotic sequence, which is random to the location of scrambling. But it cannot be exhaustive at all the possible locations. (Chang & Chen, 2001)

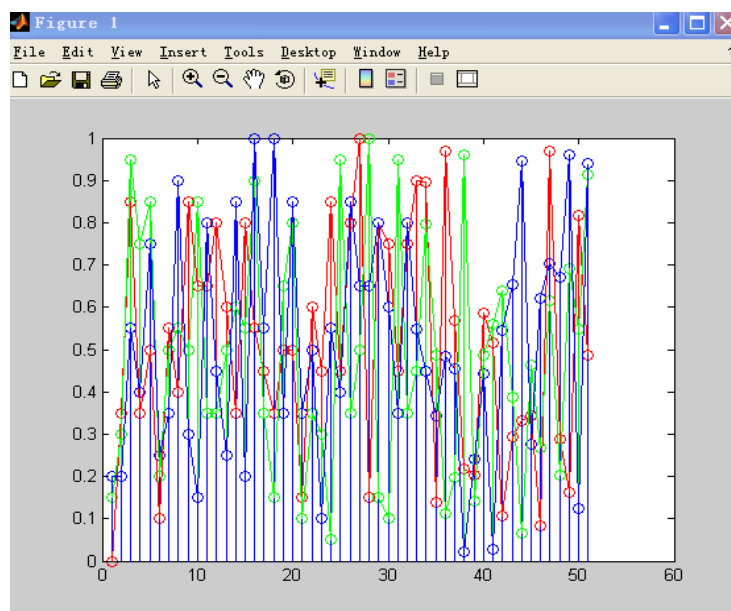


Figure 4.6 50 iterations of 3DAM

### 4.1.3 Henon Map Analysis

The 3D Lorenz and Chen's system have been introduced in section 3.13; they are continuous dynamic systems for solving differential equations and methods for numerical integration; after several iterations, different numerical sequences will be generated for a subtle change in the course of the motion. In this chapter, Henon map will be introduced as follows. 2D Henon is proposed by French astronomer Michel

Henon (1976). It is shown from analysis results that the Henon map in figure 4.7 is a 2D non-linear map (Gonchenko, 2005); its formula (4.9) is presented as follows:

$$\begin{cases} x_{n+1} = 1 - ax_n^2 + y_n \\ y_{n+1} = bx_n \end{cases} \quad (4.9)$$

Where parameters  $a$  and  $b$  are the variables of bifurcation, and  $n= 0, 2, 3 \dots$



Figure 4.7 Henon map

(Sonls, 1996)

The Jacobian matrix is applied here to calculate the system's Lyapunov exponents. Taking the derivative of the right sides of both above equations for  $X_n$  and  $Y_n$ , the Jacobian matrix is presented as follows (Gradshteyn, *et al.* 2000):

$$j = \begin{vmatrix} -2ax_n & \dots\dots & 1 \\ \dots\dots & b & 0 \end{vmatrix} = -b \quad (4.10)$$

When  $b= \pm 1$ , Henon map appears in a converged state; when  $b \in (-\infty, -1) \cup (1, +\infty)$ , Henon map appears in a bifurcated state; and when  $b \in [-1, +1]$ , Henon map appears dissipative. Two Lyapunov exponents of the system are shown in figure 4.8.

Therefore, Henon map is capable of not only generating chaotic phenomenon but also featuring outstanding cryptographic functions; it is very sensitive to initial values, and different chaotic sequences with large translation can be generated by the adjustment of parameters and initial values of Henon chaotic map, indicating that Henon map is suitable for generation of cryptographic functions, due to the capability of generating massive chaotic sequences; another but the most important point is that it is aperiodic and non-convergent, so it has excellent pseudorandomness and unpredictability.

Henon map is discrete which can be also treated as a combination of three maps, relative to the 2D Logistic map and 3DACM. Hitzel continued to study Henon map and developed it into three-dimensional phases as it refers to formula (4.11). (Richter, 2002)



Figure 4.8 The Lyapunov exponents versus for Henon map

(Sprott, 1997)

When  $1.54 < |a| < 2$ ,  $0 < |b| < 1$ ,

$$\begin{cases} x_{n+1} = a - y_n^2 - bz_n \\ y_{n+1} = x_n \\ z_{n+1} = y_n \end{cases} \quad (4.11)$$

The Henon map generated from this chaotic attractor is more complex than the maps from other chaotic attractors; when  $1.54 < |a| < 2$ ,  $0 < |b| < 1$ , chaotic phenomenon occurs and the chaotic attractor is represented as in Figure 4.9 below.

The Henon system features chaotic behaviors. Here below is an analysis of Henon data, the initial values are as follows:  $x=1$ ,  $y=0.1$ ,  $a=1.4$  and  $b=0.3$ ; conduct test as formula (4.9) for 50 iterations and get the regions of various dynamical behaviors. More sequence description is corresponding to table A-1, which is shown in figure 4.9.

	1	-0.3	1.174	-1.0195	-0.103	0.6792	0.3231
$x_{(n+1)}$	1.057	-0.468	1.009	-0.567	0.8519	-0.186	...
	0.1	0.3	-0.09	0.3522	-0.030	0.2037	0.0969
$y_{(n+1)}$	0.3172	-0.140	0.3028	-0.170	0.2555	-0.055	...

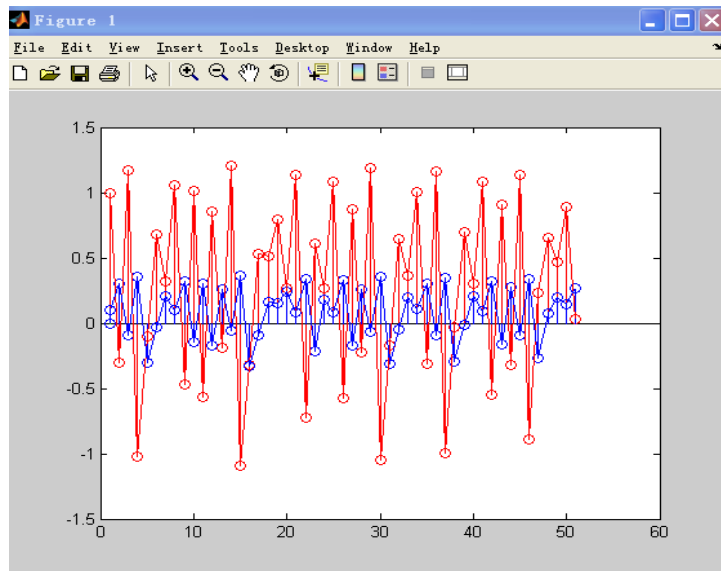


Figure 4.9 The mean and variance of 2D Henon map

The mean and variance of the 2D Henon map respectively are  $M_x=0.26080$ ,  $V_x=0.5162$ ,  $M_y=0.0782$ , and  $V_y=0.0464$ . Where  $x$  and  $y$  in the range, the sequence of uniform distribution of mean and variance should be very close to 0. Figure 4.9 shows the Henon system does not satisfy uniform distribution. In this perspective, more desirable random properties can be generated by Henon map.

As shown in figure 4.10, the chaotic attractor of generalized 3D Henon map is more complex than that of 2D Henon map, its controls and synchronization become more difficult; therefore, it is more advantaged in security and confidentiality for encryption

communication.

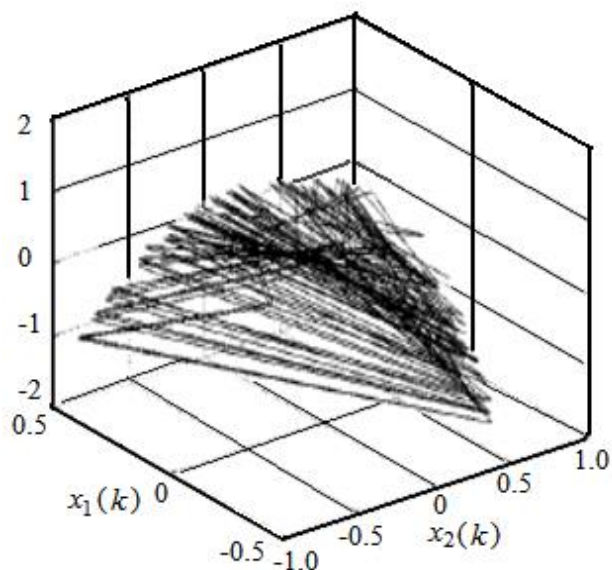


Figure 4.10 3D Henon chaotic attractor

The initial value are  $x=1$ ,  $y=0.1$ ,  $z=0$ ,  $a=1.6$ ,  $b=0.2$ ; do test as formula (4.11) for 50 times iteration. Figure 4.11 shows 3D Henon map generates  $x_{(n+1)}$ ,  $y_{(n+1)}$ , and  $z_{(n+1)}$ , which is random sequence Henon.

$x_{(n+1)}$	1	1.59	0.58	-1.1281	0.9456	0.2113	0.9314
	1.3661	0.6901	-0.452	0.8505	1.256	0.9671	...
$y_{(n+1)}$	0.1	1	1.59	0.58	-1.1281	0.9456	0.2113
	0.9314	1.3661	0.6901	-0.452	0.8505	1.256	...
$z_{(n+1)}$	0	0.1	1	1.59	0.58	-1.1281	0.9456
	0.2113	0.9314	1.3661	0.6901	-0.452	0.8505	...

In summary, properties of Henon map are (1) invertible, the Henon map is better than the Logistic map; (2) Henon map is dissipative; (3) The parameter values of a Henon map fall within a region that gets mapped inside itself; (4) the Henon map is infinity. So the Henon map is a suitable method for generating pseudo-random sequences.

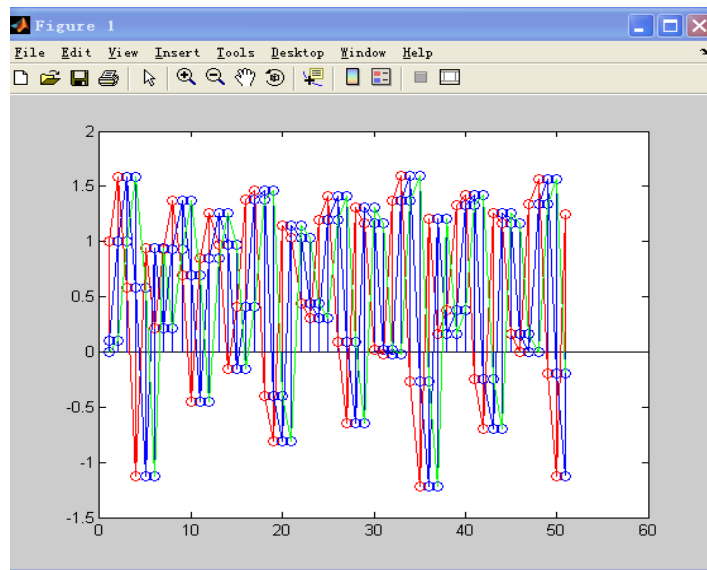


Figure 4.11 The mean and variance of 3D Henon map

## 4.2 Confirmation of Hyperchaos through Numerical Simulation

This section aims to introduce the detailed steps of a hyperchaotic encryption system which has two or more positive Lyapunov exponents. Typical examples are hyperchaotic encryption systems generated by the Logistic map, Arnold map and Henon map separately. Merging different chaotic maps can yield different arbitrary Henon hyperchaotic systems. It is shown from the above diagram (Figure 4.11) that the sequence generated by Henon map is not uniform in distribution. Therefore, it is possible to generate ideal random chaotic system by merging it with other chaotic sequences. Initial values and control parameters of Henon map are operated in chaotic state; it can be improved by other chaotic (ideally hyperchaotic) maps, which can be proven by the diagram and sequence values in the following experiment.

### 4.2.1 3D Henon Map and Logistic Map

- (1) The 3DHL of chaotic attractor is generated by the 3D Henon map and Logistic map. It is 4D hyperchaotic system when the parameters are taken  $a=1.6$ ,  $b=0.2$ ,  $\mu=3.75$ . The Lyapunov exponent shows:  $L_1=0.6376$ ,  $L_2=-1.8416$ . Generate the following program that can be used Matlab:

```

x(n+1)=1.6-y(n)^2-0.2*z(n);
y(n+1)=x(n);
z(n+1)=y(n);
x(n+1)=3.75*x(n)*(1-x1(n));

```

The last line of the program indicates disturbing signal. Consider synchronization of two chaotic system and controls, so the ' $x(n+1)$ ' is used in the proposed algorithm. The generated statistical orbit of the hyperchaotic (Henon map and Logistic map) system is shown in figure 4.12:

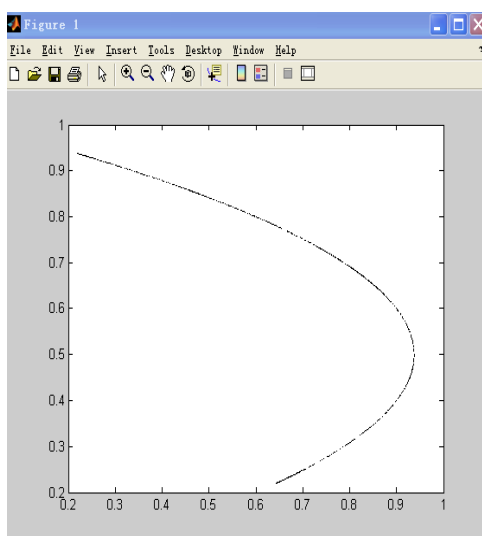


Figure 4.12 Dynamical behaviors of 3DHL system

Here, same secret key and block size for every chaotic map is used in the 3DHL system and generated three chaotic pseudo random sequences. They will show up as follows:  $x_{(n+1)}$ ,  $y_{(n+1)}$  and  $z_{(n+1)}$ .

	0.8200	0.5535	0.9267	0.2545	0.7115	0.7697	0.6646
$x_{(n+1)}$	0.8358	0.5145	0.9367	0.2223	0.6483	0.8549	...
	0.1000	0.8200	0.5535	0.9267	0.2545	0.7115	0.7697
$y_{(n+1)}$	0.6646	0.8358	0.5145	0.9367	0.2223	0.6483	...
	0.0000	0.1000	0.8200	0.5535	0.9267	0.2545	0.7115
$z_{(n+1)}$	0.7697	0.6646	0.8358	0.5145	0.9367	0.2223	...

The 3DHL system generates random sequences as shown in figure 4.13. From



this figure we see that there are no peaks in the spectra for both the small and large

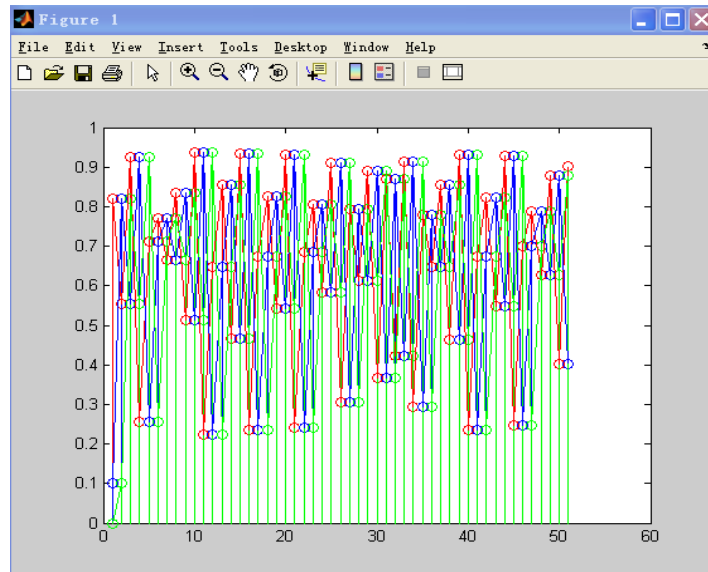


Figure 4.13 The mean and variance of 3DHL system

- (2) The 3DHL of chaotic attractor is generated by the 3D Henon map and one-dimensional Logistic with mean value =0. It is 4D hyperchaotic system when the parameters values are taken as  $a=1.6$ ,  $b=0.2$ . The Lyapunov exponent shows:  $L_1= 0.5748$ ,  $L_2=-1.7788$ .

$$\begin{aligned} x(n+1) &= 1.6 - y(n)^2 - 0.2 * z(n); \\ y(n+1) &= x(n); \\ z(n+1) &= y(n); \\ x(n+1) &= 1 - 2 * (n)^2; \end{aligned}$$

Selected sequence of the 3DHL system has the following meaning:  $x_{(n+1)}$ ,  $y_{(n+1)}$  and  $z_{(n+1)}$  three random sequences.

	0.9000	-0.620	0.2312	0.8930	-0.595	0.2914	0.8301
$x_{(n+1)}$	-0.378	0.7136	-0.018	0.9993	-0.997	-0.988	...
	0.1000	0.9000	-0.620	0.2312	0.8930	-0.5952	0.2914
$y_{(n+1)}$	0.8301	-0.3783	0.7136	-0.0186	0.9993	-0.9972	...
	0.0000	0.1000	0.9000	-0.6200	0.2312	0.8930	-0.595
$z_{(n+1)}$	0.2914	0.8301	-0.3783	0.7136	-0.0186	0.9993	...

The generated statistical orbit of the hyperchaotic system is shown in figure 4.14.

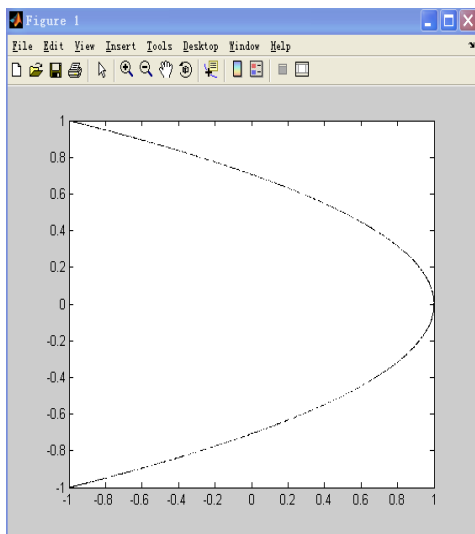


Figure 4.14 Dynamical behaviors of 3DHL system (mean value=0)

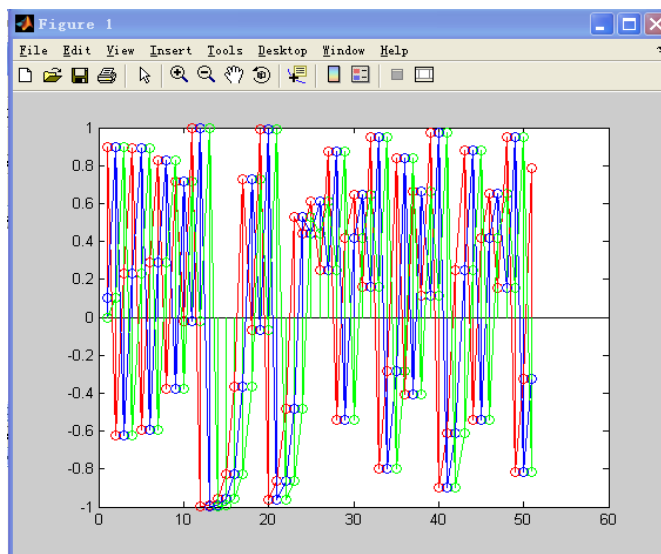


Figure 4.15 The mean and variance of 3DHL system (mean value =0)

Figure 4.15 shows the mean and variance of 3DHL system with initial values  $x = 0.9$ ,  $y = 0.1$ ,  $z = 0$ ,  $a = 1.6$ , and  $b = 0.2$ . This phenomenon can be described by another quantity.

## 4.2.2 3D Henon Map and 2D Logistic Map

The 3DH2DL of chaotic attractor is generated by the 3D Henon map and 2D Logistic map, when the parameters  $a = 1.6$  and  $b = 0.2$ . The Lyapunov exponent shows:  $L_1 =$

0.6514,  $L_2 = -1.8554$ . The corresponding mathematical Matlab code reads: with initial values  $x_{(1)}=1, y_{(1)}=0.1, z_{(1)}=0$  and  $\mu=3.75$ . Solving:

$$\begin{aligned} x(n+1) &= 1.6 - y(n)^2 - 0.2z(n); \\ y(n+1) &= x(n); \\ z(n+1) &= y(n); \\ y(n+1) &= 1 - y(n)^2; \\ x(n+1) &= 3.75x(n)(1-x(n)); \end{aligned}$$

The generated statistical orbit of the hyperchaotic system is shown in figure 4.16:

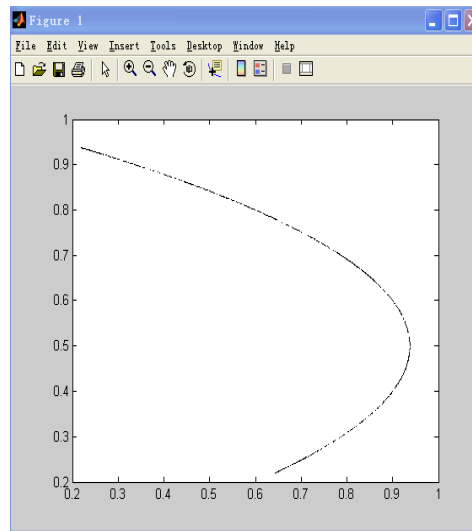


Figure 4.16 Dynamical behaviors of 3DH2DL system

$x_{(n+1)}$	0.82	0.5535	0.9267	0.2545	0.7115	0.7697	0.6646
	0.8358	0.5145	0.9367	0.2223	0.6483	0.8549	...
$y_{(n+1)}$	0.1000	0.9800	-0.9208	-0.6957	0.0318	0.9979	-0.9918
	-0.9676	-0.8726	-0.5231	0.4525	0.5903	0.3030	...
$z_{(n+1)}$	0.0000	0.1000	0.9800	-0.9208	-0.6957	0.0318	0.9979
	-0.9918	-0.9676	-0.8726	-0.5231	0.4525	0.5903	...

Figure 4.17 shows dynamical variance of hyperchaotic 3DH2DL sequence. Selected sequence of the 3DH2DL system has the following meaning:  $x_{(n+1)}, y_{(n+1)}$  and  $z_{(n+1)}$  three random sequence.

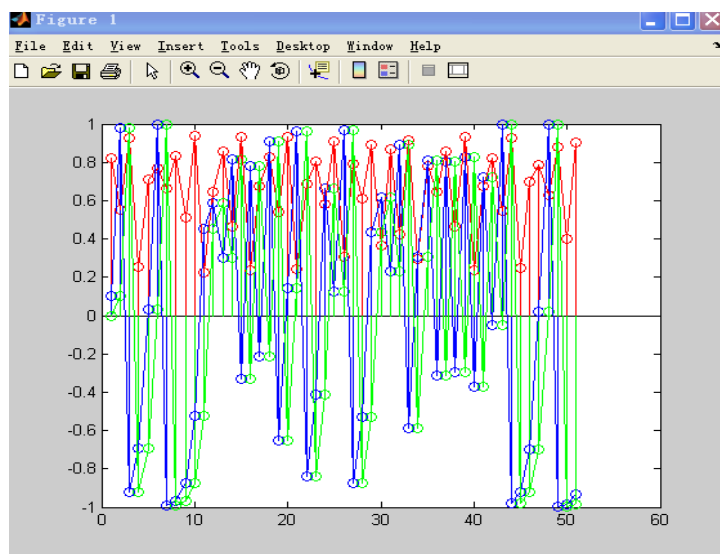


Figure 4.17 The mean and variance of 3DH2DL system

It is shown from analysis on the above experiments that adding an appropriate chaotic perturbation (such as logistic map) to the Henon map will result in a hyperchaotic state. The Henon map perturbed by different Logistic maps will give rise to different plots and different chaotic sequence values.

In general, the iterative equation for Logistic model is simple and only one parameter is applied. This dictates very fast encryption operation, in particular much faster than higher dimension chaotic systems. But its key space is small and it is less secure. Therefore, to take advantage of digital chaotic systems (Logistic and Henon) to construct cryptosystems means that hyperchaotic systems have more complex dynamical behaviors.

### 4.2.3 3D Henon Map and 2D Arnold Map

The 3DH2DA chaotic attractor is generated by the 3D Henon map and 2D Arnold map, when the parameters  $a=1.6$ ,  $b=0.2$ ,  $k=2$ . The Lyapunov exponent shows:  $L_1 = 0.4018$ ,  $L_2 = -1.6057$ . The attractors of 3D Henon map and 2D Arnold map run trajectory curve as represents figure 4.18. With initial values  $x_{(1)}=1$ ,  $y_{(1)}=0.1$ ,  $z_{(1)}=0$ ,  $a=1.6$  and  $b=0.2$ , the corresponding Matlab code reads:

$$\begin{aligned}
 x(n+1) &= a_2 - y(n)^2 - b_2 \times z(n); \\
 y(n+1) &= x(n); \\
 z(n+1) &= y(n); \\
 x(n+1) &= \text{mod}((x(n) + y(n)), 1); \\
 y(n+1) &= \text{mod}((x(n) + ky(n)), 1);
 \end{aligned}$$

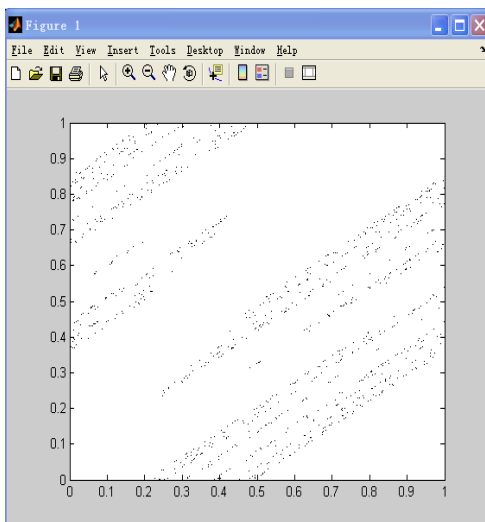


Figure 4.18 Dynamical behaviors of 3DH2DA system

Figure 4.19 is the mean and variance of 3DH2DA system which generates chaotic sequence is given in appendix A-6. The last two lines of the program indicates disturbing signal (2D Arnold map). Consider synchronization of two chaotic system and controls, so the ‘ $x(n+1)$ ’ and ‘ $y(n+1)$ ’ are used in the proposed algorithm.

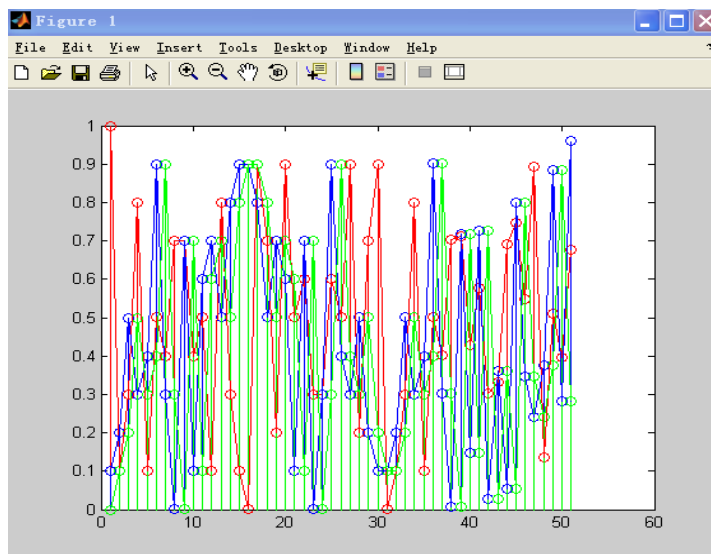


Figure 4.19 The mean and variance of 3DH2DA system

#### 4.2.4 3D Henon Map and 2D Henon Map

The 3DH2DH chaotic attractor is generated by the 3D Henon map and 2D Henon map, when the parameters  $a_1=1.4$ ,  $b_1=0.3$ ,  $a_2=1.6$  and  $b_2=0.2$ , the Lyapunov exponent shows:  $L_1 = 0.6100$ ,  $L_2 = -1.8140$ . Figure 4.20 shows the dynamical behaviors of 3DH2DH chaotic attractor. The plot function can be used to produce the double parabola.

The algorithm is used in Matlab will be generated by the following program:

```
x(n+1)=1-1.4x(n)^2+y(n);
y(n+1)=0.3x(n);
x(n+1)=1.6-x(n)^2-0.2x(z(n));
y(n+1)=x(n);
z(n+1)=y(n);
```

Under these simulation conditions, the following results are quite satisfactory. Figure 4.20 shows the trajectories of the 3D Henon map and 2D Henon map for 50 patterns before training. The last three lines of the program indicates disturbing signal (3D Henon map). Consider synchronization of two chaotic system and controls, so the ' $x(n+1)$ ', ' $y(n+1)$ ' and ' $z(n+1)$ ' are used in the proposed algorithm.

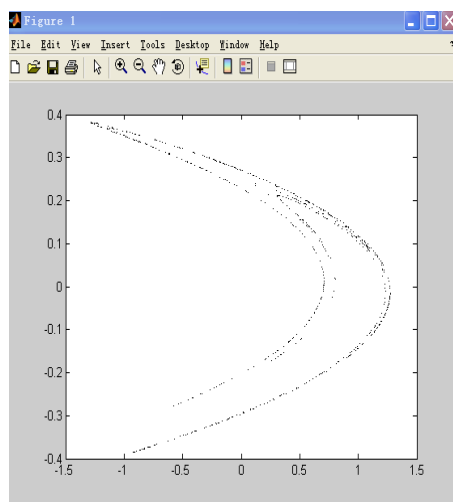


Figure 4.20 Dynamical behaviors of 3DH2DH system

$x_{(n+1)}$	1.0000	-0.300	1.1740	-1.101	-0.103	0.6792	0.3231
	1.0575	-0.468	1.0095	-0.567	0.8519	-0.186	...
$y_{(n+1)}$	0.1000	0.3000	-0.090	0.3522	-0.305	-0.030	0.2037
	0.0969	0.3172	-0.140	0.3028	-0.170	0.2555	...
$z_{(n+1)}$	0.0000	0.1000	0.3000	-0.090	0.3522	-0.305	-0.030
	0.2037	0.0969	0.3172	-0.140	0.3028	-0.170	...

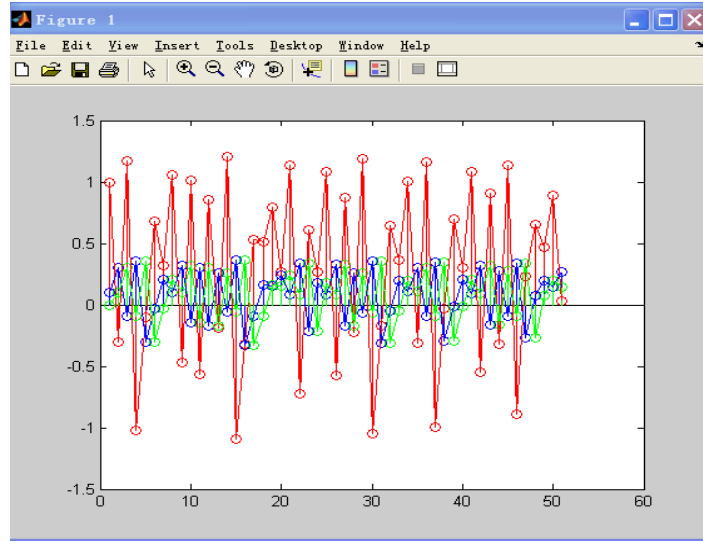


Figure 4.21 The mean and variance of 3DH2DH system

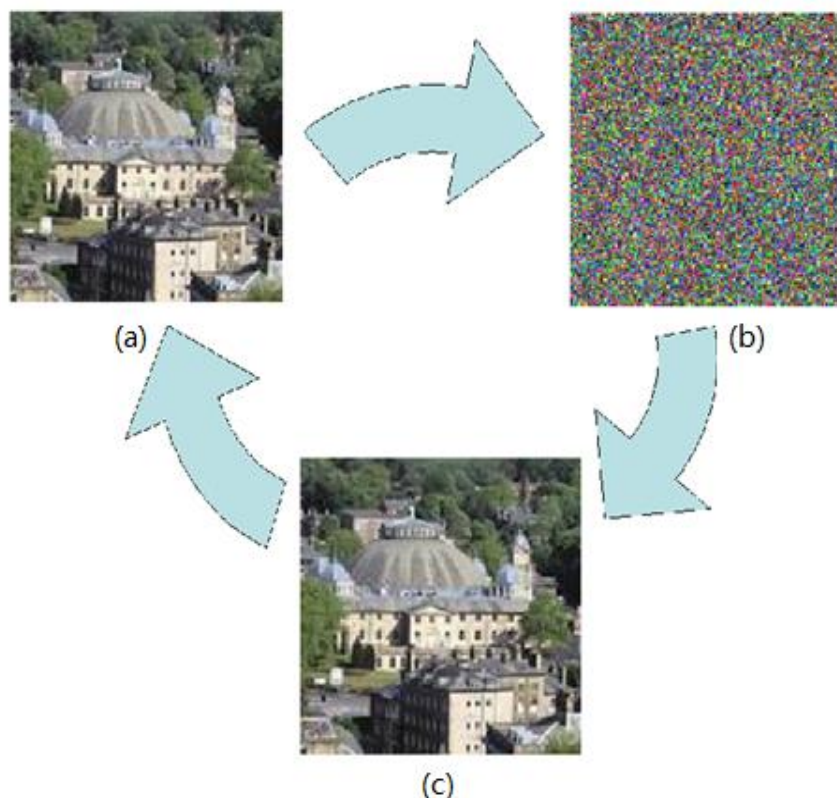
Figure 4.21 shows the corresponding mean and variance of 3DH2DH system with initial values  $x_{(1)}=1, y_{(1)}=0.1, z_{(1)}=0, a_1=1.4, b_1=0.3, a_2=1.6, b_2=0.2$ . There are three different random sequence  $x_{(n+1)}, y_{(n+1)}$  and  $z_{(n+1)}$ , where have different three status from the 3DH2DH system. It is easy to see that exists stable fluctuation in the data.

