Graphic cryptography with pseudorandom bit generators and cellular automata

Gonzalo Álvarez Marañón[1], Luis Hernández Encinas[1], Ascensión Hernández Encinas[2], Ángel Martín del Rey[3], and Gerardo Rodríguez Sánchez[2]

[1] Instituto de Física Aplicada, CSIC, C/ Serrano 144, 28006-Madrid, Spain,
{gonzalo, luis}@iec.csic.es,
WWW home page: http://www.iec.csic.es/~{gonzalo, luis}
[2] ETSII, Universidad de Salamanca, Av. Fernández Ballesteros 2,
37700-Béjar, Salamanca, Spain,
{ascen, gerardo}@usal.es,
[3] EPS, Universidad de Salamanca, C/ Sto. Tomás s/n, 05003-Ávila, Spain,
delrey@usal.es

Abstract. In this paper we propose a new graphic symmetrical cryptosystem in order to encrypt a colored image defined by pixels and by any number of colors. This cryptosystem is based on a reversible bidimensional cellular automaton and uses a pseudorandom bit generator. As the key of the cryptosystem is the seed of the pseudorandom bit generator, the latter has to be cryptographically secure. Moreover, the recovered image from the ciphered image has not loss of resolution and the ratio between the ciphered image and the original one, i.e., the factor expansion of the cryptosystem, is 1.

1 Introduction

As it is known, the goal of cryptography is to assure the secrecy and confidentiality of communications between two or more users, who use an insecure channel ([9], [10]). Opposite, the goal of cryptanalysis is to break the security and privacy of these communications. Here, we are interested in a special class of messages: colored images.

We propose to use cellular automata of dimension 2 and pseudorandom bit generators as a graphic symmetrical cryptosystem, that is, as a symmetrical cryptosystem to encrypt colored images. The proposed protocol begins with a message (the plaintext), uses an algorithm (based on a cellular automaton of dimension 2 and a pseudorandom bit generator), and ends with an encrypted message (the ciphertext). We wish that both messages, the plaintext and the ciphertext, belong to the set of colored images of the same size, i.e., the cipher-image will be defined by the same number of pixels than the plainimage. In this way, several applications to watermarking, steganography and subliminal cryptography, can be derived.

To date, there are several proposals for using images in cryptography. Some of them are based on dynamical systems ([6], [7]), but these proposals are difficult

to implement in practice due to the difference between the chaotic arithmetic defined by the dynamical systems and the discrete arithmetic of the computers. Other proposals for encrypting images are based on the use of compression methods in an iterated way ([4], [5]). In these cases, the problem is that the recovered image has less definition than the original one.

The foremost use of images in cryptography is done by means of the visual cryptography ([12], [15]). It is based on visual threshold schemes t of n, that is, the original image is divided in n shades. Each of them is photocopied in a transparency and then, the original image is recovered by superimposing any t transparencies, but no less. Moreover, no cryptographic protocol is used to recover it. Nevertheless, the recovered image also has less definition than the original one.

There are some applications of the previous protocols. Visual authentication and identification methods are proposed in [11]; the identification of documents and photograph signatures are presented in [1] and [13]; and visual secret sharing schemes for developing a method for intellectual property protection of grey level images in [3].

The rest of this paper is organized as follows: Cellular automata are briefly recalled in §2, and the main properties of pseudorandom bit generators are presented in §3. In §4 the proposal of a new graphic cryptosystem is presented, and finally, the conclusions are summarized in §5.

