Journal of Information Security and Applications 40 (2018) 134–144

Contents lists available at ScienceDirect

Journal of Information Security and Applications

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

Securing images with a diffusion mechanism based on Fractional Brownian Motion

Manish Kumar[a], Sunil Kumar[a], M.K. Das[a], Rajat Budhiraja[b], Sanjeev Singh[a,*]

[a] Institute of Informatics and Communication, University of Delhi South Campus, New Delhi, India
[b] Aricent India (H) Pvt. Ltd., Gurugram, Haryana, India

ARTICLE INFO

Article history:

Keywords:
Image security
Fractional Brownian Motion
Confusion
Diffusion

ABSTRACT

In this work, an image encryption model is proposed incorporating an intertwining logistic map, neural network based confusion algorithm and Fractional Brownian Motion (FBM) based diffusion process. Using an intertwining logistic map in the encryption model leads to a better distribution of random numbers compared to the logistic map which have the blank window observed in bifurcation diagram and further the neural network model based confusion algorithm increases the key sensitivity in the model. Finally, the use of FBM based diffusion process in the model changes the pixels in a way that any minor change in a pixel leads to a change in a large number of pixels in the cipher image. The present encryption model is shown to possess an enhanced image security and better NPCR, UACI scores. The correlation between pixels in cipher image is observed to be negligible. The performance indices of the proposed model are shown to improve in comparison with other models based on dynamic random growth method and also block scrambling and dynamic index based diffusion.

1. Introduction

Recent trends in the field of social connectivity have triggered an exhaustive use of digital images over the Internet, and its security is of prime concern. Many research groups and labs across the globe are working towards the security of such images. The images have huge information content and high redundancy in the data and hence specific algorithms are required for their security rather than the traditional data encryption algorithms. It is also challenging to develop quality algorithms as the high correlation among image pixels inspire attackers to develop successful attack mechanisms. Many different image encryption methods have been reported in the last decade implementing different algorithms. Some are chaos based due to various desirable properties [1–3], while some used Genetic Algorithms owing to their random nature [4–6] and many others used wavelet and other transformations for encrypting images [7,8]. For achieving higher complexity, many cryptographic algorithms are also found to use coupled map lattice (CML) [9,10]. A detailed review, analysis and suggestions on recent work in cryptography using chaos is given in [11].

Rostami et al. [12], proposed a cryptographic model for gray images with chaotic windows based on logistic map. The initial conditions for the logistic map were derived from plain text, so as to increase the key sensitivity to plain text. Multiple chaotic windows of size 16×16 were then generated using the out put of the logistic map. The plain image was first broken down into 16×16 blocks and then XORed with these chaotic windows. The algorithm was designed to process image pixels in a parallel mode and does not take into account the consecutive pixels for processing. Xu et al. [13] used an encryption system using block scrambling and dynamic index based diffusion. The method, first divides the image into two equal blocks in horizontal and vertical directions, then X, Y coordinates and swapping control tables were generated using a chaotic matrix. Pixel positions are swapped based on the X and Y coordinates and blocks are chosen based on swapping control table. At the end, diffusion is applied based on the dynamic index scheme. Similarly, Mandal et al. [14], proposed a symmetric image encryption technique based on chaotic Rössler system which generates three chaotic sequences. The system uses element shuffling and pixel replacement techniques for confusion. Further, Laiphrakpam et al. [15] proved the non-robustness in key generation used in [14] and suggested ways for better security.

Another cryptographic system was proposed by Pak et al. [16] based on one-dimensional chaotic maps. The coupling of two similar chaotic maps was used to generate a new chaotic map of one dimension. System parameters were used as secret key and further permutation, diffusion and linear transformation were applied to the image making it resistible against differential attacks.

* Corresponding author.
E-mail address: sanjeev@south.du.ac.in (S. Singh).

https://doi.org/10.1016/j.jisa.2018.03.007
2214-2126/

Similar to these cryptosystems many others are proposed in literature with different permutation and diffusion mechanisms [17–20].

Fractional Brownian Motion (FBM) has many applications like in connectionless networks [21], as a maximum likelihood estimator [22], in detection of objects in images [23] and in other image analysis [24,25]. Apart from image analysis, FBM is used in various other domains such as market analysis [26], data traffic modeling [27], and face recognition [28]. In literature, many algorithms based on Brownian Motion have utilised the randomness of the algorithm as the key property for image security. Wang et al. [29] utilised the brownian motion to shuffle the image pixels according to the motion of the particle observed using Monte Carlo Simulation. Similarly Chai et al. [30] used 3D brownian motion in combination of the logistic tent system to permute the image pixels. In Chai [31], a method to scramble the plain image pixels is proposed using the 1D chaotic system with brownian motion.

The present work uses FBM as a diffusion process. The idea of using FBM into the diffusion follows from the fact that it can bind only the previous pixel to the next one. The property of FBM depends on the random seed and a *Hürst* exponent *H*. Generally, encryption algorithms use Friedrich's diffusion mechanism for propagating small change in pixel to a large number of pixels in the image. However, use of such diffusion mechanism allows one to decipher the encryption key in case of chosen plaintext (CP) and known plaintext (KP) attacks [32–34]. The vulnerability of such diffusion based encryption algorithms towards CP and KP can be shown to be due to use of difference equation of modulo addition (DEA) of the form $(\alpha \dotplus k) \oplus (\beta \dotplus k) = y$, where symbols have their same meanings as in [35].