During the experiment and analysis of three chaotic systems, this section presents an algorithm for encryption and decryption based on chaotic dynamical system. The algorithm can be applied to the file encryption, real-time data stream encryption and anti-fake of commodities.

### 4.3 Experimental Analysis

Based on the above analysis, the Henon map presents a simple 2D or 3D map with

quadratic non-linearity. The proposed algorithm combined two or more chaotic maps to enhance security, space key and anti-attack. This feature is suitable for image encryption. There are images are based on 2D still image, as shown in figure 4.22. Simulation experimental and performance analysis of several hyperchaotic algorithms are provided in following has been done with 2D still image.



(a) Original image, (b) Encrypted image, (c) Recovered image

Figure 4.22 Key sensitive test in the 3DH2DL map

The complete image encryption scheme consists of several steps of operation. First step is key generation, select a select a sequence of 128 bits as the key, and split them into  $n$  groups, which are further mapped onto to several parameters of the 3D Henon map and the Logistic map. So the color castle.bmp of size  $128 \times 128$  is encrypted by using the key ' $a_x, b_x, \mu_x, a_y, b_y, \mu_y, a_z, b_z, \mu_z, k$ '. Second step is piled up the 2D image into 3-dimensional. Then, the plain-image will be scrambled and unknowable. Final is the transform the 3-dimensionioal image cube back to a 2D image. The decrypted image is can be seem clear and correct by use the key. The algorithm is 3DHLL map. Figure



4.22 shows the test result.

#### 4.4 Summary

This paper proposes a hyperchaotic encryption algorithm based on Henon maps, the simulation experiments show that this algorithm is capable of achieving good effects in encryption and decryption. It is shown from the algorithm security analysis that it has large key space, strong sensitivity of encryption key and good statistical characteristics. Therefore, the future research shall focus on the theoretical deduction of improvement method of Henon chaotic sequence and optimization of encryption algorithm, so that it can achieve better encryption/decryption effects and resist diversified attacks. After the condition of periodic or stable state is verified, it is compared with perturbed chaotic sequence, and the influences of initial value and parameter value range of Henon map on cryptographic properties of its generated chaotic sequence are compared and discusses.

- (1) Logistic map: It is easy to generate a chaotic sequence prone to attack; therefore, it is necessary to adopt two or more equations to ascend to higher dimension or combine it with other chaotic maps to build a compound chaotic system.
- (2) Arnold map: Available as 2D or 3D chaotic sequence, the Arnold map is area preserving and the determinant of its linear transformation matrix  $|C|=1$ .
- (3) Henon map: Depicted as high-dimensional, Henon map can comprise two or three chaotic sequences. It is complex, has single nonlinearity or single linearity. It is sensitive to initial values that can merge other chaotic maps such as Logistic map, Arnold map, etc. to form a hyperchaotic map.
  - a. 3D Henon and Logistic
  - b. 3D Henon and 2D Logistic
  - c. 3D Henon and 2D Henon

d. 3D Henon and 2D Arnold

These experiments can illustrate the high-dimensional hyperchaotic system which is more complex and unpredictable; through more ‘chaotic sequences’, different hyperchaotic systems are generated.

## **Chapter 5**

---

# **Experiment of the Hyperchaotic Encryption System**

Three methods are proposed to solve the problem of chaotic attractor overflow during chaotic encryption detailed in this chapter. The first method adopts the initial iteration point reasonably. The second one encrypts one image with multigroup hyperchaotic systems. And third one encrypts one image with the block cipher. These solutions can improve the unpredictability of encrypted images and are proven effective and feasible by numerous experiment researches. In addition, an encryption algorithm is proposed to merge many subimages and to process extra large-scale images, which expands the application of chaotic encryption.

## 5.1 Phenomenon of Chaotic Attractor Overflow

Henon map has significant safety and reliability; it is more complex than any other chaotic maps. So in this chapter, the Henon map is the main research object.

### 5.1.1 Henon Map of Overflow Phenomenon

The phase space of generalized Henon map is of real number set which as representable in Matlab, the maximum real number (realmax) is  $1.7977e+308$  and the minimum real number (realmin) is  $2.2251e-308$ . As same with Matlab, double-precision floating-point numbers can represents the maximum and minimum values in C language, too. Calculation of the large values needs a large number of arithmetic operators.

The 2D Henon map is a reversible iterative map as represented by formula (5.1). The simplified Poincare map (Wiggins, 1990) applies to the Lorenz model, such that:

$$\begin{cases} x_{n+1} = 1 + a x_n^2 + b y_n \\ y_{n+1} = x_n \end{cases} \quad (5.1)$$

in which ‘ $a$ ’ and ‘ $b$ ’ are bifurcation variables.

According to the second equation in formula (5.1), it follows that  $y_n = x_{(n-1)}$ , then Henon map can be written in a single variable formula:

$$x_{n+1} = 1 + a x_n^2 + b x_{n-1} \quad (5.2)$$

The parameter  $b$  is a measure of contraction rate and the Henon map is the most general 2D quadratic map with a property that the contraction is independent of  $x$  and  $y$ .

For equation (5.2), set the random initial value  $(-2, 2)$ , parameter ‘ $a$ ’  $\in (-0.5, 2.5)$ ,

parameter ' $b$ '  $\in (-1.1, 1.1)$ , and step size 0.001 to discuss range value. For Henon map over a range of  $a$  and  $b$  values, a portion of this range (about 6%) yields chaotic solutions as in figure 5.1 below. Figure 5.1 proposed by Wiggins (1990).

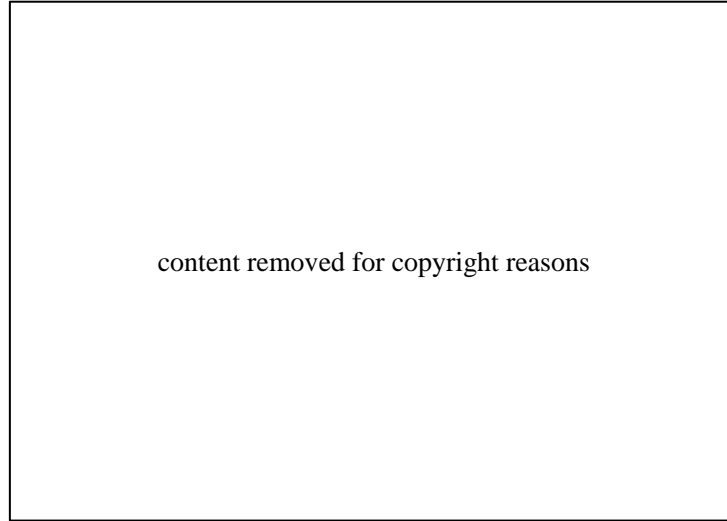


Figure 5.1 Initial values of the bifurcate variables ' $a$ ' and ' $b$ ' for Henon map  
(Wiggins, 1990)

Furthermore, the 2D Henon map is extended to a 3D Henon map, and a simple three-dimensional quadratic autonomous system can be represented by the equation system below:

$$\begin{cases} x_{n+1} = a - y_n^2 - bz_n \\ y_{n+1} = x_n \\ z_{n+1} = y_n \end{cases} \quad (5.3)$$

In order to facilitate the calculation, the map can be rewritten as:

$$x_{n+3} = a - x_{n+1}^2 - bx_n \quad (5.4)$$

Some high-dimensional chaotic maps have inverse operation which is different from one-dimensional chaotic map. The 3D Henon map has reversibility and simple geometric insights (Tangerman & Carvalho, 1999). Using the Lorenz map or other non-linear map need to consider their inverse operations in the encryption algorithm,

such as the inverse trigonometric function. Because of each computer have different precision so produce error is also different, they are difficult to ensure encryption repeatability and decryption correctness. Consequently, such the algorithm is very restricted in the practical application. However, the Henon map has uniformization invertible information processing function is suitable for encryption algorithm. The Henon map coefficient is an integer, so reduce the calculation error. When  $b=1$  the Henon map is invertible, chaotic behavior with simple of formula (5.5):

$$x_n = \frac{x_{n+3} + x_{n+1}^2 - a}{b} \quad (5.5)$$

Thus, theoretical analysis and trial tests reveal a set of parameters:  $x_0=1.59$ ,  $y_0=0.1$ ,  $z_0=0$ ,  $a=1.6$  and  $b=0.2$ . The results are subject computer simulation. See more detailed date of the Henon map in table 5.1 below.

Table 5. 1 The 3D Henon sequence generated

Time ( $t$ )	1	2	3	4	5	6	7	8	9	10
$x_{(n+1)}$	1.59	-0.9481	-1.2461	0.3831	0.2369	1.7025	1.4673	1.3458	-0.8935	-0.5046

### 5.1.2 3D Henon mapping Correlation Coefficient Analysis

Many papers mentioned the Henon map generated sequence password that, when its parameter ranges are  $1.54 < a < 2$  and  $0 < b < 1$ , the system is in the hyperchaotic state. But they did not discussed specifically about the feasibility of hyperchaotic sequence password generated within these parameter valuing ranges and key space influence. In this section, the influences of parameter variation on the hyperchaotic sequence password are mainly discussed. Afterwards, the parameter relationship of Henon map will be discussed, such as the initial value, the divergence rate and chaotic characteristics. As validated, when  $1.55 < a < 1.8$ ,  $0 < b < 0.2$ , and initial value is within  $(-1.5, 1.5)$ , the generalized Henon map system exhibits good password sequence.

The hyperchaotic system was discovered by O.E. Rössler (Rössler, 1979). The 3D Henon is similar to the Lorenz system which with small perturbation provided, is suitable for continuous system control. In any case, the Henon map has been extended to higher dimensions and has diversified control parameters available as passwords.

$$x_{k+1} = F(x_k) + dert \quad (5.6)$$

where the *dert* is the small disturbance quantity, its value is controlled by another chaotic sequence. When the value generated from Logistic map is less than a threshold, let *dert*=*m*, otherwise, let *dert*=0. Because the high-dimensional maps are associated, sequence change in one dimension will affect the entire map trajectory, and thus jump out of periodic state. In essence, the scrambling method is a method of constructing sequence password by combining multiple chaotic or hyperchaotic maps. After scrambled, the hyperchaotic sequence password has better characteristics and becomes more secure.

Divergence speed of the Henon map depends on the select parameter region. In order to determine their relationship, the experiments are designed the range 'a' and 'b' by a certain length. And then evaluate the results of the map whether has stabilized a fixed point to the end of the 10000 iterations. The number of iterations in divergence will be recorded to constant stable state for the parameter values of the Henon map.

As shown in figure 5.2, the initial values are set to [-2, 2], some calculated values will overflow while the parameters 'a' and 'b' are set the same value; in order to avoid overflow, users should provide suitably different the parameters 'a' and 'b', and initial values in [-1.5, 1.5]. Several experiments were done, and if there is no value overflow after iterating 100,000 times, then a stable random sequence can be obtained, proving that the stability of random chaotic sequence is affected by the parameters 'a' and 'b'. The results are shown in figure 5.2, the red points represent no overflow data; the blue points are more stable data; the black points are overflow data.

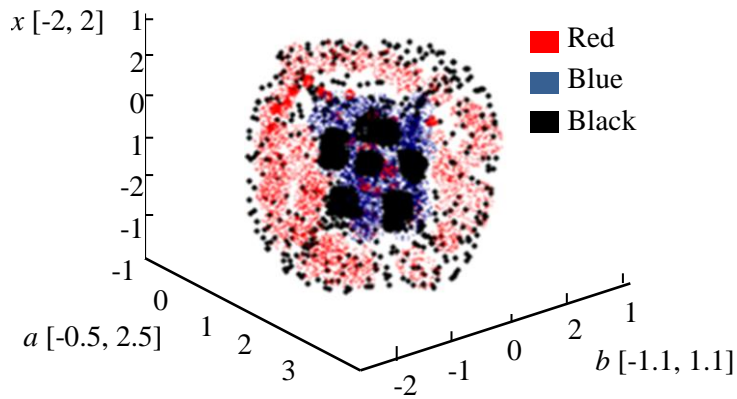


Figure 5.2 The variables 'a' and 'b' initial value of Henon map

The key to generation of the hyperchaotic sequence is the digitization of hyperchaotic map. Generally, sequence password methods are used, including real numerical sequence, bit sequence, the binary sequence, integer value sequence, etc. The focus of this study is to generate hyperchaotic sequence using the binary sequence method. A threshold function  $\Gamma$  is defined by formula (5.7), which yields a real number sequence  $x_k (k=0, 1, 2, \dots)$ .

$$\Gamma(x_k) = \begin{cases} 0 & x_k < T \\ 1 & x_k \geq T \end{cases} \quad (5.7)$$

where  $T \in [0.1, 0.2]$ , and is set depending on valuing range and data distribution of the chosen hyperchaotic system; in the experiment,  $T$  is set to 0.100001. With the same formula,  $y_k$  and  $x_k$  generate the binary sequences, then we mixed the three sequences to obtain a binary hyperchaotic sequence as follows:

$$\Gamma(x_k), \quad k = 0, 1, 2, \dots \quad (5.8)$$

In the infinite precision conditions, the chaotic sequence generates quasi-white noise characteristic. Although chaos is an infinite loop, word-length of a computer is finite. In practice, a chaotic sequence generated often tends to be periodic or converge in stable state.



In the image encryption system, the multi-dimensional Henon map is a continuous iterative equation, which can produce different chaotic sequence. At Henon map capacity increases, chaotic sequence data overflow, which is image encryption failure, will occurs after tens of thousands of iterations (repetition). Data overflow is related with the Henon map to bifurcate rate and hampers chaotic image encryption. The following used Henon-based hyperchaotic method to simulate experiment. The original image is *castle.bmp*, its size for  $128 \times 128$ .

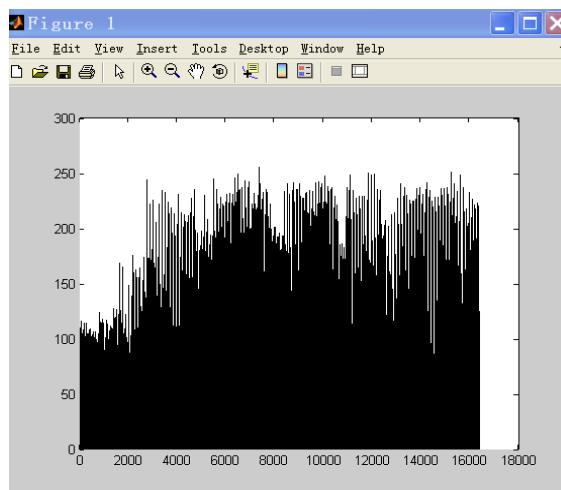
A series of experiments were done by selecting number of initial iteration points at 11,101,201,301,401,501,601,701,801,901 and 1001. Only when 1-101 points were used for the initial iteration position of chaotic sequence, the image scrambling effect was good. While the number of initial iteration points is more than 201, the chaotic attractor is overflowed. The experiment result is shown in figure 5.3.

In order to prove the universality of image encryption technology, the selected images for the experiments were downloaded from University of Derby, such as figure 5.3(a), and figure5.11(a) (University of Derby, 2013). Figure 5.3(a) is the original image. Figure 5.3(c) is the image failing to be scrambled, indicating that the encryption failed in this experiment. Because, the chaotic attractor was overflowed after iterating 100,000 times with the initial iteration position chosen at the 1001 point of chaotic sequence. These overflowed chaotic attractor lost the ability of encryption. Therefore the resulted image still maintains the feature of the original image, as shown in figure 5.3(c).

At this moment the hyperchaotic sequence is difficult to meet secret communication's requirements. In the following the image scrambling will be introduced for overcome limited accuracy. The hyperchaos can use the following program and selected initial values ( $x_1=0.9$ ,  $y_1=0.1$ ,  $z_1=0$ ,  $a_1=1.4$ ,  $b_1=0.3$ ,  $a_2=1.6$ ,  $b_2=0.2$ ). In the hyperchaotic, there are two chaotic maps to scramble the original image: One is 2D Henon map; the other is 3D Henon map. The experimental result is given in figure 5.3:



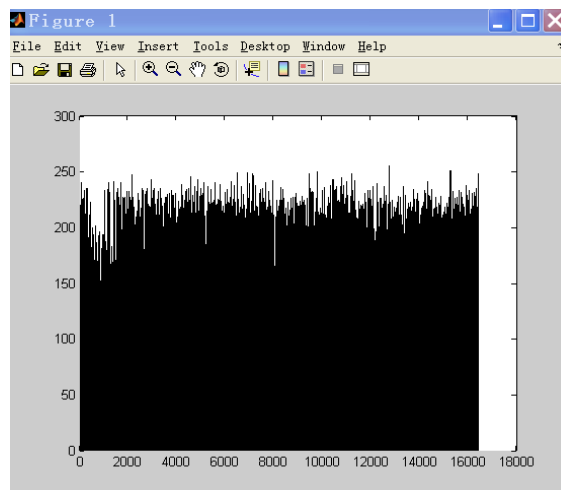
(a) Original image



(b) Histogram of original image



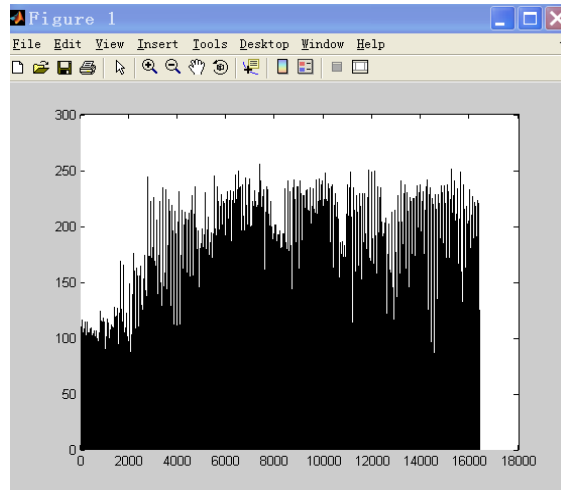
(c) Error encryption image



(d) Histogram encrypted image



(e) Decryption image



(f) Histogram decrypted image

Figure 5.3 Image encryption and decryption experimental result

$$\begin{aligned}
x1(n+1) &= 1 - 1.4 * x1(n)^2 + y1(n); \\
y1(n+1) &= 0.3 * x1(n); \\
x1(n+1) &= 1.6 - y1(n)^2 - 0.2 * z1(n); \\
y1(n+1) &= x1(n); \\
z1(n+1) &= y1(n);
\end{aligned}$$

The effectiveness of the encryption can be evaluated from histograms. An image histogram expresses the statistical characteristics of an image, and is the distribution map of pixels amongst those grayscale values. The horizontal axis ( $x$ -axis) represents the range of gray-value. The vertical axis ( $y$ -axis) represents the probability of occurrence for gray-value. Figure 5.3(d) shows histogram of encrypted image, that its pixels distribution no-uniformly at the left of histogram. Figure 5.3(c) can be seen the encryption is not successful.

In order to realize good encryption, we conducted the following experimental research in three ways, include one hyperchaotic application and adjusting iterations position, multigroup hyperchaotic sequences in RGB sub-image and multigroup hyper chaotic in block image.

## 5.2 Hyperchaotic Application and Adjusting Iterations Position

The testing pattern is a single hyperchaotic based on Henon map. And taking advantage of the properties of a composite hyperchaotic system, we proposed an encryption algorithm based on multiple three-dimensional continuous chaotic sequences.

Given an RGB color image, that can be regarded as a matrix. The matrix elements of the image are represented in the rows and the columns ' $M \times N$ ' in computing. The elements value is just the pixel gray scale (value between 0 and 255). The pixels of a color image can be represented by a mixed matrix including 'R (red)', 'G (green)' and 'B (blue)'. These components are displayed as three digital matrixes in figure 5.4 below.

The function `rgb2gray(imread(FileName))` is to divide a digital image into three of pixel-color components (red, green, and blue). The code is shown below:

```
rgb2gray(imread('Castle.bmp'))
```

Because the digital image can have many formats, such as *bmp*, *jpg*, *fig*, etc., so the storage classes are also different accordingly. The storage classes used in Matlab are unit-8, unit-16 and double.

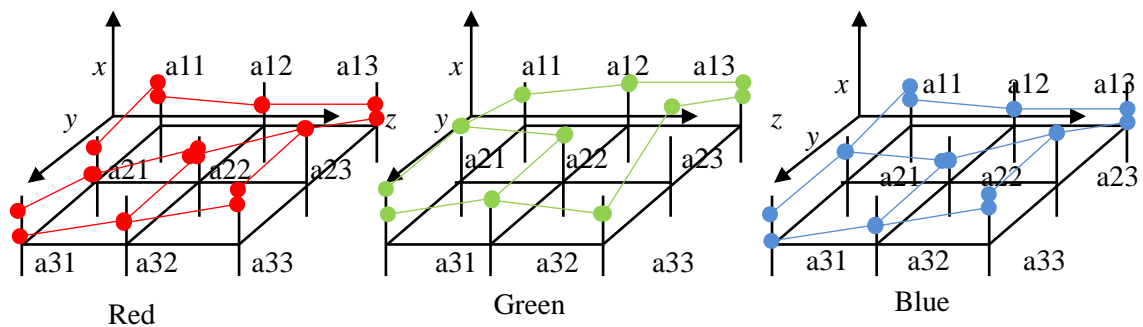


Figure 5.4 Combination of three digital matrices

Stability characteristics are different for each fundamental color. In order to overcome the chaotic attractor overflow, the initial point should be set according to the different fundamental color stability. There are three rules to observe:

- (1) The stability of red fundamental color is the weakest, and the initial point has a range of 1-11 that is aimed at *castle.bmp*. So the initial iterative position of red is set near of the origin coordinate.
- (2) The stability of the green fundamental color is relatively weak and the initial iterative point has a range of 1-1000. So the iterative point of green color is set near the 1000<sup>th</sup> point.
- (3) The stability of the blue fundamental color is the best and the initial point has a range of 1-2000. So the initial iterative position of blue color can be set at near a farther position.

The above rules determine value range according to image format and the type of chaos. So the case of *castle.bmp* used this method.

The proposed algorithm applies to four different chaotic maps named 2D Henon map, 3D Henon map, Logistic map and Arnold map respectively. The hexadecimal mode is used to define the cipher for initial value (for each chaotic map) which generates the scrambling image. The ciphers are used by the following data:

Table 5.2 Parameter values in chaotic range for the R,G,B sub-image at the initial position

Fundamental color	Initial position	Parameter values
Red	11	$x_{(11)}=0.9, y_{(11)}=0.1, z_{(11)}=0$
Green	1001	$x_{(11)}=0.9, y_{(11)}=0.1, z_{(11)}=0$
Blue	2001	$x_{(11)}=0.9, y_{(11)}=0.1, z_{(11)}=0$

In the programme, the processing of R attractor, G attractor, B attractor, and initial positions and initial parameters are defined one type of composite hyperchaos. The codes are shown below. The proposed algorithm uses four different combination of chaotic maps as cipher, where the iteration position is  $n$  where  $n=11$  and  $x_1(11)=0.9$ ,  $y_1(11)=0.1$ ,  $z_1(11)=0$  as encryption key. The 11<sup>th</sup>, 1001<sup>th</sup>, 2001<sup>th</sup> are encryption position of each R, G, B sub-images.

Red attractor

```

x1(11)=0.9;
y1(11)=0.1;
z1(11)=0;
for n=11:10+rh*rw-11;
x1(n+1)=1.6-y1(n)^2-0.2*z1(n);
y1(n+1)=x1(n);
z1(n+1)=y1(n);
x1(n+1)=mod((x1(n)+y1(n)),1);
y1(n+1)=mod((x1(n)+2*y1(n)),1);
end;

```

## Green attractor

```

x1(11)=0.9;
y1(11)=0.1;
z1(11)=0;
for n=1001:1000+rh*rw-1001;
x1(n+1)=1.6-y1(n)^2-0.2*z1(n);
y1(n+1)=x1(n);
z1(n+1)=y1(n);
x1(n+1)=mod((x1(n)+y1(n)),1);
y1(n+1)=mod((x1(n)+2*y1(n)),1);
end;

```

## Blue attractor

```

x1(11)=0.9;
y1(11)=0.1;
z1(11)=0;
for n=2001:2000+rh*rw-2001;
x1(n+1)=1.6-y1(n)^2-0.2*z1(n);
y1(n+1)=x1(n);
z1(n+1)=y1(n);
x1(n+1)=mod((x1(n)+y1(n)),1);
y1(n+1)=mod((x1(n)+2*y1(n)),1);
end;

```

The image shown in figure 5.4(a) is selected as the experimental subject. Because of cyclic iterative of 3D nonlinear Henon map, it not only rearranges the pixel location, but also changes the gray scale of each pixel. As a result, the image becomes unrecognizable and the characteristic of image has iterative function. The encryption image becomes unrecognizable and completely changes the characteristics of histogram.

The image encryption system aims to *bmp* or *jpg* file for chaotic scrambling and pixel diffusion. Figure 5.5 shows one type composite hyperchaos of image encryption system.

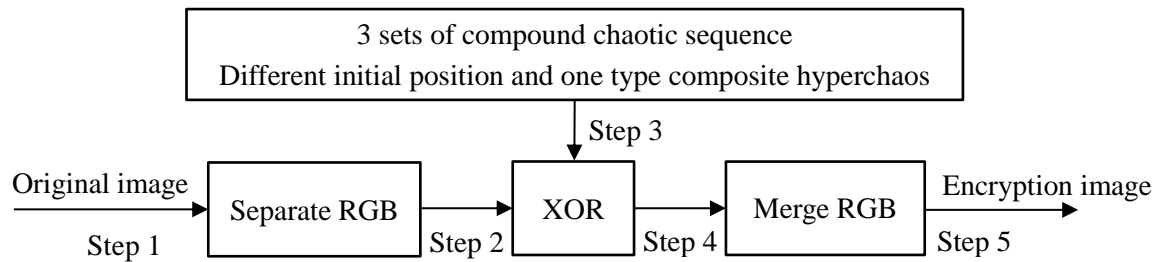


Figure 5.5 One type composite hyperchaos of image encryption system

Steps for proposed encryption algorithm:

Step 1. Select an image to encrypt, with a minimum size of 64 bit.

Step 2. Decompose the color image into three subimages of R, G, and B components, denoted as  $I_R$ ,  $I_G$  and  $I_B$  respectively.

Step 3. Input various initial parameters; perform the XOR operation in four kinds of 3D chaotic Henon maps to generate three compound chaotic sequences, which are then converted into matrices  $C_R$ ,  $C_G$  and  $C_B$  with the same size as  $I_R$ ,  $I_G$  and  $I_B$  respectively. For example, the  $I_R$  is processed with to obtain the final encrypted image  $C_R$ .

Step 4. Merge the encrypted R, G, and B subimages into the final encryption image.

Step 5. Output the encryption image.

The corresponding architecture is shown in figures 5.6, 5.7 and 5.8, in which the confusion images are obtained by 3 sets of compound chaotic sequences respectively. After encryption, each layer (R, G, B) becomes illegible and the corresponding histogram also becomes very smooth and uniformity. Figure 5.8(d) is clearly different from the original histogram, and demonstrates that all gray values have been uniformly distributed over the whole gray interval of pixels after encryption. Hence the encrypted image has good statistical characteristic, which can resist certain attacks.

