ISSN 1870-4069

Criptoanálisis y mejora a esquema de cifrado de imágenes hipercaótico

Marco T. Ramírez-Torres[1], César A. Guerra-García[1], María F. Mendoza Martínez[1], Jazmín Rangel Saucedo[1], Carlos Soubervielle Montalvo[2]

[1] Universidad Autónoma de San Luis Potosí, Coordinación Académica Región Altiplano Oeste, México

[2] Universidad Autónoma de San Luis Potosí, Facultad de Ingeniería, México

{tulio.torres, carlos.soubervielle@uaslp.mx}

Resumen. En los últimos años, debido a la demanda de seguridad los sistemas de cifrado de imágenes basados en sistemas caóticos, han ganado relevancia, debido a su posible aplicación en equipos de cómputo y dispositivos móviles para asegurar la transferencia y almacenamiento de imágenes y videos. En este trabajo, analizamos y probamos un esquema de cifrado de imágenes en escala de grises, basado en un sistema hipercaótico multienroscado. Este esquema consiste en un algoritmo de llave simétrica que utiliza un novedoso Generador de Bits Pseudo-Aleatorios (GBPA), capaz de generar secuencias binarias utilizando los cuatro estados de un sistema hipercaótico multienroscado. Aún cuando esta propuesta se evaluó a través de diversas pruebas de seguridad (análisis de histogramas, entropía, etc.) y el GBPA pasa las pruebas del NIST (National Institute of Standards and Technology), este sistema presenta una debilidad al momento de aplicar un ataque criptoanalítico, conocido como chosen-plain image attack (CPIA). En este trabajo se presenta un criptoanálisis al sistema, donde con dos imágenes arbitrarias es posible recuperar la información original, sin conocer la llave secreta. Además, se propone una mejora para evitar la fuga de información, basada en una función de sustitución de valores utilizando autómatas celulares.

Palabras clave: Criptoanálisis, cifrado imágenes, chosen-plain image attack.

Cryptanalysis and Improvement of a Hyperchaotic Image Encryption Scheme

Abstract. In recent years, due to the demand for security, image encryption systems based on chaotic systems have gained relevance due to their potential application in computers and mobile devices to secure the transfer and storage of images and videos. In this work, we analyze and test a grayscale image encryption scheme based on a multi-threaded hyperchaotic system. This scheme consists of a symmetric key algorithm that uses a novel Pseudo-Random Bit Generator (PRG), capable of generating binary sequences using the four states of

a multi-threaded hyperchaotic system. Although this proposal was evaluated through various security tests (histogram analysis, entropy, etc.) and the PRG passes the NIST (National Institute of Standards and Technology) tests, this system presents a weakness when applying a cryptanalytic attack, known as a chosen-plain image attack (CPIA). This paper presents a cryptanalysis of the system, whereby, with two arbitrary images, it is possible to recover the original information without knowing the secret key. Furthermore, an improvement is proposed to prevent information leakage, based on a value substitution function using cellular automata.

Keywords: Cryptanalysis, image encryption, chosen-plain image attack.

1. Introducción

En la actualidad, el uso generalizado de teléfonos inteligentes y otros tipos de dispositivos ha convertido a las imágenes digitales en uno de los formatos de intercambio de información, más importantes en la vida diaria de las personas.

Asimismo, estas imágenes pueden contener información sensible de diverso tipo, como la seguridad nacional, asuntos legales y/o médicos, entre otros. Según Wu [1], la exposición no planificada de imágenes privadas (fotos) acentúa la importancia de la seguridad de la imagen.

El paradigma del cifrado de cualquier tipo de imágenes (incluidos videos) tiene sus propios requisitos de implementación, principalmente por sus características intrínsecas, por ejemplo, una capacidad de datos masivos y una alta redundancia. A medida que aumenta el ancho de banda en los canales de comunicación, surge un nuevo desafío, un nuevo conjunto de soluciones criptográficas debe mantenerse al día con la gran cantidad de información que debe protegerse. Actualmente, los esquemas de encriptación de imágenes basados en sistemas hipercaóticos han ganado relevancia, siendo una opción que puede brindar seguridad.

