Encryption of Images with 2-dimensional Cellular Automata

Luis HERNÁNDEZ-ENCINAS

Dpto. Tratamiento de la Información y Codificación, Instituto de Física Aplicada, C.S.I.C.
C/ Serrano 144, 28006 Madrid, Spain
Email: luis@iec.csic.es

and

Ángel MARTÍN DEL REY and Ascensión HERNÁNDEZ ENCINAS

Dpt. Matemática Aplicada, E.T.S.I.I., Universidad de Salamanca
Avda. Fernández Ballesteros 2, 37700 Béjar, Salamanca, Spain
Emails: {ascen, delrey}@usal.es

ABSTRACT

We propose to use cellular automata of dimension 2 as graphic cryptosystems. In particular, a 256-coloured graphic cryptosystem is analyzed. Moreover, the security of the cryptosystem is studied and it is proved that it is safe against brute force and chosen-plaintext attacks. The cryptosystem can easily be extended for images defined by pixels with a different number of colours.

Keywords: Graphic cryptography, Cellular automata, Reversibility, Visual cryptography, Coloured images.

1. INTRODUCTION

As it is known, cryptography is related with all aspects of the algorithm number theory, computational theory, etc., in order to assure the secrecy and confidentiality of communications between two or more users, who use an insecure channel. To get this goal, the original message (the *plaintext*) to be exchanged is modified to obtain the encrypted message (the *ciphertext*), by using a cryptosystem, which is based on an algorithm. The parameters of the algorithm are called the *keys* of the cryptosystem. If the keys are only known for the sender and the receiver, the cryptosystem is known as *symmetric* (or *secret-key*) *cryptosystem*. The security of symmetric cryptosystems rely on keeping secret the key used. If the key for encrypting the plaintext is publicly known whereas the key for decrypting (the private key) is only known by the receiver, the cryptosystem is called *asymmetric* (or *public-key*) *cryptosystem*. In general, the security of an asymmetric cryptosystem is based on the supposed intractability of a mathematical problem, as the factorization of composite numbers or the computation of discrete logarithms (*c.f.*, [7, 14]).

Cryptanalysis tries to break the secrecy and confidentiality of the communications. The *brute force attack* consists in trying out all possible keys. Other attacks to break the secrecy of the communications ([7, §7.2]) are: (i) *Ciphertext-only*: in this case only the ciphertext is known; (ii) *Known-plaintext*: plaintext-ciphertext pairs are available; and (iii) *Chosen-plaintext*: the cryptanalyst chooses plaintexts and can obtain their corresponding ciphertexts. Moreover, it is known that if a cipher is secure against chosen-plaintext attack then it is secure against known-plaintext and ciphertext-only attacks.

On the other hand, dynamical systems have been also used in cryptography ([17]). In particular, chaotic dynamical systems have been proposed as for secret-key (*e.g.*, see [13] and the references therein) as public-key cryptography.

Moreover, cellular automata have been proposed as secret-key and as public-key cryptosystems. In the case of the streamciphers, cellular automata are used as pseudorandom bit generators in order to produce encryption sequences ([5, 6, 18]). Other cryptosystems use a permutation of the state set as secret-key ([4]), or an irreversible cellular automaton ([2]). As for public-key cryptosystems, there are several proposals. For example, to use an inhomogeneous cellular automaton ([1]), or a composition of several reversible cellular automata of dimension 1 ([4]).

In this paper, we propose to use cellular automata of dimension 2 as graphic cryptosystems, *i.e.*, as cryptosystems to encrypt images defined by pixels. These cryptosystems have several differences compared to the visual schemes proposed to date. For this reason, we denote this cryptography as *graphic cryptography*. Here, the proposed protocol begins with a message (the *plainimage*), uses and algorithm (a cellular automaton of dimension 2), and ends with a encrypted message (the *cipherimage*).

The rest of this paper is organized as follows. Below, visual cryptography is reviewed in order to compare the graphic cryptosystem that we propose and the visual one. Some definitions and properties of cellular automata are presented in §2, our proposal of a graphic cryptosystem is detailed in §3, and we present the conclusions in §4.

