KYBERNETIKA — VOLUME *62* (2026), NUMBER 1, PAGES 115–132

SECURING IMAGES VIA VASICEK PROCESS AND CHAOTIC MAP

Hajar Ahalli, Abderrahim Aslimani, Khalid Charif and Mohamed El Ouafi

We propose a novel image encryption algorithm based on the stochastic Vasicek process and a modified chaotic logistic map. The proposed algorithm relies on the Fridrich confusion-diffusion structure. In the initialization step, we generate process parameters and an initialization state from a secret key of arbitrary length. Then the Vasicek process performs several iterations, where the number of iterations is generated at each step by the secret key. The floating point numbers generated by the process are discretized to generate the permutation for pixel confusion, and then generate pseudo-random sequences to diffuse the pixel values. The confusion-diffusion process can be applied for several rounds to improve the encryption performance. Experimental results and security analysis show that our model has good performance. The algorithm has a large key space, passes statistical analysis and has a strong resistance to differential attacks, which confirms that our algorithm is safe and efficient.

Keywords: image encryption, Vasicek process, logistic map, diffusion, confusion

Classification: 65H05, 65F10

1. INTRODUCTION

The exchange of information has become a crucial thing in our daily life, especially with the evolution and growth of communication networks and the development of multimedia technologies. Indeed, image security has become an inescapable necessity and an extremely important issue for every user. Most applications require a level of security to ensure the protection of this data in different communication channels. Efficient encryption algorithms have the property of confusion and diffusion, which corresponds to those mentioned by Shannon [33]. Indeed, these two properties are considered to be the fundamental concepts of cryptographic systems. Traditional algorithms such as AES, DES, IDEA, RSA,..., fulfill these properties well for data as a data stream, but remain inefficient for digital images, which have special properties, such as high redundancy and strong correlation between adjacent pixels, which places special demands on any encryption technique. Several image encryption algorithms based on different theories have been proposed [9, 16, 27, 28, 34, 41, 42]. Indeed, researchers [2, 18, 23] have underlined the existence of a similarity between the properties of chaotic systems such as:

DOI: 10.14736/kyb-2026-1-0115

ergodicity, mixing, randomness, unpredictability and sensitivity to initial conditions, and the needs of cryptographic systems. Consequently, such a close relation gives rise to several robust algorithms for encryption of images by implementing different chaotic systems [7, 12, 17, 19, 30, 44].

Generally, secure image encryption schemes are based on the confusion-diffusion principle proposed by Fridrich [10]. In confusion, the pixels in the image are swapped using permutations generated by multiple methods [14, 15, 25], this breaks the correlation between the pixels in the image, but keeps the same histogram. While in diffusion, pixel values are changed and encrypted by random sequences [4, 6, 26, 31], to ensure higher security against statistical attacks. Indeed, the security of cryptosystems strongly depends on the systems implemented in their design. Using systems with simple behavior, their initial states and control parameters can be estimated using certain techniques [3, 8, 29, 40, 43], and thus, the corresponding image encryption scheme can be easily attacked [21, 22, 35].

Brownian motion is a mathematical description of the random motion of a particle submerged in a fluid. As its name suggests, the movement was discovered in 1827 by botanist Robert Brown (1773 – 1858). It was by observing pollen dispersed in water under a microscope that he noticed that the microscopic grains constituting it were subjected to a continuous and irregular movement. He later realized that this same phenomenon could be observed with all kinds of particles of sufficiently small size. Currently, Brownian motion is used in many applications, for example to analyze the price of financial stocks in the market. Due to the high randomness of Brownian motion, this motion has been used in the design of several cryptographic applications. For example, in 2014, Wang and al. [39] took each pixel in the image as a Brownian particle, used the Monte Carlo method to simulate Brownian motion, and effectively scrambled the image. In 2015, Zhu [45] had broken the encryption algorithm proposed in [39] because the permutation vector and the diffusion sequence are not bound to the plain text image, making the method impossible to resist the chosen clear text attack.

In 1930, Leonard Ornstein and George Eugene Uhlenbeck [37], proposed a process which is considered for the velocities of Brownian particles: the so-called Ornstein – Uhlenbeck process. This continuous-time process is the solution of a stochastic differential equation, the Langevin equation for the Brownian motion of a particle with friction [20]. A decade later, Doob studied the properties of its trajectories from those of Brownian motion using deterministic time change. Indeed, the Ornstein – Uhlenbeck process can be represented as an affine transformation of time-changed Brownian motion. This process has been used for several other applications: in finance, climatology, physics and other areas of science.

This process was originally introduced to describe the speed of a mass particle with friction. When the potential is zero, the speed V of the particle satisfies the following stochastic differential equation

$$dV_t = -\gamma V_t dt + \sqrt{2\gamma\varepsilon} dB_t,$$

where $\varepsilon > 0$ is the force of the noise (depending on the temperature and the mass of the particle), $\gamma > 0$ the friction coefficient (also called damping) and $(B_t)_{t\geq 0}$ is a standard Brownian motion.

Our purpose in this paper is to give a novel image encryption algorithm based on the Vasicek processes, which constitutes a generalization of the Ornstein – Uhlenbeck process. First, at the beginning of this paper, we will give a brief introduction to the Vasicek process and the associated fundamental properties. The second chapter is devoted to the study of logistic maps, on which we will make some modifications to have a total bifurcation. The rest of the paper is organized as follows. Section 3 briefly presents the basic preliminaries used in the design of a strong cryptosystem. Section 4 describes the proposed algorithm. Section 5 presents the simulation results and security analysis, and Section 6 concludes the paper.

X_t	Vasicek process at time t
W_t	standard Brownian motion (Wiener process)
λ, μ, σ	mean-reversion rate, long-term mean, volatility of X_t
x_n	state of the logistic map at iteration n
r	control parameter of the logistic map
K	secret key
$M \times N$	image size (rows $\times$ columns)

Tab. 1: Main notation used in the paper.