En 1989, Matthews publicó el trabajo [2], que se considera una de las primeras propuestas de un esquema de cifrado basado en el caos. En los últimos años, con el deseo de ampliar este enfoque, se han realizado algunas iniciativas y esfuerzos para aplicar diferentes algoritmos de caos a la criptografía [3, 4, 5]. Debido a que existen algunas propiedades que en ambos casos podrían ser similares, por ejemplo, propiedades como la ergodicidad y la sensibilidad de los sistemas caóticos podrían estar relacionadas con las propiedades de confusión y difusión en el cifrado [6, 7]. Entre algunas opciones para el cifrado de imágenes, hay principalmente dos tipos de arquitecturas clásicas consideradas para el cifrado de imágenes: arquitectura de permutación-sustitución y arquitectura de solo permutación.

Fridrich en [8] propuso un cifrado de imagen caótico, que posteriormente se mejoró con otras propuestas como [9, 10, 11, 12], en las que proponen algoritmos enfocados principalmente en nuevas estrategias de permutación y esquemas de difusión. Por ejemplo, en [10] se muestra un algoritmo con permutación caótica con una matriz ergódica. La difusión se lleva a cabo con un sistema dinámico caótico integrado con una operación algebraica S-box, para realizar una sustitución similar a la que se lleva a cabo con las cajas S-box. Y también cuenta con una operación 'XOR plus mod' que se aplica a cada píxel. Asimismo, en [11] los autores propusieron un esquema de cifrado

basado en mapas caóticos bidimensionales. Esta propuesta se basa en la idea de que la función de mapeo baker map bidimensional se amplía para que sea tridimensional y luego se utiliza para componer un esquema de cifrado de imágenes seguro y rápido.

Por otro lado, se ha considerado la arquitectura de solo permutación debido a las ventajas mostradas en propuestas como [13, 14]. En [13] los autores proponen un algoritmo para el cifrado jerárquico de imágenes caóticas, y utilizan una arquitectura VLSI (Very Large Scale Integration). Mientras tanto, en [14] se muestra un algoritmo de esquema de cifrado basado principalmente en las secuencias hipercaóticas mejoradas. Las propuestas antes mencionadas [8-14] tienen un inconveniente en común, son inseguras.

La propuesta presentada en [8], fue analizada por Solak et al. y como mencionan en su trabajo [15], a través del ataque chosen-ciphertext attack (CCA), el algoritmo mostrado en [8] puede ser descifrado.

Por otro lado, las propuestas [9, 10, 11, 12] se centran en la implementación de algoritmos cifrados de arquitectura permutación-sustitución, también se han mostrado inseguras, esto fue expuesto por Li et al. en [16]. Donde se puede revelar la clave equivalente de los esquemas si solo se realiza una ronda de cifrado.

Las propuestas [13, 14], que son algoritmos de solo permutación, han sido infringidas como se muestra en los trabajos [17, 18].

En la misma área de oportunidad de las propuestas de encriptación de imágenes con enfoque caótico, García-Martínez en [19] presentó un esquema de encriptación de imágenes basado en un sistema hipercaótico multienroscado, para generar secuencias pseudoaleatorias. Los autores propusieron un nuevo Generador de Bits Pseudo-Aleatorio (GBPA) capaz de generar secuencias binarias. Para esto implementaron un algoritmo capaz de utilizar los cuatro estados de un sistema hipercaótico multienroscado.

Cabe destacar que esta propuesta fue validada a través de varias pruebas de seguridad, por ejemplo, el análisis de espacios de llave, entropía, análisis de histogramas, entre otros. Dada la originalidad del esquema presentado por García-Martínez et al. y que este esquema puede ser una buena solución para mejorar las tareas de cifrado de imágenes en escala de grises, decidimos estudiar esta propuesta y realizar un criptoanálisis basado en un ataque específico para encontrar una posible vulnerabilidad. Teniendo en cuenta lo anterior, en este trabajo propusimos un criptoanálisis usando el ataque chosen-plain image attack. Con el cual pudimos recuperar las imágenes originales sin conocer la llave secreta, utilizando dos imágenes arbitrarias. Esperamos que, con este trabajo, diseñadores de criptosistemas aprendan sobre los ataques criptoanalíticos y sus consideraciones, y los contemplen en sus propuestas.

Este trabajo se organiza de la siguiente manera, en la sección 2 se describe el esquema de cifrado propuesto por García-Martínez et al. [19]. En el apartado 3 se explica a profundidad nuestra propuesta de criptoanálisis. La sección 4 presenta una mejora propuesta para el esquema de cifrado y una validación. Y finalmente, la sección 5 presenta las conclusiones de este trabajo.

