Encryption and decryption of images with chaotic map lattices

A. N. Pisarchik, N. J. Flores-Carmona, and M. Carpio-Valadez

Citation: Chaos: An Interdisciplinary Journal of Nonlinear Science **16**, 033118 (2006); doi: 10.1063/1.2242052
View online: http://dx.doi.org/10.1063/1.2242052
View Table of Contents: http://scitation.aip.org/content/aip/journal/chaos/16/3?ver=pdfcov
Published by the AIP Publishing

Encryption and decryption of images with chaotic map lattices

A. N. Pisarchik
Centro de Investigaciones en Optica, Loma del Bosque 115, Lomas del Campestre, 37150 Leon, Guanajuato, Mexico

N. J. Flores-Carmona and M. Carpio-Valadez
Instituto Tecnologico de Leon, Av. Tecnologico, Julian de Obregon, 37290 Leon, Guanajuato, Mexico

(Received 24 April 2006; accepted 30 June 2006; published online 30 August 2006)

We propose a secure algorithm for direct encryption and decryption of digital images with chaotic map lattices. The basic idea is to convert, pixel by pixel, the image color to chaotic logistic maps one-way coupled by initial conditions. After small numbers of iterations and cycles, the image becomes indistinguishable due to inherent properties of chaotic systems. Since the maps are coupled, the image can be completely recovered by the decryption algorithm if map parameters, number of iterations, number of cycles, and the image size are exactly known. © *2006 American Institute of Physics*. [DOI: 10.1063/1.2242052]

The use of chaos in cryptography is of great interest in many areas, including a database, Internet transaction banking, software, and protection of communication channels, because of the high sensitivity of chaotic systems to initial conditions and parameters, which implies strong cryptographic properties of chaotic cryptosystems and makes them robust against any statistical attacks. Although significant achievements have been reached in this field, there are still many problems that restrict the application of existing encoding/decoding algorithms to real systems. New improvements in communication technologies try to apply discrete nonlinear dynamical systems. Most traditional cryptosystems are based on a single map [i.e., a zero-dimensional (0D) system] or on a sequence of maps (1D system) and utilize a block encryption technique that is appropriate for encryption of plaintext files or blocks of bits. However, they do not allow the direct encryption of images because the latter requires the use of spatially extended dynamical systems (i.e., 2D systems). In this paper, we design a new practical algorithm based on a chaotic map lattice (CML) that allows direct encryption and decryption of color digital images. We test our algorithm with a real color image and prove that our cryptosystem incorporates necessary cryptographic properties inherent to a good cryptosystem. These are (i) high sensitivity to any changes in the image, (ii) high sensitivity to secret keys, (iii) absence of any patterns in the encoded image, and (iv) being robust against cryptographic attacks. We also discuss the possibilities for improving our cryptosystem with further developments of computer techniques and coupled schemes of the CMLs.

I. INTRODUCTION

In the past decade, methods and ideas from the theory of dynamical systems and chaos have gained wide attention in applications to communication and cryptography.[1–6] Cryptography is the art of secret writing. The main purpose of cryptography is to develop a cryptosystem that converts an original message into a nonreadable message and then recovers the message back in its original form. This involves information transformation to apparent understandable garbage so that nonauthorized people cannot understand the message.[7] High sensitivity of chaotic systems to initial conditions and parameters implies strong cryptographic properties of chaotic cryptosystems that makes them robust against any statistical attacks. Therefore, the use of chaos in cryptography is of great interest in many areas, including a database, Internet transaction banking, software, and protection of communication channels, in order to preserve confidential data from indiscretion attacks of enemies, spies, interceptors, opponent, cryptanalysts, etc.[8,9]

