ISSN 1870-4069

Algoritmo esteganográfico basado en autómatas celulares y en la sustitución de los bits menos significativos

Uriel López, Humberto Dávila, Luis Zapata, Marco Ramírez

Universidad Autónoma de San Luis Potosí,
Coordinación Académica Región Altiplano Oeste, México
tulio.torres@uaslp.mx

Resumen. En esta investigación se presenta un nuevo algoritmo de esteganografía para imágenes, utilizando la sustitución de los bits menos significativo y la regla local 90 de autómatas celulares. Este algoritmo es capaz de cifrar y ocultar los pixeles de la imagen en un proceso combinado. El factor PSNR indica que no existe una distorsión severa en la imagen esteganográfica, lo que en esta área se interpreta como a menor distorsión de la imagen esteganográfica, mayor es la probabilidad de pasar desapercibida y que no sea analizada. Sin embargo, en caso de que la imagen oculta sea extraída, ésta se encuentra protegida por una clave secreta. Este algoritmo transforma la imagen secreta a una imagen cifrada con distribución uniforme y sin correlación. Este algoritmo esteganográfico puede ser una nueva opción para ocultar y proteger imágenes relacionando ambos procesos.

Palabras clave: esteganografía, autómatas celulares, PSNR.

Steganographic Algorithm Based on Cellular Automata and the Replacement of Less Significant Bits

Abstract. This research presents a new steganography algorithm for images, using the least significant bit replacement and local rule 90 of cellular automata. This algorithm is able to encrypt and hide the image pixels in a combined process. The PSNR factor indicates that there is no severe distortion in the steganographic image, which in this area is interpreted as the less distortion of the steganographic image, the greater the probability of going unnoticed and that it is not analyzed. However, if the hidden image is extracted, it is protected by a secret key. This algorithm transforms the secret image to an encrypted image with uniform distribution and no correlation. This steganographic algorithm can be a new option to hide and protect images by relating both processes.

Keywords: steganography, cellular automata, PSNR.

1. Introducción

En nuestra vida digital actual, hay una demanda latente por seguridad para garantizar que nuestra información confidencial no sea observada por personas sin autorización. Para ello se han desarrollado diversas técnicas, resolviendo diferentes aspectos de este problema. La esteganografía es una de las metodologías con las que se cuenta para ocultar y proteger nuestra información en algún objeto llamado portador. El objetivo es que la información pase inadvertida estando oculta en el portador. La naturaleza de los datos a esconder y el objeto portador puede no ser la misma, es decir, podemos ocultar un archivo de audio en una imagen, un texto en un audio etc.

Existen diferentes técnicas para aplicar la esteganografía, una de las más conocidas es la sustitución de los bits menos significativos. Esta técnica consiste en reemplazar los bits menos significativos del portador por los bits del archivo secreto. Por lo tanto, en el caso de que el portador y la información secreta sean imágenes, la percepción de la imagen portadora puede ser distinta cuando se monte la imagen secreta, incluso puede revelar ciertos patrones de ésta. La imagen resultante de sobreponer en la imagen portadora la imagen secreta, se le llama imagen esteganográfica.

Para evitar que la información sea simplemente revelada al momento de ser extraída del portador, varios esquemas han propuesto cifrar la información secreta para después ocultarla en el portador [1-3] haciendo dos procesos independientes. Dado que las imágenes digitales pueden tener una gran cantidad de datos, una alta correlación adyacente [4] o pueden ser totalmente redundantes; los algoritmos de cifrado suelen tener varias operaciones de confusión y difusión e iteraciones para poder romper la correlación adyacente. Lo que incrementa el costo computacional.

Es por eso que en esta investigación, se ha desarrollado un algoritmo que relaciona ambos procesos: criptografía y esteganografía; capaz de sustituir los valores de la imagen secreta con influencia de los bits de la imagen portadora y una llave inicial para lograr una mayor dispersión de los valores, y al mismo tiempo ocultar la imagen secreta. Gracias a la regla local 90 de autómatas celulares, con operaciones simples se logra una gran dinámica capaz de cifrar imágenes altamente redundantes, para obtener imágenes con histograma uniforme. El algoritmo está basado en un sistema de encriptación llamado ESAC (Encriptación basada en la sincronización de autómatas celulares) en la versión que se muestra en la Ref. [5], con algunas modificaciones y mejoras, que pueden hacer de este algoritmo una nueva propuesta para el área de esteganografía.