2 Cellular automata

A 2-*dimensional finite cellular automaton*, (CA for short), $\mathsf{A}= (L, S, V, f)$ ([14]), is a 4-uplet, where L is the *cellular space* formed by a 2-dimensional array of size $r \times s$ of identical objects, called *cells*. Each cell is denoted by $\langle i, j\rangle$, with $0 \leq i \leq r-1$, and $0 \leq j \leq s-1$. We denote by S the finite set of all possible values of the cells which is called the *state set*. As the state set is finite, we take $|S| = k$. Let $V \subset \mathbb{Z}^2$ be a finite ordered subset, called the *set of indices* of L, then for every cell $\langle i, j\rangle \in L$, its *neighborhood*, $V_{\langle i,j\rangle}$ is an ordered set of n cells defined as follows:

$$V_{\langle i,j\rangle} = \{\langle i+\alpha, j+\beta\rangle : \forall\, (\alpha, \beta) \in V\} \subset L. \tag{1}$$

Moreover, the *local transition function* $f : S^n \to S$ is a function which determines the evolution of the CA throughout the time, i.e., the changes of the states of every cell taking the states of its neighbors into account. Finally, as the CA considered are finite, we take periodic boundary conditions of the form:

$$a_{ij}^{(t)} = a_{kl}^{(t)} \Leftrightarrow i \equiv k \pmod{r} \quad \text{and} \quad j \equiv l \pmod{s}, \tag{2}$$

where $a_{ij}^{(t)} \in S$ stands for the state of the cell $\langle i, j\rangle$ at time t. Hence, the cellular space can be supposed as a 2-dimensional toroidal array. The matrix

$$C^{(t)} = \left(a_{ij}^{(t)}\right), \quad 0 \leq i \leq r-1, \quad 0 \leq j \leq s-1, \tag{3}$$

is called the *configuration* of A at time t. In particular, $C^{(0)}$ is called the *initial configuration* of the CA. Hence, the evolution of A is the sequence

$$\left(C^{(0)}, C^{(1)}, C^{(2)}, \ldots\right).$$

A cellular automaton is *reversible* (RCA for short) if there exists another CA, called its inverse, that computes the inverse evolution of A.

3 Pseudorandom bit generators

As is well known, a pseudorandom bit generator (PRBG) is a deterministic algorithm which, given a truly random bit sequence of length l, outputs a binary sequence of length $g \gg l$, which appears to be random. The input to the PRBG is called the seed, whereas the output is called a pseudorandom bit sequence. They are very used in cryptography, for example, to encrypt a plaintext by using a stream cipher, where the secret key is the seed. Hence, good random properties of the generator are convenient to prevent statistical attacks; but moreover, it is necessary that the generator must be cryptographically secure (CSPRBG). The security, in this sense, means that the probability that an algorithm can produce the next bit of a given sequence in a polynomial time, is negligible.

In particular, the BBS generator (see [2]) is defined by iterating the function $x^2 \pmod n$ on the set of the quadratic residues of integers modulo n, where n is a Blum integer, i.e., $n = p \cdot q$, where p and q are very large prime numbers, both congruent to 3 modulo 4. That is, starting from a seed x_0, and iterating $x_{i+1} \equiv x_i^2 \pmod n$, one obtains a binary sequence $b_i = \text{parity}(x_i)$. This PRBS is a CSPRBG under the assumption that the integer factorization is intractable. The length of the orbits of the BBS generator were characterized in [8], hence the BBS generator is a good option to be used in the graphic cryptosystem proposed below.

4 The graphic cryptosystem

In this section we propose a graphic symmetrical cryptosystem in order to encrypt a colored image. This cryptosystem is based on a reversible 2-dimensional CA and uses a cryptographically secure PRBG.

Let I be a colored image defined by $r \times s$ pixels, P_{ij}, where the pixels are denoted by $P_{ij} = (p_{ij}^1, p_{ij}^2, \ldots, p_{ij}^{24})$, with $p_{ij}^n \in \mathbb{Z}_2$, $1 \leq i \leq r$, $1 \leq j \leq s$, $1 \leq n \leq 24$, and c colors, with $2 \leq c \leq 2^{24}$. If P_{ij} is in the m-th position from the left top corner of the image, then $m = (i-1)s + j$, with $1 \leq m \leq r \times s$.

4.1 Key generation

The key of the graphic cryptosystem is the key used for the CSPRBG to generate the pseudorandom bit sequence. Then, the security of the proposed cryptosystem

is guaranteed by the security of the PRBG. This is the reason why a CSPRBG is used. Hence, to break this graphic cryptosystem it is necessary to determine the keys used by the pseudorandom bit generator. In the suggested generator (the BBS generator), this means to determine the prime numbers p and q, or the modulus $n = p \cdot q$, and to factorize it. Moreover, the session key k_0 is also necessary. But, to date this problem is infeasible computationally.