2. Sistema de cifrado

En [19], los autores proponen un esquema de encriptación para imágenes en escala de grises utilizando un nuevo GBPA basado en las series de tiempo obtenidas de los estados del sistema hipercaótico multienroscado, dados por las ecs. 1 con 2 para 2, 4 o 10 enroscados:

$$\dot{\boldsymbol{X}} = \boldsymbol{AX} + \boldsymbol{B}(\mathbf{X}), \boldsymbol{B}(\mathbf{X}) = \begin{cases} B_1, if\ \boldsymbol{X} \in \mathcal{D}_1; \\ B_2, if\ \boldsymbol{X} \in \mathcal{D}_2; \\ \vdots \\ B_k, if\ \boldsymbol{X} \in \mathcal{D}_k, \end{cases}, \tag{1}$$

$$\boldsymbol{A} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -1.5 & -1 & -1 & 0 \\ 0 & -1 & 0 & -1 \end{pmatrix}, \boldsymbol{B} = \begin{pmatrix} 0 \\ 0 \\ B_3 \\ B_4 \end{pmatrix}, \tag{2}$$

donde $\boldsymbol{R}^4 = \cup_{i=1}^{k} \mathcal{D}_i$, $\boldsymbol{X} = [x_1, x_2, x_3, x_4]^T \in \boldsymbol{R}^4$ es el vector de estado. $\boldsymbol{A} = [a_{ij}] \in \boldsymbol{R}^{4\times4}, i, j = 1,2,3,4$ denota una matriz real y $\boldsymbol{B} = [b_1, b_2, b_3, b_4]^T \in \boldsymbol{R}^4$ representa un vector real. El proceso para generar las secuencias con el GBPA, consiste en iterar el sistema n veces para obtener una secuencia $\boldsymbol{X}$ después de 1000 iteraciones del estado transitorio. Cada conjunto de condiciones iniciales $\boldsymbol{X}_{0p}$ se puede considerar como una llave para el cifrado, donde $p \in \boldsymbol{Z}^+$, da como resultado p series de tiempo diferentes. El GBPA está definido por la ecuación 3:

$$k_i = \left\lfloor \sum_{j=1}^{4} x_j(i) \cdot 10^{14} \right\rfloor mod\ 256. \tag{3}$$

En este caso, las secuencias pseudoaleatorias producidas por GBPA se denominan k_i, donde $k_i \in \{0, 1, 2, \dots, 255\}, i = 1, \dots, n$, y n es el número total de píxeles en la imagen de entrada. La operación $\lfloor \cdot \rfloor$ implica la función floor, una función de redondeo de enteros. El factor 10^{14} permite escalar la suma, porque el simulador numérico trabaja con valores de punto flotante. Finalmente, calculando la operación $mod\ 256$, cada valor de k es un entero positivo de 8 bits.

Una vez explicada la generación de las secuencias pseudoaleatorias k, procederemos a explicar las operaciones de cifrado. Las imágenes son cifradas píxel a píxel utilizando las siguientes ecuaciones:

$$\begin{cases} C_1 = P_1 \oplus k_1 \oplus \boldsymbol{IV} \\ C_i = P_i \oplus k_i \oplus C_{i-1}, \end{cases} \tag{4}$$

donde C y P representan los pixeles cifrados y en claro respectivamente, con $i = 2 \dots n$. $\boldsymbol{IV}$ representa un vector inicial de 8 bits, k es una secuencia aleatoria de 8 bits obtenida con el GBPA. Y por último, el símbolo $\oplus$ representa la operación XOR. Para obtener el primer pixel cifrado C_1, se calcula una operación XOR entre el coeficiente del pixel P_1, la secuencia pseudoaleatoria k_1 y el vector inicial $\boldsymbol{IV}$. Los demás pixeles cifrados C_i, se

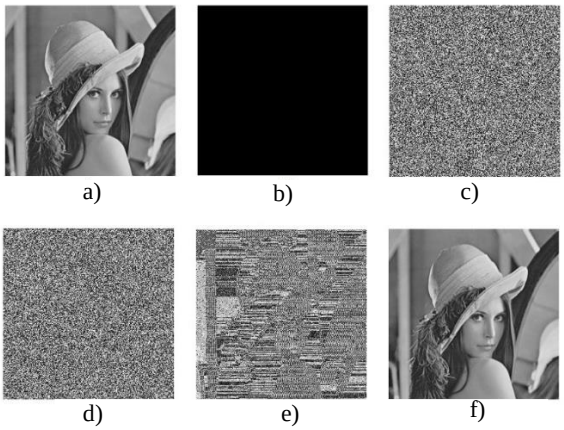


Fig. 1. Chosen-plain image attack. a) Imagen de Lena, b) imagen solida negra, c) imagen de Lena encriptada, d) máscara $\boldsymbol{I}_M$, e) imagen recuperada con la operación XOR, f) imagen recuperada usando la ec. (3).

