Signal Processing 142 (2018) 53–68

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

# Secure image cryptosystem with unique key streams via hyper-chaotic system

Hossam Diab[a,b,*], Aly M. El-semary[a,c]

[a] College of Computer Science and Engineering, Taibah University, KSA
[b] Math and computer science Department, Faculty of Science, Menoufia University, Menoufia, Egypt
[c] Systems and Computer Engineering Department, Faculty of Engineering, Al-Azhar University, Cairo, Egypt

ARTICLE INFO

*Article history:*
Received 20 February 2017
Revised 24 June 2017
Accepted 27 June 2017
Available online 28 June 2017

*Keywords:*
Image cryptosystem
Cryptanalysis
Key generation
Multimedia
Chaotic systems

ABSTRACT

This paper cryptanalyses the hyper-chaotic image encryption scheme developed by Norouzi et al. and presents a chosen-plain-image attack scenario to reveal its generated key stream. The recovered key stream can be used to decrypt any future related cipher-image without having the secret key. In addition, the paper introduces an advanced version of the underlying image encryption scheme to overcome its security shortcomings. Specifically, the proposed cipher generates a unique key stream for each distinct plain-image based on its fingerprint. This thwarts the chosen-plain-image attacks and enhances the security level of the proposed scheme. Finally, the experimental results confirm the robustness of the proposed image cipher against different types of attacks.



## 1. Introduction

The multimedia protection, especially digital images, has been recognized as a hot topic of research in the recent few years. By studying the intrinsic features of digital images, it is found that the conventional encryption algorithms are not suitable for protecting the images. These features include high correlation among pixels, high redundancy, and huge data capacity. Accordingly, several image cryptosystems based on chaos techniques were introduced into literature to consider these features [1–12]. The chaos techniques have been employed to build stronger ciphers than ciphers based on traditional cryptographic algorithms. Indeed, this is motivated by the subtle similarities between chaotic systems and cryptographic systems. Specifically, chaos characteristics such as tough sensitivity to the initial conditions and control parameters, aperiodicity, random-like behavior, and ergodicity can be utilized into the main functions of confusion, diffusion, randomness, and secret key sensitivity to provide good ciphers. Unfortunately, a large number of these image cryptosystems have been found insecure; especially, they are vulnerable to known and/or chosen-image attacks [13–20].

Recently, a private key image encryption scheme based on hyper-chaotic system was proposed in [1]. The cryptosystem consists of only one round of diffusion in which the image pixels are sequentially masked by a hyper-chaotic pseudo-random sequence. To encrypt a pixel, the scheme takes into account the sum of all pixels located after the pixel to increase the plain-image sensitivity of the scheme. Accordingly, the scheme has several advantages such as the sensitivity to the plain-image and secret keys, the sufficient large key space, and the high speed performance. Unfortunately, we argue that this scheme is vulnerable to chosen-plain-image attack. Therefore, this paper is devoted to demonstrate a serious security flaw associated with this scheme and then provide a secure version of the underlying scheme to resist several attacks including the chosen-plain-image attacks.

The remainder of this paper is organized as follows. Section 2 provides a brief description of the image cipher under study and then presents a chosen-plain-image attack that compromises the security of cipher. Section 3 suggests an enhanced version of the scheme to overcome its security shortcomings. Section 4 introduces security analysis and simulation results associated with the proposed image cipher. Finally, Section 5 concludes the paper.

## 2. Underlying image scheme cryptanalysis

The original image encryption scheme developed by Norouzi et al. [1] utilizes a hyper-chaotic system to generate a key stream that masks a gray image through one round of diffusion to enhance the encryption speed. The hyper-chaotic system obtains the

* Corresponding author.
*E-mail addresses:* hdiab@taibahu.edu.sa, dr.hosamdiab@gmail.com (H. Diab), aelsemary@taibahu.edu.sa, alyelsemary@azhar.edu.eg (A.M. El-semary).

http://dx.doi.org/10.1016/j.sigpro.2017.06.028
0165-1684/

key stream as only a function of the associated secret key (i.e., the key stream is independent of the plain-image). The hyper-chaotic system deployed by the underlying image cipher is described in Eq. (1).

$$\begin{cases} \dot{x} = a(y - x) + yz \\ \dot{y} = cx - y - xz + w \\ \dot{z} = xy - bz \\ \dot{w} = dw - xz \end{cases} \tag{1}$$

where $a$, $b$, $c$ and $d$ are the control parameters of the system while $x_0$, $y_0$, $z_0$, and $w_0$ are initial conditions. The image cryptosystem under study uses these initial conditions as secret keys for generating the key stream of the encryption/decryption process. The rest of this section briefly describes the cryptosystem under study in Section 2.1 while Section 2.2 provides the details of a chosen-plain-image attack to successfully break the scheme.

### 2.1. Image cryptosystem under analysis

This section briefly discusses both the encryption and decryption algorithms employed by the cryptosystem under study. To introduce the algorithms, it is assumed that a plain-image $F$ of size $W \times H$ needs to be encrypted, where $W$ and $H$ are the width and height of the image, respectively. The encryption algorithm described in more details in [1] can be summarized in the following steps:

**Step 1:** Scan the two dimensional plain-image, $F$, from left to right and then top to bottom to obtain one dimensional vector $P = \{p_1, p_2, \ldots, p_L : L = W \times H\}$ in which $p_i$ is the gray value of the *ith* pixel.

**Step 2:** Utilize the hyper-chaotic system given in Eq. (1) to generate the key stream $K = \{k_i\},\ i = 1, 2, \ldots, L$ in which $k_i$ is ranging from 0 to 255.