Visual Cryptography

Visual cryptosystems ([9, 15]) are based on *visual threshold schemes* t of n. This means that the original image is divided in several parts or shades, each of them is of the same size that the original image. This division is made in such way that it is necessary to join t shades to recover the original image, and no cryptographic protocol is used to recovered it. The recovery of the secret is carried out photocopying each shade on a transparency and then superimposing any t transparencies, keeping in mind that the secret cannot recover with $t-1$ transparencies or less.
A black and white image of size 310×270 pixels, the recovered image, and the corresponding two shades, obtained by using a visual threshold scheme 2 of 2 are shown in Figure 1. As it can see, the recovered image is not exactly the same that the original one. There exists a loss of resolution due to the fact that black pixels are enciphered as black pixels, but white pixels are enciphered as half-black and half-white pixels.

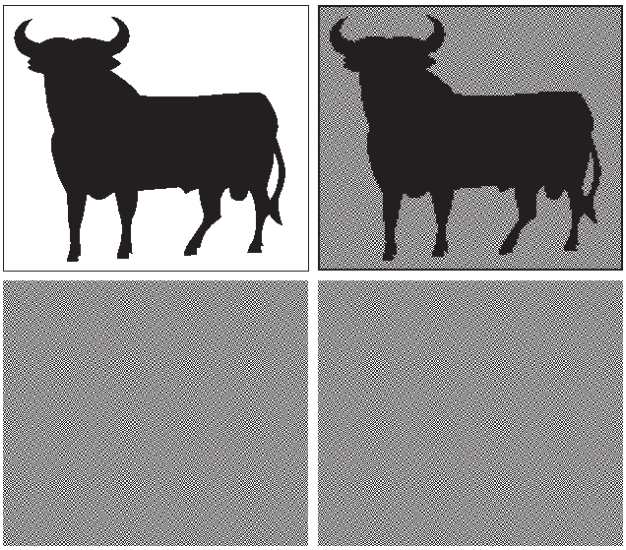

Figure 1. Example of a visual threshold scheme 2 of 2

The design of visual cryptosystems for coloured images is more difficult. However, there are several approaches to this possibility. One of them ([8]) is based on the fact that each colour is characterized by its wavelength. Other method ([16]) describes how to colour a scheme 2 of 2, for example, with r colours. A different solution for coloured images was proposed in [12]. In this proposal when two different colours are superimposing, a third colour it is obtained: Their sum.

2. CELLULAR AUTOMATA

A 2-*dimensional finite cellular automaton* ([11]) —or CA for short—, is a 4-uplet $\mathcal{A} = (I, S, V, f)$, where I 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 I, then for every cell $\langle i, j\rangle \in I$, its *neighbourhood* $V_{\langle i,j\rangle}$ is an ordered set of n cells defined as follows:

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

One of the neighbourhoods more used is the *von Neumann neighbourhood* ([10]) which considers the set $V^N = \{(0,0),(-1,0),(0,1),(1,0),(0,-1)\}$ and then

$$V^N_{\langle i,j\rangle} = \{\langle i,j\rangle, \langle i-1,j\rangle, \langle i,j+1\rangle, \langle i+1,j\rangle, \langle i,j-1\rangle\},$$

i.e., the neighbourhood of a cell are the cell itself and the four cells placed in the North, East, South and West positions:

	N	
W	*cell*	E
	S	

The *Moore neighbourhood* is defined by the set:

$$V^M = \{(0,0),(-1,0),(-1,1),(0,1),(1,1),(1,0),(1,-1),(0,-1),(-1,-1)\},$$

so that each cell has nine neighbours: the cell itself, North, North-East, East, South-East, South, South-West, West, and North-West:

NW	N	NE
W	*cell*	E
SW	S	SE

In general, each user can define the neighbourhood more appropriate for his particular interest.
Moreover, the *local transition function* $f\colon S^n \to S$ is a function which determines the evolution of the cellular automaton throughout the time, *i.e.*, the changes of the states of every cell taking the states of its neighbours into account.
Finally, as the cellular automata considered are finite, we take periodic boundary conditions of the form:

$$a_{ij}^{(t)} = a_{kl}^{(t)} \Leftrightarrow i \equiv k \bmod (r-1), \quad j \equiv l \bmod (s-1),$$