In our practical implementation we use the BBS pseudorandom generator. Hence, the key is the couple (n, K), where the number $n = p \cdot q$ is the product of two large primes numbers, each of them congruent to 3 modulo 4; and K is the seed used to generate the bit sequence (the session key). The modulo n is a long-term key, whereas the seed, i.e., K is a short-term key, that is, it must be changed for each session.

We denote by $B = (B_1, B_2, \ldots, B_{r\times s})$ the bit pseudorandom sequence of length $r \times s \times 24$ generated by the CSPRBG from the key K, where $B_m = (b_{m1}, b_{m2}, \ldots, b_{m24})$, and $b_{mn} \in \mathbb{Z}_2$.

4.2 Encryption

To construct the encryption protocol, we consider the RCA $\mathsf{A}= (L, S, V, f)$ defined by:

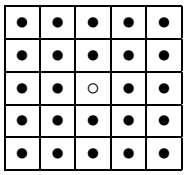
(i) The cellular space, L, is a rectangular array of size $r \times s$, i.e., L is the set of $r \times s$ pixels.

(ii) The state set is defined by $S = \mathbb{Z}_2 \times \overset{24}{\cdots} \times \mathbb{Z}_2$, with $|S| = 2^{24}$; hence, each color of the image I can be identified with an element in S. We denote by $x_m = (u_{m1}, \ldots, u_{m24})$ an element of S, where $u_{mn} \in \mathbb{Z}_2$, $1 \le n \le 24$.

(iii) The set of indices, V, is selected in a public way, such that $|V| = 24$. For example, if we consider the square of size 5×5 around the cell $\langle i, j\rangle$, except the cell itself, the set of indices is

$$V = \{(-2,-2), \ldots, (-2,2), \ldots, (0,-1), (0,1), \ldots, (2,-2), \ldots, (2,2)\}, \quad (4)$$

and the neighborhood, $V_{\langle i,j\rangle}$, can be viewed as follows:



This set can be represented by two variables h and w, $V_{\langle i,j\rangle} = \{(i+h, j+w)\}$, where:

$$h = \lfloor (n-1)/5 \rfloor - 2, \quad w = (n-1) \pmod 5 - 2, \quad 1 \le n \le 12, \quad (5)$$

$$h = \lfloor n/5 \rfloor - 2, \quad w = n \pmod 5 - 2, \quad 13 \le n \le 24. \quad (6)$$

(iv) To determine the encrypted pixel of P_{ij}, Q_{ij}, $1 \leq i \leq r$, $1 \leq j \leq s$, the 24 pixels of the neighbourhood of P_{ij}, $V_{\langle i,j \rangle}$: $P_{i-2,j-2}$, $\ldots$, $P_{i-2,j+2}$, $\ldots$, $P_{i,j-1}$, $P_{i,j+1}$, $\ldots$, $P_{i+2,j+2}$, are taken, and the transition function $f : S^{24} \to S$ is applied to them as follows:

$$\begin{aligned} P_{ij} \to Q_{ij} : & f\left(P_{i-2,j-2}, P_{i-2,j-1}, \ldots, P_{i+2,j+2}\right) \\ & = B_m \oplus \left(\pi_1\left(P_{i-2,j-2}\right), \pi_2\left(P_{i-2,j-1}\right), \ldots, \pi_{24}\left(P_{i+2,j+2}\right)\right) \\ & = \left(b_{m1}, b_{m2}, \ldots, b_{m24}\right) \oplus \left(p^1_{i-2,j-2}, p^2_{i-2,j-1}, \ldots, p^{24}_{i+2,j+2}\right) \\ & = \left(b_{m1} \oplus p^1_{i-2,j-2}, b_{m2} \oplus p^2_{i-2,j-1}, \ldots, b_{m24} \oplus p^{24}_{i+2,j+2}\right) \\ & = \left(q^1_{ij}, q^2_{ij}, \ldots, q^{24}_{ij}\right) = Q_{ij}, \end{aligned}$$

that is, $q^n_{ij} = b_{mn} \oplus p^n_{i+h,j+w}$; m denotes the position of the pixel P_{ij}, i.e., $m = (i-1)s + j$, B_m is the m-th component of the bit sequence B, the operation $\oplus$ is the XOR-operation, $\pi_n : S \to \mathbb{Z}_2^{(n)}$ is the projection onto the n-th component, and the boundary conditions are periodic conditions.