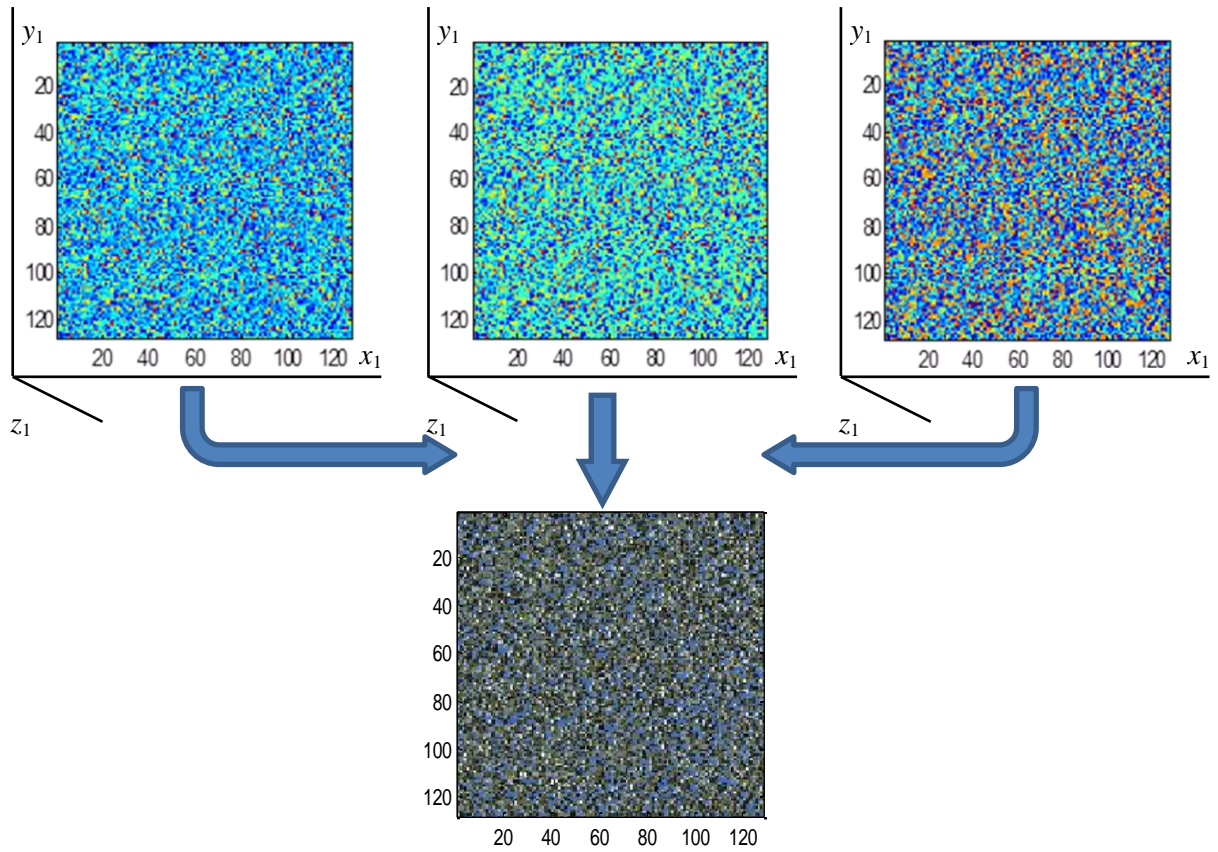


Figure 5.6 Encryption of *castel.bmp* and R, G, B pixel rearrangement

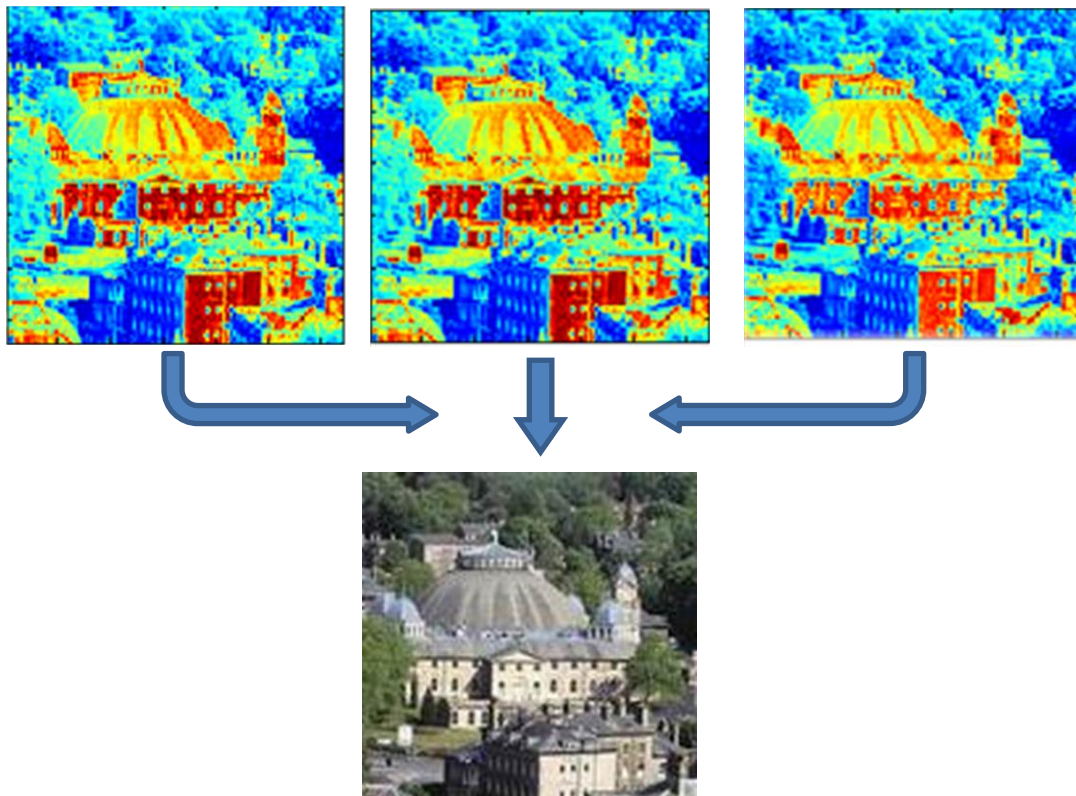
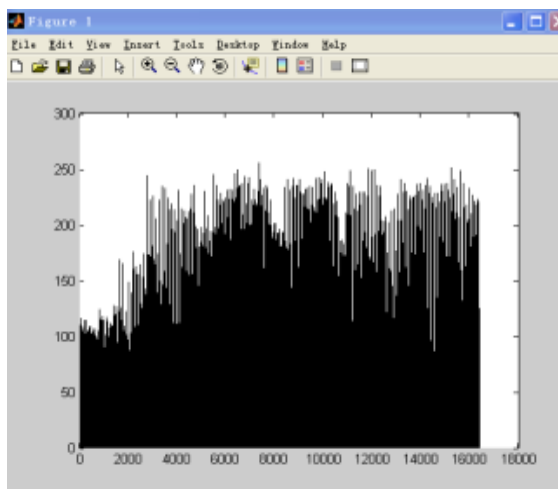


Figure 5.7 *Castle.bmp* recover process

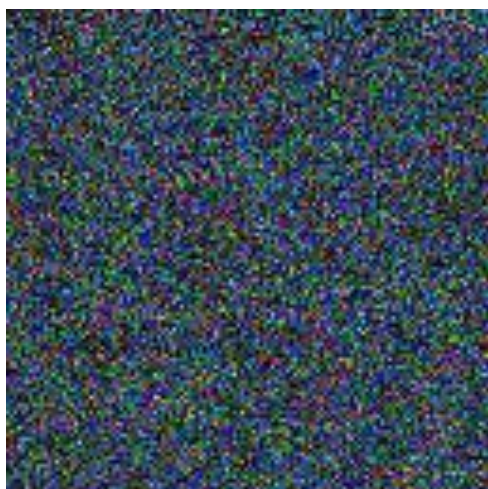




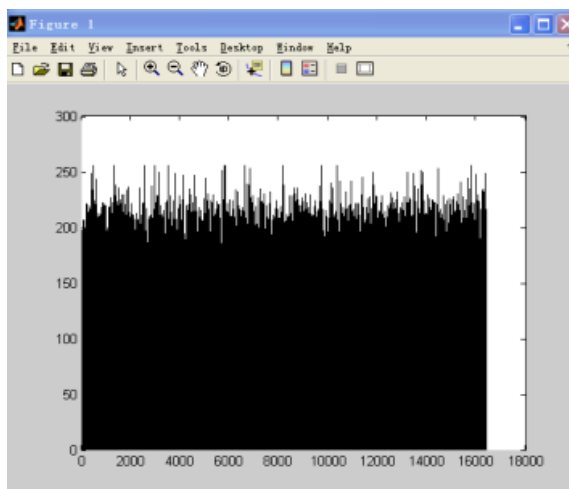
(a) Original image



(b) Histogram of original image



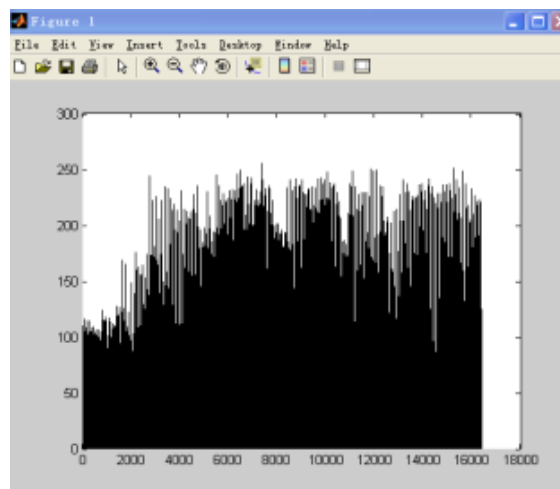
(c) Encrypted image



(d) Histogram of encrypted image



(e) Decrypted image



(f) Histogram of decrypted image

Figure 5.8 Image encryption and decryption experimental result

### 5.3 Multigroup Hyperchaotic Sequences in RGB Sub-image

The current scrambling algorithms have popular problems of insufficient encryption intensity demonstrates and small key space. To address these problems, this paper applied multigroup chaotic sequence and matrix straightening operator to digital color image scrambling, and proposed a digital color image scrambling algorithm based on multigroup chaotic sequence. The algorithm can realize both pixel position scrambling and pixel grayscale scrambling, enlarge the key space. The multigroup hyperchaotic combination is easily executable for RGB sub-images encryption, and is flexible with several different space cycles; it is highly safe with ideal scrambling effect.

According to the characteristics of a compound hyperchaotic system, a digital image encryption algorithm based on multiple three-dimensional continuous chaotic system sequences is presented. Different compound chaotic computation processes are involved during both encryption and decryption of this algorithm. Nine independent compound hyperchaotic sequences are generated according to the parameters.

In the same way, three color primary matrixes are extracted from the digital image, red matrix represents  $R(I)$ , green matrix represents  $G(I)$  and blue matrix represents  $B(I)$ . To understand that multigroup hyperchaotic map used in RGB sub-image, use the column vector coordinate  $(x, y, z)$  to represent. Therefore, R, G, B phase portraits have three complex dynamical behaviors, as will be seen figure 5.9. In order to recover better for the encryption image, RGB submatrix must make  $x_1//x_2//x_3$ ,  $y_1//y_2//y_3$ ,  $z_1//z_2//z_3$ .

Figure 5.9 is a model of hybrid chaotic system, which consists of Red coordinate system  $(x_1, y_1, z_1)$ ; Green coordinate system  $(x_2, y_2, z_2)$ , and Blue coordinate system  $(x_3, y_3, z_3)$ .

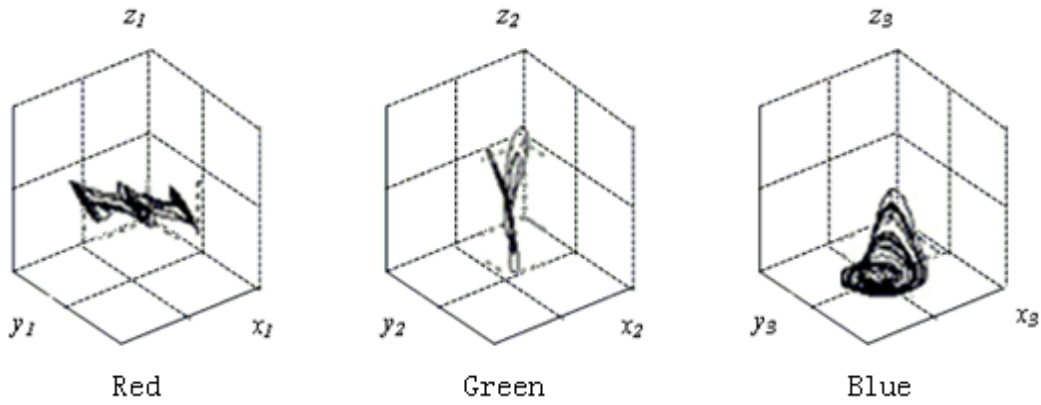


Figure 5.9 The three color of coordinate system

In programming, in order to obtain a combined chaotic system, two formulas (5.3) and (4.3) should be added in red color image.  $R(I)$  Its cipher key shows the initial value  $(x_1, y_1, z_1)$  and parameters  $a, b$ . Other for  $G(I)$  cipher keys, and  $B(I)$  cipher keys, , can be implemented in the same way and the respectively. Formula (5.9) shows the process of cipher key.

$$\begin{cases} R = \{x_1, y_1, z_1\} \\ G = \{x_2, y_2, z_2\} \\ B = \{x_3, y_3, z_3\} \end{cases} \quad (5.9)$$

The following code shows the process of three hyperchaos in program, and three sequences gather are generated by using the Runge-Kutta integration. These sequence are non-periodic, convergence, and very sensitive for initial value.

As defined earlier, the proposed symmetric-key algorithm is designed to utilize one multigroup hyperchaotic map for secret key, which will increase the strength of encryption. The proposed algorithm employs four different chaotic maps, i.e., 2D Henon map, 3D Henon map, Logistic map, and Arnold map. The following codes show the experimental chaotic maps, governing equations, and their parameter values. The proposed algorithm uses four different compound hyperchaotic maps for secret key, and each map has  $n$  keys must be greater than or equal to 4. To combine these keys, we performed XOR operation as follows:

$$K_i = 3DHK_i \oplus AK_i \quad (5.10)$$

where  $i = 1, 2, 3 \dots n$  and  $n \geq 1$ . The equation (5.10) is red sub-image secret key

```
x1(11)=1; y1(11)=0.1; z1(11)=0;
for
n=11:10+rh*rw-11;
x1(n+1)=1.6-y1(n)^2-0.2*z1(n);
y1(n+1)=x1(n);
z1(n+1)=y1(n);
x1(n+1)=mod((x1(n)+y1(n)),1);
y1(n+1)=mod((x1(n)+2*y1(n)),1);
end
```

Green and Blue used same way to generate equation (5.11) and (5.12).

$$K_i = 3DHK_i \oplus LK_i \quad (5.11)$$

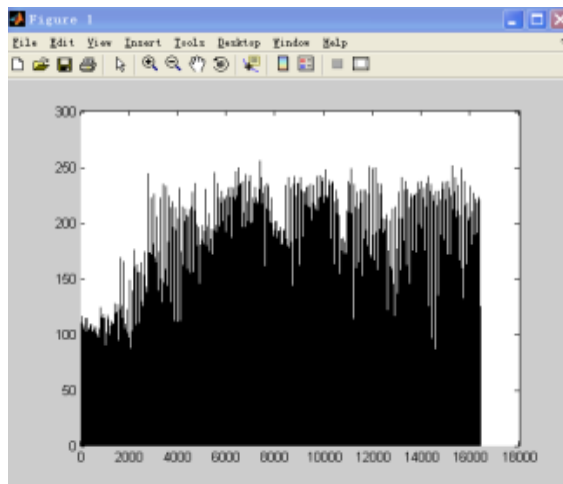
```
x2(1001)=0.82; y2(1001)=0.1; z2(1001)=0;
for
n=1001:1000+rh*rw-1001;
x2(n+1)=1.6-y2(n)^2-0.2*z2(n);
y2(n+1)=x2(n);
z2(n+1)=y2(n);
x2(n+1)=1-2*x2(n)^2;
end
```

$$K_i = AK_i \oplus 3DHK_i \quad (5.12)$$

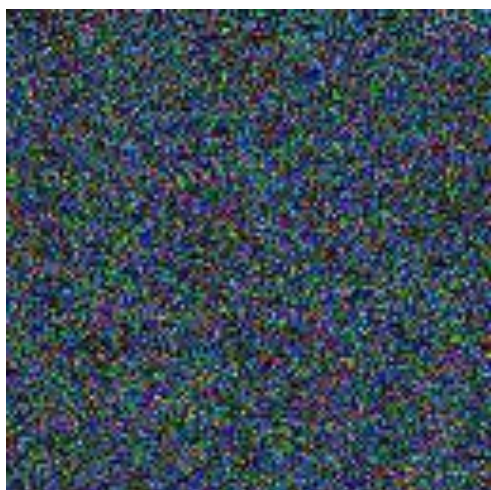
```
x3(11)=0.9; y3(11)=0.1; z3(11)=0;
for
n=2001:2000+rh*rw-2001
x3(n+1)=1-1.4.*x3(n)^2+y3(n);
y3(n+1)=0.3*x3(n);
x3(n+1)=1.6-y3(n)^2-0.2*z3(n);
y3(n+1)=x3(n);
z3(n+1)=y3(n);
end
```



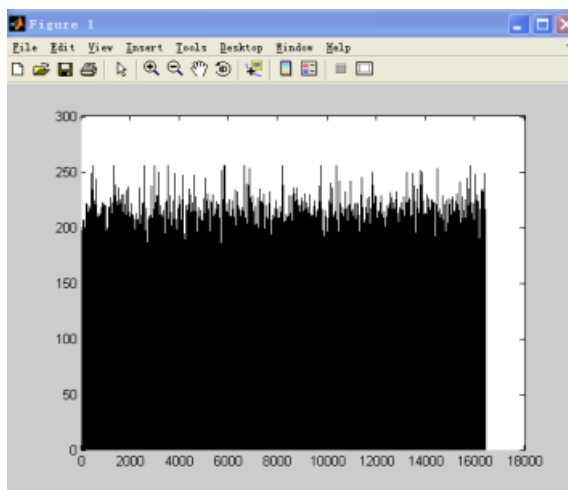
(a) Original image



(b) Histogram of original image



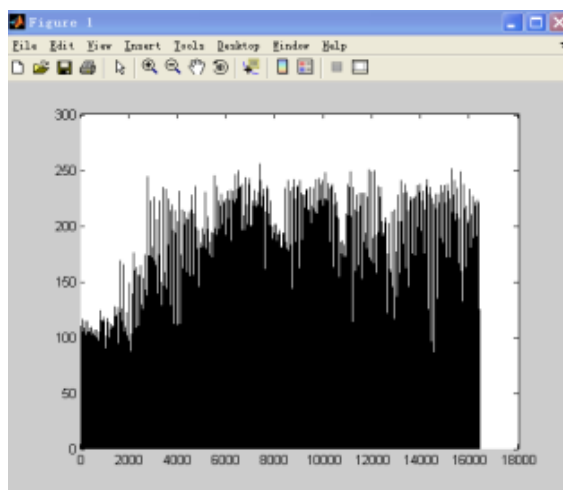
(c) Encrypted image



(d) Histogram of encrypted image



(e) Decrypted image



(f) Histogram of decrypted image

Figure 5.10 Image encryption and decryption experimental result

From the above code can be seen, each color primary image used chaos is different, and their initial position can be same or different. In this section, for standard color image castle used simulate experiment. In the experiment, set  $x_1=1$ ,  $y_1=0.1$ ,  $z_1=0$ , and bifurcate parameter  $a=1.6$ ,  $b=0.2$ ,  $k=2$  as the red subimage secret key;  $x_2=0.82$ ,  $y_2=0.1$ ,  $z_2=0$  and  $a=1.6$ ,  $b=0.2$  as the green subimage secret key and  $x_3=0.9$ ,  $y_3=0.1$ ,  $z_3=0$  and  $a_1=1.4$ ,  $b_1=0.3$ ,  $a_2=1.6$ ,  $b_2=0.2$  as the blue subimage secret key;

$$\begin{cases} R_{x_i} \Leftrightarrow G_{y_i} \\ R_{x_i} \Leftrightarrow B_{z_i} \\ G_{y_i} \Leftrightarrow B_{z_i} \end{cases} \quad (5.13)$$

The visual inspection of figure 5.10 (a)-(f) show the possibility of applying the proposed algorithm successfully and it reveals the algorithm's effectiveness during encryption/decryption process.

The encryption image figure 5.10 (c) is totally scrambled from original image figure 5.10(a) and its figure 5.10(d) histogram of encrypted image. The decrypted image is shown in figure 5.10(e) and figure 5.10(f) histogram of decrypted image.

The R, G, B pixel matrices of a color image were transformed into one-dimensional matrix. Pixel positions were arranged according to the time at which chaotic sequence elements appear. Image encryption and decryption were realized by the following Sort phrase:  $[x_a, x_d]=\text{sort}(x_1)$ ; The chaotic sequence  $x_1$  was arranged in ascending order and was assigned to matrix  $x_a$ . Remember its index matrix  $x_d$ :  $[a_x, d_x]=\text{sort}(x_d)$ ; escape address  $d_x$  is generated:  $AAd=AA(1, d_x)$ ; whichever matrix the image realizes, run  $d_x$  matrix to encrypt the image and  $x_d$  matrix to decrypt the image, with the syntax program shown as follows.

```
[xa,xd]=sort(x1,2);
[ax,dx]=sort(xd);
AAd=AA(1,xd);
```

```
[xa,xd]=sort(y2,2);
[ax,dx]=sort(xd);
AAAd=AA(1,xd);
```

```
[xa,xd]=sort(z3,2);
[ax,dx]=sort(xd);
AAAd=AA(1,xd);
```

## 5.4 Multigroup Hyperchaotic Sequences in Block-image

Whether hyperchaotic scrambling or single chaotic scrambling is applied in rendering large size image, these algorithms required much resource of CPU and graphics card. The rendering speed is slow. Therefore, the image scrambling encryption can be done in fractions to improve the image processing efficiency. How to realize efficient image processing will be stated in the following section.

The encryption image is first divided into block subimages with an appropriate size, the appropriate initial value and parameters are given as ciphers to generate chaotic sequence; then, the chaotic sequence matrix is generated for scrambling. The same procedure is applicable to subimages but the scrambling is done according to different type of chaos. Hence, the encryption effectiveness and efficiency can be improved.

Set the library.bmp as  $I_{n \times m}$ , divide the  $I_{n \times m}$  into  $k$  blocks to get the result image  $I'$  including a series of subimages  $k_i$  ( $i=0, 1, \dots$ ). Then use the hyperchaos to scramble every block subimages. These scrambling methods have been described in the section 5.2 and 5.3 of chapter 5. Figure 5.11 and 5.12 show processing result.

Experiment model: The library.bmp (512×512 in size) is divided into two block subimages ( $k=2$ ), the first subimage figure 5.11 (b) is to be scrambled by hyperchaotic 2D Henon map and 2D Arnold map with the cipher ( $a_1=1.4$ ,  $b_1=0.3$ ,  $k=2$ ,  $x_1=0.9$  and  $y_1=0.1$ ), the second subimage figure 5.11 (c) is to be scrambled using hyperchaotic 3D Henon map and 2D Arnold map with cipher ( $a_2=1.6$ ,  $b_2=0.2$ ,  $k=2$ ,  $x_1=1$ ,  $y_1=0.15$  and  $z_1=0$ ).

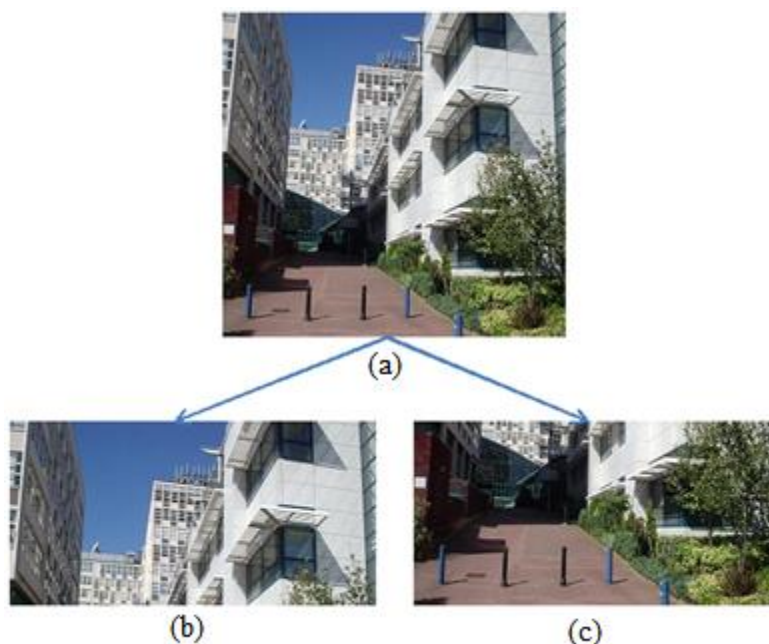


Figure 5.11 An image divided into two subimages

Main problems of block encryption are that is only internal pixel encryption. Because the relevance of a pixel and its neighboring pixels, more blocks means that its pixels will have more close relationship, resulting in worse scrambling, and worse recovery effects for the encrypted image.

Using the following formulas to calculate the relevance coefficient:

$$I = \sum_{i=1}^n \sum_{j=1}^m x_i y_j \quad (5.14)$$

where,

$I$  is pixel matrix of the image

$i$  and  $j$  represent the row and column of subimage matrix respectively.

$x_i$  and  $y_j$  are the pixel at row  $i$  and the pixel at column  $j$  respectively.

$n$  and  $m$  represent the maximum numbers of rows and columns of each matrix respectively.

The experiment yields a well encrypted image figure 5.12(c) and a well decrypted image figure 5.12 (d). The iteration equation of the chaotic sequence is shown as above. Figure 5.12(a) and figure 5.12(b) were assigned to two hyperchaotic sequence



systems with length  $m \times n$  respectively: figure 5.12(a) $\rightarrow(x_1, y_1)$  and figure 5.12(b) $\rightarrow(x_2, y_2)$ .

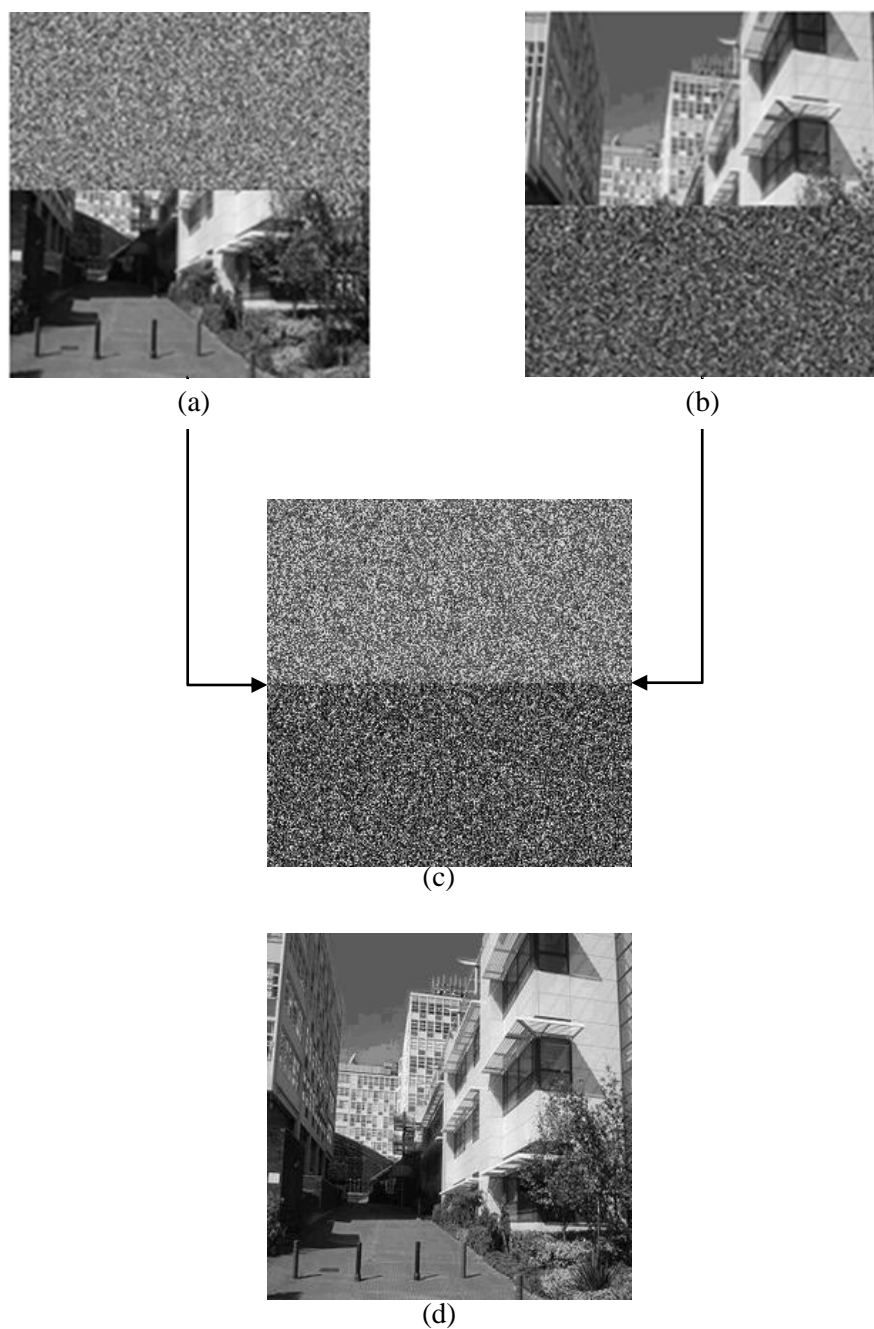


Figure 5.12 Hyperchaotic pixel rearrangement and successful restoration

Castle.bmp (256 $\times$ 256 in size) is divided into sixteen pixel blocks (64 $\times$ 64 in size each), then these blocks are combined into five submatrices A, B, C, D, and E. Finally, each block is scrambled by different composite hyperchaotic system, resulting in a full image pixel sequence shown in figure 5.13 as follows:

$$I_{256 \times 256} = \begin{bmatrix} A = \begin{bmatrix} \sum_{i=1}^{64} \sum_{j=1}^{64} x_1 y_1 & \sum_{i=1}^{64} \sum_{j=64+1}^{128} x_1 y_1 \end{bmatrix} & D = \begin{bmatrix} \sum_{i=1}^{64} \sum_{j=128+1}^{192} x_4 y_4 \\ \sum_{i=64+1}^{128} \sum_{j=128+1}^{192} x_4 y_4 \\ \sum_{i=128+1}^{192} \sum_{j=128+1}^{192} x_4 y_4 \\ \sum_{i=192+1}^{256} \sum_{j=128+1}^{192} x_4 y_4 \end{bmatrix} & E = \begin{bmatrix} \sum_{i=1}^{64} \sum_{j=192+1}^{256} x_5 y_5 \\ \sum_{i=64+1}^{128} \sum_{j=192+1}^{256} x_5 y_5 \\ \sum_{i=128+1}^{192} \sum_{j=192+1}^{256} x_5 y_5 \\ \sum_{i=192+1}^{256} \sum_{j=192+1}^{256} x_5 y_5 \end{bmatrix} \\ B = \begin{bmatrix} \sum_{i=64+1}^{128} \sum_{j=1}^{64} x_2 y_2 & \sum_{i=64+1}^{128} \sum_{j=64+1}^{128} x_2 y_2 \end{bmatrix} & & \\ C = \begin{bmatrix} \sum_{i=128+1}^{192} \sum_{j=1}^{64} x_3 y_3 & \sum_{i=128+1}^{192} \sum_{j=64+1}^{128} x_3 y_3 \\ \sum_{i=192+1}^{256} \sum_{j=1}^{64} x_3 y_3 & \sum_{i=192+1}^{256} \sum_{j=64+1}^{128} x_3 y_3 \end{bmatrix} & & \end{bmatrix}$$

Figure 5.13 Image encryption comparison of 5 blocks matrix

The submatrix  $A_{i,j}$  represents bit decomposition operator, so the  $A(i,j)$  is represented as equation (5.15):

$$A_{ij} = \begin{bmatrix} \sum_{i=1}^{64} \sum_{j=1}^{64} x_1 y_1 & \sum_{i=1}^{64} \sum_{j=64+1}^{128} x_1 y_1 \end{bmatrix} \quad (5.15)$$

' $\sum_{i=1}^{64} \sum_{j=1}^{64} x_1 y_1$ ' is the first block image whose matrix size is  $64 \times 64$  ( $n$  is 64,  $m$  is 64),

the row of pixel bit  $i$  is  $1 \cdots 64$  and the column  $j$  is  $1 \cdots 64$ . The second block image can

be processed in same way to generate ' $\sum_{i=1}^{64} \sum_{j=64+1}^{128} x_1 y_1$ ' whose matrix size is  $64 \times 64$  ( $n$

is 64,  $m$  is 128), the row of pixel bit  $i$  is  $1 \cdots 64$ , and  $j$  is  $1 \cdots 64, 64+1 \cdots 128$ . And the rest can be processed in the same manner.

The calculated image entropy values with different compound of chaotic map given in table 5.3, the different compound of chaotic maps give greater results of information entropy in different matrix size image.

Table 5.3 Information of the proposed algorithm with different compound of chaotic map

Submatrix	Initial position ( $i, j$ )	Matrix size ( $n \times m$ )	Sequence length	Different compound chaotic maps
$A_{ij}$	1,1	$64 \times 128$	$64 \times 128$	3D Henon, 2D Henon
$B_{ij}$	64+1,128	$64 \times 128$	$64 \times 128$	3D Henon, 2D Arnold
$C_{ij}$	128+1,128	$128 \times 128$	$128 \times 128$	3D Henon, Logistic, 2D Arnold
$D_{ij}$	256,128+1	$256 \times 64$	$256 \times 64$	2D Henon, 2D Arnold
$E_{ij}$	256,192+1	$256 \times 64$	$256 \times 64$	3D Henon , 2D Arnold

The image was encrypted by hyperchaotic subsystems with different chaotic variables, resulting in a fully encrypted 'Castle' image, and then the encrypted image was decrypted through inverse transformation of chaotic sequence.

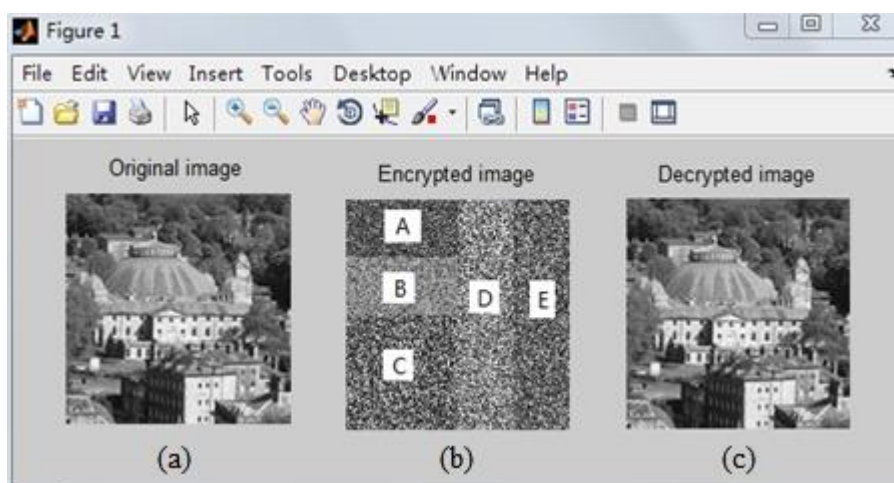


Figure 5.14 Result of encryption by using 5 blocks  
(a) Original image, (b) Encryption image, (c) Decrypted image

Figure 5.14 shows experiment results; running as tested above can obtain better quality decryption and encryption images. Calculate the similarity between the original image and decryption image,  $NC$  value=1 (Normalized Cross Correlation). So the distortion-free encryption and decryption was achieved.

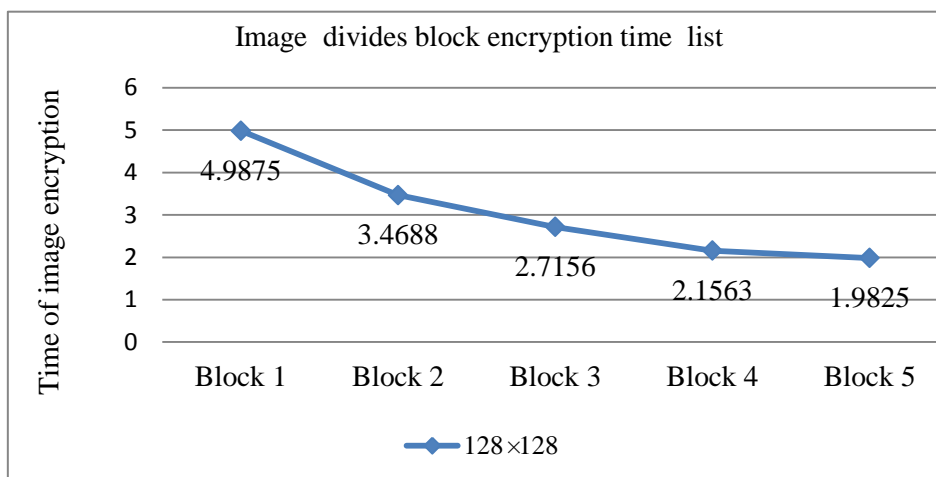
After comparison the results that were obtained can be well represented in form of figure 5.15(a), (b) and (c) that describes the encryption time and decryption time for the *castle.bmp*. From those results, 2-blocks, 3-blocks, 4-blocks and 5-blocks are faster than non-block for encryption speed.