2. THE VASICEK PROCESS

In 1977, Oldrich Alfons Vasicek [38], proposed a process which is considered as a stochastic investment model: the so-called Vasicek process. The corresponding process was originally introduced to describe the evolution of interest rates movements based on market risk.

Let us first give here the rigorous definition of the Vasicek process, which constitutes a generalization of the Ornstein – Uhlenbeck process introduced in [37]; defined as the only solution of the following stochastic differential equation

$$dY_t = -\lambda Y_t dt + \sigma dW_t, \tag{1}$$

where $(W_t)_{t\geqslant 0}$ denotes the Wiener process, $\lambda > 0$ and $\sigma > 0$ are parameters. The Vasicek process follows the stochastic differential equation

$$dX_t = \lambda(\mu - X_t)dt + \sigma dW_t, \tag{2}$$

where $\mu \in \mathbb{R}$ is a constant. This equation is used to study the evolution of interest rates in the so-called financial model of Vasicek [5]. We deduce that (2) has an explicit solution given by

$$X_t = (X_0 - \mu)\, e^{-\lambda t} + \mu + \sigma \int_0^t e^{-\lambda(t-u)} \mathrm{d}W_u.$$

It has similar properties to those of the Ornstein – Uhlenbeck process: if $X_0 \sim \mathcal{N}\left(m, \sigma_0^2\right)$ is independent of W, $X = (X_t)_{t\geq 0}$ is a Gaussian process of expectation function

$$\mathbb{E}\left[X_t\right] = me^{-\lambda t} + \mu\left(1 - e^{-\lambda t}\right),$$

of variance function

$$Var[X_t] = \frac{\sigma^2}{2\lambda}(1 - e^{-2\lambda t})$$

and of covariance function

$$Cov\left[X_s, X_t\right] = e^{-\lambda(t+s)}\left(\sigma_0^2 + \frac{\sigma^2}{2\lambda}\left(e^{2\lambda(s\wedge t)} - 1\right)\right) \text{ for } t \neq s.$$

To simulate the Vasicek process we use the Euler – Maruyama method [13]. We consider a uniform time grid $t_n = n\,\Delta t$ on $[0, t_{\max}]$, with time step $\Delta t = t_{\max}/n$, and approximate the SDE (2) by the standard Euler – Maruyama scheme

$$dX_t = \lambda\left(\mu - X_t\right)dt + \sigma dW_t$$

which leads to the recursion

$$X_{n+1} = X_n + \lambda\left(\mu - X_n\right)\Delta t + \sigma\sqrt{\Delta t}\, x_n,$$

where $(x_n)_{n\geq 0}$ are independent standard normal random variables, $x_n \sim \mathcal{N}(0,1)$. Thus,

$$W_{t_{n+1}} - W_{t_n} = \Delta W_n \sim \mathrm{N}(0, \Delta t) = \sqrt{\Delta t} N(0,1).$$

This can be simulated in Python using the algorithm below:

Data: - The parameter sigma σ; The parameter lambda λ; The parameter μ; The maximal time $t_{\max}$; The number of iterations n; The step ϵ;
Result: Simulation of the trajectory of the Vasicek process
initialization $t_0 = 0$;
Simulate n independent standard normal random variables $x_i \sim \mathcal{N}(0,1)$;
for $i = 0$ *to* $n-1$ **do**
    - Compute $X(t_{i+1})$ using

$$X(t_{i+1}) = X(t_i) + \lambda(\mu - X(t_i))\,\Delta t + \sigma\sqrt{\Delta t}\, x_i;$$

end
The discretized trajectory is given by $(t_i, X(t_i))_{i=0,\ldots,n}$.

Algorithm 1: The trajectory algorithm of the Vasicek process.

A trajectory of this process is presented for some values of λ, μ and σ in Figure 1 by applying Euler or Euler – Maruyama method.

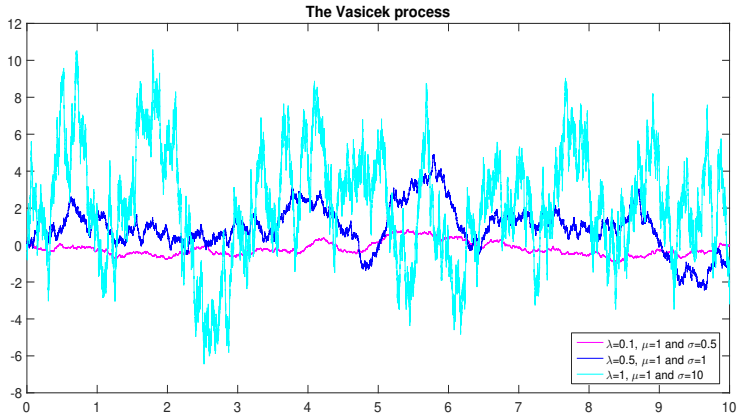

Fig. 1: Simulated trajectories of the Vasicek process.

3. LOGISTIC MAP

The logistic map is a polynomial mapping of degree 2. Its recurrence relation is

$$\begin{aligned} x_{n+1} &= L(x_n) \\ &= r x_n (1 - x_n). \end{aligned}$$

The sequence $(x_n)_{n\geq 1}$ takes values in $(0, 1)$ for $0 < r \leq 4$, and its behavior varies according to the value of the parameter r. The sequence presents a chaotic character for $r > 3.57$ except for a few isolated values of r with a behavior which is not. For example, from $1 + \sqrt{8}$. A bifurcation diagram allows to graphically summarize the different cases:

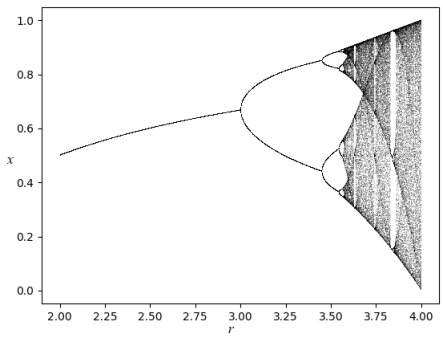

Fig. 2: Bifurcation diagram of the Logistic map.