Chaotic communication schemes are based either on discrete or continuous systems. Many cryptosystems based on continuous systems utilize the idea of synchronization of chaos.[1–4,10] However, recent studies show that the performance of these communication schemes is very poor and most models of chaos communications are insecure.[11–13] The insecurity results mainly from the insensitivity of synchronization to system parameters.[14] Recently, much attention has been given to chaotic communication schemes based on discrete systems. Most discrete chaotic cryptographic algorithms explore one or more chaotic maps as pseudorandom number generators producing a binary stream that is used for encryption of a plain text to produce a cipher text.[5,15–19] The initial conditions or parameters or both are usually used as secret keys. The existing algorithms utilize a block encryption technique that allows the encryption of plaintext files (blocks of bits), however they do not allow the direct encryption of images. The latter requires the use of spatially extended dynamical systems, e.g., 2D map lattices. The CML was introduced by Kaneko[20] as a simple model capturing the essential features of spatiotemporal dynamics of extended nonlinear systems and later was used for modeling complex spatial phenomena in diverse areas of science and engineering.[20] Recently, a one-way coupled map lattice was proposed for cryptography of a self-synchronizing stream

cipher.[14,21] Wang *et al.*[22] have shown that the communication with the CMLs is more secure than the communication with a single map, because chaotic discrete systems always generate periodic time series due to finite precision of computer calculations; however, the period increases exponentially with the number of coupled maps.

In this paper, we propose a new secure cryptosystem based on CMLs. Our cryptosystem is different from that proposed by Wang *et al.*[14] because (i) it does not utilize a block encryption technique and (ii) it does not require synchronization of a receiver with a transmitter (master-slave synchronization). Our cryptosystem utilizes only the essence of chaos: high sensitivity of a chaotic trajectory to initial conditions and to a system parameter, confinement of the motion to a finite region of the phase space, and recurrence properties of a chaotic trajectory (i.e., any trajectory originating inside the attractor remains within it and visits all points of the attractor in infinite time). This work is the first attempt, to our knowledge, of exploring the CML in a cryptosystem for direct encryption and decryption of digital images.

The paper is organized as follows. In Sec. II we describe the chaotic map lattice and our encryption/decryption algorithm. In Sec. III we prove our method with a computer experiment by encoding a real color image and study sensitivity of our cryptosystem to secret keys. In Sec. IV we analyze the security of our cryptosystem. Finally, the main results are summarized in Sec. V.

II. CHAOS-BASED CIPHER

The logistic map is one of the simplest nonlinear chaotic discrete systems known as

$$x_{n+1} = a x_n (1 - x_n), \tag{1}$$

where x_n and a are the system variable and parameter, respectively, and n is the number of iterations. For $3.57 < a < 4$, the map Eq. (1) is chaotic. The main idea of our cryptosystem is that any image can be represented as a lattice of pixels, each of which has a particular color. The pixel color is the combination of three components: red, green, and blue, each of which takes an integer value $C=(C_r, C_g, C_b)$ between 0 and 255. Thus, we can create three parallel CMLs by converting each of these three color components to the corresponding values of the map variable, $x_c=(x_c^r, x_c^g, x_c^b)$, and use these values as the initial conditions, $x_c=x_0$. Starting from different initial conditions, each chaotic map in the CMLs, after a small number of iterations, yields a very different value from the initial conditions, and hence the image becomes indistinguishable, because of an exponential divergence of chaotic trajectories. In order to be able to recover the image, the maps are coupled by the initial conditions, i.e., the initial condition x_0^i of map i depends on the final variable of the previous map $i-1$ after n iterations, x_n^{i-1}, and contains information about the pixel color.

The process of developing the chaos-based cipher can be summarized as follows. Let $a=3.9$; for this parameter the chaotic attractor occupies the phase space between $x_{\min}=0.095\,062$ and $x_{\max}=0.975$. To convert the color components C of each pixel to the variable x_c of the corresponding map in the lattice, we use the following transformation:

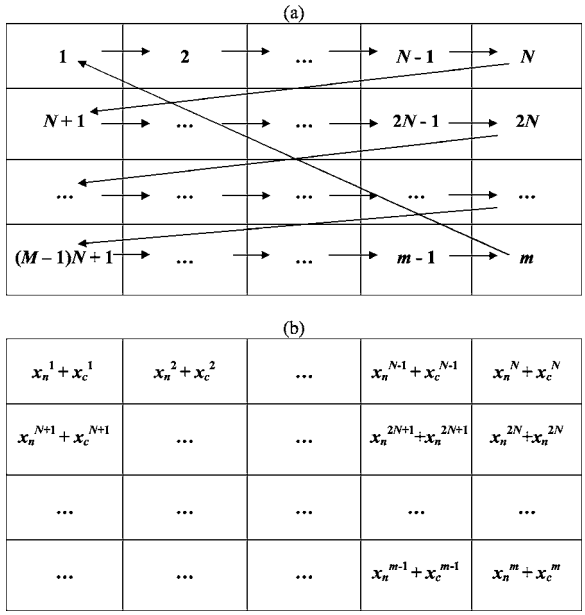

FIG. 1. (a) Indices for image pixels and (b) encoded variables. The arrows indicate coupling directions.