obtienen realizando una operación XOR entre el coeficiente del pixel P_i, una nueva secuencia aleatoria k_i y el pixel cifrado previo C_{i-1}. Este proceso se repite hasta cifrar todos los pixeles de la imagen. Las condiciones iniciales del GBPA funcionan como llave secreta.

Como podemos observar el esquema cifra directamente los coeficientes de los pixeles, sin utilizar previamente una operación de sustitución, esto permite a los atacantes introducir datos de manera arbitraria.

3. Criptoanálisis

Para definir lo que es una ruptura o debilidad de un esquema de cifrado, es conveniente mencionar antes, que para esto las reglas de la criptografía académica se relajan considerablemente. Schneier en [20] define que una ruptura en un sistema cifrado es cuando se encuentra una debilidad que se puede explotar con menos complejidad o trabajo que con fuerza-bruta. Recordemos que fuerza-bruta se refiere a los ataques donde se trata de adivinar la llave de cifrado, probando todas las posibles claves para cifrar o descifrar la información. Por lo tanto, se considera que se encontró una debilidad aún cuando en el proceso se requiera una gran cantidad de cifrados o descifrados. La limitante aquí es que sea una cantidad menor que hacerlo por fuerza bruta. Por lo que para demostrar que existe una debilidad, se puede requerir de grandes cantidades de texto plano, texto plano conocido o texto plano elegido, que pueden ser consideradas poco realistas. Es por eso por lo que mencionamos que las reglas de la criptografía se relajan en estos estudios, el propósito se vuelve solo el determinar si el sistema es seguro o no.

En el ataque chosen-plain image attack, el atacante es capaz de seleccionar las imágenes en claro y obtener sus respectivas versiones cifradas [21, 22]. Sin embargo, no posee la llave secreta. Para ilustrar este ataque y todo su proceso, utilizaremos la fig. 1. El ataque comienza seleccionando las dos imágenes en claro, en este caso la imagen de Lena, fig. 1a), y una imagen solida negra, fig. 1b). Ciframos ambas imágenes con el algoritmo de García-Martínez, con las mismas condiciones iniciales. Las imágenes que se obtienen son, fig. 1c), Lena cifrada, y la fig. 1d), la imagen solida cifrada, la cual

llamaremos Mascara $\boldsymbol{I}_M$. Para recuperar la imagen de Lena sin conocer las condiciones inciales basta con calcular una operación XOR entre ambas imágenes cifradas, píxel a píxel. El resultado lo podemos ver en la fig. 1e), que como se puede ver esta imagen revela patrones muy significativos de la fig. 1a). Aún así el proceso de recuperación se puede mejorar, y sin conocer las condiciones iniciales la imagen se puede recuperar al 100%, como se puede ver en la fig. 1f). El proceso para obtener esta imagen será explicado a detalle más adelante.

Para ilustrar la debilidad observemos la ec. (5), esta ecuación representa el paso del ataque donde se hace la operación XOR a nivel de bit, entre píxeles de ambas imágenes:

$$C_{L1} \oplus C_{M1} = (\mathrm{P_{L1}} \oplus \mathrm{k_1} \oplus \mathbf{IV}) \oplus (0 \oplus \mathrm{k_1} \oplus \mathbf{IV}) = P_{L1}, \tag{5}$$

donde C_{L1} representa el primer pixel de la imagen de Lena cifrada, C_{M1} representa el primer pixel de la Mascara $\boldsymbol{I}_M$. Y por último, P_{L1} representa el primer pixel de la imagen de Lena. Como se puede ver, al momento de calcular la operación XOR entre ambos pixeles bit a bit, los elementos de la ecuación de cifrado k_1 y $\boldsymbol{IV}$ se reducen aplicando algebra booleana (la operación XOR de elementos idénticos es igual a cero), y el resultado es el coeficiente de pixel P_{L1}. Para recuperar los demás pixeles se utiliza (6), esta versión es la usada para recuperar la fig. 1f). Para mejorar su desempeño en la recuperación, agregamos una operación XOR entre los pixeles cifrados previos, adicional a la de píxel a píxel, como se muestra a continuación:

