Signal Processing: *Image Communication* 36 (2015) 106–114

Contents lists available at ScienceDirect

Signal Processing: *Image Communication*

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

A self-synchronized chaotic image encryption scheme ☆

Amir Daneshgar [a,*], Behrooz Khadem [b]

[a] *Sharif University of Technology – Department of Mathematical Sciences P.O. Box 11155-9415, Tehran, Iran*
[b] *Kharazmi University – Faculty of Mathematics and Computer Science P.O. Box 15719-14911, Tehran, Iran*

ARTICLE INFO

Article history:
Received 10 April 2015
Accepted 11 June 2015
Available online 29 June 2015

Keywords:
Chaos
Image encryption
Self-synchronization

ABSTRACT

In this paper, a word based chaotic image encryption scheme for gray images is proposed that can be used in both synchronous and self-synchronous modes. The encryption scheme operates in a finite field where we have also analyzed its performance according to numerical precision used in implementation. We show that the scheme not only passes a variety of security tests, but also it is verified that the proposed scheme operates faster than other existing schemes of the same type even when using lightweight short key sizes.

1. Introduction

Several image encryption schemes have been proposed in the literature based on different approaches for design or implementation, while chaos-based encryption schemes have the advantage of presenting a good combination of speed and security.

It seems that Fridrich [7] is among the first contributors who has proposed an image encryption scheme based on chaotic maps, where in [7] certain invertible chaotic 2D maps on a torus or on a square have been used to create new symmetric block encryption schemes. Many other chaotic image encryption schemes have been proposed ever since with different properties and motivations for application (e.g. see [4,8,12,17,19,22,23] and references therein).

Strictly speaking, one may consider the following challenges when one is trying to design an image encryption scheme (see [16,19,20] and references therein):

- The scheme must have a relatively high speed of performance since images usually consist of large blocks of data.
- Since the information content of an image is contained in high frequencies the scheme must possess a high mixing performance.
- According to typical applications, the scheme must be relatively lightweight and should be able to operate with relatively small keys with acceptable security guaranties.
- The scheme must guaranty secure, reliable and fast rates of data transfer.

Considering the above facts, chaos-based stream ciphers may seem to be a solution while

- Although concentrating on chaotic word-based designs operating in a finite field may seem to be a solution for fast and reliable encryption, one must note that discretizing chaotic maps usually deteriorate their chaotic properties that may lead to weak security conditions (e.g. see [14]).
- Data transfer reliability can be achieved using self-synchronization, however, security guaranty is much harder in the presence of self-synchronization for the feedback structure.

It seems that one of the main problems with chaotic encryption schemes introduced so far is the direct application of the chaotic sequence which is far from being pseudorandom when it is digitized, which will definitely lead to security

☆ A preliminary version of this article has already been posted at http://arxiv.org/abs/1411.7487.

* Corresponding author.
E-mail addresses: daneshgar@sharif.ir (A. Daneshgar), std_khadem@khu.ac.ir (B. Khadem).

http://dx.doi.org/10.1016/j.image.2015.06.005
0923-5965/

weaknesses when the scheme is not design properly (e.g. see [13,21]). Therefore, to solve the above-mentioned and seemingly contradicting challenges, we introduce an image encryption scheme in which we have used a chaotic string indirectly to generate a pseudorandom permutation whose pseudorandomness is guaranteed by the results of [1]. On the other hand to compensate the weakness of discrete permutations in uniformly encrypting the high frequency image data (mainly based on correlation along edges) we use a linear feedback to achieve the acceptable uniformization. In other words we

- Use pseudorandom permutations generated by chaotic maps.
- Use word-based chaos to guaranty fast encryption.
- Compensate discretization phenomenon using a fast linear feedback.
- Make sure that the scheme can perform in both synchronous and self-synchronous modes by setting parameters, to be able to be used in different channel conditions in a reliable way.
- Make sure that the scheme has a fast receiver as an unknown input observer of the transmitter.

PLCIE[1] is an extension of PLC scheme introduced in [11] tuned to be used for image encryption. PLCIE is a word-based chaotic encryption scheme having ℓ-word state vectors that can be controlled by users, giving sufficient flexibility for multi-level security. PLCIE consists of a Initializing phase, internal state update, memory update, encryption and decryption that will be described in detail in Section 2. In Section 3, we apply various tests to verify the performance and the security of the proposed scheme.

2. Description of PLCIE

A digital image usually can be interpreted as a function $z=f(x,y)$ of physical horizontal x and vertical y coordinates that determine illumination or grayscale value of the picture element (or the pixel) at location (x,y). A pixel is the smallest addressable element in a display device. The level of illumination at each pixel has a value between 0 and 255. Thus, in a digital image, the grayscale of each pixel is presented by one byte and the whole image is presented by a large matrix of bytes. The histogram of a digital image is a discrete function $h(r_k)=n_k$, where r_k is the kth gray level and n_k is the number of pixels of the image with gray level r_k.