$$x_c = x_{\min} + \delta x (C/255), \tag{2}$$

where $\delta x = x_{\max} - x_{\min}$. To extract the value of the color component, we apply the inverse function

$$C = \text{round}[(x_n - x_{\min}) 255/\delta x]. \tag{3}$$

Equation (3) allows us to transform any state x_n of the logistic map Eq. (1) into a value between 0 and 255 and thus to visualize the pixel's color.

A. Encryption algorithm

The encryption algorithm includes the following steps.

(i) Let an image contain $N \times M = m$ pixels ($i = 1, 2, \ldots, m$) as shown in Fig. 1(a). Three color components of pixel i are converted to three values of the variable x_c with Eq. (2). For example, if the color values of the pixel i are $C^i=64$, 121, and 176 for red, green, and blue components, respectively, we obtain $x_c^i=0.315\,909\,654$, $0.512\,601\,554$, and $0.702\,392$.

(ii) The color value x_c^m of the last map m is used as the initial condition for the first map ($i=1$), i.e., $x_0^1=x_c^m$.

(iii) After n iterations of the first map, we obtain the map variable x_n^1 and add to this value the color value of the pixel x_c^1. The sum value is used as the initial condition for the subsequent map, i.e., $x_0^2=x_n^1+x_c^1$.

(iv) We iterate all maps subsequently starting from the first map and going through all image pixels, pixel by pixel, toward the last map, as shown in Fig. 1(a). In order to obtain always a stable solution and exclude transients, the trajectory should be initiated inside the chaotic attractor, i.e., $x_0 \in [x_{\min}, x_{\max}]$. Therefore, if the sum $x_n^i + x_c^i > x_{\max}$, we subtract δx, i.e., $x_0^{i+1} = x_n^i + x_c^i - \delta x$. After one cycle, going from the first map to the last one, we obtain the map lattice shown in Fig. 1(b), which can be visualized by converting the new map variables $x_n^i + x_c^i$ to the corresponding color values by using Eq. (3).

(v) We repeat steps (iii) and (iv) and make several cycles. For the next cycle, the new color value of the last map, $x_c^m(j)=x_n^m(j)+x_c^m(j)$ (j being the number of the cycle), is used as the initial condition for the first map to start the next cycle, i.e., $x_0^1(j+1)=x_c^m(j)$. After j cycles, we obtain the map lattice similar to that shown in Fig. 1(b) and can visualize the encoded image with the use of Eq. (3).

(vi) We repeat all steps for each color component (red, green, and blue) and superimpose the three images.

The encryption algorithm can be summarized as follows:

$$x_0^i(j)=x_c^m(j-1) \quad \text{if } i=1, \tag{4}$$

$$x_0^{i+1}(j)=x_c^i(j) \quad \text{if } i>1, \tag{5}$$

$$x_c^i(j)=x_n^i(j-1)+x_c^i(j-1) \quad \text{if } x_n^i(j-1)+x_c^i(j-1)\leqslant x_{\max}, \tag{6}$$

$$x_c^i(j)=x_n^i(j-1)+x_c^i(j-1)-\delta x \quad \text{if } x_n^i(j-1)+x_c^i(j-1)>x_{\max}. \tag{7}$$

B. Decryption algorithm

The encoded image is converted with Eq. (2) to the map lattice $x_c^i(j)$. For decryption, we need to recover the original image cycle by cycle in the reverse direction starting from the last map m and going to the first map ($i=1$) by making the same number of iterations for each map as for the encryption. The decryption algorithm includes the following steps:

(i) First, we need to recover the image of the $j-1$ cycle. We start from the last map. The encoded value of the penultimate map in the j cycle is the initial condition for the last map in the $j-1$ cycle, i.e., $x_0^m(j-1)=x_c^{m-1}(j)$. Starting from this initial condition, we iterate the last map n times and obtain the value $x_n^m(j-1)$. The number of iterations should be the same as for the encryption process. Subtracting this value from the color value of the last map in the j cycle, $x_n^m(j-1)+x_c^m(j-1)$, we get the color value of the last map in the $j-1$ cycle, i.e., $x_c^m(j-1)$.