$$\begin{aligned}(\mathrm{C_{Li}} \oplus \mathrm{C_{Mi}}) \oplus (\mathrm{C_{Li-1}} \oplus \mathrm{C_{Mi-1}}) = (\mathrm{P_{Li}} \oplus \mathrm{k_i} \oplus \mathrm{C_{Li-1}}) \oplus (0 \oplus \mathrm{k_i} \oplus \mathrm{C_{Mi-1}}) \\ \oplus (\mathrm{C_{Li-1}} \oplus \mathrm{C_{Mi-1}}) = P_{Li},\end{aligned} \tag{6}$$

donde C_{Li} y C_{Mi} representan los pixeles cifrados en la imagen de Lena cifrada y de la máscara, posteriores al primero. Los valores C_{Li-1} y C_{Mi-1}, representan el píxel previo cifrado, que es un elemento conocido o disponible para el atacante en la imagen cifrada, que también es usado en las ecuaciones de cifrado, por lo tanto, se puede anular. Como se puede ver, es posible recuperar la imagen original de Lena, sin conocer la llave secreta, solo se requieren dos imágenes.

Este ataque funciona en este esquema, posiblemente por varias razones, pero este trabajo se enfoca en la falta de una función de sustitución. Esto para evitar que el adversario introduzca valores en las ecuaciones a su conveniencia.

4. Mejora propuesta para el esquema

Para mejorar el esquema propuesto por García-Martínez, se sugiere agregar una función de preprocesamiento, capaz de sustituir el texto plano antes de ser cifrado. Sin importar que el texto en claro sea altamente redundante, esta función debe intercambiarlo por diferentes valores de la codificación con igual probabilidad. Si se desea utilizar una caja de sustitución, el esquema debería modificarse aún más, ya que simplemente intercambiar el texto con una S-box antes de cifrar, crearía patrones de la imagen original.

La función de preprocesamiento utilizada en esta mejora se encuentra en la patente [23], que fue presentada en el trabajo [24]. Esta función se basa en la sincronización de autómatas celulares usando la regla local 90. Gracias a su retroalimentación, la función

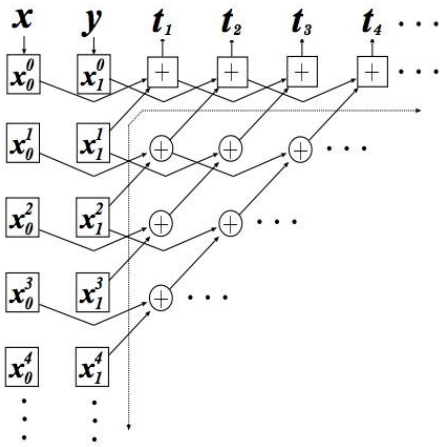


Fig. 2. Generador de secuencias pseudoaleatorias $\boldsymbol{t}$ de n bits, basado en la evolución hacia atrás de la regla 90.

de preprocesamiento puede intercambiar valores idénticos por números diferentes en cada iteración.

Para explicar su funcionamiento, en la fig. 2 se ilustra el generador pseudoaleatorio y el proceso de evolución hacia atrás usando la regla local 90. Se utiliza un vector $\boldsymbol{x}$ y un vector $\boldsymbol{y}$ de n bits y $n+1$ bits, respectivamente. Se evoluciona hacia atrás, utilizando la operación XOR como indican las flechas, hasta generar el vector $\boldsymbol{t}$. Esta función es llamada *h*.

Para aplicar esta operación como función de preprocesamiento, en la fig. 3, podemos observar a la función h como un bloque, y en el lugar del vector $\boldsymbol{x}$, entra el coeficiente del pixel $\boldsymbol{p}$, y en el lugar del vector $\boldsymbol{y}$, un nuevo vector llamado $\boldsymbol{z}$. Esto para diferenciar su funcionamiento como generador de números pseudoaletorios y como función de preprocesamiento. El vector de salida es llamado $\widehat{\boldsymbol{p}}$, para señalar que es la versión preprocesada de $\boldsymbol{p}$.

Como se puede ver existe una retroalimentación que actualiza el vector $\boldsymbol{z}$, este nuevo vector se calcula con el vector de salida $\widehat{\boldsymbol{p}}$, concatenando el bit menos significativo del vector $\boldsymbol{z}$ anterior, en la posición de bit más significativo.