2. Materiales y métodos

2.1. Autómatas celulares

Un autómata celular es un modelo matemático para un sistema dinámico, que consiste en una red organizada de celdas que adquieren distintos estados o valores. Evolucionan a pasos discretos cambiando su estado, de acuerdo a un algoritmo llamado regla local, donde intervienen los valores de la celda y los de sus vecinos.

La regla local 90 es clasificada como de clase III [6], lo que quiere decir que es de patrón caótico, por lo que ha sido utilizada para crear generadores de números pseudoaleatorios y criptosistemas [5]. Es una de las reglas de los autómatas celulares

elementales, para su cálculo se considera una vecindad de radio uno y dos estados {0,1}, puede ser descrita con la siguiente expresión:

$$x_i^{t+1} = A(x_{i-1}^t + x_{i+1}^t), \quad (1)$$

donde *t* representa el tiempo e *i* la posición del autómata. Por lo tanto, el cálculo de esta regla equivale a calcular una operación XOR entre las celdas que están a los extremos de la vecindad.

2.2. Generador de números pseudoaleatorios y función de procesamiento

En la Ref. [7] se utilizó la regla local 90 y el fenómeno de sincronización que ésta presenta, para proponer un sistema de cifrado de datos llamado ESAC y un Generador de Números Pseudo-Aleatorios (GNPA). El GNPA funciona introduciendo dos vectores booleanos **x** y **y** de longitud *n* y *n+1* bits respectivamente, para obtener la secuencia aleatoria **t**, utilizando la función *h* que es con la que se denomina a la evolución hacia atrás del autómata celular. La Fig. 1 presenta un diagrama a bloques del GNPA.

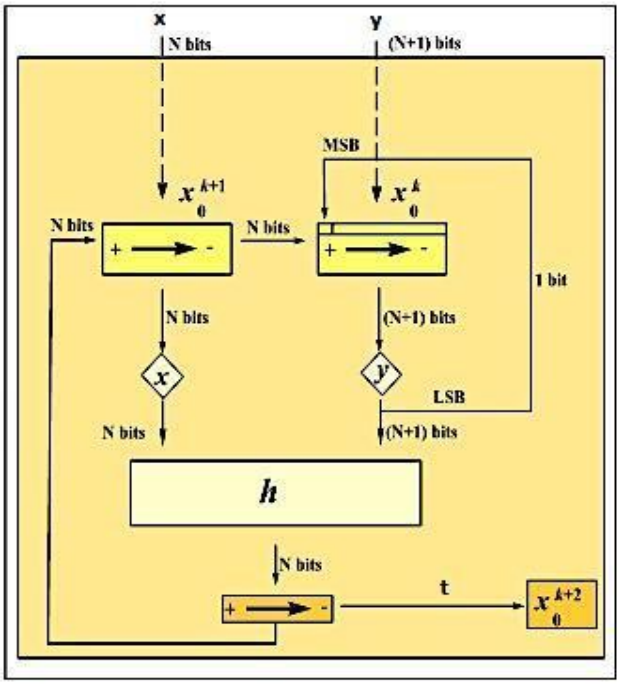

Fig. 1. Forma básica del generador de números pseudo-aleatorios.

El sistema se retroalimenta y la secuencia generada **t** ocupa el lugar del vector **x**, y la anterior **x** toma el lugar de **y**, para lo cual de la anterior versión se toma el bit menos significativo y se concatena como el bit más significativo. En la Ref. [8] se evaluó su

desempeño pasando todas las pruebas de NIST (National Institute of Standards and Technology).

En la Ref. [5] este generador se modificó para diseñar una función de procesamiento, capaz de intercambiar los coeficientes de pixeles idénticos, por valores diferentes en función del tiempo. Es totalmente reversible gracias a la regla local 90 y a las retroalimentaciones que utiliza. En este caso, el lugar del vector **x**, es utilizado por los bloques en claro **m**, y el vector **y** es llamado **z**, para diferenciar su uso. El bloque resultante se llama $\hat{m}$.