Let q be a prime power, $\mathbb{F}_q$ be the finite field on q elements[2] and $f:\mathbb{R}\to\mathbb{R}$ be a chaotic map (e.g. as in [6]). Consider a family of maps as $\pi:\mathcal{K}\times\mathbb{F}_q\to\mathbb{F}_q$ such that for any $k\in\mathcal{K}$ the map $\pi(k,.)$ is a discrete chaotic permutation on $\mathbb{F}_q$ as a discrete approximation of f (e.g. as defined in [3]). The two-variable map π gets a value $k\in\mathcal{K}$ as well as a field element $a\in\mathbb{F}_q$, and returns $\pi_k(a)\overset{\text{def}}{=}\pi(k,a)$. In Section 2.1 we will describe how one may compute this family of chaotic permutations.

For all $t\geq 1$, consider $p_t,c_t,z_t\in\mathbb{F}_q$, and let $<p_t>$, $<c_t>$, $<z_t>$ be the plain, the cipher, and the keystream sequences in time, respectively. Let ℓ be an integer. Also, define column vectors $\mathbf{p}_t,\mathbf{c}_t,\mathbf{z}_t$ in $\mathbb{F}_q^\ell$ as

$$\mathbf{p}_t\overset{\text{def}}{=}[p_t^{(1)},0,\ldots,0]^T,\quad \mathbf{c}_t\overset{\text{def}}{=}[c_t^{(1)},c_t^{(2)},\ldots,c_t^{(\ell)}]^T,$$
$$\mathbf{z}_t\overset{\text{def}}{=}[z_t^{(1)},z_t^{(2)},\ldots,z_t^{(\ell)}]^T.$$

The internal state $\mathbf{s}_t\in\mathbb{F}_q^\ell$ and internal memory $\tilde{\mathbf{c}}_t\in\mathbb{F}_q^\ell$ are also defined as column vectors

$$\mathbf{s}_t\overset{\text{def}}{=}[s_t^{(1)},s_t^{(2)},\ldots,s_t^{(\ell)}]^T,\quad \tilde{\mathbf{c}}_t\overset{\text{def}}{=}[\tilde{c}_t^{(\ell)},\tilde{c}_t^{(\ell-1)},\ldots,\tilde{c}_t^{(1)}]^T.$$

Define the map $\wp_k:\mathbb{F}_q^\ell\to\mathbb{F}_q^\ell$ as follows:

$$\wp_k([a_1,a_2,\ldots,a_\ell]^T)\overset{\text{def}}{=}[\pi_k(a_1),\pi_k(a_2),\ldots,\pi_k(a_\ell)]^T.$$

PLCIE scheme uses a set of functions introduced in Table 1 in which $\mathcal{M}$ stands for the set of all $\ell\times\ell$ matrices on $\mathbb{F}_q$. Also PLCIE has a initializing phase along with two other main phases called the kernel computation phase, and the encryption/decryption phase that will be described in what follows.

2.1. The initializing phase

In this phase, a chaotic sequence is produced that gives rise to the pseudorandom permutation π_k. Also, the initial value vector IV is set according to a uniform distribution. The secret key is a binary string consisting of

- encoding of the system precision *prec* (one bit indicating 16 or 32 bits representation of numbers).
- encodings of the initial values of the chaotic map (r_0,l_0), chosen uniformly at random, where $r_0\in_R(0,1)$ (presented in *prec* bits floating point format) and $l_0\in_R\{1,2,\ldots,2^{prec}-1\}$ (presented in *prec* bits integer format).
- encodings of a number $a\in_R\mathbb{F}_q$ and $(i_1,j_1,e_{i_1j_1}),\ldots,(i_n,j_n,e_{i_nj_n})$, in which $n<\frac{l^2}{2}$, indicating an encoding of the matrix $\mathbf{E}$ (to be used later) such that the entries not mentioned in the coding is set to the default value a.

Let ι be a constant integer of order $O(\ell)$ (e.g. for $\ell=8$ this parameter can be chosen as $\iota=32$). Then, IV is a 2ℓ word vector which is used to preset $\mathbf{s}_{-\iota}$ and $\tilde{\mathbf{c}}_{-\iota}$. Note that here one may use a random string of length ι as a prefix of plaintext for whitening.

For the chaotic map we have chosen a particular version of the Rényi map [1] with parameter $\beta=3$ which

Table 1
Functions used in PLCIE.

Title	Form
State update	$\varphi_k:(\mathrm{Fq}^\ell)2\times\mathrm{M}4\to\mathrm{Fq}^\ell$
Keystream generator	$\gamma_k:(\mathrm{Fq}^\ell)2\times\mathrm{M}2\to\mathrm{Fq}^\ell$
Encryption	$\varepsilon_k:(\mathrm{Fq}^\ell)2\times\mathrm{M}\to\mathrm{Fq}^\ell$
Decryption	$\delta_k:(\mathrm{Fq}^\ell)2\times\mathrm{M}\to\mathrm{Fq}^\ell$
Memory update	$\mu:(\mathrm{Fq}^\ell)2\to\mathrm{Fq}^\ell$

[1] PseudoLinear Chaotic Image Encryption.

[2] The cryptosystem can be defined on any finite field, however, in our real life applications with a lightweight setup we set $\mathbb{F}_q=GF(16)$ or $\mathbb{F}_q=GF(17)$.