La diferencia entre esta función y una caja de sustitución la podemos observar en la fig. 4, donde se intercambian los pixeles de la imagen de Lena, primero con una S-Box y en la segunda imagen con nuestra función.

Se puede observar en la fig. 4, que en el caso de la caja se sustitución se crean patrones, debido a que no existe ningún tipo de dinámica, los coeficientes de la imagen son sustituidos siempre por los mismos valores de la S-box.

En la fig. 5 se muestra el histograma de la imagen de Lena y su versión preprocesada. Como se puede ver el histograma se vuelve uniforme, ocultando la redundancia de la imagen original y previniendo un ataque estadístico.

Al agregar la función de preprocesamiento al algoritmo de García-Martínez et al., antes del cifrado, mejora su desempeño ante el ataque CPIA. Lo anterior se puede confirmar en la fig. 6, donde nuevamente se escogen las mismas dos imágenes de forma arbitraria, y se cifran con las mismas condiciones iniciales.

Como podemos observar el atacante no puede introducir el valor de 0 de forma arbitraria en las ecuaciones de cifrado, porque éste se intercambia. Esto evita que capture

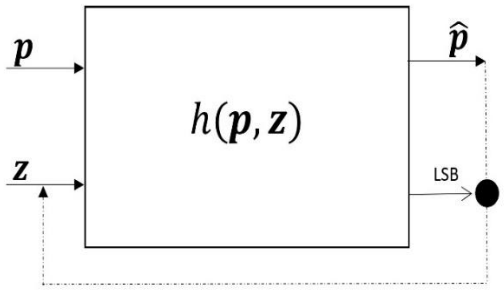
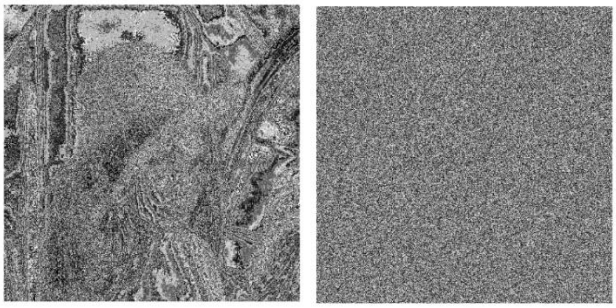


Fig. 3. Función de preprocesamiento. $\boldsymbol{p}$ es el coeficiente del pixel, $\boldsymbol{z}$ un vector aleatorio. A la salida se obtiene el vector preprocesado $\hat{\boldsymbol{p}}$.



a) b)

Fig. 4. Imagen de Lena sustituida, a) imagen de Lena sustituida con una S-box del sistema AES (Advanced Encryption Standar), b) imagen de Lena despues de ser preprocesada con la función h.

información en la Máscara $\boldsymbol{I}_M$. Además, la información no está cifrada de manera directa, el preprocesamiento intercambia los valores en ambas imágenes.

Por lo tanto, ante las condiciones que se dan en este ataque, no queda expuesta la información aún cuando se cifren los píxeles con las mismas secuencias k_i y $\boldsymbol{IV}$ y que con el ataque se anulen. Para ilustrar esto podemos analizar la ecuación (7):

$$C_{L1} \oplus C_{M1} = \left(\hat{\mathrm{P}}_{\mathrm{L1}} \oplus \mathrm{k}_1 \oplus \mathbf{IV}\right) \oplus \left(\hat{\mathrm{P}}_{\mathrm{IM1}} \oplus \mathrm{k}_1 \oplus \mathbf{IV}\right) = \hat{P}_{L1} \oplus \hat{P}_{IM1}. \qquad (7)$$

5. Conclusiones

Los sistemas caóticos pueden aportar elementos en el diseño de sistemas de cifrado. Sin embargo, su análisis debe ser profundo e interpretar de manera general las pruebas que se aplican a los esquemas de cifrado. Debido a que existe una gran variedad de

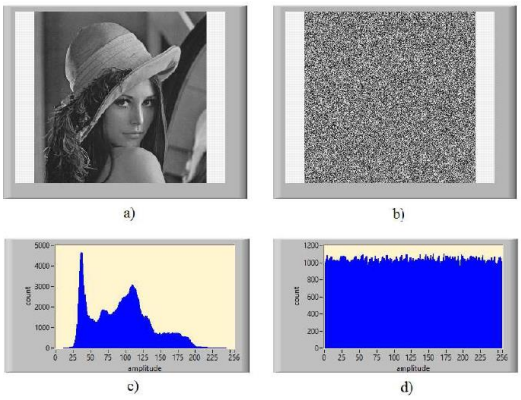


Fig. 5. Análisis de histogramas. a) Imagen de Lena, b)imagen preprocesada de Lena, c)histograma de a) y d)histograma de b).