A symmetric color image encryption algorithm is proposed in this paper. The algorithm first breaks the image into its RGB channels. Each channel is then placed in a single array and shuffled using a shuffling algorithm, resulting into completely unrelated pixels for all channels. These pixels are then subjected to perceptron based Neural Network model for increasing the degree of confusion and uses eight neurons to modify each bit of a pixel. The modification is based on the key which is also used in calculation of weights and hence modifies the pixels. Finally, the pixels are diffused using a diffusion mechanism based on FBM which helps in propagating a change in any pixel to most of the image pixels, thereby making the image more sensitive to even minor change in key or data. It may be noted that the FBM used in the diffusion process binds only two consecutive pixels and is based on two parameters namely initial seed and the fractality parameter *H*, known as *Hürst* exponent.

The rest of this paper is organized as follows: Section 2 gives an introduction to the Fractional Brownian Motion (FMB). In Section 3, key generation and shuffling process for the cryptosystem are explained. Section 4 details the Neural Network model used and diffusion model is explained in Section 5. Section 6 discusses the encryption model in brief. To verify the cryptosystem for various types of attacks resistance, detailed simulations is done and corresponding results are discussed in Section 7. Finally conclusions are drawn and discussed in Section 8.

2. Fractional Brownian Motion

Fractional Brownian Motion is like a random walk (in random direction) which when put mathematically is basically a Gaussian stochastic process depending upon the 'Hürst Exponent' $H \in (0, 1)$. It has fixed growth, self similarity and wide range of dependency. If the value of H is 0.5, the Fractional Brownian Motion results into standard brownian motion. FBM is mathematically described as a stochastic representation proposed by Mandelbrot et al. [36] as:

$$J = \int_0^{-\infty} [(t-s)^{H-1/2} - (-s)^{H-1/2}]dB(s),$$

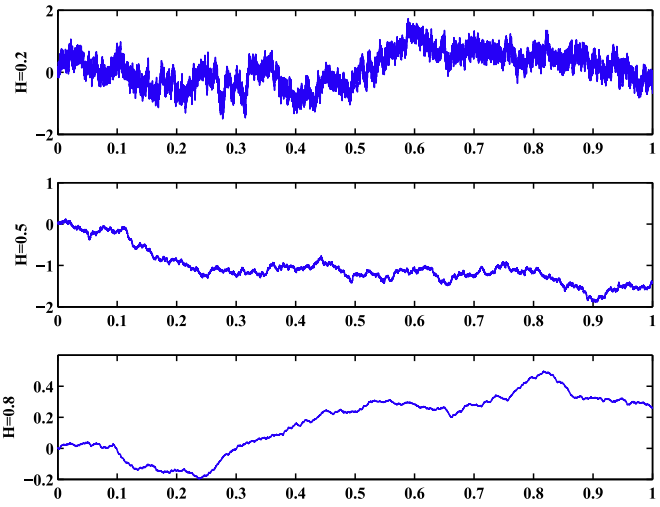

Fig. 1. Fractional Brownian Motion behavior corresponding to different Hürst exponent values.

$$K = \int_0^t [(t-s)^{H-1/2}]dB(s),$$

$$B_H(t) = \frac{1}{\Gamma(H+\frac{1}{2})}J + K.$$

where Γ is the Gamma Function $\Gamma(\alpha) = \int_0^\infty [x^{\alpha-1}\exp(-x)]dx$, and value of H is $0 < H < 1$, known as Hürst exponent.

Fig. 1 plots the output of Fractional Brownian Motion for different values of Hürst Exponent. The normalized FBM $B_H = B_H(t) : 0 \le t < \infty$ where $0 < H < 1$ has following properties:

- $B_H(t)$ has stationary increments;
- $B_H(t)$ has a Gaussian distribution for $t > 0$.

In the proposed model, sequences of random numbers corresponding to a given Hürst exponent are generated and used to diffuse the pixels of the image and such diffusion process makes the cipher images FBM dependent.

3. Key generation and shuffling

A 128 bit long random binary string *K* is used for parameter generation, the length is such that it can resist the brute force attack. This key is further used for generating different parameters of the crypto system which are further used in shuffling, confusion and diffusion processes. The subkeys k_i, where $i = 0$ to 15 and k_i represents the *i*th sub-key in decimal form, are generated from the binary string *K* as

$$k_i = [K((8 \times i) + 1) \quad \text{to} \quad K(8 \times (i+1))].$$

Here, k_i represents the subkeys derived from the randomly chosen 128 bit long binary key *K* and $k_i \in (1, 256)$.

The various Initial conditions of Logistic map used in different processes are generated as

$$x_1 = [(k_4 \times k_{13})/(k_8 + k_{10} + k_3)] \mod 1,$$
$$y_1 = [(k_{12} \times k_9)/(k_{10} + k_{15} + k_2)] \mod 1,$$
$$x_2 = [(k_6 \times k_8)/(k_5 + k_{11} + k_{14})] \mod 1,$$
$$y_2 = [(k_{14} \times k_{16})/(k_9 + k_{12} + k_4)] \mod 1,$$
$$x_3 = [(k_7)/(k_2 + k_1 + k_{10})] \mod 1,$$
$$y_3 = [(k_{11} + k_{15})/(k_4 + k_{13} + k_7)] \mod 1.$$

where x_i, y_i are the *i*th initial conditions of logistic map and $a \mod b$ refers to the remainder after dividing a by b. To remove the transients from the output of logistic map, minimum number of iterations i.e. n_{ite} are calculated as $n_{ite} = (k_1 \times k_{16} + k_7) + 1500$.