(ii) Then, we take the encoded value of the map $m-2$ as the initial condition for the map $m-1$ and find the color value of the latter map in the cycle $j-1$ and so on.

(iii) We repeat step (ii) for each map in the reverse direction from the last map to the first map and reconstruct the image of the cycle $j-1$. Note that it is not the original image.

(iv) To reconstruct the color value of the first map ($i=1$) in the $j-1$ cycle, we use the color value of the last map m in the $j-1$ cycle, $x_c^m(j-1)$, as the initial condition for the first map.

(v) We repeat all previous steps j times and obtain the map lattice $x_c^i(0)$, which is converted by Eq. (3) to the color values C^i.

(vi) We repeat all steps for each color component (red, green, and blue) and superimpose the three images to get the original image.

The decryption algorithm can be summarized as follows:

$$x_0^i(j-1)=x_c^{i-1}(j) \quad \text{if } i>1, \tag{8}$$

$$x_0^i(j-1)=x_c^m(j-1) \quad \text{if } i=1, \tag{9}$$

$$x_c^i(j-1)=x_c^i(j)-x_n^i(j-1) \quad \text{if } x_c^i(j)-x_n^i(j-1)\geqslant 0, \tag{10}$$

$$x_c^i(j-1)=x_c^i(j)-x_n^i(j-1)+\delta x \quad \text{if } x_c^i(j)-x_n^i(j-1)<0. \tag{11}$$

Thus, our cryptographic algorithm has four secret keys: system parameters, number of iterations, number of cycles, and the image size. For higher security, each map can have a differing parameter and a differing number of iterations. In the following section, we analyze the security of our cryptographic system and demonstrate its robustness against cryptographic attacks.

III. COMPUTER EXPERIMENT

Conventional cryptography deals with binary streams and utilizes the terms *plaintext* (the original text to be encoded) and *ciphertext* (the encoded text). A good cryptosystem should incorporate the following features: (i) sensitivity with respect to a plaintext (slight modification in the plaintext creates completely different ciphertext), (ii) sensitivity with respect to keys (change in a secret key produces a completely different ciphertext), and (iii) mapping a plaintext to a random ciphertext (no patterns in the ciphertext). Since our cryptosystem does not deal with binary streams and since both the original image and encoded image are digital, we will use the terms *original image* and *encoded image* instead of "plaintext" and "ciphertext." In the following, we will demonstrate that our cryptosystem combines all cryptographic properties inherent to a good cryptosystem.

As we mentioned above, our cryptosystem has four secret keys: the system parameter a, the number of iterations n, the number of cycles j, and the image size $m=N\times M$. Here, we consider the sensitivity of our cryptosystem with respect to the secret keys. The original image ($N\times M=455\times 569$ pixels) is shown in Fig. 2(a).

The sensitivity of our encryption algorithm to the number of iterations n is demonstrated in Figs. 2(b)–2(d), where we display the images encoded with $n=1$, 30, and 75, respectively. For visualization, the values of the map variables $x_c^i(j=3)$ are converted to the color numbers C^i by Eq. (3). No decryption algorithm Eqs. (8)–(11) are used. These figures illustrate a crucial dependence of chaotic trajectories on initial conditions. One can see that using only one iteration for each map, the image can still be distinguished even after three cycles [Fig. 2(b)]. The use of 30 iterations makes the image almost indistinguishable, but the colors are not uniformly distributed [Fig. 2(c)]. However, for 75 iterations all colors are completely lost in the encoded image [Fig. 2(d)]. The number of iterations n is not as critical for the encryption/decryption time as the number of cycles j. For example, the times required for the encryption/decryption of the images shown in Figs. 2(b)–2(d) vary between 165 and 170 s.