The logistic maps are used in many image encryption systems[1, 24, 30] and are becoming more familiar to the public and represent some weakness in security. Indeed, it is possible to predict some of its behavior in certain circumstances. Indeed, a chosen cipher text attack is already carried out in [36], to build the discarded version of the logistic maps which leads to the estimation of the control parameter. One solution against this kind of attack is to mix and truncate the chaotic orbit before using it for encryption.

We have kept the same map and we are going to use the random sequences part to randomize the trace of the return map. And so the modified map becomes

$$\begin{aligned} x_{n+1} &= mL(x_n) \\ &= 100rx_n(1-x_n) \mod 1. \end{aligned}$$

The modified map has an identically distributed bifurcation diagram as shown in Figure 3.

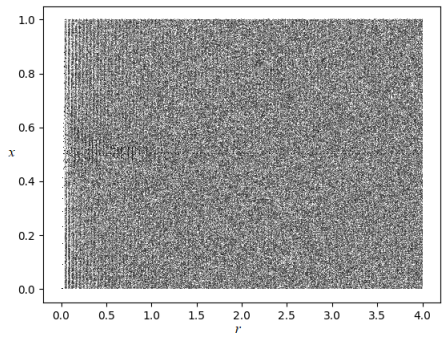

Fig. 3: Bifurcation diagram of the modified Logistic map.

The motivations for this modification can be summarized as follows:

1. This modification will allow us thereafter to have independent normal random variables.

2. For almost any $r \in [0, 4]$, the chaotic sequence is topologically dense in $[0, 1]$.

We can see that this modification in the chaotic map brings more randomness, unpredictability and sensitivity to the initial values, which can provide high security to the proposed algorithm. From a dynamical point of view, multiplying the state by 100 and taking the result modulo 1 enhances the mixing properties of the map and accelerates the loss of memory of initial conditions. Numerically, the modified map produces trajectories whose empirical distribution is closer to the invariant density on $[0, 1]$ and whose samples are less correlated. This improves the unpredictability of the generated sequences and strengthens the security of the encryption scheme compared to the standard logistic map.

4. PROPOSED IMAGE ENCRYPTION ALGORITHM

The goal is to provide a robust algorithm with exceptional resistance against statistical and differential attacks. We have based our concept on the Vasicek process and the chaotic logistic map to propose a novel algorithm for image encryption.

The proposed algorithm has two input parameters, an image I of size $M \times N$ and a secret key $K = (k_0 k_1 k_2 \dots k_{L-1})_2$ of arbitrary length L.

Our encryption scheme is based in its operation on the Fridrich principle. A confusion where it will have a permutation of the positions of the pixels of the image in a one-to-one way, and a diffusion where a masking of the pixel values will take place in a sequential way. For the execution of these operations, we take advantage of the floating point flow $\{X_i\}_{i\geq 1}$ generated by the Vasicek process.

Let f be the transition function between two states $S_i = (x_i, X_i)$ and $S_{i+1} = (x_{i+1}, X_{i+1})$ for $i \geq 0$, which simulate the Vasicek process, defined as follows

$$\begin{aligned} S_{i+1} &= f(S_i) \\ &= (x_{i+1}, X_{i+1}) \\ &= \big(mL(x_i),\ X_i + \lambda(\mu - X_i)\,\Delta t + \sigma\sqrt{\Delta t}\, x_{i+1}\big). \end{aligned}$$

The floating flow obtained by the Vasicek process is discretized by the function D_c defined for a state $S(x, X)$ as follows

$$D_c(S) = \big(\lfloor 10^c |X| \rfloor \bmod c\big), \quad \text{for a state } S(x, X),$$

so that $D_c(S) \in \{0, 1, \dots, c-1\}$. where $c \in [\![255, M \times N]\!]$. In the case where $c = 255$, we generate the random sequences for the masking and scattering of the pixel values, and for $c = M \times N$, we generate integers used in the generation of the permutations for the confusion.

The parameters and the initialization state of the process are generated from the key as indicated in the next section. Then, the process is invited to perform several iterations to generate permutations according to Algorithm 2 and pseudo-random sequences as indicated in Algorithm 3.

In this section, we give a detailed description of the algorithms.

4.1. Generation of initialization values

From the key $K = (k_0 k_1 k_2 \dots k_{L-1})_2$, considered as a character string taken on entry by the keyboard, it is preferable that the key size be greater than or equal to 384 bits (48 characters). The $\{k_i\}_{0\leq i<L}$ are the values of the representation in ASCII code of the key K. We generate the parameters of the Vasicek process X_0, λ, μ and σ, and the initiation state of the logistic function x_0.

The key-derived parameters are consistently written in the order $(x_0, r, X_0, \lambda, \mu, \sigma)$.

Let $K' = (k'_0 k'_1 k'_2 \dots k'_{383})_2$ with

$$k'_i = \begin{cases} k_{i\%L} & \text{if } L \leq 383 \\ \bigoplus_{j=1}^{\left\lceil \frac{L}{384} \right\rceil} k_{i\%(384j)} & \text{if } L > 383 \end{cases}$$

and denote

$$b_p = \frac{1}{2^{32}} \sum_{i=0}^{31} 2^i \left(k'_{L-5i-p-1} \oplus \overline{k'_{5i+p}} \right),$$

where $p \in \{0, 1, 2, 3, 4, 5\}$. The initial state is generated as follows

$$x_0 = \frac{b_5 + 4b_0}{5},\ r = 4\frac{b_0 + 4b_1}{5},\ X_0 = \frac{b_1 + 4b_2}{5},$$

$$\lambda = \frac{b_2 + 4b_3}{5},\ \mu = \frac{b_3 + 4b_4}{5} \text{ and } \sigma = b_4 + 4b_5 + 4.$$

We generate the initialization values of our process by all the bits of the key, we went through all the bits of the key, so a change bit can give a totally different initial values. This high sensitivity to the key will be illustrated in the key-sensitivity tests.

