Optics Communications 350 (2015) 77–82

Contents lists available at ScienceDirect

Optics Communications

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

Cryptanalysis of "an improvement over an image encryption method based on total shuffling"

A. Akhavan [a,*], A. Samsudin [a], A. Akhshani [b]

[a] School of Computer Sciences, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia
[b] School of Physics, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia

ARTICLE INFO

Article history:
Received 8 January 2015
Received in revised form
27 March 2015
Accepted 30 March 2015
Available online 1 April 2015

Keywords:
Chaotic cryptography
Image encryption
Cryptanalysis
Total shuffling
Chosen plain-text attack

ABSTRACT

In the past two decades, several image encryption algorithms based on chaotic systems had been proposed. Many of the proposed algorithms are meant to improve other chaos based and conventional cryptographic algorithms. Whereas, many of the proposed improvement methods suffer from serious security problems. In this paper, the security of the recently proposed improvement method for a chaos-based image encryption algorithm is analyzed. The results indicate the weakness of the analyzed algorithm against chosen plain-text.

1. Introduction

The similarities between chaotic maps and cryptographic systems have been a motivation for many of the researchers [1–12]. Some of these features such as strong sensitivity to the initial conditions and control parameters (similar to confusion and diffusion), aperiodicity, random-like behavior and ergodicity (providing uniform randomness) have been used to achieve strong cryptographic algorithms that could compete with conventional cryptographic algorithms. In addition, several hash functions and image encryption algorithms based on chaotic systems were proposed in the past two decades [13,14]. Generally, four major strategies in design of image encryption algorithms have been manipulated by the designers. The first and easiest to apply is masking the pixel values by the sequence generated from chaotic maps [15]. The second method has been the shuffling or permutation method. The shuffling method manipulates the sequences generated by the chaotic maps to permute the position of the pixels of an image [16]. The third method relies on the number of iterations of a chaotic map. The algorithms designed based on this method are usually referred as the *Baptista type* image encryption algorithms [1,17]. The fourth method or the hybrid method creates a connection between the chaotic systems and the cipher-image. The structure of this method is more complicated compared to the first three methods. The image-encryption algorithms in the first and second groups are prone to several weaknesses and usually can easily be cryptanalyzed. The image encryption algorithms in the third group, rely on the number of iteration, therefore achieving a reasonable security leads to very slow encryption speed. Several modifications of the third group algorithms are proposed after first original algorithm was cryptanalyzed, and some of these modified algorithms were broken shortly after publication. The fourth group algorithms are usually hybrid of the first three groups. There are some cases from the fourth group that are cryptanalyzed using chosen plain-text and cipher-text attacks. The Zhang and Liu works is [18] is a good example of the hybrid design of chaos based image encryption algorithm. It was cryptanalyzed by Wang and He [19] shortly after publication. Recently a new improved version is suggested by Eslami and Bakhshandeh [20]. In this paper, the security of the improved version of this algorithm [20] is analyzed. The proposed algorithm in Ref. [20] applies two separate schemes to achieve the necessary confusion and diffusion for an image encryption algorithm. Both schemes (shuffling of the pixels of the image and masking of pixels using chaotic data) do not reflect the effect of small changes in the plain image, over the cipher-image. In the other words, the presented algorithm in Ref. [20] is not sensitive enough to the plain image and therefore can be cryptanalyzed using the differential attack. Besides, the lack of existence of a transient removal step, and application of unimodal tent chaotic map, reduces the claimed key space.

* Corresponding author.
E-mail address: amir.akhavan@yahoo.com (A. Akhavan).

http://dx.doi.org/10.1016/j.optcom.2015.03.079
0030-4018/

In order to avoid these drawbacks, the algorithm should be sensitive to all the pixels of the plain image and a minor change in the plain image should lead to a completely different cipher-image. There exists several suggested strategies to achieve this goal [4,8,7]. Moreover, in Ref. [21], the inadequacy of the unimodal chaotic maps are discussed and use of unimodal chaotic maps in cryptography is prohibited. Also, in order to avoid the transient effect, the chaotic map should be iterated long enough and the results should be ignored.

The results of the security analysis indicate that the improved version of the algorithm is still vulnerable against chosen plain-text and chosen cipher-text attack and any cipher-image can be decrypted even without the access to the keys.

