UNIVERSIDAD DE ORIENTE
NÚCLEO DE ANZOÁTEGUI
ESCUELA DE INGENIERÍA Y CIENCIAS APLICADAS
DEPARTAMENTO DE COMPUTACIÓN Y SISTEMAS

DINÁMICA REVERSIBLE DE AUTÓMATAS CELULARES BIDIMENSIONALES PARA EL CIFRADO DE IMÁGENES DIGITALES A COLOR

Por:

Prof. José Luis Bastardo

Trabajo Ascenso presentado como requisito parcial para optar a la categoría de Profesor Titular

Barcelona, Febrero de 2014

RESOLUCIÓN

De acuerdo al artículo 41 del Reglamento de Trabajo de Grado:

"Los trabajos son propiedad exclusiva de la Universidad de Oriente, y sólo podrán ser utilizados para otros fines con el consentimiento expreso del Consejo de Núcleo respectivo, quien deberá participarlo previamente al Consejo Universitario para su autorización"

RESUMEN

La transferencia de la data de una imagen digital al espacio celular de un autómata celular bidimensional hace posible realizar un procesamiento sobre la imagen original mediante la aplicación reiterada de la regla del autómata sobre todas las vecindades del espacio celular. La regla del autómata y la disposición de las vecindades determinan una dinámica que hace que el espacio celular pase por diversas configuraciones, recorriendo una trayectoria que en su estado inicial refleja la imagen digital original y en su estado final contiene la imagen digital procesada. En una dinámica de autómata celular reversible, asociada a la regla, existe una regla inversa que permite recorrer la trayectoria anterior en sentido contrario. En este trabajo se hizo uso de la simplicidad de las reglas de autómatas celulares reversibles y de su capacidad para generar desorden o desinformación, sobre la configuración inicial del espacio celular, para desarrollar una dinámica reversible que permite el cifrado de imágenes digitales. La dinámica desarrollada propone un esquema de disposición de las vecindades que modifica el número de células que conforman vecindades, en el espacio celular. Se construyó una descripción formal de la dinámica propuesta, esta descripción se codificó en el lenguaje de programación Java y se realizaron diversas pruebas que demuestran la calidad del cifrado ofrecido.

DEDICATORIA

Para Alejandra, quien canta al hablar.

AGRADECIMIENTOS

A todo el personal del Departamento de Computación y Sistemas de la Escuela de Ingeniería y Ciencias Aplicadas del Núcleo de Anzoátegui de la Universidad de Oriente, por su incondicional apoyo y camaradería.

ÍNDICE GENERAL

PÁG.

ÍNDICE DE FIGURAS

PÁG.

ÍNDICE DE TABLAS

PÁG.

INTRODUCCIÓN

Con la aparición de los sistemas distribuidos y el uso de redes en las cuales la información es intercambiada entre dispositivos ubicados alrededor del mundo, surge un entorno con diversos y estrictos requerimientos de seguridad. Uno de estos requerimientos es la confidencialidad. Ofrecer confidencialidad es asegurar que la información transmitida sea accesible sólo por las partes autorizadas. Para satisfacer el requerimiento de confidencialidad se han propuesto diversos tipos de mecanismos entre los cuales el más importante, con diferencia, es el cifrado o encriptamiento de la información.

Los autómatas celulares constituyen sistemas dinámicos completamente discretos en los cuales un conjunto de células, distribuidas sobre un reticulado regular n-dimensional, reflejan estados pertenecientes a un alfabeto finito.

La transferencia de la data de una imagen digital al reticulado bidimensional de un autómata celular hace posible realizar un procesamiento sobre la imagen original mediante la aplicación reiterada de la regla del autómata celular sobre todas las vecindades del reticulado regular o espacio celular.

En este trabajo se presenta una descripción formal de una dinámica de autómata celular propuesta como criptosistema, esta descripción se codifica en el lenguaje de programación Java y el código resultante es usado para realizar diversas pruebas que permiten determinar la calidad del encriptamiento ofrecido por la dinámica.