**Step 3:** FOR $i = 1$ to $L$

3.1 Compute the characteristic value $CV_i^p$ of the *ith* pixel $p_i$ of the plain-image $P$ based on Eq. (2).

$$CV_i^p = \begin{cases} \sum_{j=2}^{L} p_j & if\ i = 1 \\ CV_{i-1}^p - p_i & if\ i > 1 \end{cases} \tag{2}$$

3.2 Compute a temporary value $v_i^p$ as a function of $CV_i^p$ and $k_i$ according to Eq. (3).

$$v_i^p = floor\left(\left(\frac{CV_i^p}{256^5} \times k_i \times 10^{10}\right) mod\ 256\right) \tag{3}$$

3.3 Encrypt the *ith* pixel $p_i$ based on Eq. (4).

$$c_i = \begin{cases} p_i \oplus (A_0 + v_i^p)\ mod\ 256 \oplus k_i & if\ i = 1 \\ p_i \oplus (k_i + c_{i-1})\ mod\ 256 \oplus v_i^p & if\ i > 1 \end{cases} \tag{4}$$

3.4 END FOR

where $p_i$ and $c_i$ are the *ith* plain-image pixel and its corresponding cipher-image pixel. Also $k_i$ is the key generated for the *ith* pixel while $A_0$ is an initial seed.

On the other hand, the decryption algorithm follows a similar procedure as the encryption process but it is carried out in reverse order. The decryption process starts from the last pixel located in the bottom right corner to the first pixel located in the upper left corner of the cipher-image $C$. Accordingly, the decryption algorithm found in [1] can be summarized in the following steps:

**Step 1:** Scan the two dimensional cipher-image from left to right and then top to bottom to obtain one dimensional vector $C = \{c_1, c_2, \ldots, c_L : L = W \times H\}$ in which $c_i$ is the cipher value of the *ith* pixel.

**Step 2:** Generate the key stream $K = \{k_i\},\ i = 1, 2, \ldots, L$ in the same way as in Step 2 of the encryption process.

**Step 3:** FOR $i = L$ to 1

3.1 Compute the characteristic value $CV_i^c$ of the *ith* pixel $c_i$ of the cipher-image $C$ as described in Eq. (5).

$$CV_i^c = \begin{cases} 0 & if\ i = L \\ CV_{i+1}^c + P_{i+1} & if\ i < L \end{cases} \tag{5}$$

3.2 Compute a temporary value $v_i^c$ as a function of $CV_i^c$ and $k_i$ according to Eq. (6).

$$v_i^c = floor\left(\left(\frac{CV_i^c}{256^5} \times k_i \times 10^{10}\right) mod\ 256\right) \tag{6}$$

3.3 Decrypt the *ith* pixel $c_i$ based on Eq. (7).

$$p_i = \begin{cases} c_i \oplus (A_0 + v_i^c)\ mod\ 256 \oplus k_i & if\ i = 1 \\ c_i \oplus (k_i + c_{i-1})\ mod\ 256 \oplus v_i^c & if\ i > 1 \end{cases} \tag{7}$$

3.4 END FOR

### 2.2. Chosen-plain-image attack scenario

This section introduces a chosen-plain-image attack scenario to recover an original plain-image from its corresponding cipher-image without having the related secret key. From the structure of the decryption algorithm, it is found that the algorithm starts the decryption from the pixel $L$ and sets its corresponding characteristic value $CV_L^c$ to zero according to Eq. (5). As a result, the corresponding temporal value $v_L^c$ becomes zero based on Eq. 6. Then the plain value $p_L$ can be recovered based on the known values of $v_L^c$, $c_{L-1}$, $c_L$, and $k_L$ according to Eq. 7. After recovering $p_L$, it will be used to obtain the values of $CV_{L-1}^c$ and $v_{L-1}^c$ of the $(L-1)^{th}$ pixel from Eqs. (5) and (6), respectively. Then, the plain value $p_{L-1}$ of the $(L-1)th$ cipher pixel is recovered from Eq. (7). This process is repeated until reaching the 1st pixel of the cipher-image. Note that in recovering the 1st pixel $p_1$, the initial value $A_0$ is used instead of the cipher value of $c_{i-1}$ (i.e., $c_0$ since $i = 1$).

According to the structure of the underlying cryptosystem, it is found that the cryptosystem is vulnerable to chosen-plain-image attacks as also briefly noted in [21]. This vulnerability can be exploited by an attacker to extract the related key stream used in the encryption process. Specifically, to extract the key stream, the attacker chooses a plain image $P$ with zero values for all its pixels $p_i$, where $i = 1, 2, \ldots, L$. Also it is assumed that the attacker has an encryption oracle of the encryption algorithm to get the corresponding cipher-image $C$ of the plain-image $P$. Since the value of the *ith* pixel is zero, each of the corresponding characteristic values $CV_i^p$ and $CV_i^c$ becomes zero based on Eqs. (2) and (5), respectively. This, in turns, evaluates each of the corresponding temporal values $v_i^p$ and $v_i^c$ to zero according to Eqs. (3) and (6), respectively. For this special chosen-plain-image $P$ and its corresponding cipher-image $C$, the decryption in Eq. (7) will be reduced to Eq. (8) by removing the variable $v_i^c$ which has the value of zero. Also the initial value $A_0$ is a value between 0 and 255 (i.e., $A_0\ mod\ 256 = A_0$).

$$p_i = \begin{cases} c_i \oplus A_0 \oplus k_i & if\ i = 1 \\ c_i \oplus (k_i + c_{i-1})\ mod\ 256 & if\ i > 1 \end{cases} \tag{8}$$

Also because the value of the *ith* pixel $p_i$ is zero for all $i = 1, 2, \ldots, L$; the key stream $K = \{k_1, k_2, \ldots, k_L\}$ can be obtained from