Después de la primera iteración, **z** se calcula con la retroalimentación, utilizando el bloque $\hat{m}$ resultante, se le concatena el bit menos significativo del vector **z** anterior. De esta manera aunque los bloques de texto en claro m_1 y m_2 sean iguales, son procesados con vectores **z** diferentes, resultando $\hat{m}_1$ y $\hat{m}_2$ diferentes entre sí.

Esta función puede ser simplificada a ecuaciones de un solo tiempo, sin la necesidad de evolucionar el autómata, utilizando lógica booleana. Las ecuaciones (2) resultantes se muestran a continuación, para el caso de 8 bits.

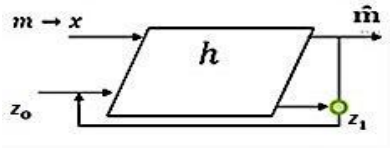

Fig. 2. Función de procesamiento, utilizando el generador de números pseudoaleatorios.

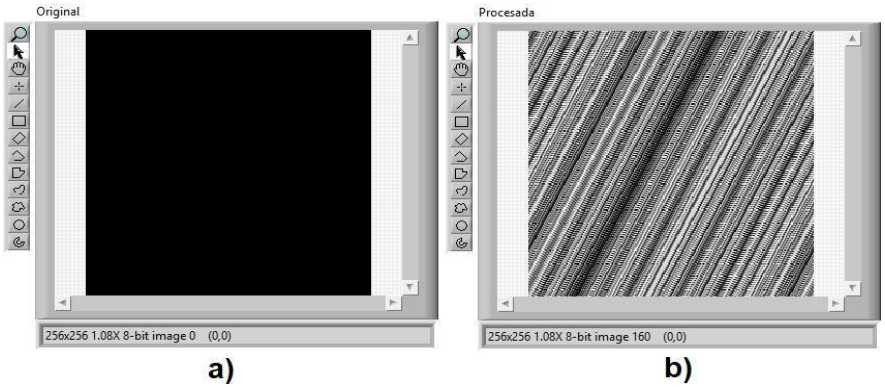

Fig. 3. Caso particular a mejorar: a) Imagen sólida con todos los pixeles **m**=**0**, b) Imagen sólida procesada.

$$
\begin{aligned}
\hat{m}_1 &= m_1 \oplus z_2, \\
\hat{m}_2 &= m_2 \oplus z_1 \oplus z_{3,} \\
\hat{m}_3 &= m_1 \oplus m_3 \oplus z_4, \\
\hat{m}_4 &= m_4 \oplus z_1 \oplus z_3 \oplus z_5, \\
\hat{m}_5 &= m_1 \oplus m_3 \oplus m_5 \oplus z_2 \oplus z_6, \\
\hat{m}_6 &= m_2 \oplus m_6 \oplus z_1 \oplus z_5 \oplus z_7, \\
\hat{m}_7 &= m_1 \oplus m_5 \oplus m_7 \oplus z_8, \\
\hat{m}_8 &= m_8 \oplus z_1 \oplus z_5 \oplus z_7 \oplus z_9,
\end{aligned}
\tag{2}
$$

donde $\oplus$ representa la operación XOR.

Uno de los aspectos a mejorar de esta función, es el hecho de que no todos los bits se combinan de igual manera como lo hacen los más significativos, donde las ecuaciones involucran más bits. Este detalle se puede ver en el caso de una imagen sólida donde todos los píxeles son igual a cero, **m=0**. La versión procesada de esta imagen se muestra en la Fig. 3.

2.3. Método propuesto

Para implementar el sistema esteganográfico se utilizó el lenguaje de programación LabVIEW de National Instruments, trabajando con imágenes de 8 bits. Para solventar los problemas antes vistos en la función de procesamiento, se realizaron cambios en la conformación de los bloques, donde se hicieron pruebas ocultando 2, 3 y 4 bits en la imagen portadora, nombrando a cada versión como V1, V2 y V3 respectivamente. La imagen portadora debe ser de dimensiones mayores a la imagen oculta, para que ésta no pierda información, es decir si se almacenan dos bits de la imagen oculta, la portadora debe ser 4 veces más grande que la imagen a ocultar.