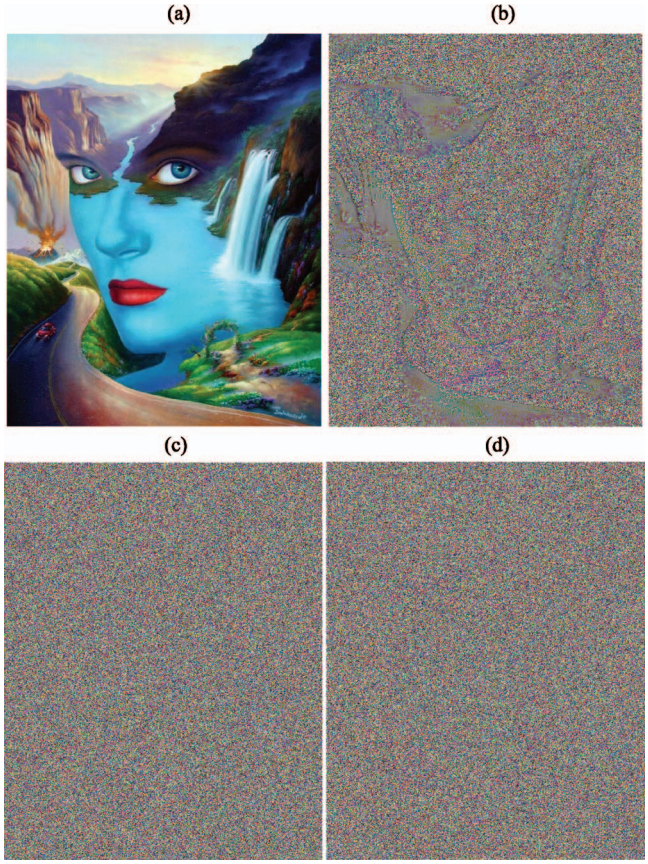

FIG. 2. (Color) Sensitivity to number of iterations. (a) Original image, (b) image encoded with $n=1$, (c) $n=30$, and (d) $n=75$. $a=3.9$ and $j=3$. The original image is the picture "Mother Nature in the new Millennium" courtesy of the artist Jim Warren.

The sensitivity of our encryption algorithm to the number of cycles j is demonstrated in Figs. 3(a) and 3(b), in which we display the images encoded with one and two cycles, respectively. As seen from the figures, only one or two cycles is not sufficient for secure encryption of the image because the original outline can still be distinguished in the encoded image. Therefore, we need to make at least three cycles to get an indistinguishable image, as shown in Fig. 2(d).

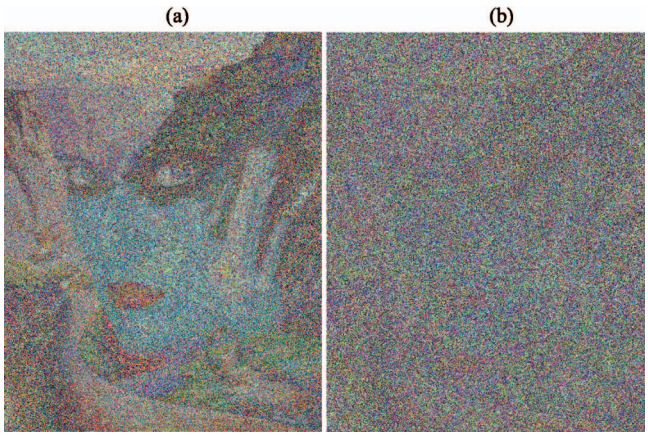

FIG. 3. (Color) Sensitivity to number of cycles. (a) Image encoded with $j=1$ and (b) $j=2$. $a=3.9$ and $n=75$.

TABLE I. Details of encryption/decryption time.

Image size ($N\times M$ pixels)	Time ($j=1$)	Time ($j=2$)	Time ($j=3$)
300×200	13.6	26.7	39.1
455×569	58.0	113.0	169.3
2400×1200	626.6	1255.9	1866.2

The larger n and j are, the higher is the security. However, with increasing n and j, the encryption/decryption time (EDT) also increases. To have reasonable EDT, we should balance between these two factors (security and time) to select adequate values for n and j. Moreover, EDT increases significantly with increasing m, as one can see from Table I, where we show EDT (in seconds) for three different image sizes and three different j. The calculations are made with a Pentium IV 3.0 GHz PC with 1.0 GB RAM and visualized with the program Microsoft Visual C#.NET 2005. One can see from Table I that to have higher resolution and security, we need to sacrifice time.