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

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

is called the *configuration* of $\mathcal{A}$ at time t. In particular, $C^{(0)}$ is called the *initial configuration* of the CA. Hence, the *evolution* of $\mathcal{A}$ is the sequence $\left(C^{(0)}, C^{(1)}, C^{(2)}, \dots\right)$.
A cellular automaton is *reversible* if there exists another CA, called its *inverse*, that computes the inverse evolution of $\mathcal{A}$. This kind of finite CA preserves the information given by the initial configuration, through the evolution. The problem of the reversibility of the 2-dimensional cellular automata is undecidable ([3]), *i.e.*, there does not exist an algorithm that would decide whether a given cellular automaton is reversible or not.

3. GRAPHIC CRYPTOSYSTEM

In this section we present a graphic cryptosystem in order to encrypt a coloured image. This cryptosystem uses a reversible 2-dimensional cellular automaton and it is based on an idea by Kari ([4]).
Given an image or graphics, G, defined by $r \times s$ pixels and c colours, where c takes one the following usual values: $c = 2^n$, $n = 4, 8, 24$. We use the standard codification of colours by means of bits. Here we will present only the case of $c = 256$ colours. The rest of mentioned cases are implemented in a similar way. Only it is neccesary to define the appropriate state set.
To construct the cryptosystem we define a reversible 2-dimensional cellular automaton $\mathcal{A} = (I, S, V, f_\pi)$ as follows:

- The cellular space I is a rectangular array of size $r \times s$, *i.e.*, I is the set of $r \times s$ pixels;
- The state set is $S = \mathbb{Z}_2 \times \overset{8}{\dots} \times \mathbb{Z}_2$, with $|S| = c = 2^8 = 216$, *i.e.*, each element of S is a colour of G;
- The neighbourhood considered here is the Moore neighbourhood without the cell itself, *i.e.*, the following:

$$V = \{(-1,0),(-1,1),(0,1),(1,1),\\ (1,0),(1,-1),(0,-1),(-1,-1)\},$$

  so that each cell has eight neighbours: North, North-East, East, South-East, South, South-West, West, and North-West;
- The transition function is based on an arbitrary non-trivial permutation of S, $\pi\colon S \to S$, and the projections $p_i\colon S \to \mathbb{Z}_2^{(i)}$, $i = 1, \dots, 8$. We set $f_\pi\colon S^8 \to S$, where

$$f_\pi(\mathbf{s}_1, \dots, \mathbf{s}_8) = \pi(p_1(\mathbf{s}_1), \dots, p_8(\mathbf{s}_8)),$$

  and $\mathbf{s}_i = (a_{i1}, \dots, a_{i8}) \in S$, $i = 1, \dots, 8$, $a_{ij} \in \mathbb{Z}_2$.

The inverse permutation of π, π^{-1}, gives the reversible automaton, $\mathcal{A}^{-1} = \left(I, S, \overline{V}, \overline{f}_\pi\right)$, where $\overline{f}_\pi\colon S^8 \to S$ is

$$\overline{f}_\pi(\mathbf{s}_1, \dots, \mathbf{s}_8) = \left(p_1\left(\pi^{-1}(\mathbf{s}_1)\right), \dots, p_n\left(\pi^{-1}(\mathbf{s}_8)\right)\right),$$

and $\overline{V} = -V$, *i.e.*

$$\overline{V} = \{(1,0),(1,-1),(0,-1),(-1,-1),\\ (-1,0),(-1,1),(0,1),(1,1)\}.$$

When this CA is applied to an element of I, *i.e.*, a coloured image, another coloured image of the same size and with the same colours, in different places, is obtained. Note that this cryptosystem has a expansion factor 1, *i.e.*, the quotient between the length of the ciphertext and the length of the plaintext is exactly 1. Moreover the recovered image is the same that the original one, and there are not loss of resolution as in the visual cryptography case.
Before to encrypt the image G, the sender, *Alice*, and the receiver, *Bob*, decide the parameter of the cryptosystem, *i.e.*, the number of iterations to be use, k. Moreover, they share the permutation π as the secret-key. The size of this secret-key is $c \log c = 2^n \cdot n = 2^8 \cdot 8 = 2048$ bits, as π is given by the following list: $\pi(0,0,0,0,0,0,0,0), \dots, \pi(1,1,1,1,1,1,1,1)$.