El método propuesto de esteganografía para el caso de 2 bits se calcula de la siguiente manera:

1) El bloque a procesar **m** de 8 bits se conforma con los 6 bits más significativos de la imagen portadora, puestos en las posiciones de m_1 a m_6. Los bits m_7 y m_8 son ocupados por bits de la imagen oculta.
2) Se genera una secuencia pseudoaleatoria **z**, con los vectores **x** y **y**.
3) Se procesan el bloque **m** y el vector **z**.
4) Al bloque $\hat{m}$ resultante se le extraen los bits $\hat{m}_7$ y $\hat{m}_8$.
5) Los bits $\hat{m}_7$ y $\hat{m}_8$ se concatenan a los bits del pixel de la imagen portadora en la posición de los bits menos significativos, formando así un pixel de la imagen esteganográfica.
6) Se repite el proceso para el siguiente par de bits de la imagen oculta.

En la Fig. 4 se puede ver un diagrama de este método.

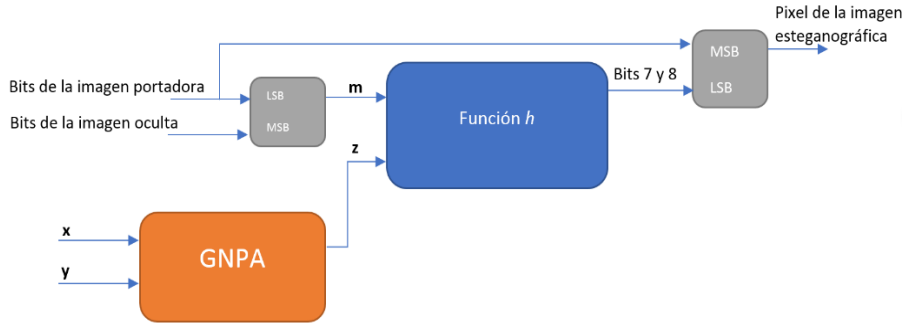

Fig. 4. Diagrama de bloques del algoritmo esteganográfico propuesto.

3. Resultados

A continuación, presentamos el análisis de las imágenes esteganográficas obtenidas, utilizando las diferentes versiones V1, V2 y V3. También se realizó un análisis

estadístico de las imágenes ocultas en su versión procesada, donde se obtuvieron los histogramas y coeficiente de correlación.

Estas pruebas se realizaron para diferentes tipos de imagen, las cuales son ampliamente utilizadas en el procesamiento de imágenes como lo son: Lena, mandril y pimientos, como imágenes ocultas de dimensiones de 256×256 pixeles a 8 bits. Como imagen portadora se utilizó la imagen de avión a 8 bits y de dimensiones de 512×512 para el caso de ocultar 2 bits por pixel, 768×256 para el caso de 3 bits por pixel, y por último de dimensiones 512×256 para el caso de 4 bits. La Fig. 5 muestra las imágenes de prueba.

Fig. 5. Imágenes de prueba utilizadas.

Fig. 6. a) Imagen portadora, b) Imagen esteganográfica con la imagen de Lena oculta.

3.1. PSNR

La métrica PSNR es utilizada para medir la calidad de imágenes tras un procesamiento. Para el caso de esteganografía, mientras mayor sea este valor menor es

la probabilidad de que por inspección visual se realice un ataque [9]. En esta investigación se utiliza para medir la calidad de la imagen esteganográfica, debido a que se suele distorsionar cuando se monta la imagen oculta, lo que puede levantar sospechas y que la imagen sea sustraída. La Fig. 6 muestra la imagen portadora utilizada y la imagen esteganográfica resultante, ocultando dos bits por pixel de la imagen de Lena.

La Tabla 1 muestra los valores de PSNR obtenidos para cada imagen en las diferentes versiones del algoritmo

En otras investigaciones [10-12] obtienen valores de PSNR mayores a los reportados, pero es muy probable que sea por las dimensiones de imágenes que se utilizan. En estas investigaciones la imagen portadora tiene una capacidad en bits mayor que la mínima requerida para ocultar la imagen secreta. En esta investigación siempre se utilizó la mínima capacidad de bits en la imagen portadora, por lo tanto, todos sus pixeles contienen datos de la imagen secreta.