To prove the first property of a good cryptosystem, we slightly modify the original image with a black square at the right inferior corner [Fig. 4(a)]. The modified image encoded with 75 iterations and three cycles is shown in Fig. 4(b) and its histogram is plotted in Fig. 4(c). The histogram of the encoded original image is shown in Fig. 4(d). Although both encoded images [in Figs. 2(d) and 4(b)] are generated by the same keys, the distribution of their colors is completely different [compare Figs. 4(c) and 4(d)].

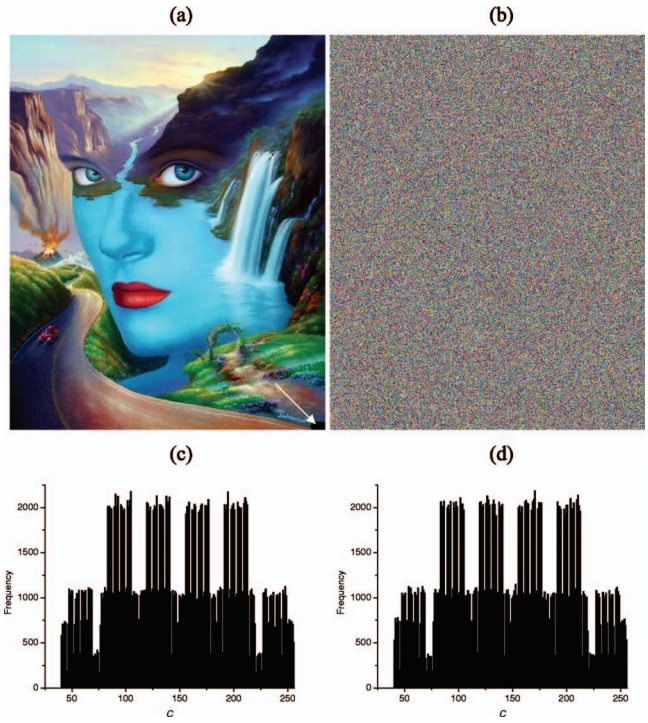

FIG. 4. (Color) Sensitivity with respect to plaintext. (a) Slightly modified image with a black square indicated by a white arrow, (b) encoded modified image, (c) histogram of encoded modified image, and (d) histogram of encoded original image shown in Fig. 2(d).

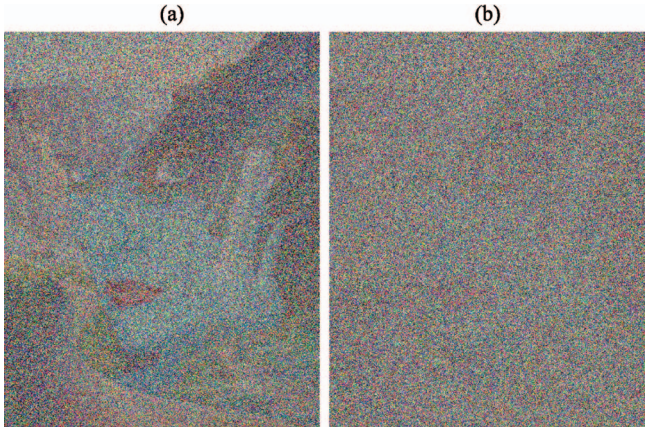

FIG. 5. (Color) Images decoded with $a=3.899\,999\,9$ and (a) $j=2$ and (b) $j=3$. The original image is encoded with $a=3.9$ and (a) $j=2$ and (b) $j=3$.

FIG. 6. (Color) Images decoded with (a) $n=74$ and (b) $n=75$. The original image is encoded with $n=75$.

Finally, our cryptosystem incorporates the third property of a good cryptosystem. The image encoded with a relatively small number of iterations and cycles does not display any patterns, as can be seen in Fig. 2(d).