In this section, various sizes of images ( $128 \times 128$ ,  $256 \times 256$ ,  $512 \times 512$ ) were encrypted and decrypted completely (without block division) and in blocks respectively, and the respective running time was measured, as shown in figures 5.15 (a), (b) and (c), wherein the horizontal axes represent the number of blocks an image is divided into 1, 2, 3, 4 and 5 blocks respectively, the vertical axes represent the time of encryption. Because every subimage involves both encryption and decryption operations, which take full advantage of computer operating capacity, so the running time gradually decreases; the more blocks an image is divide into, the less the running time, which exhibits exponential changing trends as shown in the following figures.

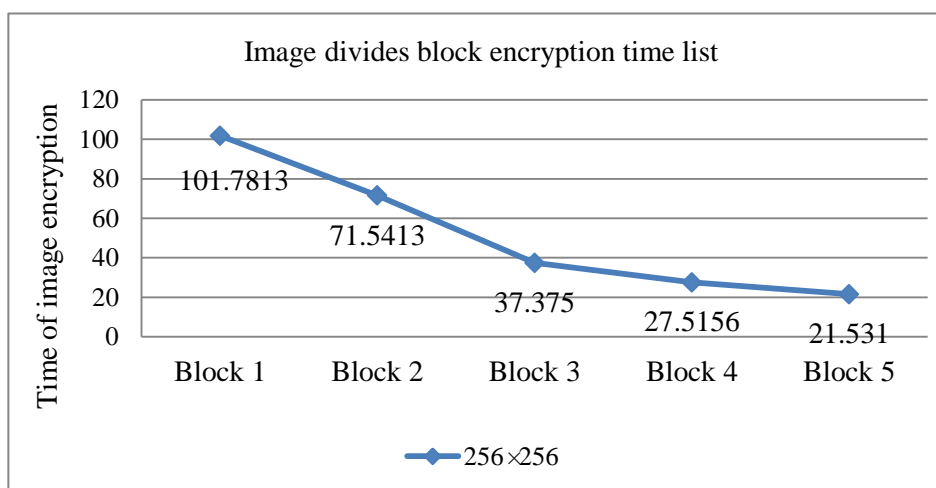
Based on the above patterns, we can divide a big image into small blocks, and scramble these small-pixel subimages. Hence, the time of encryption will be much less, because each small pixel subimage can be encrypted with a hyperchaotic subsystem with separate key (as in Figure 5.11), which correspondingly improves the randomness and unpredictability of the encryption system.

Encryption and decryption time results are compared in figure 5.15(a). The one-block scheme needs 4.9875 seconds to complete encryption and decryption without blocking. The two-block scheme needs 3.4688 seconds to complete encryption algorithm. The final scheme is in five blocks that need 1.9825 seconds. Obviously, block image scheme is faster than that not partitioned.

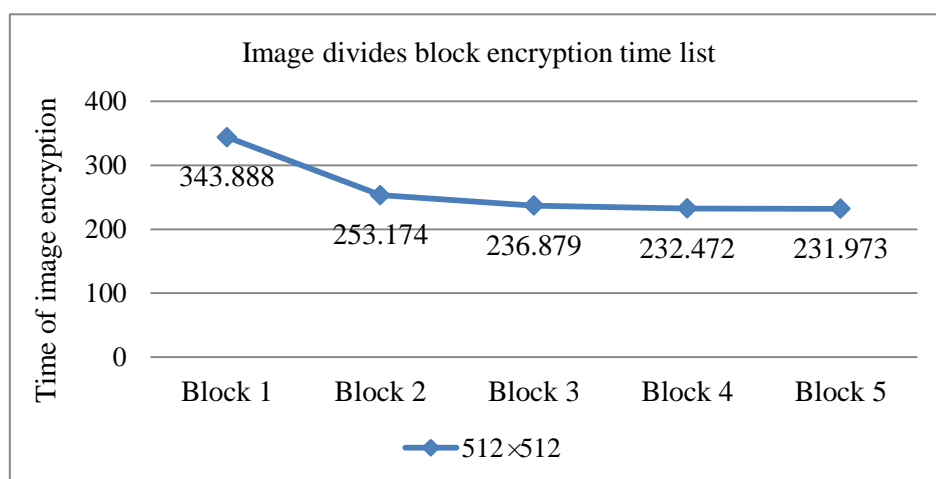
Figure 5.15(b) shows encryption and decryption result for block image scheme with size  $256 \times 256$ . The results are same that is faster than that not partitioned. Figure 5.15(c) has the same result. This means that the optimization of encryption is successful.



(a)



(b)



(c)

Figure 5.15 Encryption and decryption time comparison of block scrambling algorithm

## 5.5 Hyper Large Size Image Encryption Algorithm

In a world with a lot more digital information to transmission, what defines an ‘hyper large’ image. In the encryption technology, we need to adjust non-adaptive and adaptive image scrambling techniques. Regardless of how much size image have, which in practical image technology is limited in computer. So we want to use digital mosaic technique in encryption of digital image. The hyper-large images or files are encrypted by scrambling a lot of subimages in batches followed by integrating

One end point on the edge of a hyper-large image can act as the origin of the XYZ coordinate system, and truncate value points  $i$  ( $i=1, 2, 3, \dots$ ) on  $x$  axis with width of each subimage  $\Delta x$  (e.g.,  $\Delta x = 512$  pixels) as the unit width, truncate value points  $j$  ( $j=1, 2, 3, \dots$ ) on  $y$  axis with height of each subimage  $\Delta y$  ((e.g.,  $\Delta y = 256$  pixels) as unit height; finally, take  $(x_i, y_j)$  as the origin. Hence, a coordinate space of super large numbers with the origin  $(x_i, y_j, z_k)$  is established, that is

$$S_{ijk}=F(x_i, y_j, z_k) \quad (5.16)$$

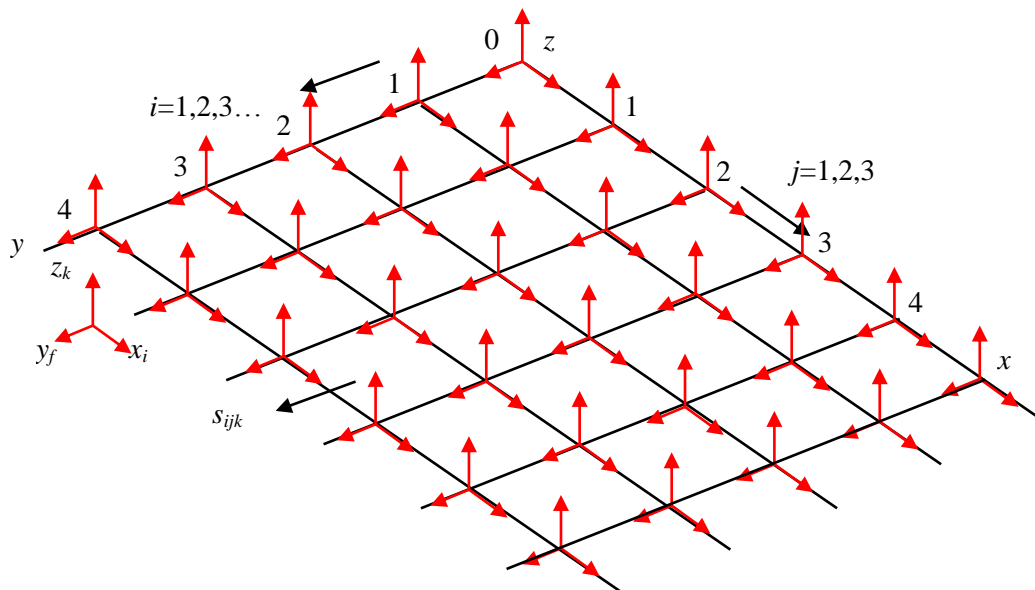


Figure 5.16 The coordinates space in large size image

Thus, the coordinate space of super-large number can provide flexible operating environment for hyperchaotic encryption image. To improve confidential performance, some operations can be performed in each coordinate system including (1) selecting various appropriate iteration initial points to scramble; (2) combining multiple chaos to perform scrambling; (3) flexibly combining the above two operations; thereby an encryption system with strong anti-attack capability comes into being. Subimages are scrambled in batches until the whole image encryption is completed. Such encrypting operation plays an important role in encrypting super-large images used for cities, marine geology, aviation, etc. The encryption effect shows in Appendix B.

## 5.6 Summary

This chapter studies the chaotic map and chaotic attractor overflow problems through analysis and experiment, and proposes several solutions which yield good encryption effects; the main work is summarized as follows:

- (1) The rational initial iteration point has been proposed to prevent chaotic attractor overflow and speed bifurcation; a chaotic system with the same coordinate system, Red (R), Blue (B) and Green (G) primary color images have been encrypted with good effectiveness in the experimental study.
- (2) The multigroup hyperchaotic system;

A number of hyperchaotic maps have been chosen to build the respective Cartesian coordinates while running multiple different hyperchaotic maps (e.g., running hyperchaotic sequences of three primary colors, namely, red, green and blue). As each chaotic system corresponds to an independent chaotic sequence which the brute-force attack decryption difficulty increases exponentially. For instance, an image has pixels ( $n \times n$ ), if  $M$  is the number of chaotic systems, then it can be decrypted (Exhaustive Attack method) with the number of attacks

$\left[ (n^2)! \right]^M$ , so the image anti-attack capability is greatly improved.

(3) Encrypting an image in blocks;

An image is divided into  $k$  subimages and is encrypted by the differently hyperchaotic which can improve robustness and resistance for the encryption system.

(4) Encryption algorithm for large-scale image;

It is an encryption method with the combination of multi-dimensional hyperchaotic mentioned above; it divides large-scale image into multi-region to build extra-large Cartesian coordinate space. It includes a variety of algorithm for different chaotic systems meanwhile operating in areas adjacent to the image encryption. Therefore, the image scrambling encryption can be done in to improve the image processing efficiency, which full play to the encryption algorithm flexibility, the randomness. It not only improves the robustness for resisting attacks and difficulty to forced decryption, but increases the complex relation among subimages. It has potential development and application value in the fields of large-scale image encryption (e.g. landscape, marine geology, satellite images, and etc.).

According to the principle proposed by Shannon that one-time pad encryption algorithm is safe. As the chaotic sequences generated from a chaotic system are unrepeatable, and the sequences generated from multiple chaotic systems together would be more unrepeatable, so the multiple chaotic color encryption algorithm belongs to one-time pad algorithm. It can be known, therefore, that the safety of this algorithm is very high.

In addition, the compound chaotic image encryption algorithm proposed in this chapter has the following merits:



- (1) For the sake of higher safety, this algorithm does not use the initial part of the generated chaotic sequence, but abandons the first  $N$  terms and adopts the middle part of the chaotic sequence. The initial point is random in this chapter, which increases the prediction difficulty.
- (2) The compound chaotic sequence is generated selectively from individual chaotic sequences and follows nonlinear change; furthermore, its histogram, auto-correlation and cross correlation have the above characteristics. By encrypting the image in this way, its resistance to statistical analysis is greatly enhanced.
- (3) In the image encryption, compared with immediate one time encryption, the separate encryption in three primary colors enhances the safety to much extent, but its encryption and decryption speeds are not affected.
- (4) The algorithm has numerous initial parameters, that is, many cipher keys. Accordingly the formed cipher space will be huge, which greatly improves the resistance to brute-force attack.

The multi-dimensional chaotic system serves as the basis of compound chaotic system generation in this algorithm; the algorithm employs linearly correlated parameters for three chaotic systems to improve the system sensitivity to parameters. The parameter change of any chaotic system will directly impact on other two chaotic systems. Hence, the generated compound chaotic sequence will be extremely sensitive to initial parameters

## **Chapter 6**

---

### **DWT-based Information Hiding Algorithm**

The design idea of image transform domain is introduced into a kind of image embedding algorithm based on wavelet transform. This algorithm adopts the image format transform, discrete wavelet transform and Human Vision System (HVS), and overcomes the flaws of existing algorithms in security and robustness. It is shown by theoretical analysis and experimental results that it has strong immunity to common attack.

## 6.1 Watermark Image Implementation Processes

The decomposition is done with ‘Haar’, which is applied in the watermark embedding and extraction processes. The proposed algorithm functions with 2D still image independent on Matlab to prove security and robustness performance.

From perspective of digital signal processing, the embedded watermark signal can be regarded as the superposition of a weak signal in strong background, as longer as its intensity is lower than that of the HVS or the Human Auditory System (HAS). Therefore, it is likely to embed some watermark signals without change visual effect through certain adjustments of the original signal.

For a color image, each pixel of it can be divided into three primary colors R, G, and B. From perspective of brightness function, human eye are most sensitive to green color, less sensitive to red color, and least sensitive to blue color. Therefore, when embedding watermark into different primary colors of a color image, one can adjust the watermark’s weighting coefficient according to HVS characteristics, that is, the weight of blue subimage is the largest, followed by red subimage, and the green subimage shall be superposed with the weakest signal.

In the following implementation, the watermark signals have low invisibility, but strong anti-attack capability and thus become robust at this time. In contrast, other watermarks have much better invisibility, but weaker anti-attack capability.

The common digital image formats include *Bmp* and *Jpg*. An indexed image or a palette-based image consists of two parts: one is a list of index  $(i, c_i)$  where a color vector  $c_i$  is assigned to index  $i$ , the other is the actual image data that assigns a palette index rather than the color value itself to every pixel.

## (1) Watermark embedding process

The embedding technology takes a color image with  $512 \times 512$  pixels as the host image, converts the image to YUV space and considers 'DWT coefficient', and its watermark is color image 'gate', the transform domain and the detail steps are shown follows:

- Step 1. DWT transform: The host image is decomposed by 2-levels using 2D DWT. Then a approaching subimage (low frequency band) and 2-level detail subimage (high-frequency band) are generated.
- Step 2. Embedding watermark: The watermark image also using 1-level 2D DWT to obtain four subimages and wavelet coefficients; the wavelet coefficient of watermark image is embedded into the wavelet coefficient of the original image to generate new coefficients.
- Step 3. Inverse transform: After embedding the watermark coefficients, the wavelet transform of the image is inverted by 2D inverse discrete wavelet transformation (IDWT), and the new watermarked image is obtained. Refer to figure 6.1.

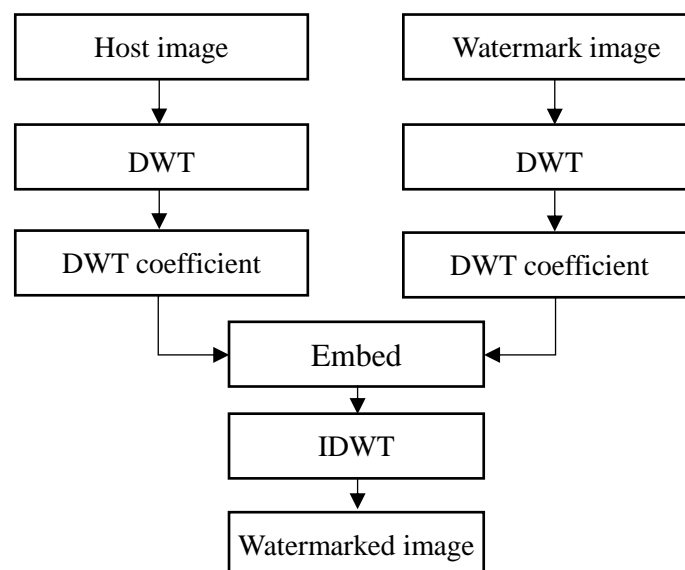


Figure 6.1 The watermark embedding procedure

The watermark signal  $W = \{w(a)\}$  is embedded into the host image  $X = \{x(a)\}$ , to achieve the watermarked image  $X' = E(x, w, a)$  accordingly; letter  $E$  is the embedding function and  $a_k$  is the cipher.

$$\text{Embedding function:} \quad X' = X + \alpha_k \times W \quad (6.1)$$

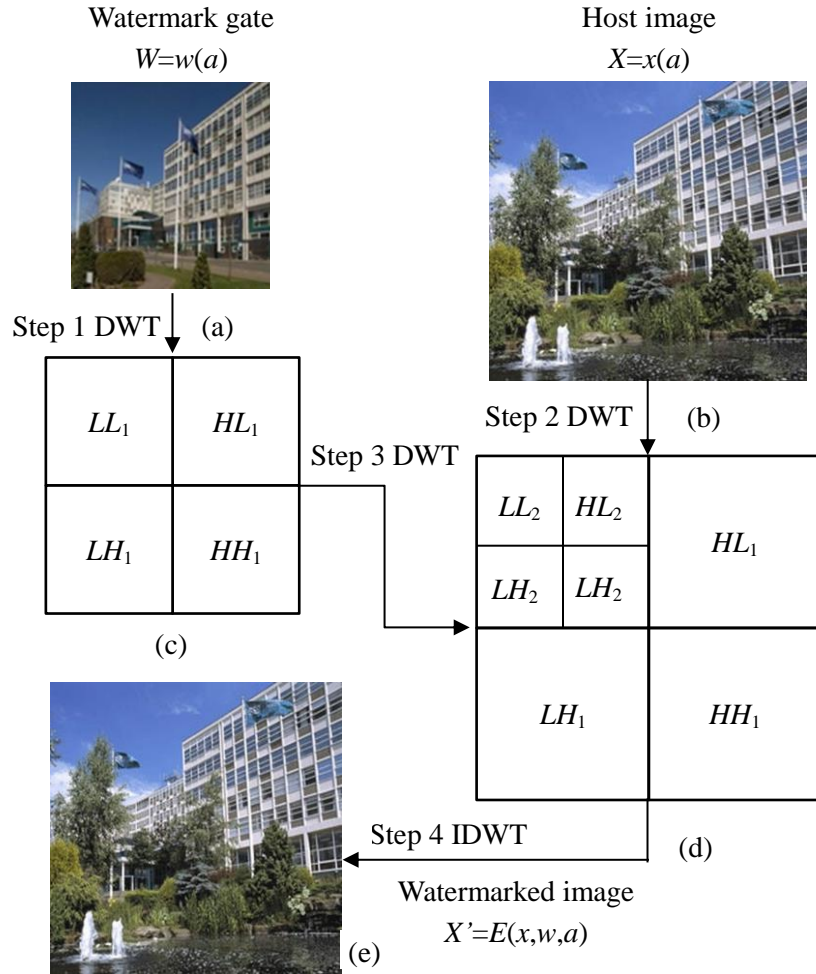


Figure 6.2 An example of the embedding system

(a) Watermark gate image; (b) Host image; (c) Wavelet representation of 1<sup>st</sup> layer DWT of the image; (d) Wavelet representation of two resolution layers of the image; (e) Watermarked image.

The embedding method selects the 2<sup>th</sup> level transformation for host image and 1<sup>st</sup> level for watermark image. Watermark pixels are inserted into blocks of of the host image. The insertion is done in a pseudo-random manner. For  $n \times n$  blocks, the value of  $n$

depends on the size of the host image and the number of watermark pixels. Figure 6.2 shows an example of the embedding system.

(2) Watermark extraction process

Step 1. DWT transform: The host image and the watermarked image are transformed by DWT to generate the coefficients  $X = \{x(a)\}$  and

$$X' = E(x, w, \alpha).$$

Step 2. Extraction watermark image: To extract out the wavelet coefficients of the host image, calculate wavelet coefficients of the watermarked image

$$X' = E(x, w, \alpha)$$

$$\text{image } W = D[X' - x(a)].$$

Step 3. Inversing transform: Then the coefficient of watermark image is inversed by discrete wavelet transformation (IDWT) and the watermark image is obtained. Figure 6.3 shows the process of the watermark extraction procedure.

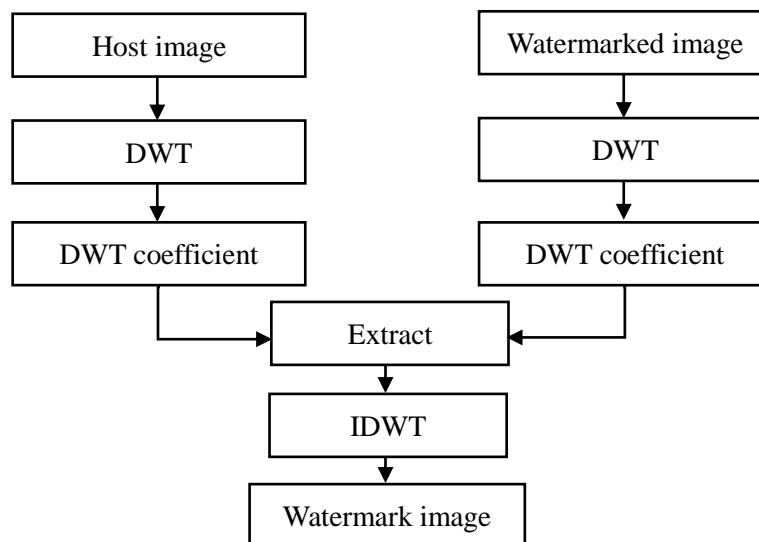


Figure 6.3 The watermark extraction procedure

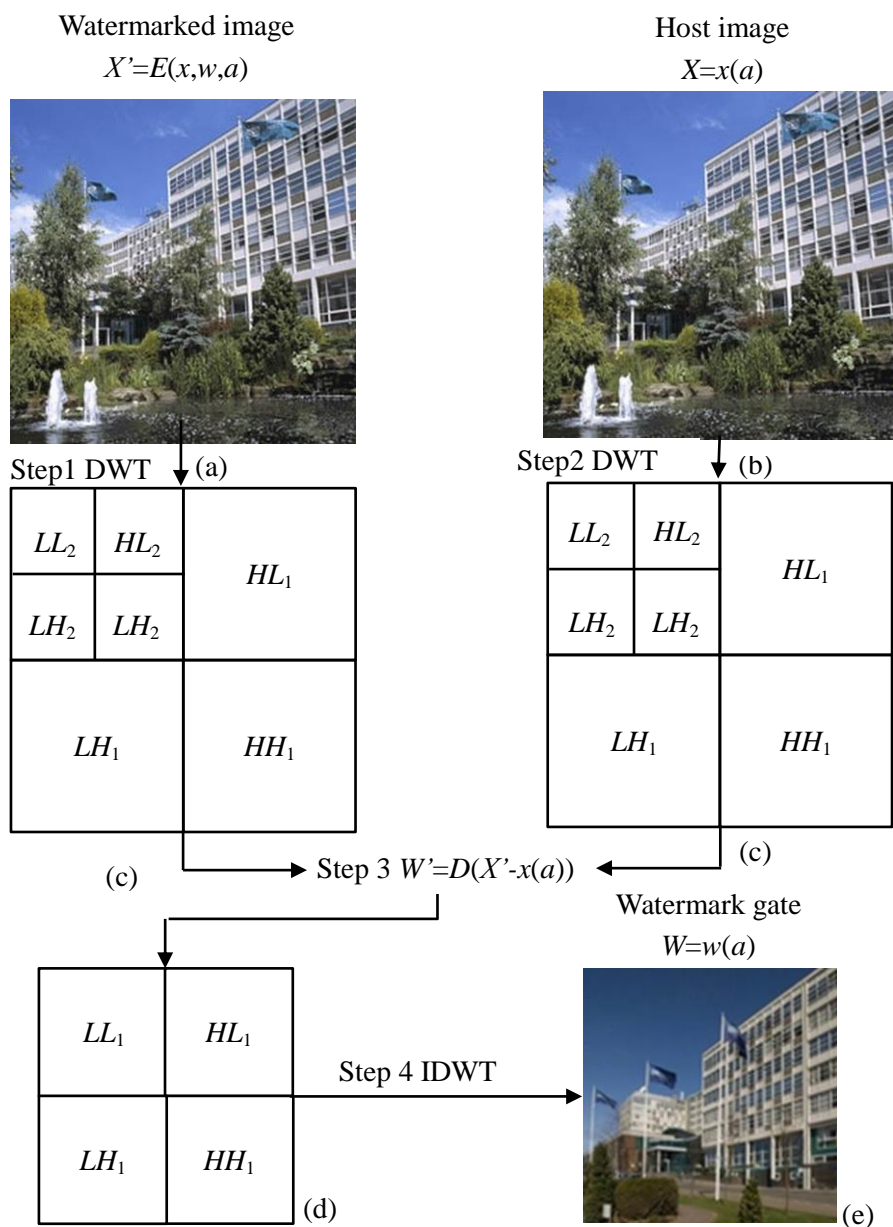


Figure 6.4 An example of the extraction system

(a) Watermarked image, (b) Original image, (c) Wavelet representation on 2-layer DWT of image, (d) Wavelet representation 1-layer DWT of image, (e) Watermark gate.jpg

Watermark extracted from the watermarked image:

$$W = D[X' - x(a)] \tag{6.2}$$

The watermarked image is as follows:

$$X' = E(x, w, \alpha) \tag{6.3}$$

The watermark image is extracted or measured from the watermarked image. Figure 6.4 indicates an example of the extraction system.

Which place is robust enough? Watermark can be robust enough; the wavelet coefficient must be in compliance with the following necessary conditions (1) after a common signal processing and noise, it is still well-preserved; (2) the original host image does not deteriorate significantly after embedding an image with higher capacity.

## 6.2 The Implement of Wavelet Algorithm

In this section, the watermark embedding algorithm is described as follows:

### (1) Preprocessing:

The original color image  $(m, n, 3)$  is decomposed into  $R$ ,  $G$  and  $B$  subimages, denoted as  $CR(m,n)$ ,  $CG(m,n)$ , and  $CB(m,n)$  respectively; meanwhile, the watermark image  $(p,q,3)$  is also decomposed into three subimages denoted as  $WR(p,q)$ ,  $WG(p,q)$  and  $WB(p,q)$  respectively.

### (2) Wavelet decomposition:

For  $CR$ ,  $CG$  and  $CB$  perform  $k$  ( $k \geq 2$ ) layers of 2D multi-resolution wavelet decomposition (MWD) respectively; for  $WR$ ,  $WG$  and  $WB$  perform one layer of 2D MWD respectively. The criterion of  $k$  selection is that, the size of the lowest-frequency region  $cA_k$  of the image after  $k$  layers of 2D MWD is coincidentally greater than or equal to the size of the lowest-frequency region  $cA_1$  of the watermark image after one layer of wavelet decomposition. Suppose after three layers of 2D MWD, the original image is decomposed by 3 times of 2D DWT. The  $A_3$  is the top layer (similar to the low frequency region),  $H_k$  is the  $k$ -layer of horizontal edges,  $V_k$  is the  $k$ -layer of vertical edges and  $D_k$  is the



$k$ -layer of high frequency in both directions (the corners).

Procedures are shown as below:

$$\begin{aligned} [cA1,cH1,cV1,cD1] &= \text{dwt2}(c,'db1') \\ [cA2,cH2,cV2,cD2] &= \text{dwt2}(cA1,'db1') \\ [cA3,cH3,cV3,cD3] &= \text{dwt2}(cA2,'db1') \end{aligned}$$

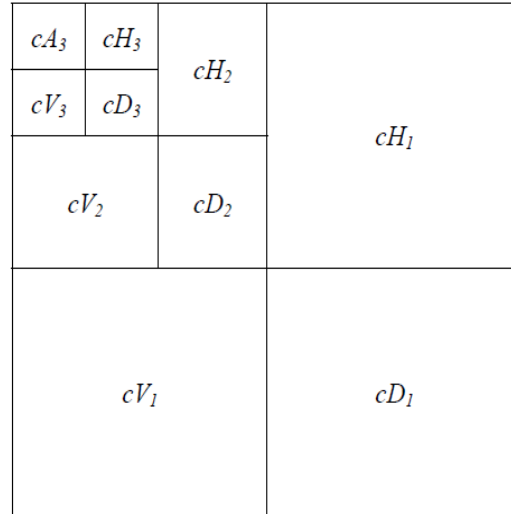


Figure 6.5 The image wavelet representations  
Low-frequency wavelet coefficient  $cA$  (approximate), and high-frequency wavelet coefficient  $cH$  (horizontal),  $cV$  (vertical),  $cD$  (diagonally)

Figure 6.5 shows the image wavelet representations. In frequency decomposition, assume that the image  $cA_3$  is  $cA_k$  and the corner image is the detail image  $D_k$ . The image  $cA_3$  corresponds to the low horizontal and vertical frequencies of  $cA_k$ .  $cH_k$  gives the vertical high frequencies and horizontal low frequencies.  $cV_k$  is the horizontal high frequencies and vertical low frequencies.  $cD_k$  is the high frequencies in both horizontal and vertical directions.

The 2D 3-layer discrete wavelet inversion transform is shown by syntax toolbox function of MATLAB.

$$\begin{aligned} cA2 &= \text{idwt2}(cA3, cH3, cV3, cD3,'db',[Mc/4,Nc/4]); \\ cA1 &= \text{idwt2}(cA2, cH2, cV2, cD2,'db',[Mc/2,Nc/2]); \\ \text{image} &= \text{idwt2}(cA1,cH1,cV1,cD1,'db',[Mc Nc]); \end{aligned}$$

IDWT is used for image recovery.

## (3) Processing of wavelet coefficients:

According to the above context with reference to the statement of Cox *et al.*, first, it is available to use weight factor applied to the watermark of low frequency signal region  $cA_3$ ; then use weight factor applied to the second layer of horizontal high-frequency region  $cH_2$  and vertical high-frequency region  $cV_2$ . Finally, while  $k \geq 2$ , the same high-frequency signals as in  $k-1$  layer can be selectively applied to  $cH_1$  and  $cV_1$  in the first layer to enhance the robustness of the entire watermarking system. Nevertheless, this will inevitably cause invisibility decline. Attributed to a robust watermark embedding with wavelet domain formula:

$$c_{(r,g,b)}A_k(a_k + i, j + \delta_k) = c_{(r,g,b)}A_k(a_k + i, j + \delta_k) + \varepsilon_{(r,g,b)}gw_{(r,g,b)}A_k(i) \quad (6.4)$$

$$c_{(r,g,b)}H_{k-1}(a_{k-1} + i, j + \delta_{k-1}) = c_{(r,g,b)}H_{k-1}(a_{k-1} + i, j + \delta_{k-1}) + \varepsilon_{(r,g,b)}gw_{(r,g,b)}H_{k-1} \quad (6.5)$$

$$c_{(r,g,b)}V_{k-1}(a_{k-1} + i, j + \delta_{k-1}) = c_{(r,g,b)}V_{k-1}(a_{k-1} + i, j + \delta_{k-1}) + \varepsilon_{(r,g,b)}gw_{(r,g,b)}V_{k-1}(i) \quad (6.6)$$

$$c_{(r,g,b)}D_{k-1}(a_{k-1} + i, j + \delta_{k-1}) = c_{(r,g,b)}D_{k-1}(a_{k-1} + i, j + \delta_{k-1}) + \varepsilon_{(r,g,b)}gw_{(r,g,b)}D_{k-1}(i) \quad (6.7)$$

$$c_{(r,g,b)}H_1(a_1 + i, j + \delta_1) = c_{(r,g,b)}H_1(a_1 + i, j + \delta_1) + \frac{\varepsilon_{(r,g,b)}}{10} gw_{(r,g,b)}H_1(i) (k \geq 3) \quad (6.8)$$

$$c_{(r,g,b)}V_1(a_1 + i, j + \delta_1) = c_{(r,g,b)}V_1(a_1 + i, j + \delta_1) + \frac{\varepsilon_{(r,g,b)}}{10} gw_{(r,g,b)}V_1(i) (k \geq 3) \quad (6.9)$$

$$c_{(r,g,b)}D_1(a_1 + i, j + \delta_1) = c_{(r,g,b)}D_1(a_1 + i, j + \delta_1) + \frac{\varepsilon_{(r,g,b)}}{10} gw_{(r,g,b)}D_1(i) (k \geq 3) \quad (6.10)$$

$$a_1 \leq \frac{Width_c - Width_w}{2}, \quad a_k \leq \frac{Width_c}{2^k} - \frac{Width_w}{2}, \quad a_{k-1} \leq \frac{Width_c}{2^{k-2}} - \frac{Width_w}{2},$$

$$\delta_1 \leq \frac{Heigh_c - Heigh_w}{2}, \quad \delta_k \leq \frac{Heigh_c}{2^k} - \frac{Heigh_w}{2}, \quad \delta_{k-1} \leq \frac{Heigh_c}{2^{k-2}} - \frac{Heigh_w}{2},$$

$$i \in \left[1, \frac{Width_w}{2}\right], \quad j \in \left[1, \frac{Heigh_w}{2}\right]$$

Principle of selecting weighting factors:  $\alpha_k$  is the embedding coefficients corresponding to the  $k$  layer of horizontal scrolling position, so  $\alpha_{k-1}$  and  $\alpha_1$  are

embedding coefficients of the  $k-1$  layer and the 1<sup>st</sup> layer respectively. Now, the wavelet coefficient of the watermarked image is equal to the wavelet coefficient of the host image plus the corresponding weighting factor of the watermark. Dynamically take point values for different image features, this can increase reasonably the imperceptibility, security and robustness.

$$\mathcal{E}_{r,g,b} \Rightarrow C_r; C_g; C_b \quad (6.11)$$

After the red component of the watermark image is embedded into the red component of the original image, proceed as follows:

$$\begin{aligned} \text{rdcA3} &= \text{rdeA3} + \text{cr} \times \text{rcA3}; \\ \text{rdcH3} &= \text{rdeH3} + \text{cr} \times \text{rcH3}; \\ \text{dcV3} &= \text{rdeV3} + \text{cr} \times \text{rcV3}; \\ \text{rdcD3} &= \text{rdeD3} + \text{cr} \times \text{rcD3}; \\ \text{rdcH2} &= \text{rdeH2} + \text{cr} \times \text{rcH2}; \\ \text{rdcV2} &= \text{rdeV2} + \text{cr} \times \text{rcV2}; \\ \text{rdcD2} &= \text{rdeD2} + \text{cr} \times \text{rcD2}; \\ \text{rdcH1} &= \text{rdeH1} + \text{cr} \times \text{rcA1}; \\ \text{rdcV1} &= \text{rdeV1} + \text{cr} \times \text{rcV1}; \\ \text{rdcD1} &= \text{rdeD1} + \text{cr} \times \text{rcD1}; \end{aligned}$$

‘rdcA3, rdcH3, rdcV3, rdcD3’ are 3-layers discrete wavelet coefficients of the red component of the watermarked image. ‘rdcH2, rdcV2, rdcD2’ are 2-layer discrete wavelet coefficients of the red component of the watermarked image. ‘rdcH1, rdcV1, rdcD1’ are 1-layer discrete wavelet coefficients of the red component of the watermarked image. ‘rdeA3, rdeH3, rdeV3, rdeD3’ are 3-layer discrete wavelet coefficients of the red component of the host image. ‘rdeH2, rdeV2, rdeD2’ are 2-layer discrete wavelet coefficients of the red component of the host image. ‘rdeH1, rdeV1, rdeD1’ are 1-layer discrete wavelet coefficients of the red component of the host image.