Encryption

Before to encrypt an image G, the sender A counts the exact number of colours used in G, say g, because if $g < 256$, the state set, S, which contains the g colours of G, must to be completed with $256 - g$ colours such that $|S| = 256$. These colours can be taken from the set of the 256 standard colours.
To encrypt an image G, A carries out the following protocol:

E1. A takes the 8 neighbours of a pixel (cell) of G in the time $t(= 0)$ to determine its colour (state) in the time $t + 1$.

E2. A applies the function f_π, to the neighbours $\mathbf{s}_i = (a_{i1}, \dots, a_{i8})$, $i = 1, \dots, 8$, as follows:

$$\begin{aligned} f_\pi(\mathbf{s}_1, \dots, \mathbf{s}_8) &= \pi(p_1(\mathbf{s}_1), \dots, p_8(\mathbf{s}_8)) \\ &= \pi(a_{11}, a_{22}, \dots, a_{88}) \\ &= (b_{11}, b_{22}, \dots, b_{88}) \end{aligned}$$

and obtain the new colour for the pixel.

E3. A repeats the previous steps for all $r \times s$ pixels of the image G.

A obtains the encrypted image of G: $\overline{G}$, after iterates k times this protocol, and sends $\overline{G}$ to B.
Note that the number of colours of $\overline{G}$ can be smaller than 256. In fact, this is what happens in general. Moreover, the colours of G will be different from the standard set of colours. As a consequence,
in these cases, A must to send to B the state set S, or the complementary set of the colours of $\overline{G}$ in S, in this way B will recover the original image with its original colours.

Decryption

To recover the plainimage G from the cipherimage $\overline{G}$, the receiver B proceeds as follows:

D1. B takes the 8 neighbours of a pixel (cell) of $\overline{G}$ in the time $t(=k)$ to determine its colour (state) in the time $t-1$.

D2. User B applies the function $\overline{f}_\pi$, to the 8 neighbours, $\overline{\mathbf{s}}_1, \ldots, \overline{\mathbf{s}}_8$, of a each pixel of $\overline{G}$ as follows:

$$\begin{aligned}&\overline{f}_\pi\left(\overline{\mathbf{s}}_1, \ldots, \overline{\mathbf{s}}_8\right)\\ &\quad = \left(p_1\left(\pi^{-1}\left(\overline{\mathbf{s}}_1\right)\right), \ldots, p_8\left(\pi^{-1}\left(\overline{\mathbf{s}}_8\right)\right)\right)\\ &\quad = \left(p_1\left(\overline{b}_{11}, \ldots, \overline{b}_{18}\right), \ldots, p_8\left(\overline{b}_{81}, \ldots, \overline{b}_{88}\right)\right)\\ &\quad = \left(\overline{b}_{11}, \overline{b}_{22}, \ldots, \overline{b}_{88}\right)\end{aligned}$$

and obtain the new colour for the pixel.

D3. B repeats the previous steps for all pixels of the image $\overline{G}$.

B obtains the plainimage G after iterate k times this protocol.

Figure 2 shows an image of 252 colors of size 162×168 pixels and its corresponding encrypted image after $k = 5$ iterations, by using the cryptosystem described above.