IV. ROBUSTNESS AGAINST CRYPTOGRAPHIC ATTACKS

The encoded image shown in Fig. 2(d) is sent to other computers where the decryption algorithm Eqs. (8)–(11) is applied and where the original image is completely recovered if all secrete keys are exactly known. Thus, to recover the original image [Fig. 2(a)] from the encoded image [Fig. 2(d)], the receiver should know a, n, j, and m and the decryption algorithm. However, the encoded image can be exposed by indiscretion attacks. There are many attack models.[7] Among them we will consider here only the worse case, namely known-plaintext attack, where the attacker has samples of both the plaintext and its encrypted version and tries to reveal the secret keys. Classical ciphers are typically vulnerable to known-plaintext attack. Instead, our cryptosystem is highly resistant against this attack because the attacker simply trying all possible combinations of the secret keys will require too much time.

For example, let us suppose that the cryptoanalyst has the encoded image shown in Fig. 2(a) and its encoded image shown in Fig. 2(d). It is supposed that the attacker knows the encryption/decryption algorithm. The principal secret key of our cryptosystem is the system parameter a. Any changes in a result, first, in changes in the size of the chaotic attractor, δx, and, second, in changes in the map variables. Suppose that the attacker knows all secret keys except a. Let the decryption be performed with a_{dec} differing from the encryption parameter $a_{\mathrm{enc}}=3.9$ on 10^{-7}, i.e., $a_{\mathrm{dec}}=3.899\,999\,9$. In Figs. 5(a) and 5(b), we show the images resulting from decoding the CMLs, with this slightly different parameter, after two and three cycles, respectively. One can see from Fig. 5(a) that the original image encoded with only two cycles can be partially recovered with this wrong parameter. However, after three cycles, the error of 10^{-7} is too large and the image cannot be recovered [see Fig. 5(b)]. If we look at Table I, we see that EDT for our image (455×569 pixels) encoded and decoded with three cycles is 169.3 s. This is the test time for one fixed value of the parameter a. The variation of all possible values for a in the whole chaotic range $3.57<a<4$ with the step of 10^{-7} will require 22 years.

Another secret key, the number of iterations n, is also important. Due to an exponential divergence of chaotic trajectories from initial conditions, the error in only one iteration will result in a completely different image. In Fig. 6(a), we show the image decoded with wrong n. The original image was encoded with $n=75$ and decoded with $n=74$. The same image decoded with the correct number of iterations is shown in Fig. 6(b). However, the secret numbers n and j are not very secure and can be recovered relatively fast by the known-plaintext attack, if these values are not very large. The security of our cryptosystem can be improved not only by increasing the numbers of iterations and cycles but also by using different a and n for each map in each cycle. In this case, the cryptosystem will have as many secret keys as the number of pixels or the resolution of the image. The main advantage of this approach is that the increasing security does not lead to enlarging EDT.

Other attack models, such as the ciphertext-only attack, the chosen-plaintext attack, and the chosen-ciphertext attack, are less effective and therefore they are not dangerous for our cryptosystem. The last two models do not work because of high sensitivity of our cryptosystem to the number of pixels m. Due to the coupling of all maps, the lack of only one pixel produces a completely different encoded image. Even if all secret keys are known, the attacker should go through all image pixels to recover the original image.

V. CONCLUSIONS

In this paper, we have described a new cryptosystem for encoding and decoding digital images. Our cryptosystem is based on a chaotic map lattice. Chaotic maps are simple nonlinear dynamical systems that can display complex behavior. Our encryption/decryption algorithm explores the important properties of chaos: localization of a chaotic attractor in a particular region of the phase space, recurrence of a chaotic trajectory (it visits all points of the chaotic attractor in infinite time), and its high sensitivity to initial conditions

and parameters. The first two properties allow us to decode an infinite number of colors, and the second one provides a very high security of the chaotic cryptosystem, because after a small number of iterations the trajectory occurs far away from the initial state. We have demonstrated how a color image can be directly converted to lattices of chaotic logistic maps one-way coupled by initial conditions and how the image can be completely recovered by using the decryption algorithm. In our encryption algorithm, the coupling is necessary for both further decoding of the image and conserving information about the pixel's colors. Our encryption/decryption algorithm complies with all essential properties for a good cryptosystem and can be easily adapted to other chaotic maps (e.g., tent map, sine map, cubic map, etc.) including two-dimensional maps, such as the Hénon map or the Baker map.