4.2. Description of the pixel permutation algorithm (confusion)

In this section, we propose a new permutation for the confusion of the pixels.

Let $S^0 = (x_0, X_0, t_0)$ be an initiation state on the Vasicek process f. The process f performs a set of iterations, to generate a set of states

$$S = \left\{ S^1, S^2, \ldots S^{M \times N} \right\}$$

with

$$S^i = f^{n_i+1} \left(S^{i-1} \right) \text{ for } i \geq 1 \tag{3}$$

n_i is an integer generated from the key K according to

$$n_i = 2k_j + k_{j-1}$$

where

$$j = 2 \times i \bmod (L - 1).$$

In order to get rid of the existing correlation between consecutive states, at each step i a number n_i of the iterations performed, this number is controlled by the key K. Indeed, between two successive states S^i and S^{i+1}, it can have up to 4 iterations $(1 \leq n_i + 1 \leq 4)$.

With this technique, we did not use all the states to produce random sequences, so it is impossible for an attacker to estimate the control parameters if he does not know the secret key K.

Let consider the identity permutation P

$$P = (1, 2, \ldots, M \times N)$$

and let $\{Q_i\}_{0 \leq i < M \times N}$ be a set of integers with $Q_i \in [\![0, M \times N]\!]$ where

$$Q_i = D_{M \times N} \left(S^i \right).$$

The final permutation is P after some modifications in the following loop

$$\begin{cases} \text{for } i = 1 \text{ to } M \times N \\ \quad \text{swap}(\text{temp}, P_i). \end{cases}$$

Let S^2 be the initialization state for this algorithm.

1 Input: A secret key $K = (k_0 k_1 k_2 \dots k_{L-1})_2$, a state $S_0 = (x_0, X_0)$ and image size M and N
2 Output: A permutation P of size $M \times N$ and state $S^{M \times N}$
3 $j \leftarrow 0, n \leftarrow 2 \times p_{j+1} + p_j$
4 **for** $i = 1$ *to* $M \times N$ **do**
5 $\quad \begin{cases} S^i \leftarrow f^{n_i+1}\left(S^{i-1}\right) \\ Q_i \leftarrow D_{M \times N}\left(S^i\right) \\ P_i \leftarrow i \\ j = (2 \times i) \bmod (L-1) \\ n_i \leftarrow 2 \times p_{j+1} + p_j \end{cases}$
6 **end**
7 **for** $i = 1$ *to* $M \times N$ **do**
8 $\quad \{$ swap(P_i, P_{Q_i})
9 **end**
10 **return** $P, S^{M \times N}$

Algorithm 2: The permutation algorithm (Permutation).

The location change pixels only breaks the correlation between adjacent pixels. As the pixels remain in the image itself, this keeps the histogram and does not increase the entropy. Therefore, the image may be vulnerable to statistical attacks, and for this, we need to mask the pixel values with pseudo-random sequences in the diffusion step.

4.3. Pseudo-random generator (diffusion)

Diffusion consists in modifying the values of the pixels, by masking pseudo-random sequences according to the Vernam encryption [44]. In this section, we propose a pseudo-random number generator based on the Vasicek process. We generate a set $\{d^e\}_{1 \le e < M \times N}$ with $0 \le d^e < 2^8$:

$$d^e = D_{256}(S^e),$$

where $S^e = f^{n_e+1}\left(S^{e-1}\right)$ with an initial state S^0. Between two sub-sequences, the process is iterated several times, which adds more randomness to the generated sequences. The pixels of the CI image after the confusion, will be masked with $\{d^e\}_{1 \le e < M \times N}$ where the CI pixels CI (floor $\left(\frac{e}{N}\right), e\%N$) will be masked with d^e according to the CBC mode of the encryption.

The generation of random sequences is described in detail in Algorithm (3).

Input: A secret key $K = (k_0 k_1 k_2 \dots k_{L-1})_2$, a state S_0 and the image size $M \times N$.
Output: A set of pseudo-random number sequences $\{d^e\}_{1 \leq e < M \times N}$ and a state $S^{M \times N}$
$l \leftarrow 0, n \leftarrow 2 \times k_1 + k_0, i \leftarrow 0, j \leftarrow 0$
$S^0 \leftarrow f^{1+n}(S_0)$
for $e = 1$ *to* $M \times N$ **do**
$$\begin{cases} S^e \leftarrow f^{1+n}\left(S^{e-1}\right) \\ d^e \leftarrow D_{256}\left(S^e\right) \\ l \leftarrow (2 \times e) \bmod (L-1) \\ n \leftarrow 2 \times k_{e+1} + k_e \end{cases}$$
end
return $\{d^e\}_{1 \leq e \leq M \times N}, S^{M \times N}$

Algorithm 3: Pseudo-random number generator (PRNG).

4.4. Encryption of a gray level image

We simulate the proposed algorithm for the image I (see Figure 4) at gray level of size $M \times N$. The input of the algorithm is: an image I of size $M \times N$, a secret key K and an integer CD indicating the number of confusion-diffusion rounds. The steps involved in the proposed algorithm are described in order and in detail as follows:

1. Calculate the parameters and the initialization state of the process from a secret key K
$$(x_0, r, X_0, \lambda, \mu, \sigma) = \text{ initialize } (K)$$
let S_0 be the initial state
$$S_0 = (x_0, X_0)\,.$$

2. Perform n_0 iterations for the three particles and get
$$S^0 = f^{n_0}(S_0) \text{ with } n_0 = \sum_{i=0}^{20} 2^i \times k_i.$$

3. Generate CD permutations $\{P_i\}_{1 \leqslant i \leqslant CD}$ and CD sets of random sequences $\left\{\{d_i^e\}_{1 \leq e < M \times N}\right\}_{1 \leqslant i \leq CD}$.
   - For $i = 1$ to CD:
     - Generate the permutation P_i
     $$\left(P_i, S_1^i\right) = \text{Permutation}\left(K, S^{i-1}, M \times N\right)$$
     - Generate a set of sub-suite $\{d_i^e\}_{1 \leq e < M \times N}$
     $$\left(\{d_i^e\}_{0 \leq e < M \times N - 1}, S^i\right) = GNA\left(K, S_1^i, M \times N\right).$$