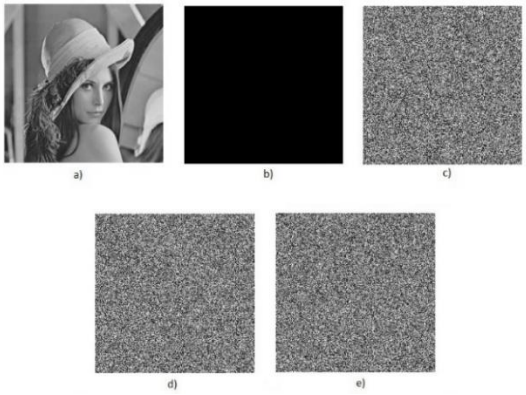


Fig. 6. Chosen-plain image attack. a) Imagen de Lena, b)imagen solida negra, c)imagen de Lena encriptada, d)máscara, e) imagen recuperada con la operación XOR.

ataques con diversos supuestos que permiten concluir cuando un esquema se puede considerar seguro acorde a alguna definición; los diseñadores deben incluir más pruebas en sus propuestas y ser muy explícitos en la forma en que se aplicaron.

En este trabajo se mostró como se puede llevar a cabo una prueba de criptoanálisis. En algún momento se puede pensar que el adversario tiene muchas ventajas, como el hecho de que se utilizan las mismas condiciones iniciales para cifrar ambas imágenes. Estas pruebas con estos mismos supuestos son aprobadas por los esquemas de cifrado que son referente, como lo es el sistema AES [25]. Como se mencionó antes, las reglas de criptografía se relajan cuando se aplica un criptoanálisis, y el objetivo simplemente es definir si el algoritmo analizado es seguro o no.

El cifrado de imágenes sigue presentado áreas de oportunidad y desarrollo, porque cuando un sistema logra solventar las problemáticas de seguridad incrementa su latencia. Por lo que la búsqueda de nuevos métodos se mantiene aún en auge y los sistemas caóticos pueden brindar soluciones a este problema.