CAPÍTULO I: EL PROBLEMA

1.1 Planteamiento del Problema

Los sistemas distribuidos y la Internet ofrecen un entorno en el cual la información es intercambiada entre dispositivos ubicados alrededor del mundo. Para el correcto funcionamiento de este entorno es necesario el cumplimiento de estrictos requerimientos de seguridad. Uno de estos requerimientos es la confidencialidad. Ofrecer confidencialidad es asegurar que la información transmitida sea accesible sólo por las partes autorizadas. Para satisfacer el requerimiento de confidencialidad se han propuesto diversos tipos de mecanismos entre los cuales el más importante, con diferencia, es el cifrado o encriptamiento de la información.

El encriptamiento convencional, también denominado encriptamiento simétrico o encriptamiento de clave simple, fue el único tipo de encriptamiento usado antes del desarrollo del encriptamiento de clave pública. De estos dos tipos principales de encriptamiento, el encriptamiento simétrico continúa siendo el más utilizado (Stallings, 2011).

En el proceso de encriptamiento simétrico, el mensaje legible original denominado *texto plano*, es convertido en un mensaje aparentemente aleatorio y sin sentido denominado *texto cifrado*. El proceso de encriptamiento consta de un algoritmo y una clave. La clave es un valor independiente del texto plano. Para un mismo texto plano el algoritmo producirá salidas diferentes dependiendo de la clave específica que se haya utilizado. Al cambiar la clave cambia la salida del algoritmo.

Una vez que el texto cifrado ha sido producido, este puede ser transmitido. Luego de la recepción, el texto cifrado puede ser transformado en el texto plano original usando un algoritmo de desencriptamiento y la misma clave que fue usada para el encriptamiento.

La seguridad del encriptamiento simétrico depende de varios factores. Ante todo, el algoritmo de encriptamiento debe ser lo suficientemente robusto como para que resulte impráctico intentar desencriptar un mensaje contando sólo con el texto cifrado. Aún más, la seguridad del encriptamiento simétrico depende del secreto de la clave, no del secreto del algoritmo de encriptamiento. Es decir, se asume que debe resultar impráctico intentar desencriptar un mensaje a partir del mensaje cifrado y el conocimiento del funcionamiento del algoritmo de encriptamiento/desencriptamiento. No es necesario mantener el secreto del algoritmo; sólo es necesario mantener el secreto de la clave.

No tener necesidad de mantener el secreto del algoritmo es una de las razones fundamentales del amplio uso del encriptamiento simétrico. Dado que el algoritmo no es mantenido en secreto, los fabricantes han podido desarrollar chips de bajo costo que implementan estos algoritmos. Estos chips están ampliamente disponibles y son incorporados a numerosos dispositivos.

El amplio campo de aplicación del encriptamiento simétrico ha orientado los esfuerzos de numerosos investigadores impulsando la creación de esquemas de encriptamiento simétrico basados en los más diversos fundamentos. En 1986, Steven Wolfran propuso un esquema de encriptamiento simétrico basado en el uso de Autómatas Celulares (Wolfran, 1986a).

Un Autómata Celular de tamaño *m* es un sistema dinámico con *m* localidades o *celdas* $(x_1^t, x_2^t, \ldots, x_m^t) = x^t$, junto a un conjunto de funciones $\{F_i^t\}$, tal que en cada instante de tiempo discreto t, $x_i^{t+1} = F_i^t(x_1^t, x_2^t, \ldots, x_m^t)$, donde *x* asume valores tomados de un conjunto finito *S*.

Debe destacarse que, en general, para el cálculo de x_i^{t+1} la función F_i^t usa como argumentos los elementos de un subconjunto de $\{x_1^t, x_2^t, \cdots, x_m^t\}$, este subconjunto es denominado la *vecindad* de x_i^{t+1}.