4. Reshape the image I as an array ID^0 of size $M \times N$

$$ID^0[i \times N + j] = I(i,j) \text{ with } 0 \leq i < M \text{ and } 0 \leq j < N.$$

5. Perform the confusion-diffusion round CD on the image DI^0 as shown in the following loop:
   - for $i = 1$ to CD, we perform the following operations:
     - For $e = 0$ to $M \times N - 1, DI^0$
       $CI^{i-1}[e] = DI^{i-1}[P_i(e)]$
     - Encrypt the CI^{i-1} data according to the CFB mode to haveDI^{i-1}.
       $DI^i[0] = CI^{i-1}[0] \oplus d_i^0$
       for $e = 1$ to $M \times N - 1$
       $DI^i[e] = CI[e] \oplus d_i^e \oplus DI^i[e-1]$.

6. Convert TD^{CD} into an encrypted digital image C of dimension $M \times N$, where the value of the pixel $C(i,j) = TD^{CD}[i \times N + j]$.

5. SECURITY ANALYSIS

We simulate our algorithm using Python. The two images tested are "Lena" and "Baboon" with image size 256×256 and in 256 gray levels. The simulation result after a round of the confusion-diffusion process is shown in Figure 4.

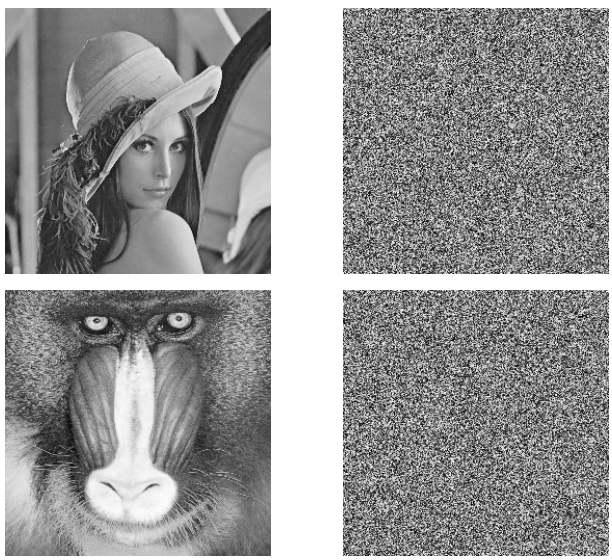

Fig. 4: Lena and Baboon original and encrypted images by our algorithm.

The various security aspects are described in detail in the following paragraphs.

5.1. The key space

The proposed crypto-system is characterized by a string of arbitrary length as key. The initialization parameters are generated by a technique that covers all the bits of the ASCII representation of the key. As a result, the proposed crypto-system is impossible to be attacked exhaustively.

5.2. Histogram analysis

Histogram analysis displays the frequency of pixel values in an image. However, the histogram of an encrypted image with a good scheme of encryption, must be uniformly distributed and different from those of the original image to avoid statistical attacks. Figure 5 shows the histograms of the original and encrypted images. The frequency of the pixel values over the 256 gray levels are uniformly distributed, so the encrypted image does not reveal any statistical information about the original image, and so statistical attacks are infeasible against our algorithm.

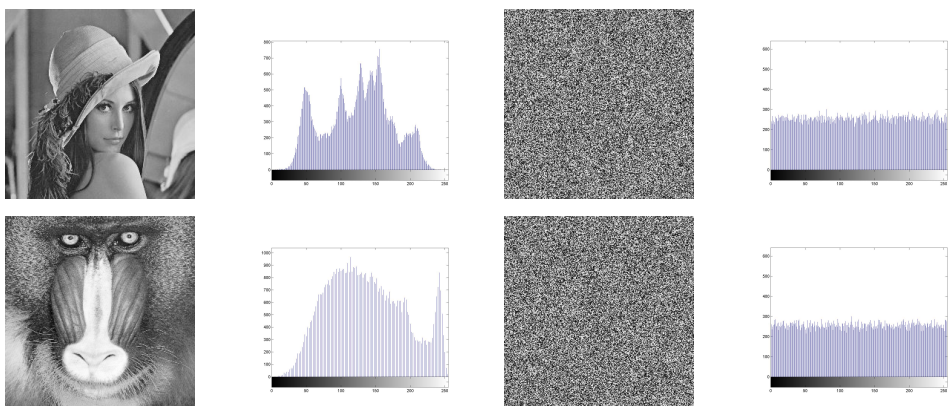

Fig. 5: Lena and Baboon original and encrypted images by our algorithm with their histograms.

5.3. Entropy analysis

Entropy [32] is defined to express the degree of uncertainty or randomness in a cipher system. The entropy of image information is a criterion that measures the effect of an image encryption algorithm. It reflects whether the distribution of pixel values is random or not. The entropy H of a gray image is calculated by the following equation

$$H(s) = \sum_{i=0}^{gl} P(s_i) \log \left(\frac{1}{P(s_i)} \right),$$

where

$$P\left(s_i\right) = \frac{\text{number of pixels at level } s_i}{M \times N}$$

with gl the gray level of the image and $P(s_i)$ the probability of the appearance of the level s_i in the image of size $M \times N$.

For an encrypted image of gray level $gl = 256$, the ideal value of the information entropy of the image should be closer to $8 = \log_2 256$.

In our simulation, we calculated the information entropy of the encrypted images by our algorithm for a single and two rounds. Table 2 summarizes the values obtained for the entropy of the encrypted images.

round	1	2
The entropy (Lena)	7.997	7.999
The entropy (Baboon)	7.997	7.998

Tab. 2: Entropies on one and two rounds.

The values are all very close to 8, and therefore our algorithm provides high security and has very good encryption performance.