Por otra parte, la imagen oculta resultante del procesamiento se puede ver en la Fig. 7, donde se ilustra el caso de una imagen sólida en color negro, con todos sus coeficientes igual a **m=0**.

Tabla 1. Cálculo del PSNR de la imagen esteganográfica en las diferentes versiones del algoritmo, con diferentes imágenes ocultas.

Imagen	**V1 (dB)**	**V2 (dB)**	**V3 (dB)**
Lena	41.4259	36.4046	29.2256
Mandril	41.4327	36.3845	29.8222
Pimientos	41.4135	36.4111	29.6882

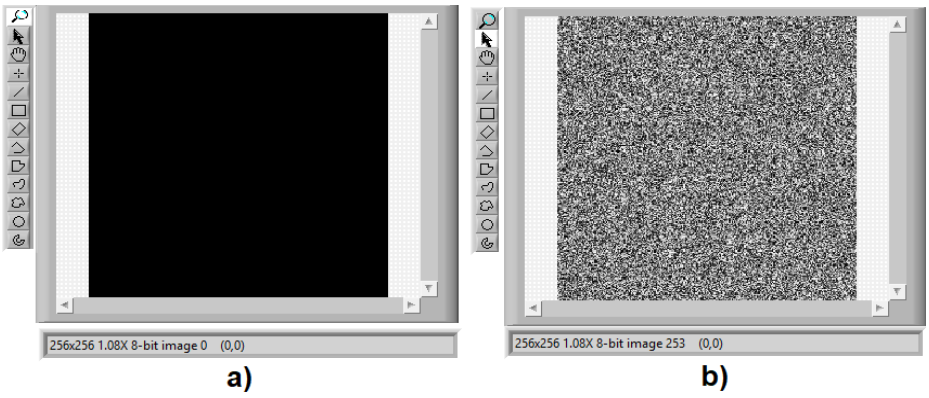

Fig. 7. Resultado de la nueva función de procesamiento, a) Imagen original sólida y b) Versión procesada de a).

Como podemos ver, ahora no muestra patrones, evoluciona de una manera más uniforme.

3.2 Análisis de histogramas

El análisis de histogramas revela la frecuencia con la que se presentan los valores de los coeficientes de los pixeles de una imagen. Por lo tanto, una imagen cifrada debe

presentar un histograma uniforme, lo que indica que logró ocultar la redundancia de la imagen original, que puede usarse como estadística para revelar información de la imagen oculta. En este análisis, todas las imágenes de prueba que se procesaron presentaron un histograma uniforme para todos los valores de la codificación, en este caso 8 bits. La Fig. 8 muestra el histograma de la imagen de los pimientos y su versión procesada.

3.3 Correlación

El cálculo de correlación nos ayuda a determinar si una imagen cifrada es independiente de su imagen original. Utilizamos el cálculo del coeficiente de correlación de Pearson, el cual señala que si el valor del índice es menor a |0.3| no existe una correlación entre las señales [13]. La Tabla 2 muestra el resultado de calcular el coeficiente de correlación a las imágenes procesadas con su respectiva imagen original, para todas las versiones.

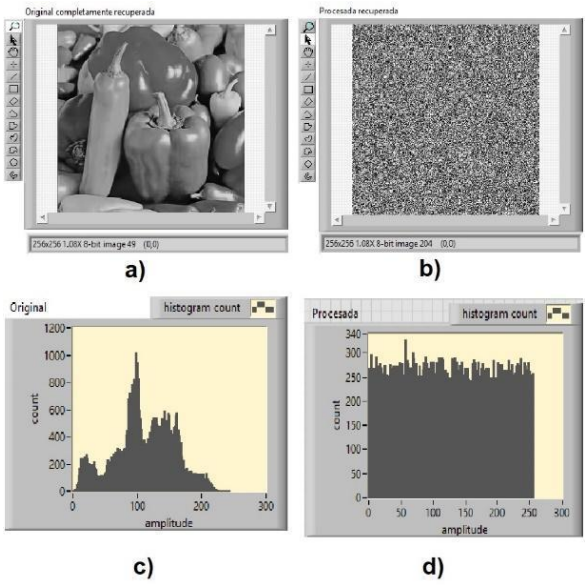

Fig. 8. Análisis de histogramas. a) Imagen original, b) Versión procesada de a), c) histograma de a) y d) histograma de b).