The cipherimage, C, is obtained by applying only once the above transition function to each pixel of the plainimage I. In this way, the cipherimage is defined by $r \times s$ pixels and d colors, with $2 \leq d \leq 2^{24}$. Moreover the expansion factor for this cryptosystem is 1, i.e., the ratio between the cipherimage and the plainimage is 1. This cryptosystem considers the original image as the initial configuration of the CA, and the ciphered image as the configuration of the CA at time $t = 1$. It is not necessary to iterate the CA more than once since the security of the cryptosystem is not increased in other case.

4.3 Decryption

For the decryption protocol, the receiver considers the inverse CA of A, that is, $\mathsf{A}^{-1} = (L, S, W, g)$. The cellular space L and the state set S of A^{-1} are the same that for A. The set of indices of A^{-1} is $W = -V$, i.e., W is the same set of indices of A, but they are taken in their inverse order. Then, the neighbourhood of the pixel $\langle i, j \rangle$ is:

$$W = \{(2,2), \ldots, (2,-2), \ldots, (0,1), (0,-1), \ldots, (-2,2), \ldots, (-2,-2)\}. \quad (7)$$

Moreover, as in the previous case, the neighbourhood of the cell $\langle k, l \rangle$, $W_{\langle k,l \rangle}$, can be codified by the same two variables h and w: $W_{\langle k,l \rangle} = \{(k-h, l-w)\}$.

The transition function $g : S^{24} \to S$ to decrypt a pixel is defined in the following way:

$$\begin{aligned} Q_{kl} \to R_{kl} : & g\left(Q_{k+2,l+2}, Q_{k+2,l+1}, \ldots, Q_{k-2,l-2}\right) \\ & = B'_t \oplus \left(\pi_1\left(Q_{k+2,l+2}\right), \pi_2\left(Q_{k+2,l+1}\right), \ldots, \pi_{24}\left(Q_{k-2,l-2}\right)\right) \\ & = \left(b'_{t1}, b'_{t2}, \ldots, b'_{t24}\right) \oplus \left(q^1_{k+2,l+2}, q^2_{k+2,l+1}, \ldots, q^{24}_{k-2,l-2}\right) \\ & = \left(b'_{t1} \oplus q^1_{k+2,l+2}, b'_{t2} \oplus q^2_{k+2,l+1}, \ldots, b'_{t24} \oplus q^{24}_{k-2,l-2}\right) \\ & = \left(b'_{t1} \oplus b_{m1} \oplus p^1_{k,l}, b'_{t2} \oplus b_{m2} \oplus p^2_{k,l}, \ldots, b'_{t24} \oplus b_{m24} \oplus p^{24}_{k,l}\right) \\ & = \left(p^1_{kl}, p^2_{kl}, \ldots, p^{24}_{kl}\right) = P_{kl}, \end{aligned}$$

where $r^n_{kl} = b'_{tn} \oplus q^n_{k-h,l-w} = p^n_{kl}$; t denotes the position of $Q_{k-h,l-w}$, i.e., $t = (k-1-h)s + l - w$, and hence $b'_{t,n} = b_{(k-1-h)s+l-w,n}$, $\pi_n : S \to \mathbb{Z}^{2(n)}$ is the projection onto the n-th component, the symbol $\oplus$ stands for the XOR-operation, and the contour conditions are periodic conditions.

Hence, to recover the t-th decrypted pixel, P_{kl}, one has to apply the transition function g to its corresponding encrypted pixel, Q_{kl}. The plainimage, I, is recovered by applying once the transition function to each pixel of the cipherimage C. The plainimage I is identical to the original one (pixel by pixel), that is, the recovered image has no loss of resolution.