5.4. Correlation analysis of two adjacent pixels

Adjacent pixels in an original image are highly correlated. Indeed, we have randomly selected PA pairs of adjacent pixels, according to the horizontal direction $(r, s)/(r+1, s)$, vertical $(r, s)/(r, s+1)$, or diagonal $(r, s)/(r+1, s+1)$ of the image. We compute and compare the correlation coefficients using the following formula

$$r(u, v) = \frac{\operatorname{cov}(u, v)}{\sqrt{\operatorname{Var}(u)} \times \sqrt{\operatorname{Var}(v)}}$$

where u and v are the respective values of the two adjacent pixels, with

$$\operatorname{cov}(u, v) = E((u - E(u)) \times ((v - E(v))))$$

$$E(u) = \frac{1}{PA} \sum_{i=1}^{PA} u_i \text{ and } \operatorname{Var}(u) = \frac{1}{PA} \sum_{i=1}^{PA} (u_i - E(u))^2 .$$

Adjacent pixels in a natural image can have a correlation coefficient close to 1 i. e., adjacent pixels have similar values. Whereas, the pixels of an encrypted image should be close to 0. The pixels should be distributed randomly. Therefore, an image encryption algorithm must effectively reduce the correlation between adjacent pixels.

In our study we chose $PA = 2000$ pixels, a fairly representative number for an image size 256×256. We compared the correlations of the pixels of the original images in Figure 4 and those of the encrypted image in Figure 5 on the chosen sample.

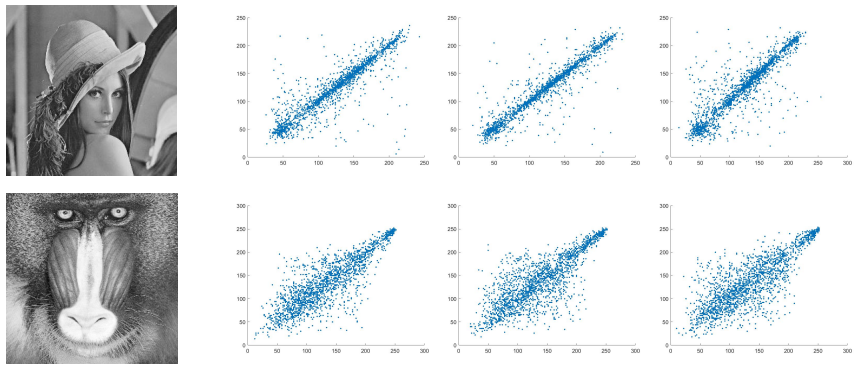

Fig. 6: Correlation distribution of adjacent pixels of original images in directions, Horizontal, Vertical and Diagonal.

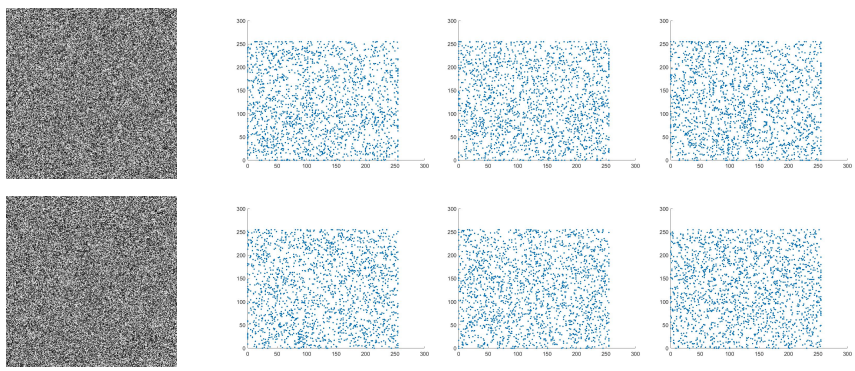

Fig. 7: Correlation distribution of adjacent pixels of encrypted images in directions, Horizontal, Vertical and Diagonal.

It is clear that the dots are located along the diagonal (Figure 6) indicating a high correlation of adjacent pixels in all three directions, while those in the encrypted image are scattered over the entire plane (Figure 7), which indicates that the correlation is strongly reduced. Correlation analysis proves that our algorithm successfully reduces correlation.

5.5. The differential attack

A differential attack involves making a change, usually a pixel, in the clear image and comparing the encrypted images obtained by the same key, and finding out if there is a relationship between the clear image and the encrypted image, which can further facilitate the determination of the secret key. An encryption scheme is invulnerable to differential attacks, if a minor modification of the original image causes significant changes in the encrypted image. The two most common criteria for measuring the ability to differential attack are: the number of pixel change rates ($NPCR$) and the unified

modified mean intensity ($UACI$). The $NPCR$ and the $UACI$ between two encrypted images C_1 and C_2 of the same size $M \times N$ are defined by

$$\mathrm{UACI}\,(C_1, C_2) = \frac{\sum_{j=1}^{M}\sum_{k=1}^{N}\frac{|C_1(j,k)-C_2(j,k)|}{L-1}}{M \times N} \times 100\%$$

$$\mathrm{NPCR}\,(C_1, C_2) = \frac{\sum_{j=1}^{M}\sum_{k=1}^{N} D_{(C_1,C_2)}(j,k)}{M \times N} \times 100\%$$

where $C_1(j,k)$ is the pixel corresponding to the jth row and the kth column of the image C_1 and $D_{(C_1,C_2)}(j,k)$ is a function defined as follows

$$D_{(C_1,C_2)}(j,k) = \begin{cases} 1 & \text{if} \quad C_1(j,k) = C_2(j,k) \\ 0 & \text{if} \quad C_1(j,k) \neq C_2(j,k). \end{cases}$$

Recently, $UACI$'s $NPCR$ values for two random images, which is an expected estimate for a good cryptographic system, are proven in [11] and given by

$$NPCR_{\text{expected}} = \left(1 - \frac{1}{2^{\log_2 gl}}\right) \times 100\%$$

and

$$UACI_{\text{expected}} = \frac{1}{gl^2}\left(\frac{\sum_{i=1}^{gl-1} i(i+1)}{gl-1}\right) \times 100\%.$$

So for a gray level $gl = 256$, we can have $NPCR_{\text{expected}} = 99.609\%$ and $UACI_{\text{expected}} = 33.464\%$.