$C_r$  represents the weight coefficient of red watermark embedded, so do the green component and blue component.

In addition to output watermarked image, the program will result in the checksum file.

- a. During watermark embedding in step (3), the host image is modified via wavelet transform, and the wavelet coefficient position and original values will be recorded and saved as *data.mat* to make watermark recovery convenient, not needing to use the original color image;
- b. '*parameter.mat*' records necessary parameters required for watermark extraction, such as wavelet type, number of decomposition layers, weight indicator, watermark displacement and so on, and can be considered as a set of cipher keys to ensure high accuracy in watermark extraction.

In addition, based on the following three points, it is necessary to create an extra matrix '*tail. Mat*' to save error: 1) For DWT image processing in MATLAB, the "*double*" structure must be used to reduce the errors; ultimately, when the image is saved and packaged in bmp format, it should be converted to uint8 or uint16 type, which can be less prone to error. 2) In actual operation, the embedded bits may overflow. When the original image is of odd-numbered dimension, its pixel value will be greater than 255 after wavelet transform and embedding. If the image is of *Bmp* format, this approach will truncate the portion greater than 255. 3) To embed such susceptible watermark, the error has to be recorded. The embedding process is as follows:

```
wimg(:,:,1)=rc;
wimg(:,:,2)=gc;
wimg(:,:,3)=bc;
[m n l]=size(wimg);
Mat= imshow(uint8(wimg));
Imwrite(mat,'image')
Imshow(uint8(mat)),title('recovery watermark')
```

### 6.3 Evaluation and Improve Implement of Watermarked Image

The scheme applies multi-resolution DWT to host image and watermarked image. The wavelet coefficients are used to determine complexity of image texture. And then, with the HVS characteristics, the watermark signal is embedded into low-frequency part of the image. Selecting the watermark embedding intensity and different texture thresholds at various resolution layer of the image, and the digital watermark image is embedded into the corresponding resolution layer of the decomposed host image; as a result, robustness of the watermarking algorithm is improved.

The information embedding process can be deemed a weak signal (embedded information) superposed in the strong background (host image). As long as the superimposed signal is lower than the contrast threshold, the HVS will fail to perceive the presence of such signal. This is the main basis for selection of information embedding region.

The watermark image is embedded into the bright energy-intensive low frequency region. Since the wavelet transform obeys conservation of energy. Therefore, the total energy remains constant after the image transformation. Watermark image embedded in the region is hardly perceivable.

In a binary wavelet transform, the sampling is done based on the wavelet function. Therefore, it exhibits a self-similarity. Retention of such self-similarity can yield a watermarked image with better overall performance, so the self-similarity shall be kept when the strength coefficient  $a(r, g, b)$  is embedded into different frequency domains, and the digital color wavelet watermark image is embedded into the host image.

Wavelet tree coefficients maintain the characteristics of image among the image space points. After discrete wavelet transform, sub-space is the coefficient from top to

bottom (sub-band) between the image space points relative seat, there is a relationship, such as the  $LL3$  sub-band, if it is the parent coefficient (Parent coefficient), there are four sub-factors in the  $LL2$  (Child coefficient) corresponding, there are 16 sub-factors in the  $LL1$  sub-factor corresponding, so this correspondence that a wavelet tree structure can be established in watermark embedding process, and we should pay attentions to maintaining the original image wavelet tree coefficients characteristics between points of image spatial which is capable to improve the quality of the embedded image.

## 6.4 Experimental Research of Watermark Image Embed and Extract

Because the 2D discrete feature of a digital image, it is suitable for 2D DWT. Every run of 2D DWT can generate the low-frequency approximated subimage and three high-frequency detail subimages, including horizontal subimage, vertical subimage and diagonal subimage. This is repeated for the next layer DWT to generate low-low frequency subbands. The image has transformed by  $n$ -layer DWT that produces  $3n+1$  blocks. The size of  $n$ -class subimage is  $1/2^n$  that of the original image, for example, an image is decomposed by 3-layer wavelet ( $n=3$ ) and results in 10 subimages, then the size of 3-layer subimage is  $1/8$  that of the original image. The number of layers of decomposition in these experiments equals to 3, as shown in figure 6.6.

In order to improve quality of the extracted watermark image, the embedding weight factor can be set as follows: red weight coefficient  $C_r = 0.2$ , blue weight coefficient  $C_b = 0.4$ , and green weight coefficient  $C_g = 0.1$ . Decompose the host image and apply simple 'Haar' into the cross multi-resolution coefficient sets:  $LL_1 (cA_1)$ ,  $HL_1 (cH_1)$ ,  $LH_1 (cV_1)$ , and  $HH_1 (cD_1)$ , as shown in figure 6.6 (a).

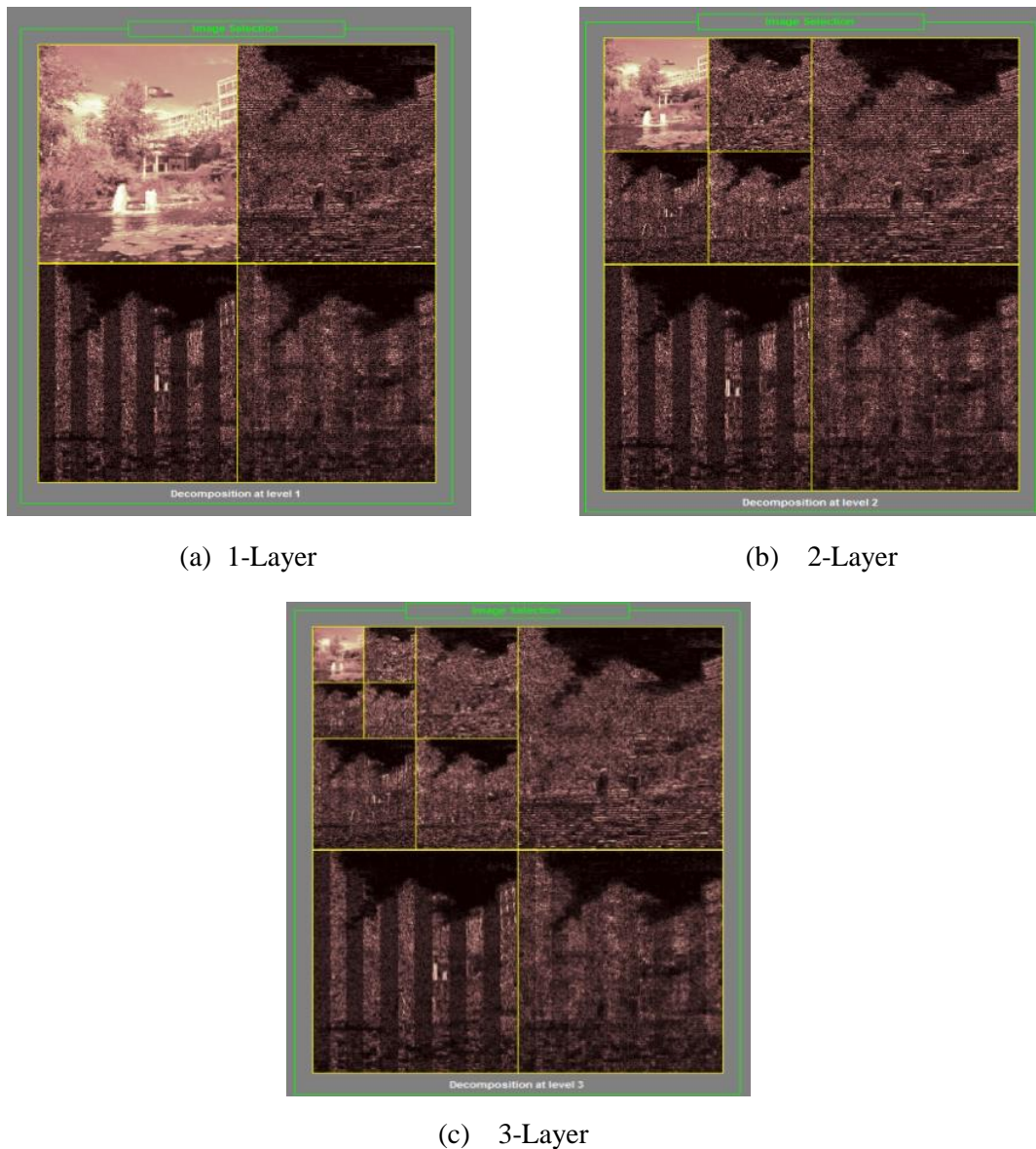


Figure 6.6 Decomposition of 3-layer DWT

Perform 2-layer DWT on  $LL_1$  to give four coefficients:  $LL_2$  ( $cA_2$ ),  $HL_2$  ( $cH_2$ ),  $LH_2$  ( $cV_2$ ) and  $HH_2$  ( $cD_2$ ), as shown in figure 6.6 (b). Repeat DWT for  $LL_2$  to yield 3-layer components:  $LL_3$  ( $cA_3$ ),  $HL_3$  ( $cH_3$ ),  $LH_3$  ( $cV_3$ ) and  $HH_3$  ( $cD_3$ ), as shown in figure 6.6 (c). The wavelet coefficient of watermarked is embedded to  $HL_3$  coefficients of the host image as per the following program :

The wavelet coefficients of the watermark image (as ‘ $m$ ’ code) :

```
rmA1; rmH1; rmV1; rmD1
```

Wavelet coefficients of the host image (as 'de' code) :

rdeH1; rdeV1; rdeD1;  
rdeH2; rdeV2; rdeD2;  
rdeA3; rdeH3; rdeV3; rdeD3;

Wavelet coefficients of the watermarked image (as 'dc' code) :

rdcH1; rdcV1; rdcD1;  
rdcH2; rdcV2; rdcD2;  
rdcA3; rdcH3; rdcV3; rdcD3;

In order to prove that the method is suitable for any size of image embedded, we adopt three different sizes when embedding the watermark image in this section, i.e., the first size is  $128 \times 128$ , the second one is  $256 \times 256$ , the third one is  $512 \times 512$ ; their formats are *jpg* and they are justified by experimental data.

Peak signal-to-noise ratios (*PSNR*) values are between 20 *db* and 40 *db*. In the experiment, *PSNR* values are used to measure similarity level between the watermarked image and the host image. Noise criterion (*NC*) values are used for detecting invisibility, explicitness and robustness of the watermarked image. The above two kinds of standard evaluation indicators are used for the watermarked image and the recovered image. In addition, the root mean square error (*RMSE*) is used to measure the fidelity of the image.

(1) The 3-layer DWT algorithm used the host image (*pool.jpg*  $512 \times 512$ ) and the watermark image (*castel.jpg*  $128 \times 128$ ). The host image is decomposed by 3-layer wavelet decomposition. The watermark image is decomposed by 1-layer DWT as shown in figure 6.7 where embedding is successful. Syntax embedding program is as follows:



```

rdcA3=rdeA3+cr×rmA1;
rdcH3=rdeH3+cr×rmH1;
rdcV3=rdeV3+cr×rmV1;
rdcD3=rdeD3+cr×rmD1;
rdcH2=rdeH2;
rdcV2=rdeV2;
rdcD2=rdeD2;
rdcH1=rdeH1;
rdcV1=rdeV1;
rdcD1=rdeD1;

```

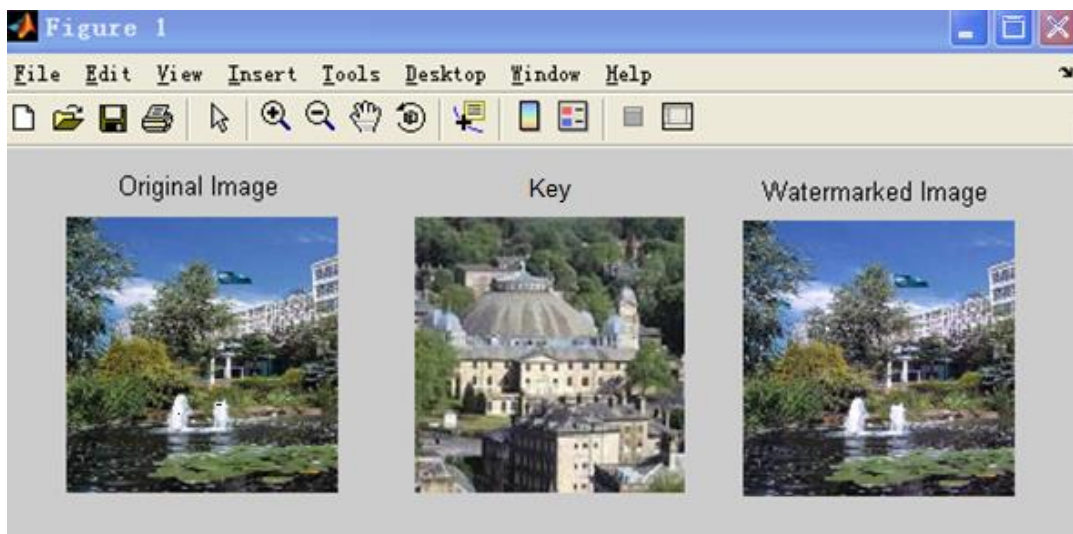


Figure 6.7 The 3-layer DWT of the watermark embedded into host image pool  
(between the watermarked image  $512 \times 512$  and the host image  $512 \times 512$ )

$$\begin{aligned}
 PSNR &= 36.5772, & SNR &= 29.4320, \\
 RMSE &= 14.3007, & NC &= 0.9503
 \end{aligned}$$

Figure 6.8 shows the result of extracting the watermark from image pool.

The syntax of the extracting watermark image is as follows:

```

rcA1=(rdcA2-rdeA2)/cr;
rcH1=(rdcH2-rdeH2)/cr;
rcV1=(rdcV2-rdeV2)/cr;
rcD1=(rdcD2-rdeD2)/cr;

```

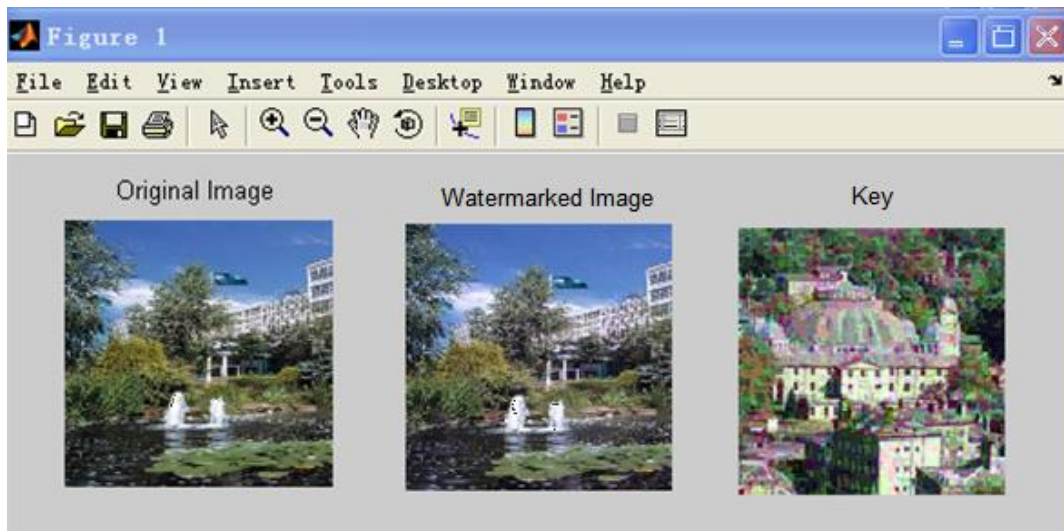


Figure 6.8 Watermark extracted from watermarked image pool

$$PSNR = 32.5842, \quad SNR = 27.4791,$$

$$RMSE = 117.7894, \quad NC = 0.9987$$

- (2) The second group is the host image (*campus.jpg* 512×512) and watermark image (*gate.jpg* 256×256) as the encryption key. It is shown in figure 6.9. *Campus.jpg* is decomposed through 2-layers wavelet decomposition, while *gate.jpg* is decomposed through 1-layer DWT. The watermark weight factors are  $C_r=0.2$ ,  $C_g=0.1$  and  $C_b=0.4$ .

The syntax of embedding is as follows:

```

rdcA2=rdeA2+rcA1×cr;
rdcH2=rdeH2+rcH1×cr;
rdcV2=rdeV2+rcV1×cr;
rdcD2=rdeD2+rcD1×cr;
rdcH1=rdeH1;
rdcV1=rdeV1;
rdcD1=rdeD1;

```

Figure 6.9 shows results of watermark embed into campus image.

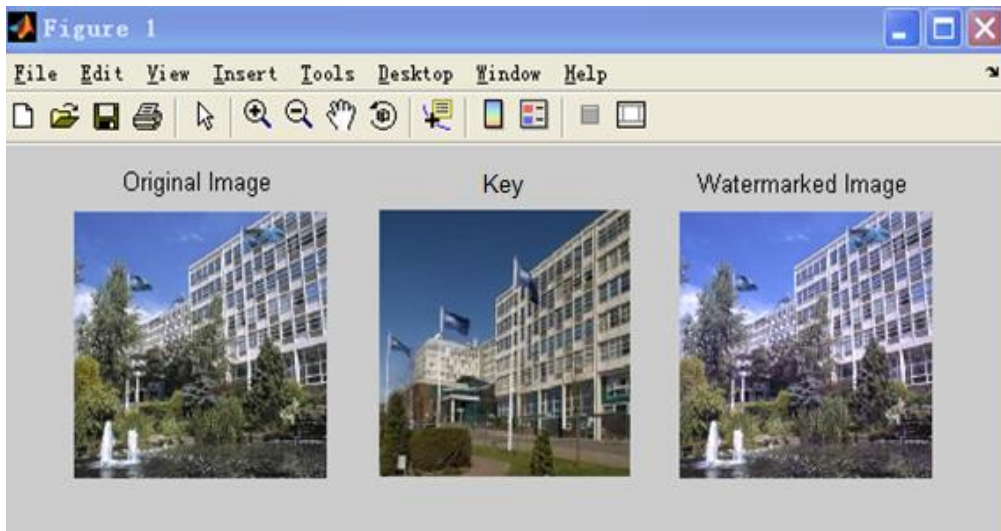


Figure 6.9 The 2-layer of DWT embedded into image campus

$$PSNR=32.7721, \quad SNR =26.5769,$$

$$RMSE=34.3458, \quad NC=0.9341$$

The syntax of watermark image extracting program is as follows:

```
rcA1=(rdcA2-rdeA2)/cr;
rcH1=(rdcH2-rdeH2)/cr;
rcV1=(rdcV2-rdeV2)/cr;
rcD1= (rdcD2-rdeD2)/cr;
```

Figure 6.10 shows result of watermark extracted from campus image

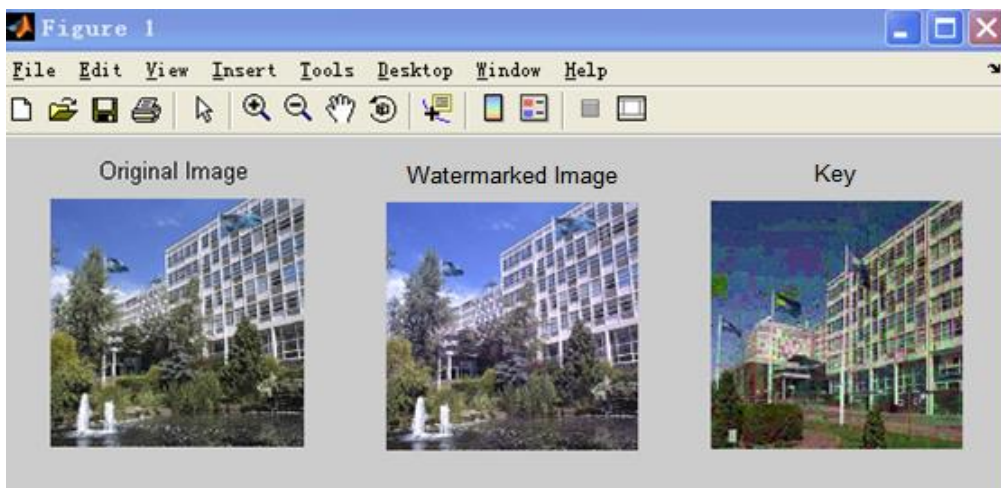


Figure 6.10 IDWT to extract from image gate

(extracted watermark image 256×256)

$$PSNR =29.9014, \quad SNR =22.4457,$$

$$RMSE =66.5186, \quad NC =0.9996$$

- (3) The final group is the host image (*campus.jpg* 512×512) and watermark image (*library.jpg* 512×512) that serves as the encryption key. It is shown in figure 6.11 that *campus.jpg* is decomposed by 2-layer wavelet decomposition, and *library.jpg* is decomposed by 2-layer DWT. The watermark weight coefficients are  $C_r=0.2$ ,  $C_g=0.2$ , and  $C_b=0.2$ .

The syntax of embedding watermark image program is as follows:

```
rdcA2=rdeA2+rCA2×cr;
rdcH2= rdeH2+rCH2×cr;
rdcV2= rdeV2+rCV2×cr;
rdcD2=rdeD2+rCD2×cr;
rdcH1=rdeH1+rCH1×cr;
rdcV1=rdeV1+rCV1×cr;
rdcD1=rdeD1+rCD1×cr;
```

The syntax extracted watermark as follows:

```
rCA2=(rdcA2-rdeA2)/cr;
rCH2=(rdcH2-rdeH2)/cr;
rCV2=( rdcV2- rdeV2)/cr;
rCD2=(rdcD2-rdeD2)/cr;
rCH1=(rdcH1-rdeH1)/cr;
rCV1= (rdcV1-rdeV1)/cr;
rCD1=(rdcD1-rdeD1)/cr;
```

The size of the watermark image is the same as that of the host image. In order to obtain a better watermarked image, the number of wavelet transform layers of watermark image is the same as that of the original host image, and accordingly their wavelet coefficients have certain correlation. The watermarked image generated by combining wavelet coefficients is capable of maintaining self-similarity among various layers of subimages and wavelet tree structure. Meanwhile, better watermarked image and the recovered watermark image are obtained as shown in figure 6.11 and figure 6.12.

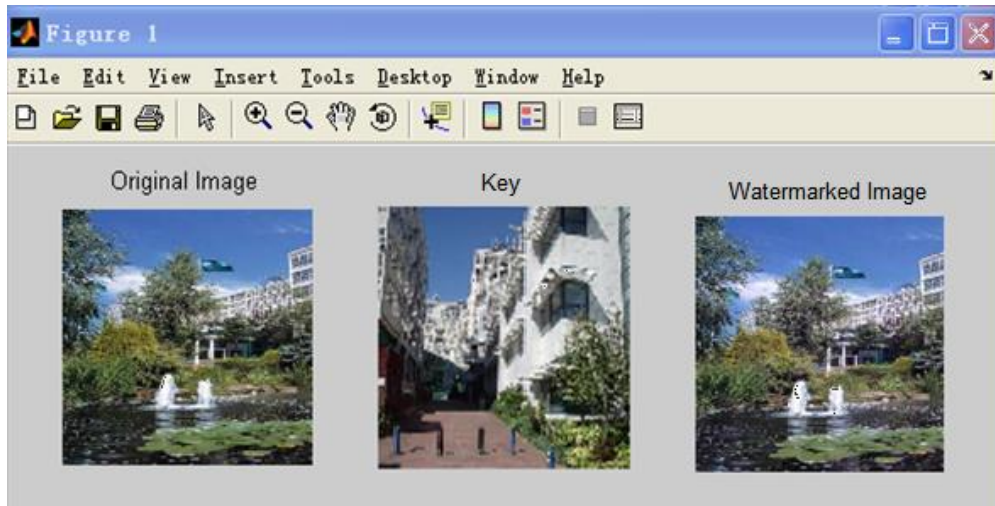


Figure 6.11 The 2-layer of DWT embedded into image pool  
 (between watermarked image  $512 \times 512$  and original (host) image  $512 \times 512$ )  
 $PSNR = 28.9713$ ,  $SNR = 23.7306$ ,  
 $RMSE = 82.4051$ ,  $NC = 0.9147$

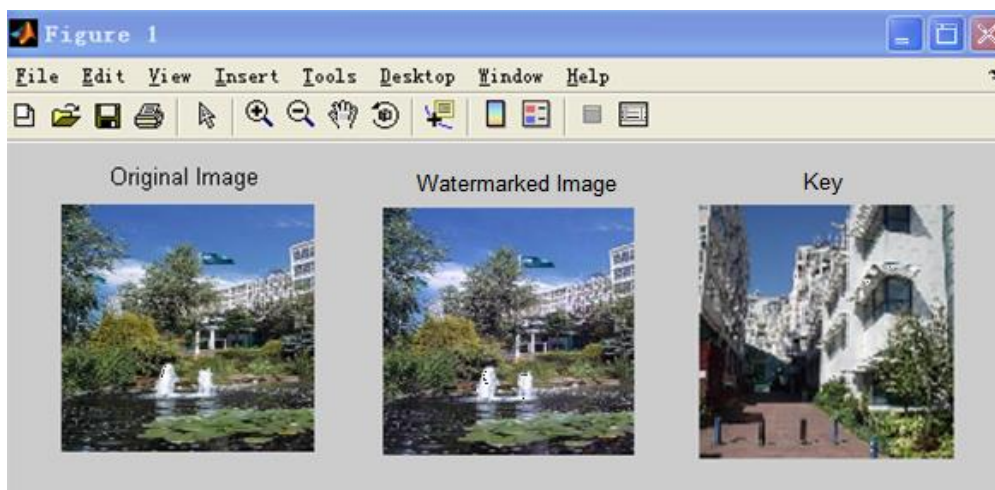


Figure 6.12 IDWT to extract from image library  
 (extracted watermark image  $512 \times 512$ )  
 $PSNR = 29.474$ ,  $SNR = 20.8304$ ,  
 $RMSE = 73.3911$ ,  $NC = 0.9738$

The greater the selected weight coefficient of a watermark image, the better the watermark image is recovered. But the watermarked image will reveal the trace of watermark image. Further encryption of the watermarked image with a hyperchaotic system can improve information hiding reliability and security, see appendix C in which hiding information via chaotic encryption and digital watermarking prevents

the watermarked images ‘pool’ and ‘library’ from being disclosed, meanwhile the extracted watermark image (or hidden cipher text) ‘library’ can be recovered more clearly, hence, the human visual perception is better, and calculated *PSNR* and *NC* values can satisfy the assessment criteria of image watermarking. Those images (figure 6.2(a) and figure 6.2(b)) for the experiment were downloaded from the University of Derby. (University of Derby, 2013)

## 6.5 Summary

The chapter describes how a color image is scrambled and embedded into the host image as a watermark; the important features are as follow

- (1) According to literature review, existing watermarking algorithms hide information in the space domain of index image, but this algorithm is carried out in the wavelet domain of the image.
- (2) The algorithm integrates HVS characteristics to choose effective information-embedding region and quantification factor below the visual distortion value. It not only ensures the invisibility of the embedded information but also enhances its robustness.
- (3) The algorithm has strong robustness and good immunity to JPEG compression, image cutting and other attacks.
- (4) The algorithm eliminates the traces of embedded information efficiently with good security, and obeys the Kerckhoffs’ principle. Even if the specific details of the algorithm are known, it would be very difficult for the attacker to crack the content of confidential information without the cipher key.
- (5) The algorithm is strongly adaptable. It can be used not only for digital watermark to protect copyright, but also for safe transmission of confidential information during covert communication.

Only the brightness signal of the information-hiding watermark image is used in the low-frequency subband coefficient of wavelet transform, meanwhile, the detail component and the embedded information of the hue and saturation signal are not used. To a certain extent, it avoids the possible visual distortion caused by repeated changes of region, but also limits the capacity of information embedded.

## Chapter 7

---

### **Hyperchaotic Encryption Based on Combined Wavelet Transform Algorithm**

The digital watermarking algorithm is based on DWT, and the watermark image is scrambled by hyperchaos. The multiresolution discrete wavelet with 'Haar decomposition' is adopted in watermark embedding. The scheme is simple, symmetric and highly secure, resulting in different ways of detection, such as cropping, JPEG compression and rotation.



## 7.1 Combination of Hyperchaotic and DWT Image Encryption

There are many properties owned in hyperchaotic, which are suitable for the encryption. Both initial value and control parameters are useful for generating the encryption key and rearranging pixels of a color image. The watermark adopting scrambled image can be embedded into the host image by wavelet transformation.

The world is multicolored in the human eyes, and without color we cannot find important information from images. Therefore, the embedding and extraction is without any loss of information represented by colors. The advantage of digital watermark algorithm is that encryption information is hidden into watermarked image, in order to premise invisible watermark, which can increase the amount of secret information embedded.

Firstly, it adopts hyperchaotic map systems to scramble the watermark image, which is capable to improve the safety and reliability of information encryption.

Secondly, it used wavelet transformation to decompose host images and scrambled images.

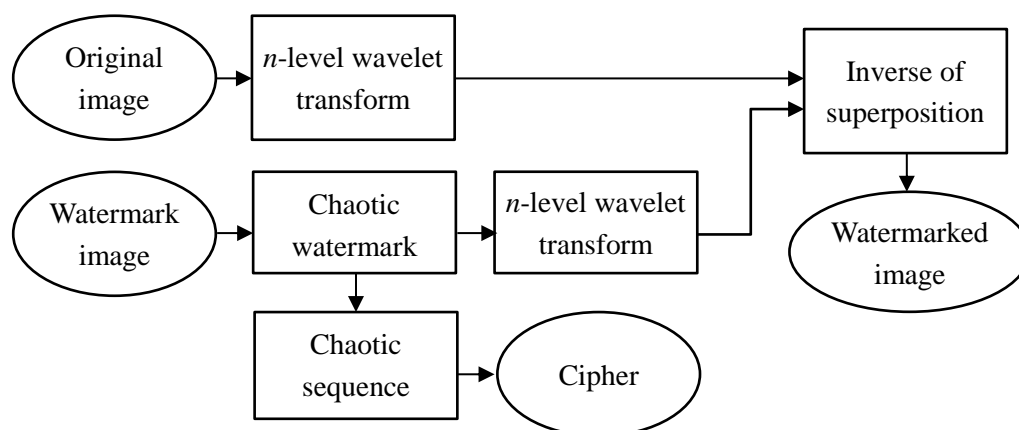


Figure 7.1 The encryption process on basis of hyperchaos and DWT

Finally, the watermark is inserted into  $LL$  part of the host image. The process diagram of the algorithm by using DWT with hyperchaotic is shown as below. Figure 7.1 shows the encryption process on basis of hyperchaotic and DWT

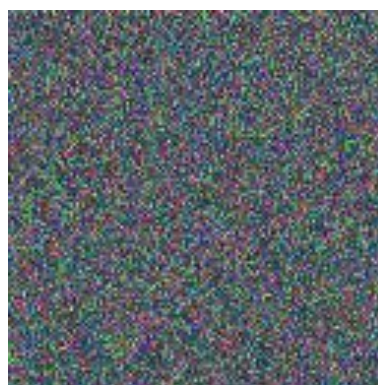
The  $128 \times 128$  pixel 'gate.jpg' image (a) is watermark image. This image was encrypted with initial parameters  $x_1=0.9$ ,  $y_1=0.1$ ,  $z_1=0$ ,  $a=1.6$ , and  $b=0.2$ . The watermark was scrambled by hyperchaos (3D Henon, 1D Logistic and 2D Arnold) to improve security levels. The scrambling image (b) is as shown in figure 7.2.