4.4 Example

In this subsection we present an example of a colored plainimage ("Self-Portrait" by Tamara de Lempicka) and its corresponding cipherimage. Both images are defined by 602×800 pixels, and they are shown reduced in Fig. 1. The number of colors of the first image is 85803, whereas the second one has 474750 colors.

For this example we have used a key with artificially small parameters. The bitlength of the prime numbers are 64, hence the bitlength of the modulus n is 128. The values of the modulus and the key are, respectively:

$$n = p \cdot q = 609490657811550215868356152313472597421, \tag{8}$$

$$K = 430146343670092314107950454676296640957. \tag{9}$$

Fig. 1. Example of a plainimage and its cipherimage

5 Conclusions

A new graphic symmetrical cryptosystem is presented in order to encrypt a colored image defined by pixels and by any number of colors. This cryptosystem is based on a reversible bidimensional cellular automaton and uses a cryptographically secure pseudorandom bit generator. The key of the cryptosystem is the same that for the CSPRBG and the session key is the seed used to generate the pseudorandom bit sequence. Moreover, the decrypted image is identical to the original one, i.e., no loss of resolution takes place.

Acknowledgement. This work was partially supported by Ministerio de Ciencia y Tecnología (Spain) under grant TIC2001–0586, and by Consejería de Educación y Cultura del Gobierno de Castilla y León (Spain) under grant SA052/03.

References

1. Bellamy, B., Mason, J.S., Ellis, M.: Photograph signatures for the protection of identification documents. Proc. of Crypto & Coding'99, LNCS **1746** (1999) 119–128.
2. Blum, L., Blum, M., Shub, M.: A simple unpredictable pseudo-random number generator. SIAM J. Comput. **15** (1986) 364–383.
3. Chang, C.C., Chuang, J.C.: An image intellectual property protection scheme for gray-level images using visual secret sharing strategy. Pattern Recogn. Lett. **23** (2002) 931–941.
4. Chang, C., Hwang, M., Chen, T.: A new encryption algorithm for images cryptosystems. J. Syst. Software **58** (2001) 83–91.
5. Chang, C., Liu, J.L.: A linear quadtree compression scheme for image encryption. Signal Process. Image **10** (1997) 279–290.
6. Fridrich, J.: Image encryption based on chaotic maps. Proc. IEEE Int. Conf. Systems, Man Cybern. Comput. Cybern. Simul. (1997) 1105–1110.
7. Fridrich, J.: Symmetric ciphers based on two-dimensional chaotic maps. Internat. J. Bifur. Chaos **8**, 6 (1998) 1259–1284.
8. Hernández Encinas, L., Montoya Vitini, F., Muñoz Masqué, J., Peinado Domínguez, A.: Maximal periods of orbits of the BBS generator. Proc. 1998 Int. Conf. on Inform. Secur. & Cryptol. (1998) 71–80.
9. Menezes, A., van Oorschot, P., Vanstone, S.: Handbook of applied cryptography. CRC Press, Boca Raton, FL, 1997.
10. Mollin, R.A.: An introduction to cryptography. Chapman & Hall/CRC, Boca Raton, FL, 2001.
11. Naor. M., Pinkas, B.: Visual authentication and identification. Proc. of Crypto'97, LNCS **1294** (1997) 322–336.
12. Naor, M., Shamir, A.: Visual cryptography. Proc. of Eurocrypt'94, LNCS **950** (1995) 1–12.
13. O'Gorman. L., Rabinovich, I.: Secure identification documents via pattern recognition and public-key cryptography. IEEE Trans. Pattern Anal. Mach. Intell. **20**, 10 (1998) 1097–1102.
14. Packard, N.H., Wolfram, S.: Two-dimensional cellular automata. J. Statist. Phys. **38** (1985) 901–946.
15. Stinson, D.: Cryptography. Theory and Practice. 2nd ed. CRC Press, Boca Raton, FL, 2001.