In our analysis, we change the first pixel in the image $I_{Lena,1}$ (Lena) and $I_{Baboon,1}$ (Baboon) of Figure 4, to get $I_{Lena,2}$ and $I_{Baboon,2}$, then we encrypt the four images by the same key through a round of the confusion-diffusion process to have $C_{Lena,1}$, $C_{Lena,2}$, $C_{Baboon,1}$ and $C_{Baboon,2}$.

Images	NPCR	UACI
$C_{Lena,1}$ and $C_{Lena,2}$	99.187%	33.351%
$C_{Baboon,1}$ and $C_{Baboon,2}$	99.574%	33.249%

Tab. 3: The values of NPCR and UACI obtained.

The values are close to the expected values, which indicates that the proposed algorithm is resistant to differential attacks.

6. CONCLUSION

In conclusion, we proposed in this paper a new image encryption algorithm based on a Vasicek process and the logistic chaotic map. In our work, we adopted Fridrich's structure to encrypt images. For the security-analysis, a various methods have been employed, including histogram analysis, correlation analysis, resisting differential attack analysis and information entropy. The corresponding experimental results have shown that the proposed cryptographic system has a high level of security and has important practical applications in securing digital images.

(Received July 7, 2025)

REFERENCES

[1] A. Akhshani, A. Akhavan, S.-C. Lim, and Z. Hassan: An image encryption scheme based on quantum logistic map. Commun. Nonlinear Sci. Numer. Simul. *17* (2012), 4653–4661. DOI:10.1016/j.cnsns.2012.05.033

[2] G. Alvarez and S. Li: Some basic cryptographic requirements for chaos-based cryptosystems. Int. J. Bifurcation Chaos *16* (2006), 2129–2151. DOI:10.1142/S0218127406015970

[3] G. Alvarez, F. Montoya, M. Romera, and G. Pastor: Cryptanalysis of an ergodic chaotic cipher. Phys. Lett. A *311* (2003), 172–179. DOI:10.1016/S0375-9601(03)00469-9

[4] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche: Sponge-based pseudo-random number generators. In: Cryptographic Hardware and Embedded Systems (CHES), Springer, Berlin 2010, pp. 33–47.

[5] T. Björk: Arbitrage Theory in Continuous Time. Oxford Univ. Press, Oxford 2009.

[6] K. Charif, A. Drissi, and Z. E. A. Guennoun: A pseudo random number generator based on chaotic billiards. A. A. *3* (2017), 2.

[7] K. Charif and Z. E. A. Guennoun: A novel image encryption algorithm based on chaotic billiards. J. Discrete Math. Sci. Cryptogr. *24* (2021), 129–154.

[8] S. Chen and J. Lü: Parameters identification and synchronization of chaotic systems based upon adaptive control. Phys. Lett. A *299* (2002), 353–358. DOI:10.1016/S0375-9601(02)00522-4

[9] P. P. Deepthi, V. S. Nithin and P. S. Sathidevi: Implementation and analysis of stream ciphers based on elliptic curves. Comput. Electr. Eng. *35* (2009), 300–314. DOI:10.1134/S0361768809060024

[10] J. Fridrich: Symmetric ciphers based on two-dimensional chaotic maps. Int. J. Bifurcation Chaos *8* (1998), 1259–1284.

[11] C. Fu, J. J. Chen, H. Zou, W. H. Meng, Y. F. Zhan, and Y. W. Yu: A chaos-based digital image encryption scheme with an improved diffusion strategy. Opt. Express *20* (2012), 2363–2378. DOI:10.1364/OE.20.002363

[12] M. Hamdi, R. Rhouma, and S. Belghith: Selective compression-encryption of images based on SPIHT coding and Chirikov standard map. Signal Process. *131* (2017), 514–526. DOI:10.1016/j.sigpro.2016.09.011

[13] D. J. Higham: An algorithmic introduction to numerical simulation of stochastic differential equations. SIAM Rev. *43* (2001), 525–546. DOI:10.1137/S0036144500378302

[14] S. P. Indrakanti and P. S. Avadhani: Permutation based image encryption technique. Int. J. Comput. Appl. *28* (2011), 45–47. DOI:10.1007/s10778-011-0441-6

[15] A. Jolfaei, X.-W. Wu ,and V. Muthukkumarasamy: On the security of permutation-only image encryption schemes. IEEE Trans. Inf. Forensics Security *11* (2015), 235–246. DOI:10.1109/TIFS.2015.2489178

[16] M. Joshi, C. Shakher, and K. Singh: Image encryption and decryption using fractional Fourier transform and radial Hilbert transform. Opt. Laser Eng. *46* (2008), 522–526.

[17] J. Khan, J. Ahmad, and S. O. Hwang: An efficient image encryption scheme based on Henon map, skew tent map and S-Box. In: ICMSAO 2015, IEEE, pp. 1–6.

[18] L. Kocarev, G. Jakimoski, T. Stojanovski, and U. Parlitz: From chaotic maps to encryption schemes. In: ISCAS (1998), IEEE, pp. 514–517.

[19] P. R. Krishna, C. V S. Teja, and V. Thanikaiselvan: A chaos-based image encryption using Tinkerbell map functions. In: ICECA (2018), IEEE, pp. 578–582.

[20] P. Langevin: Sur la théorie du mouvement Brownien. C. R. Acad. Sci. Paris *146* (1908), 530–533.