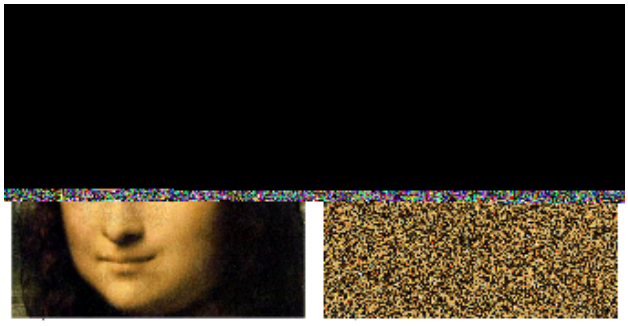

Figure 2. Example of an encrypted coloured image

Proof that the cryptosystem works

To prove that the previous cryptosystem works, *i.e.* that the encryption and decryption process are inverses, we analyze the case of a single cell $x = (x_1, \ldots, x_8) \in S$. We consider the following distribution of its neighbourhood:

NWNW	NNW	NN	NNE	NENE
WNW	NW	N	NE	ENE
WW	W	x	E	EE
WSW	SW	S	SE	ESE
SWSW	SSW	SS	SSE	SESE

Let $\overline{x} = f_\pi(x)$ the encrypted cell of x, then we have

$$\begin{aligned}\overline{x} &= f_\pi\left(\mathrm{N}, \mathrm{NE}, \mathrm{E}, \mathrm{SE}, \mathrm{S}, \mathrm{SW}, \mathrm{W}, \mathrm{NW}\right)\\ &= \pi\left(\mathrm{N}_1, \mathrm{NE}_2, \mathrm{E}_3, \mathrm{SE}_4, \mathrm{S}_5, \mathrm{SW}_6, \mathrm{W}_7, \mathrm{NW}_8\right),\end{aligned}$$

where the subindex i, $1 \leq i \leq 8$, of each position around x denotes the i-th projection of the state of the cell located in that position, *i.e.*, the i-th bit of the binary expression of its colour.
If we consider $\overline{\mathrm{P}} = f_\pi(\mathrm{P})$, we have

$$\begin{aligned}\overline{\mathrm{N}} &= \pi\left(\mathrm{NN}_1, \mathrm{NNE}_2, \mathrm{NE}_3, \mathrm{E}_4, x_5, \mathrm{W}_6, \mathrm{NW}_7, \mathrm{NNW}_8\right)\\ \overline{\mathrm{NE}} &= \pi\left(\mathrm{NNE}_1, \mathrm{NENE}_2, \mathrm{ENE}_3, \mathrm{EE}_4, \mathrm{E}_5, x_6, \mathrm{N}_7, \mathrm{NN}_8\right)\\ \overline{\mathrm{E}} &= \pi\left(\mathrm{NE}_1, \mathrm{ENE}_2, \mathrm{EE}_3, \mathrm{ESE}_4, \mathrm{SE}_5, \mathrm{S}_6, x_7, \mathrm{N}_8\right)\\ \overline{\mathrm{SE}} &= \pi\left(\mathrm{E}_1, \mathrm{EE}_2, \mathrm{ESE}_3, \mathrm{SESE}_4, \mathrm{SSE}_5, \mathrm{SS}_6, \mathrm{S}_7, x_8\right)\\ \overline{\mathrm{S}} &= \pi\left(x_1, \mathrm{E}_2, \mathrm{SE}_3, \mathrm{SSE}_4, \mathrm{SS}_5, \mathrm{SSW}_6, \mathrm{SW}_7, \mathrm{W}_8\right)\\ \overline{\mathrm{SW}} &= \pi\left(\mathrm{W}_1, x_2, \mathrm{S}_3, \mathrm{SS}_4, \mathrm{SSW}_5, \mathrm{SWSW}_6, \mathrm{WSW}_7, \mathrm{WW}_8\right)\\ \overline{\mathrm{W}} &= \pi\left(\mathrm{NW}_1, \mathrm{N}_2, x_3, \mathrm{S}_4, \mathrm{SW}_5, \mathrm{WSW}_6, \mathrm{WW}_7, \mathrm{WNW}_8\right)\\ \overline{\mathrm{NW}} &= \pi\left(\mathrm{NNW}_1, \mathrm{NN}_2, \mathrm{N}_3, x_4, \mathrm{W}_5, \mathrm{WW}_6, \mathrm{WNW}_7, \mathrm{NWNW}_8\right)\end{aligned}$$