Tabla 2. Cálculo de correlación de las imágenes ocultas procesadas.

Imagen	**V1**	**V2**	**V3**
Lena	-0.0002887	-0.0021166	-0.0019959
Mandril	0.0015039	-0.0002760	0.0004766
Pimientos	-0.0032853	-0.0005435	-0.0015039

Como podemos ver no existe correlación entre las imágenes, lo cual nos indica que no hay una dependencia entre las imágenes procesadas y su versión original.

4. Conclusiones

En este trabajo se presentó una propuesta para un algoritmo esteganográfico capaz de esconder y cifrar la imagen oculta. El utilizar la sincronización de autómatas celulares con los bits de la imagen portadora y el generador de números pseudoaleatorios, proporcionó una dinámica mayor a la de la función de procesamiento, logrando trabajar con imágenes altamente redundantes. La imagen procesada pasó un análisis estadístico básico y la imagen portadora se mantuvo en los límites de calidad. Por lo que podemos concluir que esta propuesta puede representar una opción viable para aplicaciones de seguridad. La regla local 90 para autómatas celulares brinda una dinámica interesante para aplicaciones de esteganografía debido a su comportamiento y a que es reversible.

A futuro esperamos estudiar más a fondo de este algoritmo y de ser necesario mejorar el sistema. Una de las metas es lograr que con el fenómeno de sincronización la imagen oculta tienda a ser similar a la imagen portadora, para que al momento de ser extraída, no se visualice como una imagen cifrada, sino como una imagen similar a la portadora.

Referencias

1. Gupta, S., Goyal, A., Bhushan, B.: Information hiding using least significant bit steganography and cryptography. International Journal of Modern Education and Computer Science 4(6), 27 (2012)
2. Chauhan, S., Kumar, J., Doegar, A.: Multiple layer text security using variable block size cryptography and image steganography. In: 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT), pp. 1–7, IEEE (2017)
3. Joshi, K., Yadav, R.: A new LSB-S image steganography method blend with Cryptography for secret communication. In: 2015 Third International Conference on Image Information Processing (ICIIP), pp. 86–90, IEEE (2015)
4. Lian, S.: Multimedia content encryption: techniques and applications. Auerbach Publications (2008)
5. Ramirez-Torres, M.T., Murguia, J.S., Carlos, M.M.: Image encryption with an improved cryptosystem based on a matrix approach. International Journal of Modern Physics C, 25(10), 1450054 (2014)
6. Wolfram, S.: A new kind of science. Vol. 5, p. 130, Champaign, IL: Wolfram media (2002)
7. Urias, J., Ugalde, E., Salazar, G.: A cryptosystem based on cellular automata. Chaos, Woodbury, NY, 8(4), 819–822 (1998)
8. Murguía, J.S., Carlos, M.M., Rosu, H.C., Flores-Eraña, G.: Improvement and analysis of a pseudo-random bit generator by means of cellular automata. International Journal of Modern Physics C, 21(06), 741–756 (2010)
9. Karim, S.M., Rahman, M.S., Hossain, M.I.: A new approach for LSB based image steganography using secret key. In: 14th international conference on computer and information technology (ICCIT 2011), pp. 286–291, IEEE (2011)
10. Jana, B., Giri, D., Mondal, S.K., Pal, P.: Image steganography based on cellular automata. International Journal of Pure and Applied Mathematics 83(5), 701–715 (2013)
11. Solís, I.S.: Esteganografía en imágenes digitales aplicando autómatas celulares bidimensionales como generadores seudoaleatorios. Revista de Investigaciones (Puno)-Escuela de Posgrado de la UNA PUNO, 6(1), 66–77 (2017)
12. Bhardwaj, R., Sharma, V.: Image steganography based on complemented message and inverted bit LSB substitution. Procedia Computer Science 93, 832–838 (2016)

13. Mukaka, M.M.: A guide to appropriate use of correlation coefficient in medical research. Malawi Medical Journal 24(3), 69–71 (2012)