2. The improved image encryption method based on total shuffling

The original algorithm is based on two major steps: one shuffling step and one diffusion step. The shuffling step is plain-text independent meaning that the permutation map is the same for any plain-image if the same keys are implemented. Whereas, in the second step, in order to make the algorithm sensitive to the plain image, the shuffled plain-image pixels are masked with a function which depends directly on the plain-image. In the improved version of the algorithm, this dependence to the plain-image pixels is modified to the dependence on the previously ciphered pixel. Also, the number of iterations of the chaotic map relies on the ciphered pixel value.

2.1. Permutation step

In the improved algorithm [20], the original permutation steps in [18] are kept unchanged. In the permutation step, the skew tent map is iterated equal to the number of pixels in the plain-image. Moreover, the resulting sequence is indexed and sorted based on order of the sequence values. The generated matrix is then used to permute the pixels position. The following steps describe the permutation step in more detail:

1. Iterate skew tent map equation (1) using the initial conditions x_0 and control parameter p for $M \times N$ times. Store the generated sequence $X = \{x_1, x_2, \ldots, x_{M\times N}\}$.
2. Index and sort the generated sequence T.
3. Transform the image into one dimensional sequence of pixels $P = \{p_1, p_2, \ldots, p_n\}$. Manipulate the sorted indexes (T) to shuffle the position of the pixels in the plain-image:

$$f(x_{n+1}) = \begin{cases} \dfrac{x_n}{p} & \text{if } x_n \in [0, p] \\ \dfrac{1 - x_n}{1 - p} & \text{if } x_n \in (p, 1] \end{cases} \quad (1)$$

After the permutation process, the newly generated shuffled image $P' = \{p'_1, p'_2, \ldots, p'_n\}$ is processed one more time in the diffusion step.

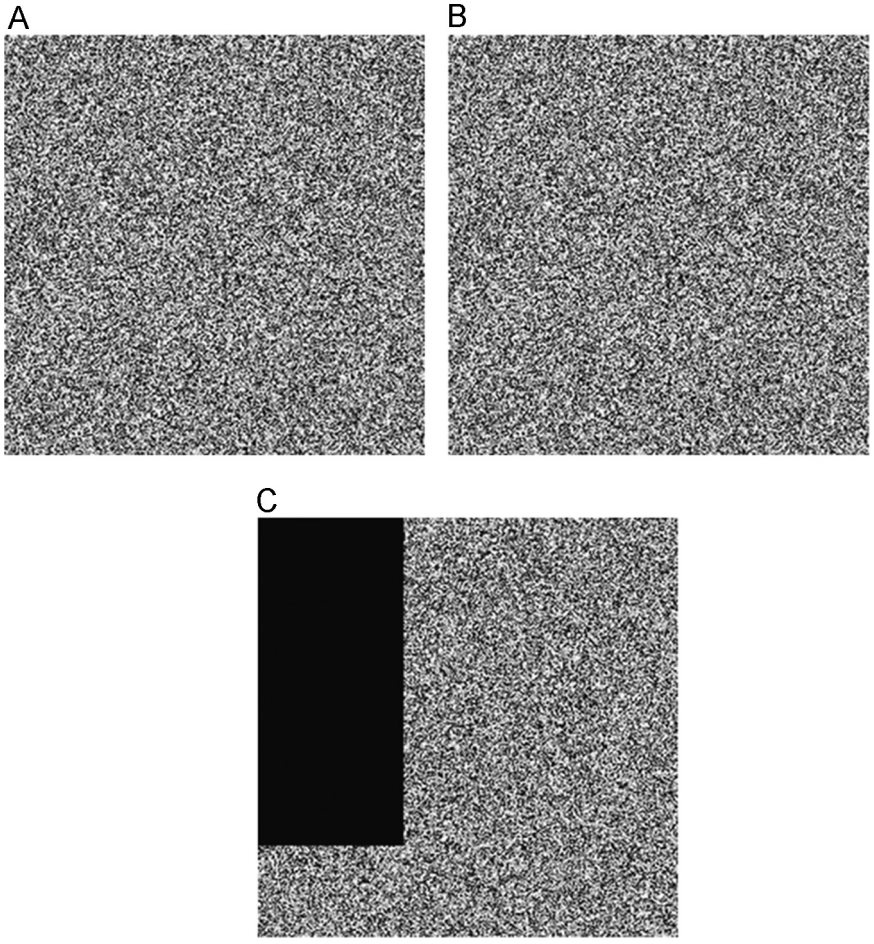

Fig. 1. (A) Ciphered black image (C_{black}). (B) Modified black image ($P^i_{M\times N}$) at location of i = (200, 89). (C) Result of XORed images A and B: D^i where i = (200, 89).