Hence, the recovered cell $\overline{\overline{x}}$ from $\overline{x}$ is determined as follows:

$$\begin{aligned}\overline{\overline{x}} &= \overline{f}_\pi(x) = \overline{f}_\pi\left(\overline{\mathrm{S}}, \overline{\mathrm{SW}}, \overline{\mathrm{W}}, \overline{\mathrm{NW}}, \overline{\mathrm{N}}, \overline{\mathrm{NE}}, \overline{\mathrm{E}}, \overline{\mathrm{SE}}\right)\\ &= \left(p_1\pi^{-1}\left(\overline{\mathrm{S}}\right), p_2\pi^{-1}\left(\overline{\mathrm{SW}}\right), \ldots, p_8\pi^{-1}\left(\overline{\mathrm{SE}}\right)\right)\\ &= (x_1, x_2, \ldots, x_8) = x.\end{aligned}$$

Cryptanalysis

The first attack is the brute force attack, but to determine the permutation π used in the cryptosystem is out of question because the number of such permutations is

$$c! = 256! \simeq 8.5 \cdot 10^{506}.$$

Another possible attack is the chosen-plaintext attack. In this case, the cryptanalyst has access to the encryption machine in such a way he can obtain the cipherimage corresponding to a chosen image (without knowing the key). The option for the cryptanalyst is to encrypt an image with all pixels of the same colour: l. Then the cipherimage is also homogeneous, that is, all its pixels have the same colour: $\pi^k(l)$, where k is the number of iterations. Nevertheless knowing π^k for each colour does not give more information about the

value of π. Observe that the security of this cryptosystem increases if k is bigger, as the initial configuration affects a larger amount of cells. The disadvantage of increasing k is that the cryptosystem would be more slower.

The fact of modifying a pixel of the original image becomes alter the encrypted image it is know as the *avalanche effect*. This effect is important in our proposal because the change in the colour of a unique pixel in the original image affects the result obtained. In Figure 3 we present an example of this effect. The two first images only differ in one pixel. The rest of images present by means of black pixels, the positions for which both images have different colours after $k = 5, 10$ and 20 iterations, respectively. In this way, it is possible to appreciate the avalanche effect for the two first images presented when only a pixel is modified.

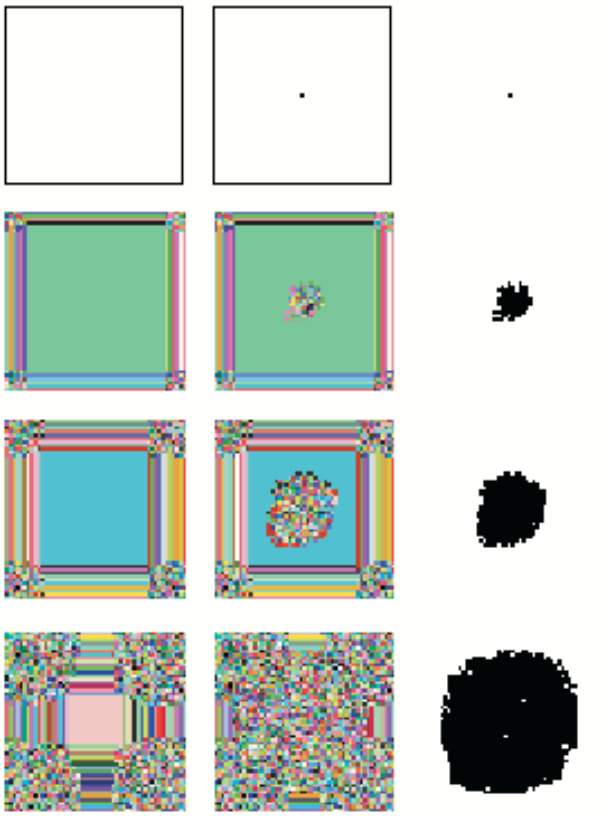

Figure 3. The avalanche effect

Encryption of a black and white image

To encrypt a black and white image we cannot use the previous cryptosystem because the only permutations of $c = 2$ colours are trivials. The best possibility is to consider the list of 256 standard colours (in fact, black and white are two of these colours), and then to use the cryptosystem described above. In this case, both, the sender and the receiver, do not need to share the set S, because it is standard.