Then, it is available to decompose the scrambling watermark image using 'haar' DWT into the 1-layer overlapping multi-resolution coefficient set:  $LL1$ ,  $HL1$ ,  $LH1$  and  $HH1$ . Using the same method, as for the  $256 \times 256$ -pixel host image 'campus.jpg', we can obtain the 2-layer overlapping multi-resolution coefficient set:  $LL2$ ,  $HL2$ ,  $LH2$ , and  $HH2$ .

The strength coefficients chosen by bright equation, the scrambling watermark image figure 7.2 (b) as the low frequency wavelet coefficients embedded into the host image; and a new watermarked image will be obtained.



(a) Watermark image  
gate.bmp 128×128



(b) Watermark scrambled



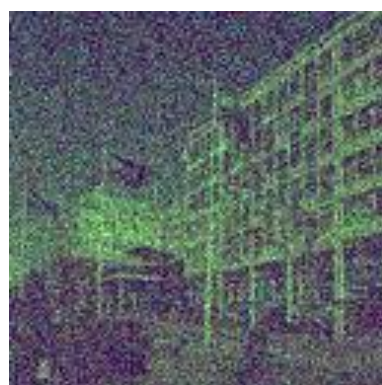
(c) Host image  
campus.bmp 256×256



(d) Watermarked image



(e) Extract watermark



(f) Watermark recovered

Figure 7.2 The scrambling image embed into the host image process

Figure7.2(a) is scrambled by hyperchaos as watermark image figure7.2(b). The host image figure7.2(c) and the watermark image are decomposed by using wavelet transformation, which generate wavelet coefficients of watermarked image then it is available to adopt wavelet reconstruction to obtain watermarked image figure7.2 (d). The watermark image figure7.2 (f) is recovered by chaotic inversely transformation. Algorithm achieve a certain level of restoration for watermark image, it is difficult to achieve no-distortion decryption.

However, for the images transform in  $n$ -layer wavelet decomposition, the size of low frequent sub-image is  $\left(\frac{1}{2^n}\right)^2$  of the host image size. The most energy of image is concentrated in the low frequency  $n$ -layer sub-image. Because it is capable, the coefficient weighed of watermark image can adopt larger value. Not only the

watermarked image has better no-visibility, but also the extracted watermark image can obtain higher *PSNR* value and *NC* value between watermark image and recovery watermark image. In the Chapter, the following to be proposed is about a hyperchaotic map combined with wavelet transform on the color image encryption algorithm.

## 7.2 Improved Implementation of Method

In the wavelet analysis, it is important to select the best wavelet basis. The wavelet variance is spatial scales of variability in geophysical data. For recognizing, the difference of the wavelet basis can be adopted to identify the number and level of wavelet variance. The hyperchaotic watermark image is embedded into the low frequency wavelet of host image by 2D wavelet after experimental adopted right wavelet basis. The digital watermarked image has the best comprehensive performance and high *SNR*.

### (1) Watermark embedding by adaptive quantization

Given  $N \times M$  watermark image (scrambling image) and  $N \times M$  host image apply the standard DWT up to  $k$ -layer, which generate a total of  $k$  sub-bands of wavelet coefficients, then wavelet coefficients are selected for an approaching sub-band coefficients and three detailed sub-band coefficients. Adaptively embed the scrambling watermark image into the host image corresponding wavelet coefficients of low-frequency. Make the low-frequency block  $U_k$ , which is obtained from the low frequency band of original host image after IDWT transform. The embedding formula is shown as following:

$$C'_k = C_k + \alpha \times V_k, \quad k = 1, 2, \dots \quad (7.1)$$

Its production rules  $C'_k$  are given by:

$C_k \rightarrow$  The former wavelet coefficient value of low-frequency sub-block  $U_k$ .

$V_k \rightarrow$  The  $No.k$  component weight of one-dimensional digital watermark sequence  $V_k$ .

$C_k' \rightarrow$  The new wavelet coefficient value of low-frequency sub-block  $U_k'$ .

$\alpha \rightarrow$  The embedding depth weight for digital watermark image

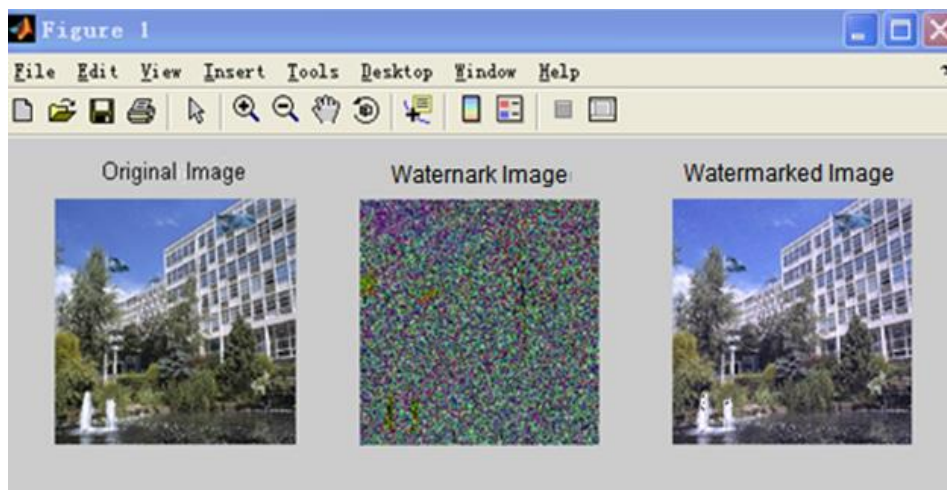


Figure 7.3 Result after embedding the scrambling image  
( between watermarked image  $512 \times 512$  and host image  $512 \times 512$  )  
 $PSNR = 32.6290$ ,  $NC = 0.9341$ ,

If watermarked signal is lower than the HVS contrast threshold, the HVS will not be able to feel the watermark presence in accordance with the vision of its minor change to the more sensitive, because the HVS can distinguish the strength of image edge and texture information.

It adopts DWT derived features for digital image texture and image edge; the image combines the strength of watermark to achieve the adaptive embedding. It is capable of not only improving the strength of embedded watermark, but also ensuring the visibility. In accordance with wavelet analysis theory, the image texture and edge contains maximum wavelet coefficient, which can be applied to the embedding strength.

(2) Watermark extraction by quantization

For  $N \times M$  watermarked image, apply the DWT up to  $n$ -layer, which gets four sub-bands of wavelet coefficient, which of one is extracted out the original host image's the wavelet coefficients, from the watermarked image's wavelet coefficients to obtain the watermark image's wavelet coefficients. The extraction formula is as following:

$$V_k = C_k - C_k / \alpha; \quad k = 1, 2, \dots \quad (7.2)$$

The watermark image's wavelet coefficients are reconstructed by 2D-IDWT inverter transformation that obtains scrambled watermark image

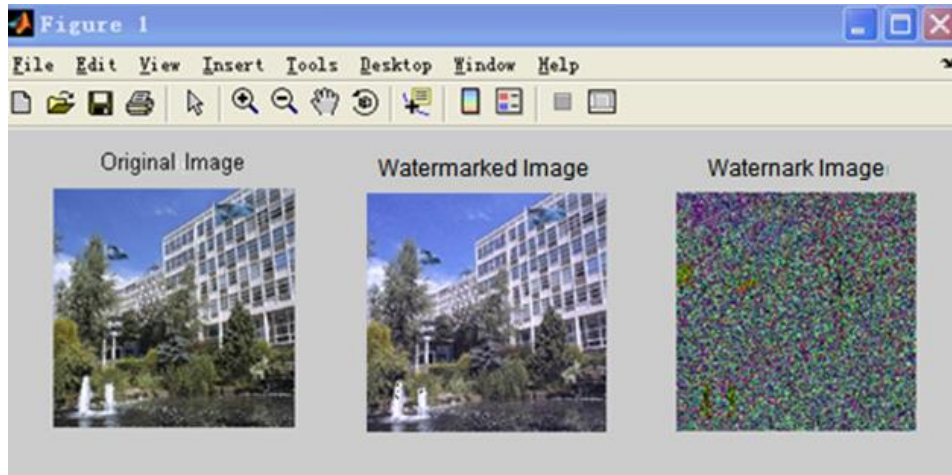


Figure 7.4 Result after extraction by IDWT  
( between host image  $256 \times 256$  and watermarked image  $256 \times 256$  )  
 $PSNR = 35.0078$ ,  $NC = 0.9826$ ,

### 7.3 Parameters Setting and Attack Test

In order to improve the robustness of watermark, it is needed to detect for watermarked image (cropping, *jpg* compression, adding noise signal and rotation), and the following experiments adopt  $PSNR$ ,  $NC$  to measure relevancy among host image, watermark image and watermarked image.  $NC$  values are much closer to 1, which indicates that the extracted watermark and original watermark are similar. The watermarking similarity is a standard to measure watermarked image quality. Image

distortion can be used with  $PSNR$  to indicate. So, the stability of watermark image is tested through a series of perturbations.

### 7.3.1 General Watermarking Testing

It adopts  $512 \times 512$  color image 'campus' (figure 7.5(a)) as the host image and  $256 \times 256$  color image 'gate' (figure 7.5 (b)) as the watermark, using the method introduced above to construct and detect watermark information. Those images (figure 7.5(a) and figure 7.5(b)) for the experiment were downloaded from the University of Derby. (University of Derby, 2013)



(a) Host image



(b) Watermark



(c) Watermarked image


Figure 7.5 Watermark embedded image ( $T=0.06$ )

For extraction of unattached watermark, figure 7.5(b) is the watermark image directly extracted from watermarked image figure 7.5(c) and its  $PSNR=32.77$  and  $NC=0.9341$ . After repeated experiments, it is found that the watermarked image effect is relatively good when  $T$  ranges from 0.04 to 0.08.

(1) 'jpg' lossy compression

Most of the images transmitted via the internet are currently compressed in *jpg* format in order to reduce the file size and increase transmission rate. Thus it is quite necessary to guarantee the robustness of an algorithm against *JPEG* compression, for *JPEG* formation is widely applied for image transmission over the networks. It adopts quality factor of diminishing to compress a *JPEG* image, and then extracts the watermark from the compressed image. *PSNR*, *MER* and *NC* can be used to measure the effects of *JPEG* compression. The watermarked images after *JPEG* compression and the results of comparison are shown in table 7.1 below.

Table 7.1 Detection values of watermarked image after *Jpeg* lossy compression

Watermarked image 1024×1024	Watermarked image 512×512	Watermarked image 256×256
		
Recovered watermark	Recovered watermark	Recovered watermark
		
<i>PSNR</i> =23.2125 <i>MER</i> =310.3326 <i>NC</i> =0.9839	<i>PSNR</i> =28.6545 <i>MER</i> =88.6392 <i>NC</i> =0.9844	<i>PSNR</i> =30.0514 <i>MER</i> =64.2606 <i>NC</i> =0.9850

The *NC* value can remain at  $NC \approx 1$  so as to effectively keep high compression ratio without distorted image. As a result, when the image 1024×1024 is compressed to the size of 256×256; the detected *NC* ranges from 0.9839 to 0.9850. The experimental result shows that such method has good robustness to resist *JPEG* compression attacks.

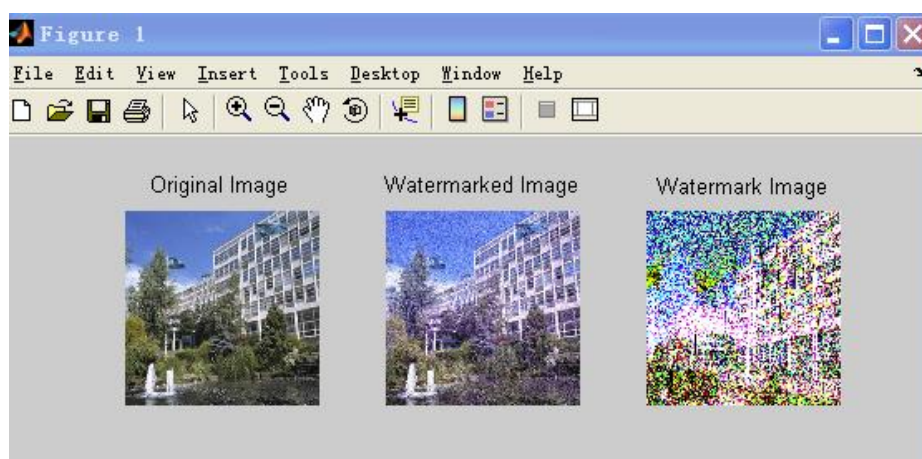


## (2) Add noise signal

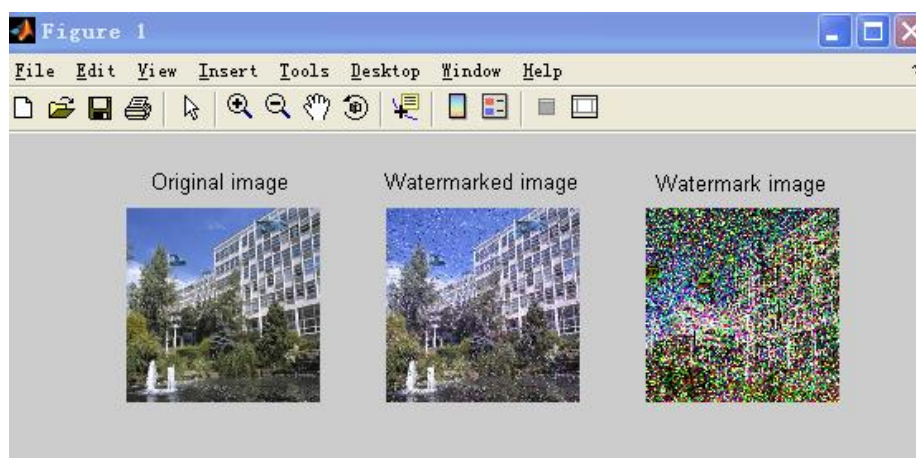
To test the robustness of DWT watermarking algorithm against noise attacks, it is needed to process the watermarked image by adding noise signal. (Nikolova, 2005) Normally, the greater the variance of noise is, the higher the energy of noise. Make white noise and salt pepper noise are adopted in the experiment to perform the detection; the result is shown in table 7.2. Figure 7.6 shows the effect after adding noise.

Table 7.2 Detection values results for watermarked images after adding noise

Type of noise	Gaussian (0,0.005)	Gaussian (0,0.001)	Salt and pepper (0.02)	Salt and pepper (0.04)
<i>NC</i>	0.8473	0.8239	0.8645	0.8276



(Gaussian)



(Salt and pepper)

Figure 7.6 Result after adding noise

With increasing noise, the distortion is becoming larger, but the detected similarity coefficient is bigger than  $T$  (that is ' $sim > T$ '), which indicates that such an algorithm still has strong resistance against noise attacks.

(3) Rotation

Rotation is a kind of powerful attack; it tests the image by rotating the watermarked image at different angles. Figure 7.7(a) shows the image rotating by 10 degrees; and figure 7.8(a) shows the image after reversely rotating by 30 degrees. The extraction results after rotation are shown in figure 7.7(b) and figure 7.8(b). The  $NC$  values are less than threshold  $T$ , and they are relatively close.

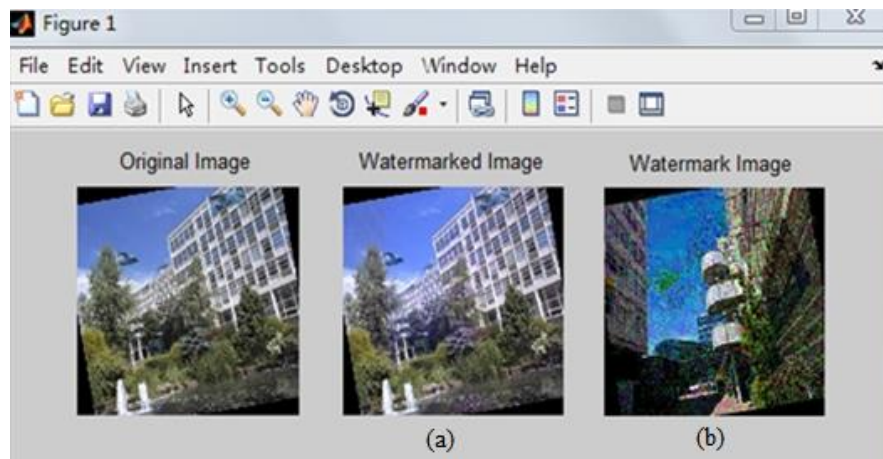


Figure 7.7 Detection values and extraction for watermarked image result after rotation 10°  
Rotated 10°  $NC=0.9206$

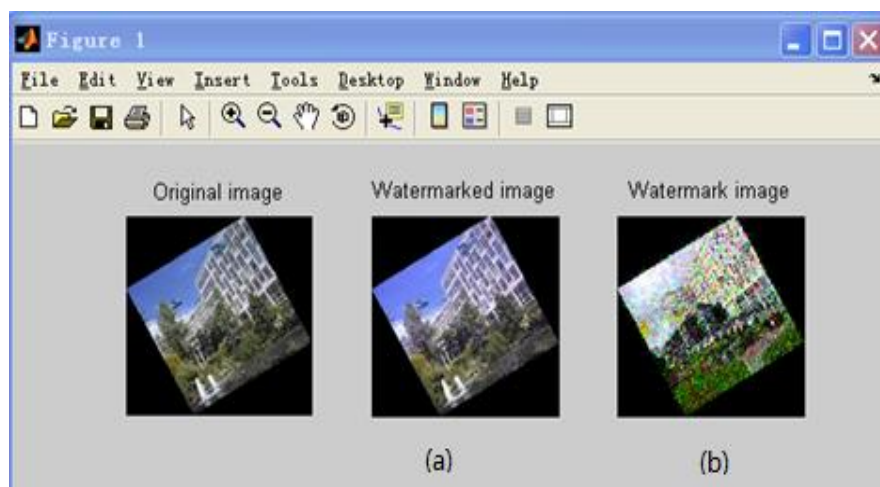


Figure 7.8 Detection values and extraction for watermarked image result after rotation 30°  
Rotated 30°  $NC=0.9206$

We rotated the watermarked image with its center as the circle center, and then cropped the rotated image to retain the central part of it to the maximum extent; to extract watermark, we also rotated the host image by the same angle and got the watermark image (figure 7.8).

#### (4) Cropping

Crop watermark embedded image from different position and by different ratios, as shown in figure 7.9. The detection results are as shown in table 7.3.

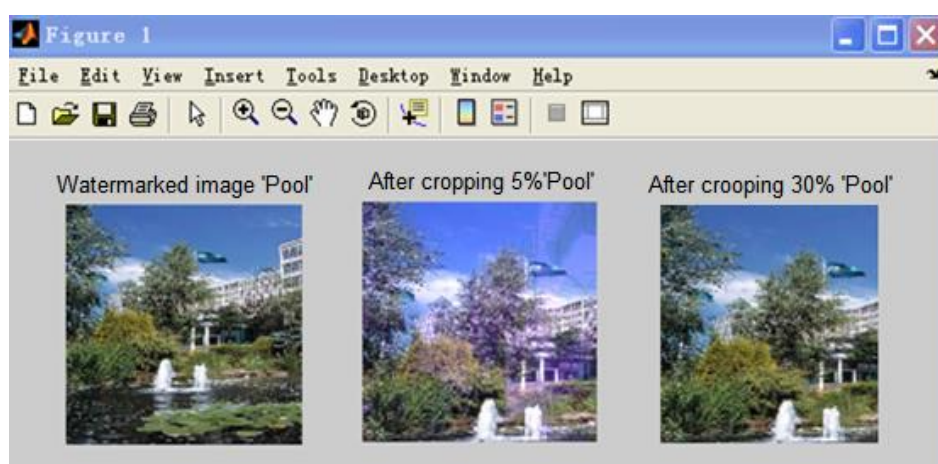


Figure 7.9 Detection value for watermark image after cropping

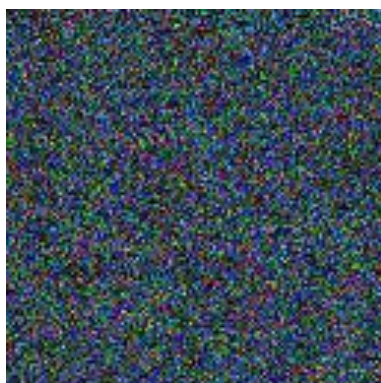
As the cropping area increasing, the distortion of image is larger, and the effect of watermark detection and extraction is becoming worse; however, even cutting area has reached up to 30%, the similarity value  $sim$  is still larger than threshold value  $T$ , which suggests such an algorithm remains better robustness against cut attacks. Therefore with twice detector, this proves its efficiency.

Table 7.3 The detection result after crop watermark embedded image

Shear strength	5% Watermarked image	5% Watermark image	30% Watermarked image	30% Watermark image
$PSNR$	26.8929	26.8239	26.7402	26.8276
$NC$	0.8473	0.8239	0.8645	0.8276

### 7.3.2 Chaotic Watermark Testing

As the above, figure 7.5(a) serves as the host image while figure 7.5(b) is the scrambled image after chaotic encryption; figure 7.10(a) is embedded into the host image to obtain the chaotic encrypted image figure 7.10(b).



(a) Scrambled image



(b) Watermarked image

Figure 7.10 Chaotic watermarking result

Given that all probable attacks can be resisted and the quality of the original image can be guaranteed, the lowest bound of test registration rate can be selected as the threshold. Hence, when the attack-incurred registration rate is lower than the threshold, such attack makes no sense because the image must have been disrupted greatly.

At  $T=0.04$ , the embedding time used is 1.5313s. The result shows no much change in host image. In the experiment,  $T$  is scale factor used to control the watermark embedding strength.

#### (1) 'jpg' lossy compression

The chaotic watermarked image is processed with *JPEG* lossy compression and then detected; the detection result is shown in table 7.4

Table 7.4 Detection values of chaotic watermarked image after *Jpeg* lossy compression

QF	75%	50%	25%	10%	5%
<i>sim</i>	1	1	1	1	1

The watermarked image is compressed in standard *JPEG* format with different quality factors QF, and then detected for zero watermarks. For compression attacks with different QFs, the smaller the QF, the larger the compression ratio, the more distorted the image becomes, and the more difficult it is to detect watermark. But even when the compression ratio is equal to 5, the detected *sim* is still 1. Therefore, this algorithm has very strong robustness against *JPEG* compression attacks.

(2) Add noise signal

In order to detect how this algorithm can resist attacks of Gaussian white noise and salt and pepper noise, we superposed Gaussian white noise and salt and pepper noise onto the watermark-embedded image respectively. Although the image has been distorted conspicuously after superposed with noise, it is still possible to extract the watermark. The corresponding *PSNR* and *NC* values of the extracted watermark images are shown in table 7.5.

Table 7.5 Detection values results for chaotic watermarked images after adding noise

Type of noise	Gaussian (0, 0.001)	Gaussian (0, 0.002)	Salt and pepper (0.01)	Salt and pepper (0.02)
<i>PSNR</i>	31.91	26.11	36.42	35.52
<i>NC</i>	0.8074	0.7073	0.9006	0.8477

The above experiment reveals that, this algorithm is fairly robust to noise attacks with moderate intensity, with better resistance to salt and pepper noise attack than to Gaussian white noise. However, in the event of high intensity noise attacks, the watermark image is very likely to be damaged, hence the algorithm robustness is poor at this time and certainly the host image becomes meaningless.

## (3) Rotation and cropping

Cropping operation has remarkable threat to watermark detection because it deletes part of the image and causes loss of image information. To test the cropping resistance of this watermarking algorithm, the watermark-embedded image has been cropped at different ratios in this experiment.

The comparison of the above three cases shows that, the extracted watermark becomes more and more seriously damaged with increasing ratio of cropping, furthermore, the original watermarked image becomes damaged very severely, too, and almost unusable.

We rotated the chaotic watermarked image with its center as the circle center, and then cropped the rotated image to retain the central part of it to the maximum extent; to extract the watermark, the rotated image was reversely turned to restore the original direction, and the cropped blank was filled with original watermarkless image. The watermark image is extracted after rotating the watermark-embedded image by  $1^\circ$ .

Table 7.6 Detection values results for chaotic watermarked images after rotation and cropping

Type of attack	13%	25%	50%	Rotation $1^\circ$
<i>PSNR</i>	30.78	28.97	27.14	34.72
<i>NC</i>	0.9436	0.8878	0.7708	0.6057

The above experimental result shows that, this algorithm has strong ability to resist cropping attack; even the watermarked image was cropped in half, the watermark image can be detected very well by using the algorithm. Rotation not only destroys visual quality of the image but also its internal structure and attributes were destructed. As can be seen from the experimental result, the algorithm has weak robustness to rotation attack; and after the cropping angle increases, it becomes very difficult to detect the watermark again and the image distortion becomes severe.

## 7.4 Summary

The Chapter summarizes the main principles and algorithms described in the previous chapters; a systematic combination of scrambling, embedding, and extracting was implemented. The main research works of this chapter are as follows:

- (1) A digital watermark embedding and extracting algorithm and its process have been proposed. It is shown from massive experimental results that the method is feasible and highly confidential
- (2) Haar wavelet basis has been adopted during research and experiments, characterized by linear phase; it can be reconstructed by wavelet inverse transformation. Its watermark image can generate perfect recovery effect; the extracted watermark image is highly related to the original one, so this is an ideal algorithm.
- (3) This chapter has summarized methods of embedding a watermark image into 1/4 or 1/16 of the host image.
  - a. Perform two-dimensional 1-layer wavelet transform for the watermark image to embed.
  - b. Perform two-dimensional 2-layer wavelet transform for the host image when the watermark image is 1/4 of the host image, and two-dimensional 3-layer wavelet transform for the host image when the watermark image is 1/16 of the host image.
  - c. Finally, embed the corresponding wavelet coefficient of the watermark image into that of 2-layer or 3-layer of the host image to complete the 1/4 or 1/16 embedding operation.

In this chapter, anti-attack experiments of digital watermark images have been carried out, such as cropping attack test, rotation attack test, random noise attack, and high-pass filter attack; the experiments verifies that digital watermark has strong resistance to attacks, and indicates that the combination of chaotic scrambling watermark and wavelet (high-dimensional) transform watermark can generate secure ciphertext and recover information clearly and reliably..



## **Chapter 8**

---

### **Conclusions and Future Work**

The chapter summarizes the major focuses and contribution of the thesis, and proposes some potential future directions of research.

## 8.1 Summary of the Thesis

In summary, this thesis has proposed an improved digital image watermarking system based on the Henon chaotic system, multidimensional chaotic system and DWT. In order to indicate the research clearly, its focus are summarized as follows:

- (1) The main focus of this thesis is to develop a hyperchaotic watermarking system. By reviewing the background and important of the watermark encryption technologies, the present study has explored a hyperchaotic digital image watermarking system. The study is divided into two major parts: hyperchaotic scrambling and the encryption /decryption of watermarked image.
- (2) The second focus of the thesis of is a comparison of the development and application of several chaotic systems (Chapter 4). Through the comparison, thesis proposed that the Henon map be used for digital watermarking.
- (3) The third focus of this thesis is to analyze the overflow of the Henon attractor and to propose three solutions to the problem (chapter 5): a. selecting the appropriate initial position of the iteration from the R, G, and B colors of the image; b. devising a combined system of several chaotic maps; and c. forming a flexible system by combining a and b.
- (4) The forth focus of this thesis is to compare three watermarking algorithms based on the characteristics of the human visual system (HVS). They include DFT, DCT (Chapter 3) and DWT. This thesis proposed a DWT watermarking method and used experiments to show that the use of HVS can significantly improve the visual quality of the image and the robustness of the watermark (Chapter 6)

By synthesizing the above study focuses and making a large number of experiments all the theoretical bases have be verified and the presented research has shown that the characteristics of hyperchaotic mapping can be used in robust watermarking technology.

## 8.2 Research Contributions

According to all the research work we have done so far, it is clear that the goal is to embed scrambled watermark in the picture content. As a result of our method, hyperchaotic scrambling can be flexibly applied in color image and the scrambled watermark is again embedded by DWT. In order to increase the key space, high security, fast scrambling speed and strong robustness for the image scrambling, the algorithm is proposed not only at the region where the watermark is embedded, with a certain neighborhood. The major contributions are shown as below:

- (1) The main contributions of this thesis development of frame-work (chapter 7) in integrate a hyperchaotic scrambling and embedding/extracting of watermark image. The digital image encryption has good random characteristics, also improve the performance of confidentiality and practical significance.
- (2) The second contribution of this thesis is to analyze different kinds of chaotic maps (Lorenz, Chen's, Logistic, Arnold, Henon), and generated a lot of hyperchaos it can also be said that the compound chaos. A chaotic system has high sensitivity to its initial condition, parameter value, ergodictiy, random behavior and unstable periodic orbit with long periods. According to these properties required in conventional cryptography algorithms are achieved through iterative processing. So the hyperchaotic system is associated with synchronization of two or more chaotic systems and controls. The scrambling watermark can be produced by a random number generator. It is easy to produce a complex and unpredictable cipher-text.
- (3) The third contribution of thesis is to find the Henon map is used for image encryption, the sequence generated by the generalized the Henon system does not satisfy uniform distribution. Therefore, it must be modified in order to achieve better-random statistical properties. The results of several experiments show that the Henon map can provided larger cipher space for large-scale image, with

higher sensitivity to the encryption key and better statistical properties.

- (4) The fourth contribution is the development and application of hyperchaos for large-scale image; it divides large-scale image into areas to build large Cartesian coordinate space, which accommodates a variety of different chaotic sub-systems while operating in areas adjacent to the image encryption algorithm that give full play to the encryption algorithm flexibility. The randomness not only increases the robustness and decryption difficulty against attacks, but increases the complex relation among sub-images; it has potential development and application value in the fields of large-scale image encryption (e.g. landscape, marine geology, space images, and etc.).

Experiments were designed to validate the methods (Chapters 4, 5, 6, 7) and these projects are implemented in Matlab. They can play an important reference role in future research work of hyperchaotic encryption.

- (5) According to the characters of human vision and the suitability of DWT based on watermarking scheme has presented, it was good behavior of the watermark detector. In this algorithm, DWT is similar to the theoretical model of HVS. The goal is that the quality of the host image must not be affected. Therefore, the fifth contribution delineates the desirable properties of a watermark and creates a program file in the MATLAB. An oblivious watermark is generated using spread spectrum techniques. The watermark image is added to the low-frequency of the host image by DWT. Therefore, the watermarking technique has high energy and good robustness with four important parameters: the wavelet coefficient, the weight of the  $k^{th}$  component, the new wavelet coefficient and the weight of embedding depth.

### 8.3 Discussion and Future Work

During the thesis work of its content on the research achieved some results, but time and technology constraints there are some problems not research and mainly:

- (1) The future research study will focus on the realization of encryptions for both video and audio files. Shifting to further strengthen the security of chaotic encryption.
- (2) The combination of chaotic encryption and image data compression technology. We can consider the image of information is encrypted and appropriate to introduce a certain loss of data. How to use chaotic system properties of the design more secure, more efficient chaotic image data. It is a promising research direction.
- (3) The study of hardware realization of chaotic encryption technology has provided an effective tool for both chaotic encryption and chaotic confidential communications. Some researchers have proposed to implement chaotic confidential communications and chaotic encryption by using very-large-scale integration (VLSI) technology; such a technology is believed to have tremendous potential for future research work.

---

## References

- Abu-Errub, A., et al. (2008), Optimized DWT-based image watermarking, *Proceeding of First International Conference on Applications of Digital Information and Web Technologies, IEEE*, pp.4-6.
- Addison, P.S. (2002), *The Illustrated Wavelet Transform Handbook*, IOP Publishing Ltd., ISBN:0-7503-0692-0.
- Alligood, K.T., et al. (1996) *Chaos: An Introduction to Dynamical Systems*, *Proceeding of Berlin: Springer-Verlag*.
- Andrew, B., Watson, J., and Solomon A. (1997), A Model of Visual Contrast Gain Control and Pattern Masking, *Journal of the Optical Society of America* A.87.
- Arnold, L. (1998), *Random Dynamical Systems*, *Proceeding of Berlin: Springer-Verlag*.
- Argyris, J., et al. (1994), *An Exploration of Chaos*, *Proceedings of North-Holland*, NY.
- Arivazhagan, S. and Ganesan, L. (2003), Texture Segmentation Using Wavelet Transform, *Proceeding of Patter Recognition Letters*, pp.3197–3203.
- Azzaz, M.S., et al. (2010), Real-time image encryption based chaotic synchronized embedded cryptosystems, *Proceedings of the 8th IEEE International NEWCAS Conference (NEWCAS '10)*, pp. 61–64.