[21] C. Li, D. Arroyo, and K.-T. Lo: Breaking a chaotic cryptographic scheme based on composition maps. Int. J. Bifurcation Chaos *20* (2010), 2561–2568. DOI:10.1142/S0218127410027192

[22] C. Li, Y. Liu, L. Y. Zhang and M. Z. Q. Chen: Breaking a chaotic image encryption algorithm based on modulo addition and XOR operation. Int. J. Bifurcation Chaos 23 (2013), 1350075. DOI:10.1142/S0218127413500752

[23] S. Li, Q. Li, W. Li, X. Mou, and Y. Cai: Statistical properties of digital piecewise linear chaotic maps and their roles in cryptography and pseudo-random coding. In: Cryptography and Coding (IMA), Springer, Berlin (2001), pp. 205–221.

[24] M. K. Mandal, G. D. Banik, D. Chattopadhyay, and D. Nandi: An image encryption process based on chaotic logistic map. IETE Tech. Rev. *29* (2012), 395–404. DOI:10.4103/0256-4602.103173

[25] A. Mitra, Y. V. S. Rao and S. R. M. Prasanna: A new image encryption approach using combinational permutation techniques. Int. J. Comput. Sci. *1* (2006), 127–131.

[26] C. Nouar and Z. E. A. Guennoun: A pseudo-random number generator using double pendulum. Appl. Math. *14* (2020), 977–984. DOI:10.2147/OPTH.S237757

[27] F. Omary: Application des algorithmes évolutionnistes à la cryptographie. Univ. Mohammed V-Agdal, Rabat 2006.

[28] D. Pandey, U. S. Rawat, and A. Kumar: Robust progressive block based visual cryptography with chaotic map. J. Discrete Math. Sci. Cryptogr. *19* (2016), 1025–1040. DOI:10.1080/09720529.2015.1132040

[29] H. E. Papadopoulos and G. W. Wornell: Maximum-likelihood estimation of a class of chaotic signals. IEEE Trans. Inf. Theory *41* (1995), 312–317. DOI:10.1109/18.370091

[30] N. K. Pareek, V. Patidar, and K. K. Sud: Image encryption using chaotic logistic map. Image Vision Comput. *24* (2006), 926–934. DOI:10.1016/j.imavis.2006.02.021

[31] W. Schindler: Random number generators for cryptographic applications. In: Cryptographic Engineering, Springer, Berlin (2009), pp. 5–23.

[32] C. E. Shannon: A mathematical theory of communication. Bell Syst. Tech. J. *27* (1948), 379–423. DOI:10.1002/j.1538-7305.1948.tb01338.x

[33] C. E. Shannon: Communication theory of secrecy systems. Bell Syst. Tech. J. *28* (1949), 656–715. DOI:10.1002/j.1538-7305.1949.tb00928.x

[34] M. Sharma, M. Kowar, and M. Sharma: An improved evolutionary algorithm for secured image using adaptive genetic algorithm. J. Discrete Math. Sci. Cryptogr. *11* (2008), 673–683. DOI:10.1080/09720529.2008.10698397

[35] A. Skrobek: Cryptanalysis of chaotic stream cipher. Phys. Lett. A *363* (2007), 84–90. DOI:10.1016/j.physleta.2006.10.081

[36] A. Skrobek: Approximation of a chaotic orbit as a cryptanalytical method on Baptista's cipher. Phys. Lett. A *372* (2008), 849–859. DOI:10.1016/j.physleta.2007.08.041

[37] G. E. Uhlenbeck and L. S. Ornstein: On the theory of the Brownian motion. Phys. Rev. *36* (1930), 823. DOI:10.1103/PhysRev.36.823

[38] O. Vasicek: An equilibrium characterization of the term structure. J. Financ. Econ. *5* (1977), 177–188.

[39] X. Wang and D. Xu: A novel image encryption scheme based on Brownian motion and PWLCM chaotic system. Nonlinear Dyn. *75* (2014), 345–353. DOI:10.1007/s11071-013-1070-x

[40] X. Wu, H. Hu, and B. Zhang: Parameter estimation only from the symbolic sequences generated by chaos system. Chaos Solitons Fractals *22* (2004), 359–366. DOI:10.1016/j.chaos.2004.02.008

[41] Y. Wu, Y. Zhou, S. Agaian, and J. P. Noonan: A symmetric image cipher using wave perturbations. Signal Process. *102* (2014), 122–131. DOI:10.1016/j.sigpro.2014.03.015

[42] G. Xiao, M. Lu, L. Qin, and X. Lai: New field of cryptography: DNA cryptography. Chin. Sci. Bull. *51* (2006), 1413–1420. DOI:10.1007/s11434-006-2012-5

[43] W. Xiaofu and S. Songgeng: A general efficient method for chaotic signal estimation. IEEE Trans. Signal Process. *47* (1999), 1424–1428. DOI:10.1109/78.757236

[44] W. Zhen, H. Xia, L. Yu-Xia, and S. Xiao-Na: A new image encryption algorithm based on the fractional-order hyperchaotic Lorenz system. Chin. Phys. B *22* (2013), 010504. DOI:10.1088/1674-1056/22/1/010504

[45] C. Zhu, S. Xu, Y. Hu, and K. Sun: Breaking a novel image encryption scheme based on Brownian motion and PWLCM chaotic system. Nonlinear Dyn. *79* (2015), 1511–1518. DOI:10.1007/s11071-014-1757-7

Hajar Ahalli, Dept. of Mathematics, Faculty of Sciences, Mohammed First University, BP 717, 60000, Oujda. Morocco.
e-mail: ahallihajar@gmail.com

Abderrahim Aslimani, Dept. of Mathematics, Multidisciplinary Faculty, Mohammed First University, BP 300, Selouane, 62700 Nador. Morocco.
e-mail: a.slimani@ump.ac.ma

Khalid Charif, Dept. of Mathematics, Faculty of Sciences, Mohammed V University, BP 1014 RP, Rabat. Morocco.
e-mail: Khalidcharif@gmail.com

Mohamed El Ouafi, Dept. of Mathematics, Multidisciplinary Faculty, Mohammed First University, BP 300, Selouane, 62700 Nador. Morocco.
e-mail: mohamed.elouafi10@ump.ac.ma