The important problem in computer communications is communications in real time, as well as telecommunications. Unfortunately, the speed of modern personal computers is still not sufficient for communications in real time with our algorithm. However, we believe that progressing development of computer technology and further improvement of our algorithm will enable a decrease in the encryption/decryption time. The future possible research in this direction is the use of mutual (spacial) coupling instead of one-way coupling. This will make an image indistinguishable after smaller numbers of iterations and cycles. Other developments of cryptosystems based on CMLs might be the use of transmitter-receiver coupling, i.e., each map of a transmitter could be coupled with the corresponding map of a receiver, and then, to recover an image, we may apply a conventional chaotic communication technique based on complete synchronization. The latter approach does not require reverse calculations and therefore will save time and probably will allow computer communications and telecommunications in real time.

Finally, our encryption/decryption algorithm can be extended to a 3D chaotic map system, i.e., to a volume of maps. This will allow us to encode directly 3D images, such as holograms, which we believe will be used in communications in the future.

ACKNOWLEDGMENTS

This work was supported by the Mexican Council of Science and Technology (CONACYT), Project No. 46973. A.N.P. acknowledges support from the Spanish Ministry of Education and Science, Project No. SAB2004-0038. We thank Jim Warren for permission to use his picture in our work.

[1] L. M. Pecora and T. L. Carroll, Phys. Rev. Lett. **64**, 821 (1990).
[2] K. M. Cuomo and A. V. Oppenheim, Phys. Rev. Lett. **71**, 65 (1993).
[3] L. Kocarev and U. Parlitz, Phys. Rev. Lett. **74**, 5028 (1995).
[4] D. G. Van Wiggeren and R. Roy, Science **279**, 1198 (1998).
[5] N. K. Pareek, V. Patidar, and K. K. Sud, Phys. Lett. A **309**, 75 (2003).
[6] L. Kocarev, M. Sterjev, A. Fekete, and G. Vattay, Chaos **14**, 1078 (2004).
[7] A. J. Mendezes, P. C. V. Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography* (CRC Press, New York, 1997).
[8] L. Shujun, *Analyses and New Designs of Digital Chaotic Ciphers* (Xi'an Jiaotong University Press, Xi'an, 2003).
[9] D. Bishop, *Introduction to Cryptography with Java Applets* (Jones and Bartlett Publishers, Boston, 2003).
[10] S. Boccaletti, J. Kurths, G. Osipov, D. L. Valladares, and C. Zhou, Phys. Rep. **366**, 1 (2002).
[11] G. Perez and H. A. Cerdeira, Phys. Rev. Lett. **74**, 1970 (1995).
[12] K. M. Short and A. T. Parker, Phys. Rev. E **58**, 1159 (1998).
[13] C. Zhou and C. H. Lai, Phys. Rev. E **60**, 320 (1999).
[14] S. Wang, J. Kuang, J. Li, Y. Luo, H. Lu, and G. Hu, Phys. Rev. E **66**, 065202(R) (2002).
[15] Z. Kotulski and J. Szczepanski, Ann. Phys. (Paris) **6**, 381 (1997).
[16] M. S. Baptista, Phys. Lett. A **240**, 50 (1998).
[17] E. Alvarez, A. Fernandez, P. Garcia, J. Jimenez, and A. Marcano, Phys. Lett. A **263**, 373 (1999).
[18] K. W. Wong, S. W. Ho, and C. K. Yung, Phys. Lett. A **310**, 67 (2003).
[19] N. K. Pareek, V. Patidar, and K. K. Sud, Commun. Nonlinear Sci. Numer. Simul. **10**, 715 (2005).
[20] K. Kaneko and I. Tsuda, *Complex Systems: Chaos and Beyond: A Constructive Approach with Applications in Life Sciences* (Springer-Verlag, Berlin, 2001), and references therein.
[21] H. Lu, S. Wang, X. Li, G. Tang, J. Kuang, W. Ye, and G. Hu, Chaos **14**, 617 (2004).
[22] S. Wang, W. Liu, H. Lu, J. Kuang, and G. Hu, Int. J. Mod. Phys. B **18**, 2617 (2004).