- 
- Batterman, R. W. (1993), Defining Chaos, *Philosophy of Science*, pp. 43-66.
- Banks, J. (1992), On Devaney's Definition of Chaotic for Discontinuous Dynamical Systems, *Recent Advances in applied Mathematics and Computational and Information Sciences*.
- Banks, J., et al. (1992), On Devaney's Definition of Chaos, *Proceeding of Mathematical Association of America*, pp.332-334.
- Baptista, MS. (1998), Cryptography with chaos, *Phys Lett A*, 240(1-2):50-4.
- Bas, P., et al. (1999), A Geometrical and Frequently Watermarking Scheme using Similarities, *Proceeding of SPIE Conf. on Security and Water-marking of Multimedia Contents*, Vol. 3657, pp. 264-272.
- Bender, W., et al. (1996), Techniques for Data Hiding, *IBM System Journal*, pp. 313-335.
- Behnia, S., et al. (2009), Cryptography based on chaotic random maps with position dependent weighting probabilities [J]. *Chaos, Solitons and Fractals*, pp. 362-369.
- Berghel, H. (1998), Digital watermarking makes it marks, *Proceeding of netWorker: The craft of network computing*, 2(4), pp.30-39.
- Bianco, M.C. (1991), Encryption system based on chaos theory, *Proceeding of US Patent*, No.5048086.
- Bourbakis, N., et al. (1992), Picture Data Encryption Using SCAN Patterns, *Proceeding of Pattern Recognition*, pp.567-581.

- 
- Brock, W., et al. (1995), A Test for Independence based on the Correlation Dimension, Working Papers 9520, *Proceeding of Wisconsin Madison Social Systems*.
- Brooks, J., et al. (1992), On Devaney's Definition of Chaotic, *Proceeding of the American Mathematical of Chaotic*, pp. 332-334.
- Shannon, C.E. (1948), A mathematical theory of communication, *Proceeding of The Bell System Technical Journal*, 27 (3) 379-423. 623-656.
- Calderon, A. P. (1964), Intermediate Spaces and Interpolation, the Complex Method, *Proceeding of Studio Mathematical*, 24, pp.113-90.
- Charles, K., et al. (1992), An Introduction to Wavelets, *Proceeding of San Diego: Academic Press*
- Chapeau-Blondeau, F., et al. (2006), Noise-aided SNR amplification by parallel arrays of sensors with saturation, *Proceeding of Physical Lett. A 351*, pp.231-237.
- Chapeau-Blondeau, F., et al. (2008), Signal-to-noise ratio of a dynamical saturating system: Switching from stochastic resonator to signal processor, *Proceeding of Physical 387*, pp.2394-2402.
- Chang, C.C., et al. (2001), A new encryption algorithm for image cryptosystems. *Proceeding of J System Software*.
- Chen, G., et al. (1998), From Chaotic to order: methodologies, perspectives and applications. *Singapore: World Scientific*.
- Chen, G. (1999), Controlling Chaotic and Bifurcations in Engineering Systems, *CRC Press, Boca Raton, FL*.



- 
- Chen, G. (1999) Yet another chaotic attractor, *Proceeding of Int. J. Bifurc. Chaos*, Vol. 9, No. 7, pp. 1465-1466.
- Chen, G., and Ueta, T. (2000), Bifurcation analysis of Chen's equation, *Proceeding of Int J of Bifurcation and Chaos*.
- Chen, G., et al. (2003), Dynamics of the Lorenz System Family, *Proceeding of Analysis Control and Synchronization*.
- Chen, G., et al. (2004), A symmetric image encryption scheme based on 3D chaotic cat maps, *Proceeding of Chaos, Solitons & Fractals*, Vol. 21, Issue 3, pp. 749-761.
- Chen, G., et al. (2004), Chaos based image encryption, *Proceeding of in Handbook of Computational Geometry for Pattern Recognition, Computer Vision, Neural Computing and Robotics*, E. B. Corrochano, Ed., Springer, Heidelberg, Germany
- Chen, M. (2006), Image steganography based on Arnold transform, *Proceeding of Appl. Res. Comput.*, pp. 235-237.
- Chia, P., et al. (2003), Understanding Organizational Security Culture. Information Systems: The Challenges of Theory and Practice. *Las Vegas, USA, Information Institute*.
- Christopoulos, C., et al. (2000), The jpeg2000 still image coding system: an overview, Vol. 46, pp.1103-1127.
- Cox, I.J., et al. (1997), Secure Spread Spectrum Watermarking for Multimedia, *Proceeding of IEEE Trans on Image processing*, pp.1673-1687.

- 
- Craver, S., et al. (1997), On the Inevitability of Invisible Watermarking Techniques, *Proc. of IEEE Int. Conf. On Image Processing*, Vol. 1, pp. 540-543.
- Daubechies, I. (1992), Ten Lectures on Wavelets, *Proceeding of CBMS-NSF Regional Conference Series in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM)*, Vol.61.
- Daubechies, I., and Sweldens, W. (1998), Factoring Wavelet Transforms into Lifting Steps, *Proceeding of J. Fourier Anal. Appl*, pp.245–267.
- Davis, G., and Nosralinia, A. (1996), Wavelet-Based Image Coding: An Overview, *Proceeding of IEEE Trans. on Image Proc.*
- Dittmann, J., Stabenau, M., and Steinmetz, R., (1998), Robust MPEG video watermarking technologies, *Proceeding of the 6th ACM international conference on multimedia*.
- Duan, F., et al. (2005), Evaluation of bistable systems versus matched filters in detecting bipolar pulse signals, *Proceeding of Fluctuation and Noise Lett.5*, pp. L127-L142.
- Emanuele, Application Note 236-An introduction to the Sampling Theorem, *National Semiconductor Corporation*, available at <URL: <http://www.noise.physx.u-szeged.hu/DigitalMeasurements/Sampling/SamplingTheorem.pdf>>, 1989.
- Estrada, R., et al. (2009), The multi-dimensional coordinate system.
- Fridrich, J. (1997), Image encryption based on chaotic maps, *Proceeding of IEEE International Cybernetics and Simulation*, pp.1105-1110.

- 
- Fridrich, J., (1998), Symmetric ciphers based on two-dimensional chaotic maps, *Processing of International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, Vol. 8, No. 6, pp. 1259-1284.
- Fujii, H., et al. (1995), Digital Image Scrambling Method for Copyright Protection, *IPSJ SIG Notes GroupWare Abstract*, pp. 021-009.
- Forre, R. (1991), The Henon Attractor as Key Stream Generator, *Abstracts of Eurocrypt*, pp. 76-80.
- Fridrich, J. (1997), Image Encryption Based on Chaotic Maps, *IEEE*, pp.1105-1110.
- Ghanbari, M. (2008). Scope of validity of PSNR in image/video quality assessment, *Proceeding of Electronics Letters* 44 (13), pp. 800-801.
- Gavlasov, A., Proch, A., and Mudrov M. (2005), Wavelet Use for Image Classification, *Proceeding of 15<sup>th</sup> International Conference on Process Control*.
- Gershenfeld, N. (1999), The Nature of Mathematical Modeling, *Cambridge University Press*.
- Germano, T., Wavelets in multi-resolution analysis, *Presented for CS563*, available at <URL: <http://davis.wpi.edu/~matt/courses/wavelets/#Haar Example>>, 1999.
- Glenn Elerbt, Measuring Chaotic, available at <URL: <http://hypertextbook.com/chaotic/43.shtml>>, 2007.
- Ghouti, L., Bouridane, A., and Ibrahim, MK. (2006), Digital image watermarking using balanced multi-wavelets, *Proceeding of IEEE Transactions on Signal*, 192

---

54(4), pp. 1519-1536.

Gonchenko, S.V., et al. (2005), Three-dimensional Henon-like maps and wild Lorenz-like attractors, *Proceeding of Internet J. Bifur. Chaos Appl. Sci. Engrg.*, pp.3493-3508.

Graham, J. (2002), Preserving the Aftermarket in Copyrighted Works: Adapting the First Sale Doctrine to the Emerging Technological Landscape, *Proceeding of Stanford Technology Law Review*.

Gradshteyn, I.S. et al. (2000), Jacobian Determinant, *Proceeding of Tables of Integrals, Series, and Products, 6th ed. San Diego*, pp.1068-1069.

Grossmann, A., and Morlet, J. (1984), Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape, *Proceeding of SIAM Journal of Mathematical Analysis*, 15, pp.723-736.

Guckenheimer, J., and Holmes, P. (1990), Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields, (*New York: Springer*) *First citation in article*

Gulati, K. (2003), Information Hiding Using Fractal Encoding, *Proceeding of Indian Institute of Technology Bombay*.

Gunjal, B.L., et al. (2010), Discrete Wavelet Transform Based Strongly Robust Watermarking Scheme for Information Hiding in Digital Images, *Proceeding of Third International Conference- Emerging Trends in Engineering and Technology*, pp.12-21.

- 
- Gvan, R., Schyndel, et al. (1994), A digital watermark, *In Int. Conf. On Image Processing of IEEE*, pp.86-90.
- Haar, A. (1910), Zur Theorie der Orthogonalen Funktionensysteme, *Proceeding of Mathematics Annalen*, 69, pp.3371-71.
- Henon, M. (1976), A Two-dimensional mapping with a Strange Attractor, *Proceeding of Communication in Mathematical Physics*.
- Hossam, H., Kalash, H.M., and Farag Allah O.S. (2007), An efficient chaos-based feedback stream cipher (ECBFSC) for image encryption and decryption, *Proceeding of Informatica*, Vol. 31, No. 1, pp. 121-129.
- Hernandez, J.R., et al. (1998), Statistical Analysis of Watermarking Schemes for Copyright Protection of Images. *Proceeding of Image and Signal Processing*, pp.1142-1166.
- Hsu, C.T., and Wu, J.L. (1999), Hidden digital watermarks in images, *Proceeding of IEEE Trans. on Image Processing*, Vol. 8, Issue 1, pp.58-68.
- Hua, X.S., and Shi, Q.Y. (2001), Local Watermarking Scheme, *Proceeding of Journal of Image and Graphics*, 6(7), pp.642-647.
- Huang, F., and Feng, Y. (2007), Novel 2D chaotic map based on image segmentation and image encryption approach, *Proceeding of Opt. Précis. Eng.*, pp. 1096-1103.
- Huang, L., et al, (2009), The best image scrambling degree of binary image based on Arnold transform, *Proceeding of J. Comput. Appl.*, pp. 474-476.

- 
- Jiang, J., and Armstrong, A. (2002), Data hiding approach for efficient image indexing, *Proceeding of Electronics letters 7th*, 38(23), pp. 1424-1425.
- Joo, S., et al. (2002), A new robust watermark embedding into wavelet DC components, *Proceeding of ETRI Journal*, pp. 401-404.
- John, J. (2008), Histogram dialog, *GIMP User's manual*, pp.245.
- Kachris, C. (2003), Chapter 3 The SCAN Algorithm, *Design and FPGA Implementation of the SCAN Encryption Algorithm*.
- Kachris, C., et al. (2003), A Reconfigurable Logic-based Processor for the SCAN Image and Video Encryption Algorithm, *International Journal of Parallel Programming*, Vol.31, No.6, pp.489-504.
- Kahng, B. (2009), On Devaney's Definition of Chaos for Discontinuous Dynamical Systems, *Proceeding of Recent Advances in Applied Mathematics and Computational and Information Sciences*, Volume I.
- Katherine, S. (2009), A Chaotic Image Encryption, *Proceeding of Mathematics Senior Seminar*, 4901.
- Kocarev, L., et al. (1998), From Chaotic Maps to Encryption Schemes, *Proceeding of IEEE International Symposium on Circuits and Systems*.
- Kraft, R.L. (1999), Chaos, Cantor Sets, and Hyperbolicity for the Logistic Maps, *Proceeding of Amer. Math. Monthly* 106, pp. 400-408.
- Kutter, M., et al. (1999), Towards Second Generation Watermarking Schemes, *International Conference on Image Processing*, pp.320-323.

- 
- Lorenz, E.N. (1963), Deterministic nonperiodic flow, *Proceeding of J. Atmos. Sci.* 20, pp. 130-141.
- Li, T.Y., and Yorke, J. A. (1975), Period three implies chaos, *Proceeding of American Mathematical Monthly*, 82, pp. 985-992.
- Li S., et al. (2003), On the security of a chaotic encryption scheme: problems with computerized chaotic in finite computing precision, *Proceeding of Computer Physics Communications*.
- Li, X., et al. (1997), Image compression and encryption using tree structures, *Pattern Recognition Letters*, Vol. 18, No. 11-13, pp. 1253–1259.
- Li, Y., et al. (2008), Image Encryption Algorithm Based on Self-adaptive Symmetrical-coupled Toggle Cellular Automata, *Proceeding of Image and Signal Processing*, Vol.3, pp.32-36.
- Li, Y.X., et al. (2005), Controlling a Unified Chaotic System to Hyperchaotic, *Proceeding of IEEE Transactions on circuits and systems*, Vol.52, No.4.
- Lin, Q.H., et al. (2004), Secure Image Communication Using Blind Source Separation, *Proceeding of IEEE 6th CAS Symp.on Emerging Technologies: Mobile and Wireless COMM*, pp.261-264.
- Lu, W., and Kanzo, O. (2004), A Digital Watermarking Technique Based on ICA Image Features, pp. 906-913.
- Nikolova, M., et al. (2005), Salt-and-Pepper noise removal by median-type noise detectors and edge-preserving regularization, *IEEE Transactions on Image*

---

*Processing 14*, pp.1479 - 1485.

Maniccam, S.S. (2001), Lossless image compression and encryption using SCAN [J],  
*Proceeding of Pattern Recognition*, pp.1229-1245.

Marc, V., et al. (2004), Encryption of Images for Real-time Application, *In: Fourth IEEE Benelux Signal Processing Hilvarenbeek*, pp.11-15.

Marek, B., et al. (2002), Chaotic and Non-chaotic Mixed Oscillations in a Logistic System with Delay and Heat-integrated Tubular Chemical Reactor,  
*Proceeding of Chaotic, Solitons & Fractals*.

Mark, D., et al.(1999), Information System Incidents: The Development of A Damage Assessment Model.

Mallat, S.G. (1989), A Theory for Multiresolution Signal Decomposition: A Wavelet Representation, *Proceeding of IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.11, No. 7, pp.674-693.

Mallat, S.G. (1989), Multiresolution Approximation and Wavelet Orthonormal Bases of  $L_2(\mathbb{R})$ , *Proceeding of Transactions of the American Mathematical Society*, 315, pp. 69-87.

Mallat, S.G. (1998), A Wavelet Tour of Signal Processing, *San Diego: Academic Press*.

MathWorld-A Wolfram Web Resource, Arnold's Cat map, available at  
< URL: <http://mathworld.wolfram.com/ArnoldsCatMap.html>>, 2007.

Maniccam, SS., et al. (2001), Lossless Image Compression and Encryption Using SCAN, *Proceeding of Pattern Recognition*, pp.1229-1245.



- 
- Mauro, B., et al. (1998), Copyright Protection of Digital Images by Embedded Unperceivable Marks. *Proceeding of Image and Vision Computing*, pp.897-906.
- Masaki, M., et al. (1999), Truncated Baker transformation and its extension to image encryption, *Proceeding of SPIE on Advanced Materials and Optical Systems for Chemical*, Vol. 3858.
- Melnikov, V.K. (1963), On the Stability of the Center for Time Periodic Perturbations, *Processing of Trans. Moscow Math. Soc.*, pp.1-57.
- Meyer, Y. (1992), Wavelets and Operators, *Proceeding of Cambridge Studies in Advanced Mathematics. Cambridge University Press, vol. 37, Translated from the 1990 French original by D. H. Salinger.*
- Murray, H. (1992), Ten Lectures on Wavelet, *Proceeding of SIAM, Philadelphia, Pennsylvania.*
- Pareek, N.K., et al. (2005), Cryptography using multiple one-dimensional chaotic maps. *Communications Nonlinear Science and Numerical Simulation*, pp.715-723.
- Patrizia, C., et al. (2008), Chaotic and Coarse Graining in Statistical Mechanics, 2-*Dynamical Indicators for Chaotic Systems: Lyapunov Exponents, Entropies and Beyond*, pp.24-57.
- Patrick, E., et al. (2003), Asymptotic angular stability in non-linear systems: rotation numbers and winding numbers, *Proceeding of Dynamical Systems*, Vol.18, pp.191-200.

- 
- Puate, J., and Frederic, J. (1996), Using fractal compression scheme to embed a digital signature into an image, *In Proceedings of the SPIE, Video Techniques and Software for Full-Service Networks*, Vol.2915, pp.108-118.
- Ponomarenko VI and Prokhorov MD. (2002), Extracting information masked by the chaotic signal of a time-delay system, *Proceeding of Phys Rev E*.
- Rabbani, M., and Joshi, R. (2002), An overview of the jpeg 2000 still image compression standard, *Elsevier*, pp. 3-48.
- Richard, A. (1986), A Diagnostic Test for Non-linear Serial Dependence in Time Series Fitting Errors, *Proceeding of Journal of Time Series Analysis*, pp.165-178.
- Richter, H. (2002), The Generalized Henon maps: Examples for higher-dimensional chaotic, *Proceeding of International Journal of Bifurcation and chaotic*.
- Rong, J., et al. (2005), Architecture Design of the Re-configurable 2-D Von Neumann Cellular Automata for Image Encryption Application, *Proceeding of IEEE Circuits and Systems*, pp. 3059-3062.
- Rosenstein, M.T., et al. (1993), A Practical Method for Calculating Largest Lyapunov Exponents from Small Data Sets, *Proceeding of Physionet D*, pp.117-134.
- Rossler, O.E. (1976), An equation for continuous chaos, *Physics Letters. A. Vol. 57*, pp. 397-398.
- Rossler, O.E. (1979), An equation for hyperchaos, *Physics Letters A, Vol.71*, pp.155-157.

- 
- Rowlands, T., et al. (2002), A More Resilient Approach to Chaotic Encryption, *Proceeding of the ICITA2002*, pp. 1-4.
- Rukhin, A. (2001), A Statistical Test Suite for Random and pseudorandom Number Generator for Cryptographic Applications, *NIST Special Publication 800-22*.
- Ruanaidh, J.J.K.O., et al. (1996), Watermarking Digital Images for Copyright Protection, *Image and Signal Processing, Proceeding of IEEE proceedings on Vision*.
- Refregier, P., et al. (1995), Optical image encryption based on input plane and Fourier plane random encoding, *Proceeding of Opdet*, pp.767-769.
- Satish, S., Classifying image data, Image Compression, available at <URL:<http://www.debugmode.com/imagecmp/classify.htm>>, 2010.
- Schwartz, C. (1991), A new graphical method for encryption of computer data [J], *Proceeding of Cryptology*, pp.43-46.
- Seripeariu, L., (2005), A New Image Encryption Algorithm Based on Inversible Functions Defined on Galois Fields, *Proceeding of IEEE Signals, Circuits and Systems*, pp.243-246.
- Shannon, C.E. (1948), A Mathematical Theory of Communication, *Proceeding of Bell System Technical Journal*, pp. 379–423 & 623–656.
- Shan, A., et al. (2002), Real-Time Digital Video Watermarking, *Proceeding of the IEEE TUAM 2.1 ICCE*, pp.12-13.
- Sobhy, M., et al. (2001), Methods of Attacking Chaotic Encryption and

- 
- Countermeasures, *Proceeding of Acoustics Speech and Signal*, Vol.2, pp.1001-1004.
- Sociedade Natureza, & Uberlândia (2005), Analysis of precipitation time series using the wavelet transform, *Proceeding of Sociedade & Natureza, Uberlândia*, Special Issue, 736-745.
- Sonls, M. (2001), Once more on Henon map: analysis of bifurcations, *Proceeding of Pergamum Chaos, Sotilons Fractals*, Vol. 7, No. 12, pp. 2215-2234.
- Sprott, J.C., Henon Map Correlation Dimension, available at <URL:<http://sprott.physics.wisc.edu/chaotic/henongp.htm>>, 1997.
- Sprott. J.C., Numerical calculation of largest Lyapunov Exponent, available at <<http://sprott.physics.wisc.edu/chaos/lyapexp.htm>>, 1997.
- Sprott, J.C. (2003), Chaos and Time-Series Analysis, *Proceeding of Oxford University*, pp.116-117.
- Stephen R., et al. (2005), Chaotic Encryption and Decryption: Involving ergraduates in Authentic Research, *IEEE*, pp.244-248.
- Steeb, W.H. (1991), Chaotic behavior in systems, *Proceedings of a Handbook of Terms used in Chaotic and Quantum Chaotic*.
- Stefan, W. (2000), Vision Models and Quality Metrics for Image Processing Applications.
- Strogatz, S.H. (1994), Nonlinear Dynamics and Chaos, *Proceeding of Cambridge, MA, Perseus Books Publishing*.
- Sikorski. R. (1960), Boolean algebras, *Proceeding of Springer-Verlage*.

- Taubman, D. (1999), High performance scalable image compression with ebcot, *Proceeding of International Conference on Image Processing ICIP 99*, Vol. 3, pp. 344-348.
- Takashi, K., Wavelet Decomposition of Image, available at <URL: <http://brain.cc.kogakuin.ac.jp/~kanamaru/WaveletJava/Swing/JavaWaveletImage-e.html>>, 2003.
- Taylor, K. (2008), Diffusion Maps, *Proceeding of Dates of Involvement Faculty Advisor Francois Meyer*.
- Theiler, J., et al. (1992), Testing for Nonlinearity in Time Series: The Method of Surrogate Data, *Proceeding of Physical D*, pp.77-94.
- Turner, L.F. (1989), Digital Data security system, *Proceeding of Patent IPNWO*: 89-08915.
- Yang, X., et al. (2007), Image Encryption Algorithm Based on Universal Modular Transformation, *Proceeding of the National Natural Science Foundation of China*.
- Yamazaki, M., et al. (2001), Optimization of encrypted holograms in optical security systems, *Processding of OptEng*, pp. 132-137
- Yen, J.C., et al. (1998), A new chaotic image encryption algorithm, *Proceeding of National Symposium on Telecommunications*, pp. 358-362.
- Yen, J.C., et al. (2000), A new chaotic mirror-like image encryption algorithm and its VLSI architecture, *Pattern Recognition and Image Analysis*, pp.236-247.
- Yves, M. (1985), Principed' incertitude bases hilbertiennes at algebres d' operateurs,

- 
- Proceeding of Bourbaki semimar*, No. 662, pp.209 -223.
- Van, R.G., Schyndel, et al. (1994), A Digital Watermark, *Proceeding of IEEE International Conf. on Image Processing*, pp.86-90.
- Vaidyanathan, P. P. (1993), Multirate Systems and Filter Banks, *Proceeding of IEEE Signal Processing Magazine*.
- Verhulst P.F., (1845), Recherches mathématiques sur la loi d'accroissement de la population, *Nouvmem. de l'Acad'emie Royale des Sci. et Belles-Lettres de Bruxelles*, pp.1-41.
- Visualization of Dynamical Systems: Recurrence, Planet.pk, available at <URL: <http://www.mpipks-dresden.mpg.de/mipi-doc/kantzgruppe/wiki/projects/Recurrence.html>>, 2009.
- Voloshynovskiy. S.S., et al. (2001), Attacks on Digital watermarks: classification, Estimation-Based attacks and Benchmarks, *Proceeding of Comm, Magazine*, pp.118-126.
- Watson, A.B. (1993), DCT quantization matrices visually optimized for individual images, *Proceedings of SPIE*, pp. 202-216.
- Wang, C.Y. (2009), A computation structure for 2-D DCT watermarking, *Circuits and Systems*, pp. 577-580.
- Wikipedia, Sampling Signal Processing, available at <URL: [http://en.wikipedia.org/wiki/Sampling\\_\(signal\\_processing\)](http://en.wikipedia.org/wiki/Sampling_(signal_processing))>, 2010.
- Wiggins, S. (1990), Introduction to Applied Nonlinear Dynamical Systems and Chaos, *Proceeding of Springer*.

- Wong, KW. (2002), A fast chaotic cryptography scheme with dynamic look-up table. *Phys. Lett A*, 298:238-42.
- Wolfram Research, Inc. Logistic Map, available at <URL:<http://documents.wolfram.com/mathematica/Demos/SoundGallery/LogisticMap.html>>, 2010.
- Woods, J.W., et al. (1986), Subband coding of images, *Proceeding of IEEE Trans. Aoust., Speech, Signal Processing*, Vol. ASSP-34.
- Wu, G. Z., et al. (2005), Robust watermark embedding/detection algorithm for H.264 video, *Proceeding of Journal of Electronic Imaging*, Vol. 14, No. 1, pp.1-9.
- Zhang, Y.S. (2008), Multiresolution analysis for image by generalized 2D wavelets, Master's thesis.
- Zheng, D., et al. (2007), A survey of RST Invariant Image Watermarking Algorithm, *Proceeding of ACM Computing Surveys*, Vol. 39, No.2.
- Zhou, J.C., et al, (2005), Some Novel Image Scrambling Methods Based on Affine Modular Matrix Transformation, *Journal of Information & Computational Science*.
- Zhu L.H., et al. (2006), A Novel Image Scrambling Algorithm for Digital Watermarking Based on Chaotic Sequences, *Proceeding of International Journal of Computer Science and Network Security*, pp. 125-130.

## Appendix A

### Input Entropy Data of the Proposed Algorithm with Different Combinations of Hyperchaotic Maps

Table A-1 2D Henon chaotic sequence

$x_{(n+1)}$	$y_{(n+1)}$
1	0.1
-0.3	0.3
1.174	-0.09
-1.0195864	0.3522
-0.103178997890945	-0.30587592
0.679219812151908	-0.030953699367283
0.323170926125173	0.203765943645573
1.05755071715593	0.096951277837552
-0.46882764926227	0.317265215146779

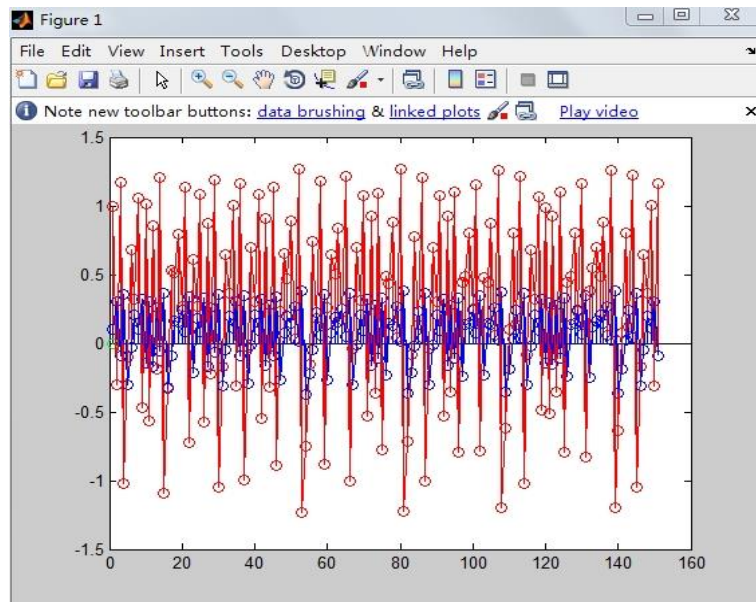


Figure A-1 Different behaviors in 2D Henon



Table A-2 3D Henon chaotic sequence

$x_{(n+1)}$	$y_{(n+1)}$	$z_{(n+1)}$
1	0.100000000000000	0
1.590000000000000	1	0.100000000000000
0.580000000000000	1.590000000000000	1
-1.128100000000000	0.580000000000000	1.590000000000000
0.945600000000000	-1.128100000000000	0.580000000000000
0.211390390000000	0.945600000000000	-1.128100000000000
0.931460640000000	0.211390390000000	0.945600000000000
1.36619410301565	0.931460640000000	0.211390390000000
0.690102998130790	1.36619410301565	0.931460640000000
-0.452778455114731	0.690102998130790	1.36619410301565
0.850519031367765	-0.452778455114731	0.690102998130790
1.25697107095776	0.850519031367765	-0.452778455114731
0.967173068304185	1.25697107095776	0.850519031367765
-0.150080079498249	0.967173068304185	1.25697107095776
0.413182041755516	-0.150080079498249	0.967173068304185
1.38404135607696	0.413182041755516	-0.150080079498249

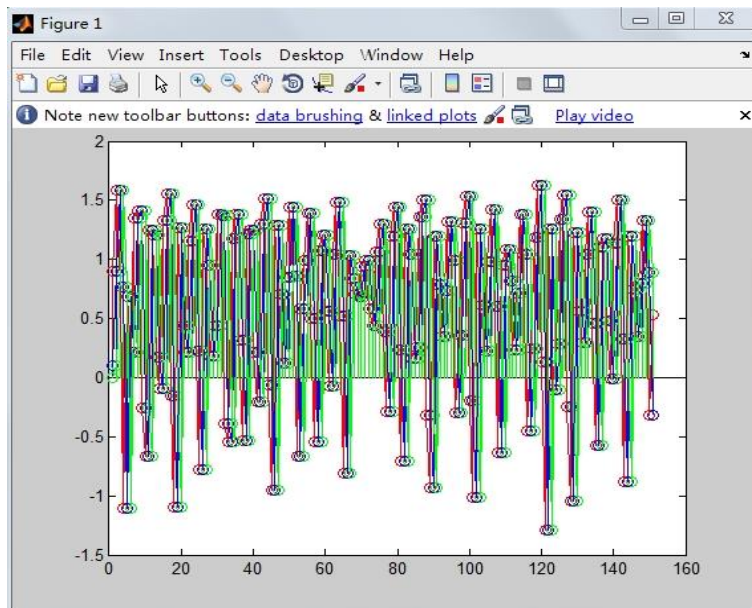


Figure A-2 Different behaviors in 3D Henon

Table A-3 3D Henon and 1D Logistic chaotic sequence

$x_{1(n+1)}$	$y_{1(n+1)}$	$z_{1(n+1)}$
0.9000000000000000	0.1000000000000000	0
-0.6200000000000000	0.9000000000000000	0.1000000000000000
0.2312000000000000	-0.6200000000000000	0.9000000000000000
0.8930931200000000	0.2312000000000000	-0.6200000000000000
-0.595230641982670	0.8930931200000000	0.2312000000000000
0.291400965689797	-0.595230641982670	0.8930931200000000
0.830170954390107	0.291400965689797	-0.595230641982670
-0.378367627025962	0.830170954390107	0.291400965689797
0.713675877637485	-0.378367627025962	0.830170954390107
-0.0186665166432676	0.713675877637485	-0.378367627025962
0.999303122312813	-0.0186665166432676	0.713675877637485
-0.997213460528275	0.999303122312813	-0.0186665166432676
-0.988869371717554	-0.997213460528275	0.999303122312813
-0.955725268642140	-0.988869371717554	-0.997213460528275
-0.826821578242182	-0.955725268642140	-0.988869371717554
-0.367267844493787	-0.826821578242182	-0.955725268642140
0.730228660801775	-0.367267844493787	-0.826821578242182
-0.0664677941127088	0.730228660801775	-0.367267844493787
0.991164064691581	-0.0664677941127088	0.730228660801775

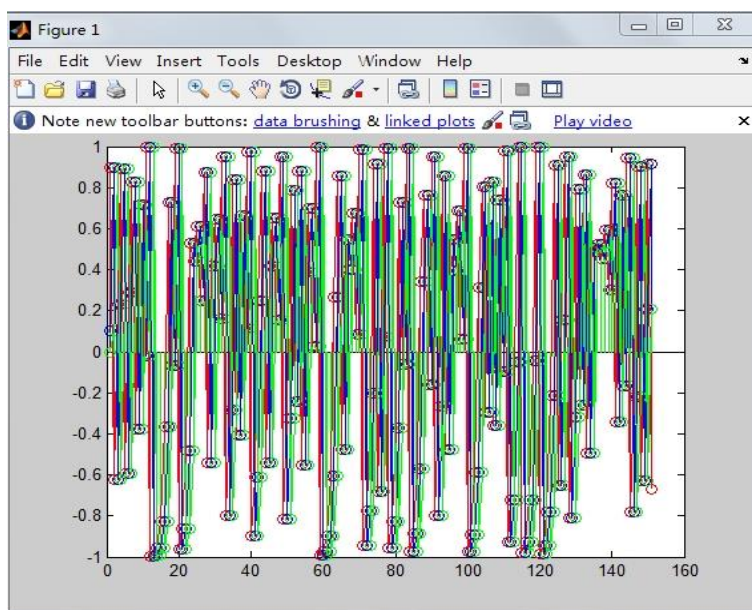


Figure A-3 Different behaviors in 3DHL

Table A-4 3D Henon and 1D Logistic chaotic sequence (mean value=0)