As an example about how to use this possibility to encrypt a black and white image, in Figure 4 we present a black and white plainimage, of size 310×270 pixels, and its corresponding 256-coloured cipherimage, after $k = 75$ iterations. Note that the number of iterations in this case is bigger that in the example of Figure 2 because this back and white image has big zones of homegeneous colours.

Figure 4. An encrypted black and white image

4. CONCLUSIONS

We have proposed the use of cellular automata of dimension 2 for encryption coloured images defined by pixels, *i.e.*, as graphic cryptosystems. In this cryptosystem, a unique image is obtained and its size is the same than the original one. Hence, the expansion factor is equals to 1. Nevertheless, the time for computing the cipherimage from a plainimage is bigger than in text cryptosystems. Moreover, the cryptosystem for 256-coloured images presented can be extended to a different number of colours like: $2, 2^4$, and 2^{24}, which are considered as standard, or to any other number of colours. On the other hand, the security of these cryptosystems have been analyzed and it is proved that the cryptosystems are safe against brute force and chosen-plaintext attacks if the number of colours used is equal or bigger than 256.

Acknowledgement. Supported by "Samuel Solórzano Barruso" Foundation (Spain).

5. REFERENCES

[1] P. Guan, "Cellular automaton public-key cryptosystem", Complex Systems 1, 1987, pp. 51–57.

[2] H. A. Gutowitz, "Cryptography with dynamical systems", Proceedings of the NATO Advanced Study Institute, 1993, pp. 237–274.

[3] J. Kari, "Reversibility of 2-D cellular automata is undecidable", Physica D 45, 1990, pp. 379–385.

[4] —, "Cryptosystems based on reversible cellular automata", University of Turku, Preprint 1992.

[5] M. Matsumoto, "Simple cellular automata as pseudorandom m-sequence generators for built-in seft-test", ACM Trans. Mod. Comput. Simul. 8 (1), 1998, pp. 31–42.

[6] W. Meier and O. Staffelbach, "Analysis of pseudo random sequences generated by cellular automata", Proceedings of EUROCRYPT'91, LNCS 547, 1991, pp. 186–199.

[7] A. Menezes, P. van Oorschot, and S. Vanstone, Handbook of applied cryptography, Boca Raton, FL.: CRC Press, 1997.

[8] D. Naccache, "Colorful cryptography -a purely physical secret-sharing scheme based on chromatic filters-", Coding and Information Integrity, French-Israeli Workshop, 1994.

[9] M. Naor and A. Shamir, "Visual cryptography", Proceedings of EUROCRYPT'94, LNCS 950, 1995, pp. 1–12.

[10] J. von Neumann, "Theory of selft-reproducing automata", A. W. Burks (ed.), University of Illinois Press, 1966.

[11] N. H. Packard and S. Wolfram, "Two-dimensional cellular automata", J. Statist. Phys. 38, 1985, pp. 901–946.

[12] V. Rijmen and B. Preneel, "Efficient colour visual encryption or Shared colours of Benetton", Rump Session of Eurocrypt'96, available in htpp://www.iacr.org/conferences/ec96/rump/preneel.ps.

[13] R. Schmitz, "Use of chaotic dynamical systems in cryptography", J. Franklin Inst. 338, 2001, pp. 429–441.

[14] B. Scheiner, Applied cryptography, New York: John Willey & Sons Inc., 2nd. edition, 1996.

[15] D. R. Stinson, Cryptography. Theory and practice, Boca Raton, FL.: CRC Press, 2nd. edition, 2001.

[16] E. R. Verheul and H. C. A. van Tilborg, "Constructions and properties of k out of n visual secret sharing schemes", Designs, Codes and Cryptography 11, 1997, pp. 179–196.

[17] S. Wolfram, "Cryptography with cellular automata", Proceedings of CRYPTO'85, LNCS 218, 1986, pp. 429–432.

[18] —, "Random sequence generation by cellular automata", Adv. Appl. Math. 7, 1986, pp. 123–169.