Referencias

1. Wu, Y., Noonan, J.P., Yang, G.: Image Encryption Using the Two-Dimensional Logistic Chaotic Map. Journal of Electronic Imaging, 21(1), 013014 (2012) doi: 10.1117/ 1.JEI.21.1.013014.
2. Matthews, R.: On the Derivation of a "Chaotic" Encryption Algorithm. Cryptologia, 8(1), pp. 29–41 (1984) doi: 10.5555/67071.67073.
3. Li, Z., Peng, C., Tan, W.: A Novel Chaos-Based Image Encryption Scheme by Using Randomly DNA Encode and Plaintext Related Permutation. Applied Sciences, 10(21), 7469 (2020) doi: 10. 10.3390/app10217469.
4. Liao, T.L., Lin, H.R., Wan, P.Y.: Improved Attribute-Based Encryption Using Chaos Synchronization and its Application to MQTT Security. Applied Sciences, 9(20), 4454 (2019) doi: 10.3390/app9204454.
5. Datcu, O., Macovei, C., Hobincu, R.: Chaos Based Cryptographic Pseudo-Random Number Generator Template with Dynamic State Change. Applied Sciences, 10(2), 451 (2020) doi: 10.451.10.3390/app10020451.
6. Alvarez, G., Montoya, P., Pastor, G.: Chaotic Cryptosystems. In: Proceedings IEEE 33rd Annual 1999 International Carnahan Conference on Security Technology, pp. 332–338 (1999) doi: 10.1109/CCST.1999.797933.
7. Kocarev, L., Jakimoski, G., Stojanovski, T.: From Chaotic Maps to Encryption Schemes. In: ISCAS'98. Proceedings of the 1998 IEEE International Symposium on Circuits and Systems, 4, pp. 514–517 (1998)
8. Fridrich, J.: Symmetric Ciphers Based on Two-Dimensional Chaotic Maps. International Journal of Bifurcation and chaos, 8(06), pp. 1259–1284 (1998) doi: 10.1142/S0218 12749800098X.
9. Chen, G., Mao, Y., Chui, C.K.: A Symmetric Image Encryption Scheme Based on 3D Chaotic Cat Maps. Chaos, Solitons & Fractals, 21(3), pp. 749–761 (2004) doi: 10.1016/j.chaos.2003.12.022.
10. He, X., Zhu, Q., Gu, P.: A New Chaos-Based Encryption Method for Color Image. In: International Conference on Rough Sets and Knowledge Technology Springer, Berlin, Heidelberg, pp. 671–678 (2006) doi: 10.1007/11795131_98.
11. Mao, Y., Chen, G., Lian, S.: A Novel Fast Image Encryption Scheme Based on 3D Chaotic Baker Maps. International Journal of Bifurcation and chaos, 14(10), pp. 3613–3624 (2004) doi: 10.1142/S021812740401151X.
12. Shen, J., Jin, X., Zhou, C.: A Color Image Encryption Algorithm Based on Magic Cube Transformation and Modular Arithmetic Operation. In: Pacific-Rim Conference on Multimedia Springer, pp. 270–280 (2005) doi: 10.1007/11582267_24.
13. Yen, J.C., Guo, J.I.: Efficient Hierarchical Chaotic Image Encryption Algorithm and its VLSI Realisation. In: IEE Proceedings-Vision, Image and Signal Processing, 147(2), pp.167–175 (2000) doi: 10.1049/ip-vis:20000208.
14. Zhu, C.: A Novel Image Encryption Scheme Based on Improved Hyperchaotic Sequences. Optics communications, 285(1), pp. 29–37 (2012) doi: 10.1016/j.optcom.2011.08.079.
15. Solak, E., Cokal, C., Yildiz, O.T.: Cryptanalysis of Fridrich's Chaotic Image Encryption. International Journal of Bifurcation and Chaos, 20(05), pp. 1405–1413 (2010) doi: 10.1142/S0218127410026563.
16. Li, C., Chen, G.: On the Security of a Class of Image Encryption Schemes. In: 2008 IEEE International Symposium on Circuits and Systems IEEE pp. 3290–3293 (2008) doi: 10.1109/ISCAS.2008.4542161.
17. Li, C.: Cracking a Hierarchical Chaotic Image Encryption Algorithm Based on Permutation. Signal Processing, 118, pp. 203–210 (2016) doi: 10.1016/j.sigpro. 2015.07.008.

18. Li, C., Liu, Y., Xie, T.: Breaking a Novel Image Encryption Scheme Based on Improved Hyperchaotic Sequences. Nonlinear Dynamics, 73(3), pp. 2083–2089 (2013) doi: 10.1007/s11071-013-0924-6.
19. García-Martínez, M., Ontañón-García, L.J., Campos-Cantón, E.: Hyperchaotic Encryption Based on Multi-Scroll Piecewise Linear Systems. Applied Mathematics and Computation, 270, pp. 413–424 (2015) doi: 10.1016/j.amc.2015.08.037.
20. Schneier, B.: A Self-Study Course in Block-Cipher Cryptanalysis. Cryptologia, 24(1), pp. 18–33 (2000) doi: 10.1080/0161-110091888754.
21. Munir, N., Khan, M., Shah, T.: Cryptanalysis of Nonlinear Confusion Component Based Encryption Algorithm. Integration, 79, pp. 41–47 (2021) doi: 10.1016/j.vlsi.2021.03.004.
22. Li, S., Li, C., Chen, G.: Cryptanalysis of the RCES/RSES Image Encryption Scheme. Journal of Systems and Software, 81(7), pp. 1130–1143 (2008) doi: 10.1016/j.jss. 2007.07.037.
23. Mejía Carlos, M., Murguía Ibarra, J.S., Ramírez Torres, M.T.: Sistema de cifrado de datos digitales IMPI (MX2015008665A) (2015)
24. Ramírez-Torres, M.T., Murguia, J.S., Mejía-Carlos, M.: FPGA Implementation of a Reconfigurable Image Encryption System. In: 2014 International Conference on ReConFigurable Computing and FPGAs (ReConFig14) IEEE, pp. 1–4 (2014) doi: 10.1109/ReConFig.2014.7032524.
25. Heron, S.: Advanced Encryption Standard (AES). Network Security, 2009(12), pp. 8–12 (2009) doi: 10.1016/S1353-4858(10)70006-4.