$x_{(n+1)}$	$y_{(n+1)}$	$z_{(n+1)}$
0.820000000000000	0.100000000000000	0
0.553500000000000	0.820000000000000	0.100000000000000
0.926766562500000	0.553500000000000	0.820000000000000
0.254513629244751	0.926766562500000	0.553500000000000
0.711511656650312	0.254513629244751	0.926766562500000
0.769735571628903	0.711511656650312	0.254513629244751
0.664660205242609	0.769735571628903	0.711511656650312
0.835826313035482	0.664660205242609	0.769735571628903
0.514577578023729	0.835826313035482	0.664660205242609
0.936703103321108	0.514577578023729	0.835826313035482
0.222338998311427	0.936703103321108	0.514577578023729
0.648391380529868	0.222338998311427	0.936703103321108
0.854924993191650	0.648391380529868	0.222338998311427
0.465105934529653	0.854924993191650	0.648391380529868
0.932934015731067	0.465105934529653	0.854924993191650
0.234630517586147	0.932934015731067	0.465105934529653
0.673421391762764	0.234630517586147	0.932934015731067
0.824718828296498	0.673421391762764	0.234630517586147

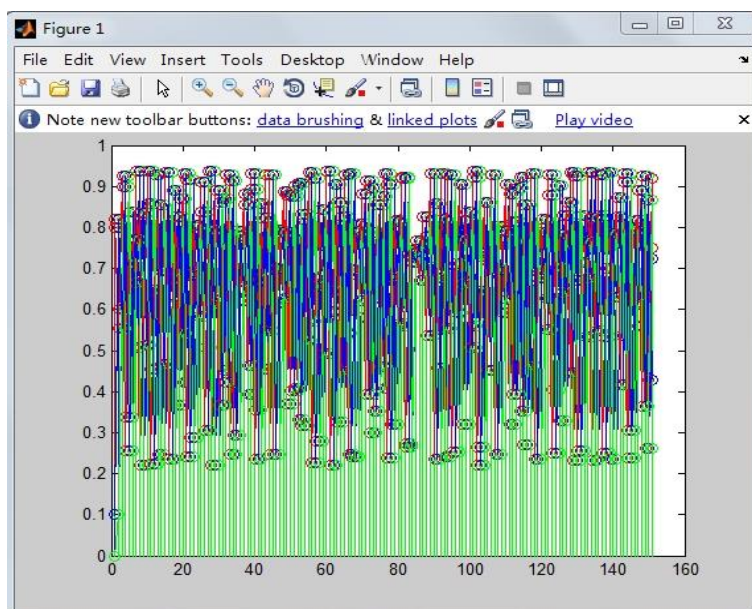


Figure A-4 Different behaviors in 3DHL (mean value=0)

Table A-5 3D Henon and 2D Logistic chaotic sequence

$x_{(n+1)}$	$y_{(n+1)}$	$z_{(n+1)}$
0.820000000000000	0.100000000000000	0
0.553500000000000	0.980000000000000	0.100000000000000
0.926766562500000	-0.920800000000000	0.980000000000000
0.254513629244751	-0.695745280000000	-0.920800000000000
0.711511656650312	0.0318770107154447	-0.695745280000000
0.769735571628903	0.997967712375695	0.0318770107154447
0.664660205242609	-0.991879109888755	0.997967712375695
0.835826313035482	-0.967648337267419	-0.991879109888755
0.514577578023729	-0.872686609232799	-0.967648337267419
0.936703103321108	-0.523163835868481	-0.872686609232799
0.222338998311427	0.452599201678753	-0.523163835868481
0.648391380529868	0.590307925279510	0.452599201678753
0.854924993191650	0.303073106704400	0.590307925279510
0.465105934529653	0.816293383985086	0.303073106704400
0.932934015731067	-0.332669777475648	0.816293383985086
0.234630517586147	0.778661638308606	-0.332669777475648
0.673421391762764	-0.212627893946886	0.778661638308606
0.824718828296498	0.909578757431424	-0.212627893946886

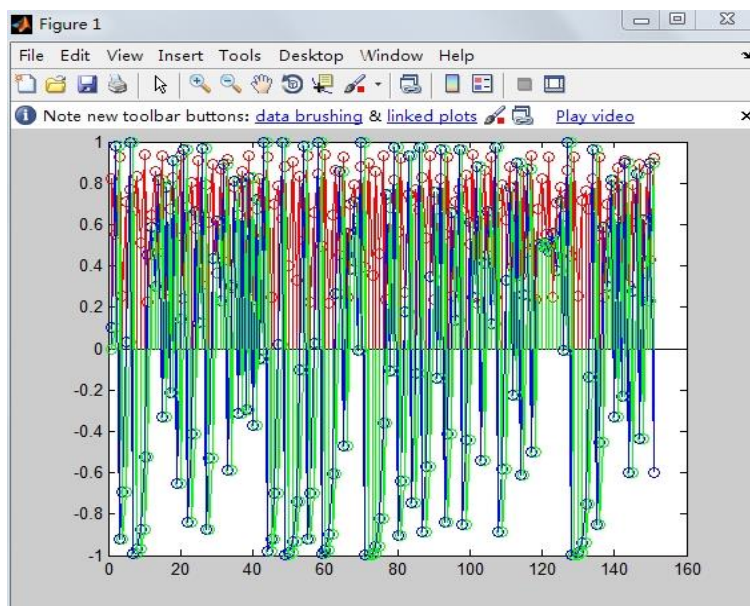


Figure A-5 Different behaviors in 3DH2DL

Table A-6 3D Henon and 2D Arnold chaotic sequence

$x_{(n+1)}$	$y_{(n+1)}$	$z_{(n+1)}$
1	0.100000000000000	0
0.100000000000000	0.200000000000000	0.100000000000000
0.300000000000000	0.500000000000000	0.200000000000000
0.800000000000000	0.300000000000000	0.500000000000000
0.100000000000000	0.400000000000000	0.300000000000000
0.500000000000000	0.900000000000000	0.400000000000000
0.400000000000001	0.300000000000001	0.900000000000000
0.700000000000001	1.99840144432528e-15	0.300000000000001
0.700000000000003	0.700000000000005	1.99840144432528e-5
0.400000000000009	0.100000000000014	0.700000000000005
0.500000000000022	0.600000000000036	0.100000000000014
0.100000000000059	0.700000000000095	0.600000000000036
0.800000000000154	0.500000000000249	0.700000000000095
0.300000000000402	0.800000000000651	0.500000000000249
0.100000000001053	0.900000000001705	0.800000000000651
2.75801603777381e-12	0.900000000004463	0.900000000001705
0.900000000007221	0.800000000011683	0.900000000004463
0.700000000018904	0.500000000030587	0.800000000011683
0.200000000049491	0.700000000080078	0.500000000030587

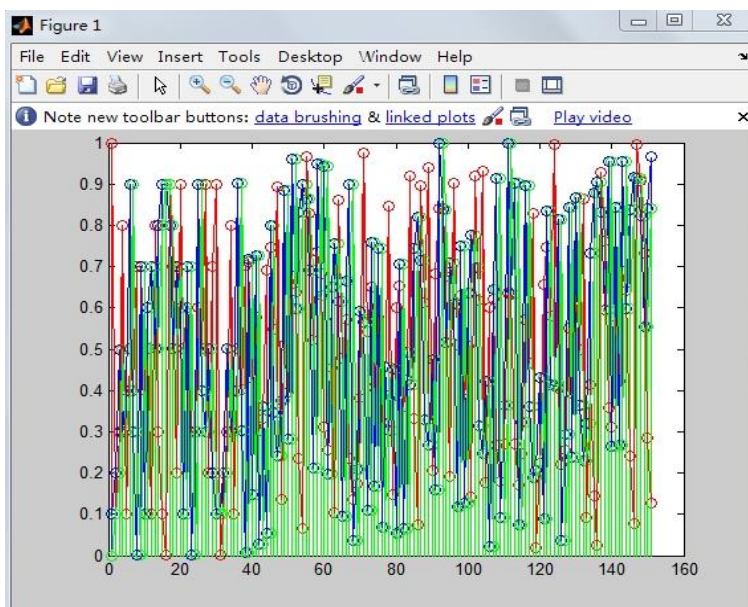


Figure A-6 Different behaviors in 3DH2DA

Table A-7 3D Henon and 2D Henon chaotic sequence

$x_{(n+1)}$	$y_{(n+1)}$	$z_{(n+1)}$
1	0.100000000000000	0
-0.300000000000000	0.300000000000000	0.100000000000000
1.174000000000000	-0.090000000000000	0.300000000000000
-1.019586400000000	0.352200000000000	-0.090000000000000
-0.103178997890945	-0.305875920000000	0.3522 00000000000
0.679219812151909	-0.030953699367283	-0.30587592 000000
0.323170926125173	0.203765943645573	-0.030953699367283
1.05755071715593	0.096951277837552	0.203765943645573
-0.468827649262277	0.317265215146779	0.096951277837552
1.00954610454887	-0.140648294778683	0.317265215146779
-0.567504966872398	0.302863831364661	-0.140648294778683
0.851977188969883	-0.170251490061719	0.302863831364661
-186462672796753	0.255593156690965	-0.170251490061719
1.206917497005852	-0.055938801839026	0.255593156690965
-1.09524858424945	0.362075249101756	-0.055938801839026
-0.317321996718828	-0.328574575274834	0.362075249101756
0.530454875282892	-0.095196599015648	-0.328574575274834
0.510868076388407	0.159136462584868	-0.095196599015648
0.271192857042237	0.153260422916522	0.159136462584868

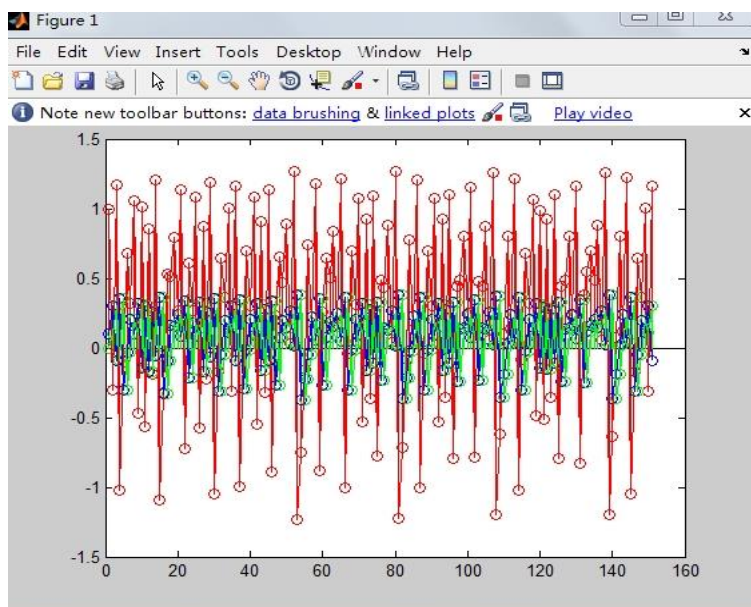


Figure A-7 Different behaviors in 3DH2DH

Table A-8 2D Henon and 3D Henon chaotic sequence

$x_{(n+1)}$	$y_{(n+1)}$	$z_{(n+1)}$
1	0.100000000000000	0
1.590000000000000	1	0.100000000000000
0.580000000000000	1.590000000000000	1
-1.128100000000000	0.580000000000000	1.590000000000000
0.945600000000000	-1.128100000000000	0.580000000000000
0.211390390000000	0.945600000000000	-1.128100000000000
0.931460640000000	0.211390390000000	0.945600000000000
1.36619410301565	0.931460640000000	0.211390390000000
0.690102998130790	1.36619410301565	0.931460640000000
-0.452778455114731	0.690102998130790	1.36619410301565
0.850519031367765	-0.452778455114731	0.690102998130790
1.25697107095776	0.850519031367765	-0.452778455114731
0.967173068304185	1.25697107095776	0.850519031367765
-0.150080079498249	0.967173068304185	1.25697107095776
0.413182041755516	-0.150080079498249	0.967173068304185
1.38404135607696	0.413182041755516	-0.150080079498249
1.45929661627039	1.38404135607696	0.413182041755516
-0.398206883682460	1.45929661627039	1.38404135607696
-0.806354885473610	-0.398206883682460	1.45929661627039

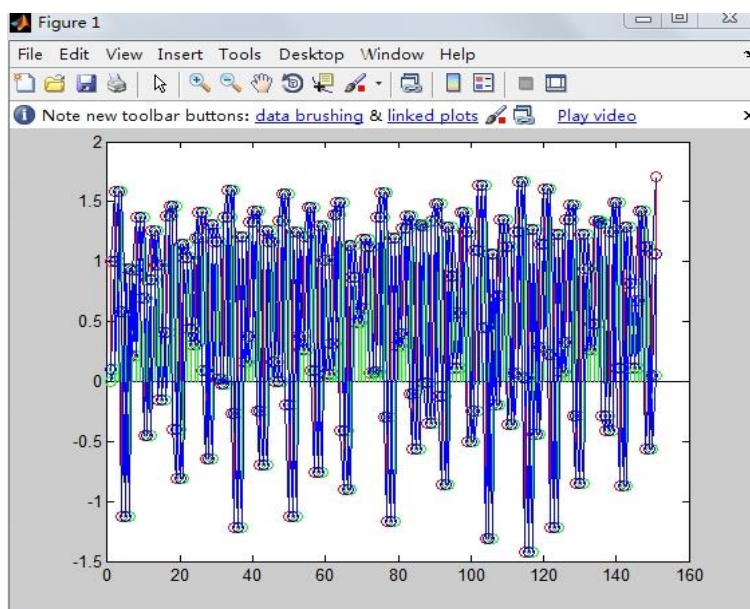


Figure A-8 Different behaviors in 2DH3DH

## Appendix B

### Large-scale Image Scrambled by Different Hyperchaos

The large-scale image was divided two subimages which are scrambled by different hyperchaos.

The library as original image was divided into two blocks (up and down) and used two kind of hyperchaotic systems to scramble. One of block is 3D Henon and 2D Henon hyperchaotic system, second is 2D Arnold chaotic system. The result shows below figure.

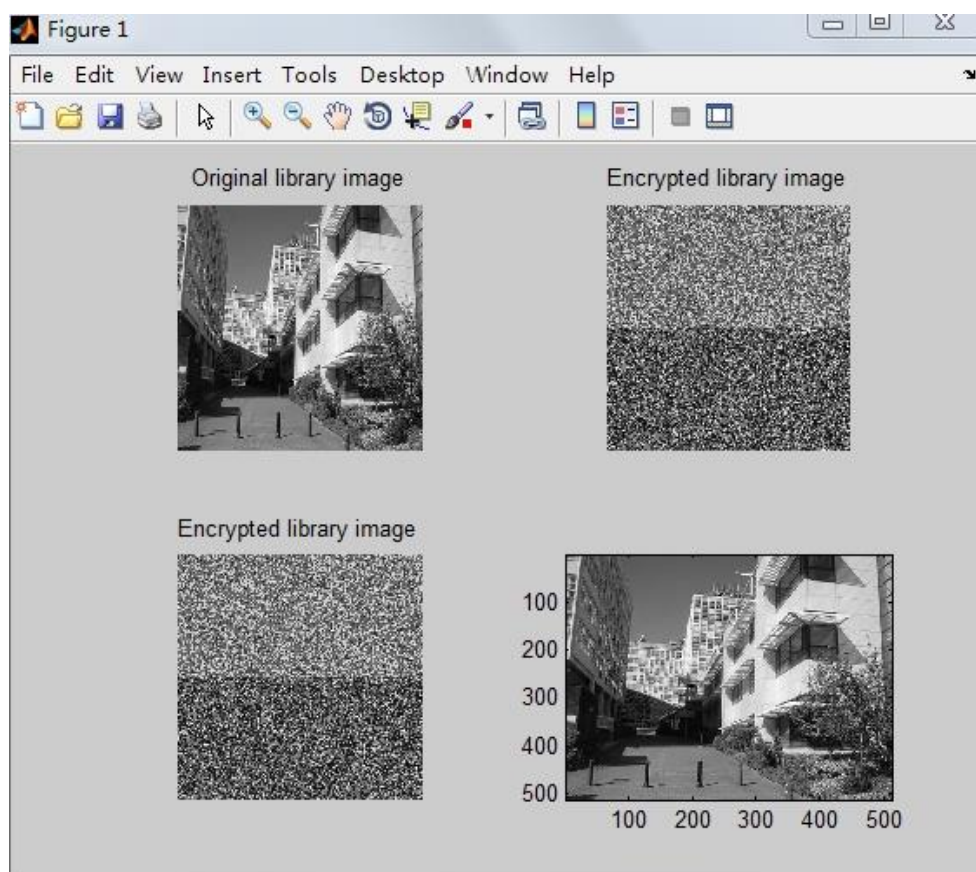


Figure B-1 The two blocks of library image encryption and decryption



The library image as original image was divided into two blocks (left and right) and used different hyperchaotic systems to scramble. One of block used 3D Henon and 2D Henon hyperchaotic system, second used 2D Arnold chaotic system. The result shows below figure.

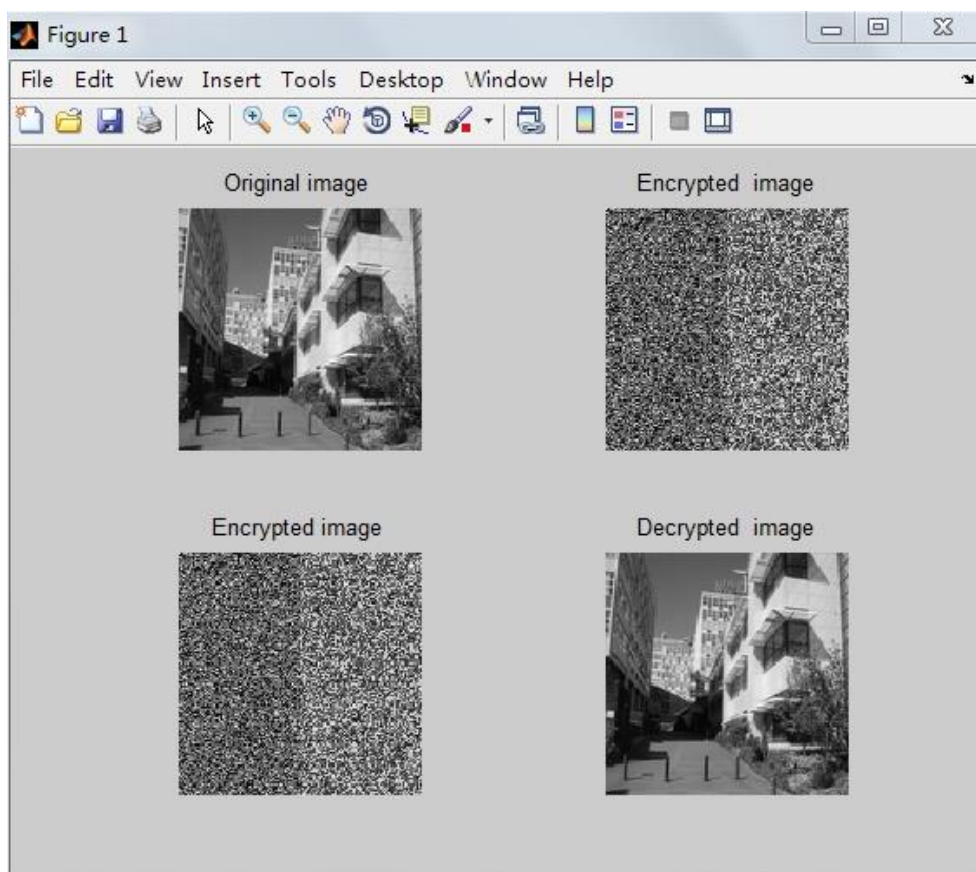


Figure B-2 Two kinds of chaotic system used in the library image

The castle image was divided into three blocks and used three kinds of hyperchaotic to encryption. One of block used 3D Henon and 2D Henon hyperchaotic system, second used 2D Arnold chaotic system. Third used 2D Arnold chaotic system

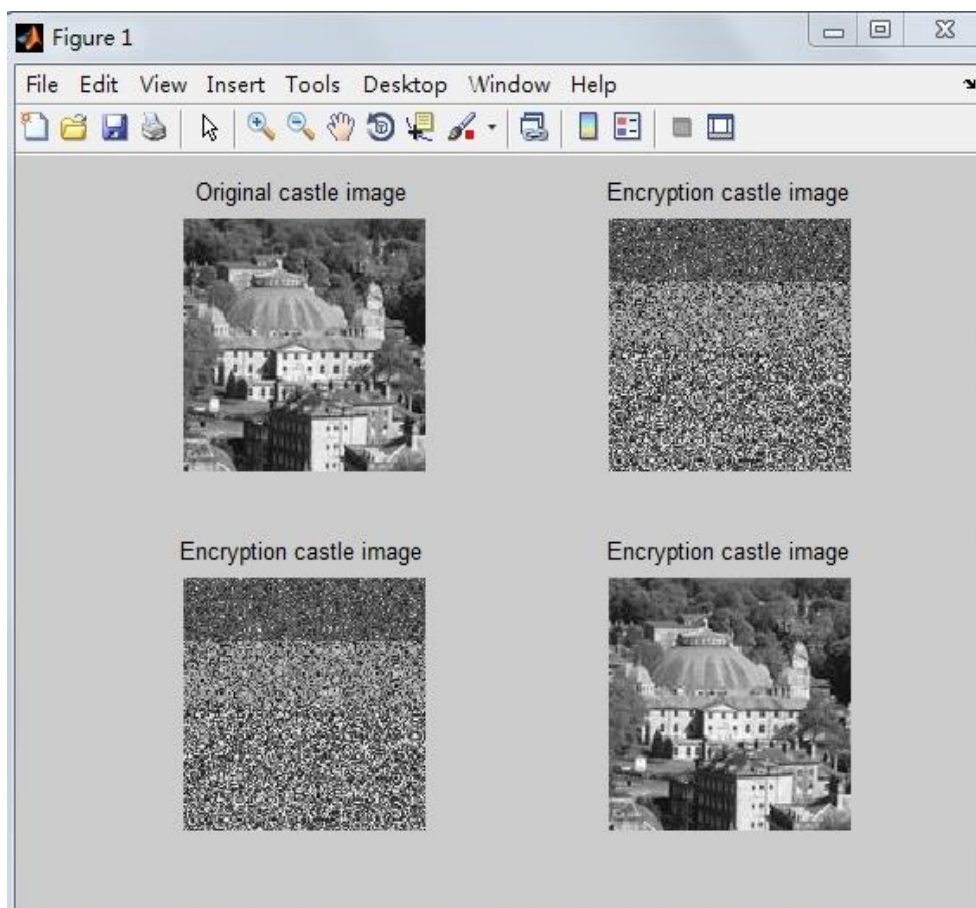


Figure B-3 Three kinds of chaotic system used in the castle image

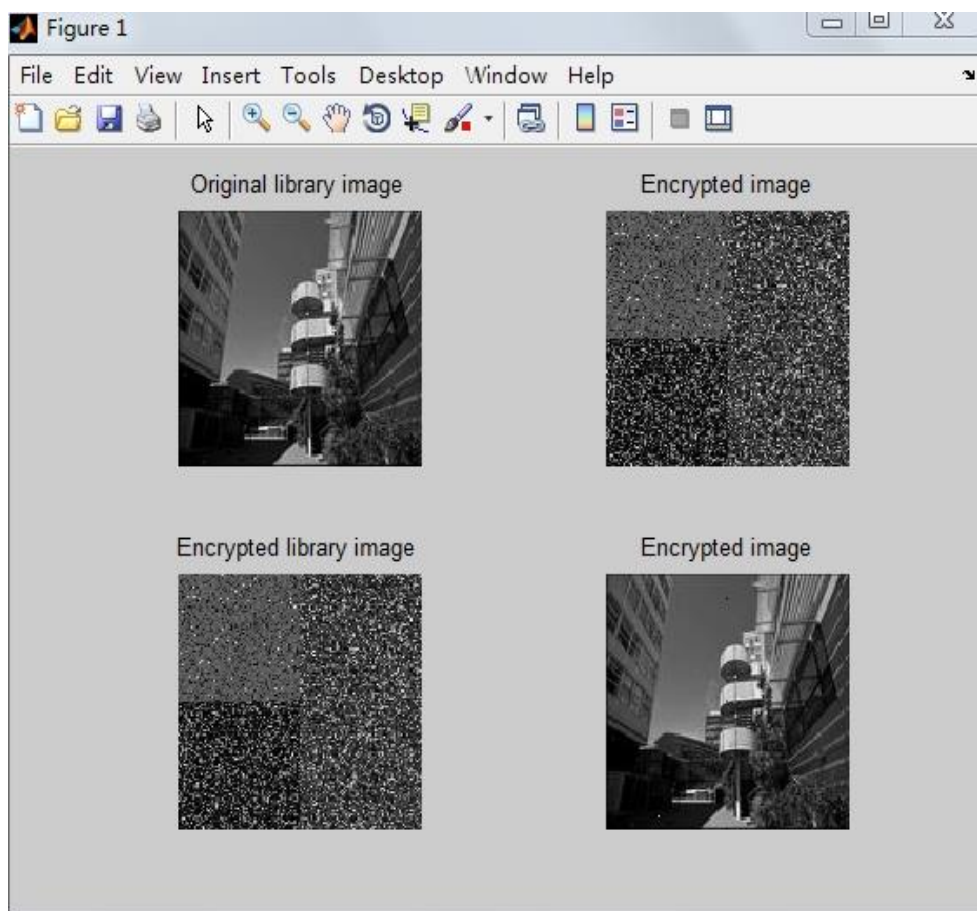


Figure B-4 The three of kinds chaotic system used in the library image

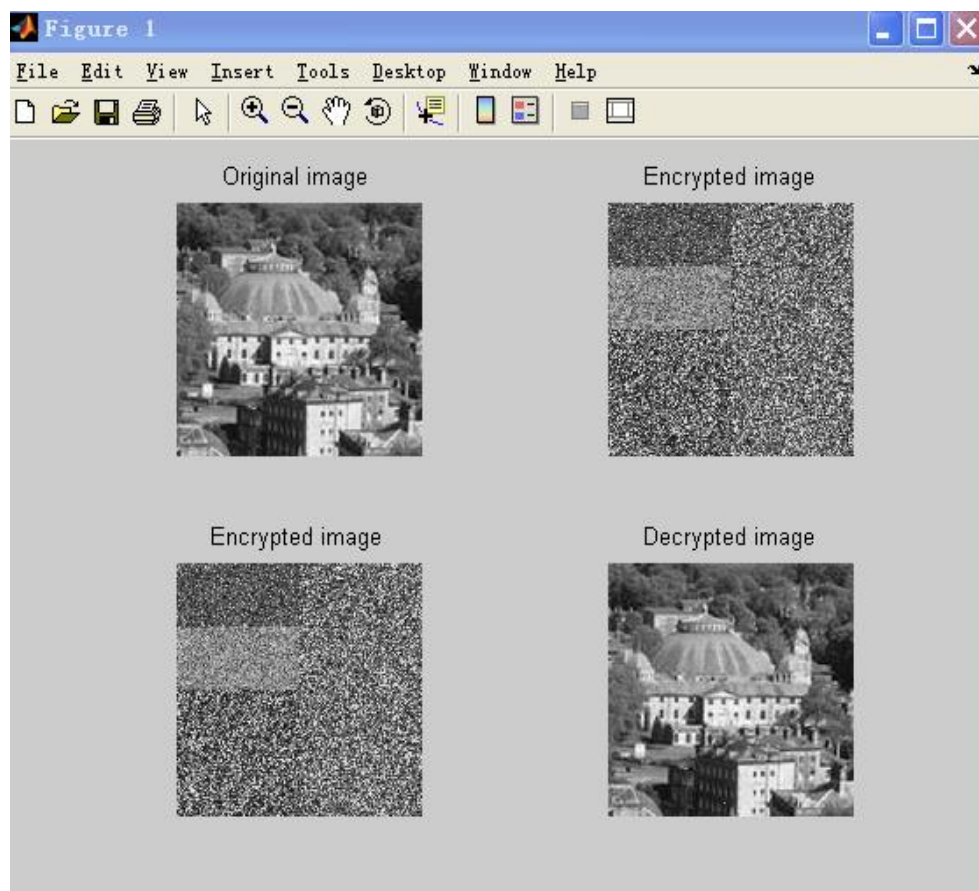


Figure B-5 The four kinds of chaotic system used in castle image

## Appendix C

### Combining DWT with Hyperchaotic Encryption Method

The following experiment data shows that we can get a better secured picture by combining DWT with hyperchaotic encryption method.

DWT method:

Figure C-1 (b) watermark image (*library.jpg* with  $128 \times 128$ ) is embedded into figure C-1 (a) host image (*pool.jpg* with  $512 \times 512$ ) generates figure C-1 (c) watermarked image.

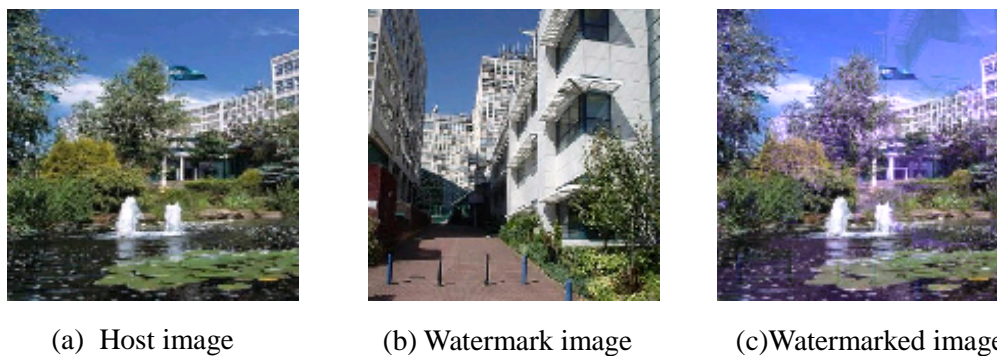


Figure C-1 Combining DWT with hyperchaotic encryption method

Hyperchaotic encryption method:

Figure C-2 (c) watermarked image can be encrypted to figure C-2 (d) scrambled watermarked image by applying hyperchaotic encryption method. The chaotic decryption method can recover figure C-2 (e) back to figure C-2 (c).

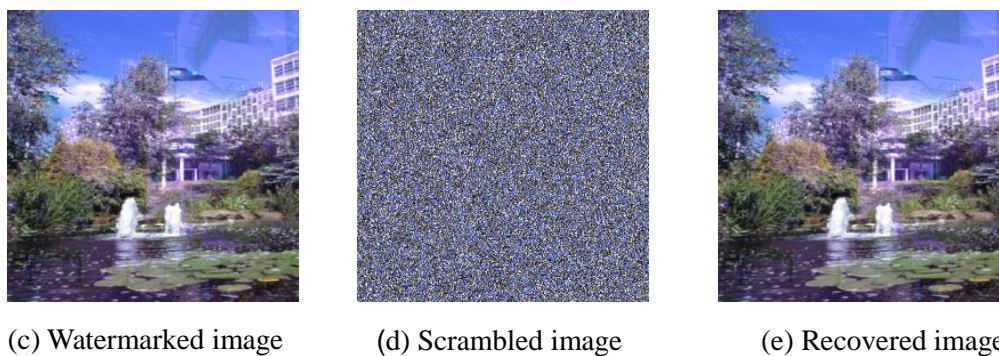


Figure C-2 Hyperchaotic encryption method

Figure C-3(c) watermark image is extracted by inverse DWT from the watermarked image. Hidden information in figure C-1 (b) is recovered.

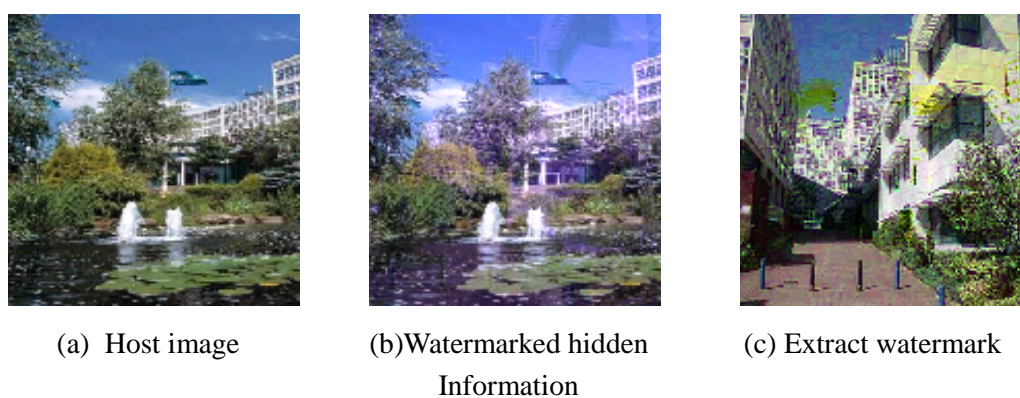


Figure C-3 DWT to recover the watermark image