Si $F_i^t = F_j^t$ para todo i, j, se dice entonces que el autómata celular es *homogéneo* y en caso contrario, es decir, cuando existen i ≠ j tales que $F_i^t \neq F_j^t$, se dice que el autómata celular es *no homogéneo*. Si $F_i^t = F_i^s$ para todo t, s e i, se dice que el autómata celular es *estable en el tiempo* y en caso contrario, cuando existen t y s tales que $F_i^t \neq F_i^s$, se dice que el autómata es *variable en el tiempo* (Guan, 1987).

La secuencia de estados por las que atraviesa el sistema a lo largo del tiempo es denominada *trayectoria*. Se denomina *dinámica* a las características de las posibles trayectorias del sistema. Toda dinámica se manifiesta siguiendo un conjunto de reglas. Este conjunto de reglas constituye una descripción o caracterización de la dinámica.

Aquellas dinámicas que permiten el recorrido en sentido contrario de todas sus trayectorias, desde el estado final hasta el estado inicial, se denominan *dinámicas reversibles*.

Steven Wolfran primero propone un autómata celular unidimensional como generador de números pseudoaleatorios (Wolfran, 1986b). Wolfran estudia en profundidad las secuencias generadas por la regla 30 en su esquema de numeración para autómatas celulares uniformes unidimensionales binarios de radio 1, donde el número de la regla representa en formato decimal los números binarios que están codificados en la tabla de la regla. Por ejemplo, f(111) = 1, f(110) = 0, f(101) = 1, f(100) = 0, f(011) = 1, f(010) = 0, f(001) = 1, f(000) =0 es denotada como la regla 170.

La regla 30 puede ser escrita en forma booleana de la siguiente manera:

$$x_i^{t+1} = x_{i-1}^t \text{ XOR } (x_i^t \text{ OR } x_{i+1}^t),$$

donde x_i^t es el estado de la celda i en el tiempo t. La fórmula da el estado de la celda *i* en el tiempo *t+1* como una función booleana de los estados de las celdas vecinas en el tiempo *t*. La secuencia de números pseudoaleatorios a^t es obtenida por el muestreo de los valores que una celda particular alcanza como función del tiempo. El texto cifrado C es obtenido a partir del texto plano P mediante la realización de la operación:

$$C_t = P_t \text{ XOR } a^t$$

El texto plano puede ser recuperado repitiendo la misma operación, sólo si se dispone de la secuencia a^t. En (Meier y Staffelbach, 1991) se describe un ataque, contra este esquema de encriptamiento basado en la regla 30, que resulta exitoso en un tiempo razonable para claves de tamaño menor o igual a 500. Se destaca la uniformidad del autómata celular como la causa de la debilidad del esquema de encriptamiento. En (Nandi, Kar y Chaudhuri, 1994) se presenta un esquema de encriptamiento más robusto

usando autómatas celulares no uniformes que combinan las reglas 90 y 150. Otro esquema de encriptamiento exitoso basados en el uso de autómatas celulares no uniformes es reportado en (Tomassini y Perrenoud, 2000).

Los resultados anteriores sugieren que la no uniformidad es un camino plausible a la hora de proponer esquemas de encriptamiento basados en dinámicas de autómatas celulares.

Debido a la amplia difusión de las cámaras digitales y a su incorporación a los dispositivos móviles, las imágenes digitales constituyen un tipo de data de transito común en la Internet. En una imagen digital a color un conjunto de píxeles dispuestos según una distribución particular sobre una matriz bidimensional representan una imagen. La correlación entre los píxeles vecinos es uno de los indicadores fundamentales de la información contenida en la imagen. Por lo que un método de encriptamiento robusto aplicado a imágenes digitales debe construir una imagen cifrada en la cual se minimiza la correlación entre los píxeles vecinos, ocultando la información de la imagen original.

En este trabajo se propone el desarrollo de una dinámica reversible de autómatas celulares bidimensionales no uniformes como método de encriptamiento robusto de imágenes digitales a color.

1.2 Objetivos

1.2.1 Objetivo General

Desarrollar una dinámica reversible de autómatas celulares bidimensionales no uniformes como método de encriptamiento simétrico robusto de imágenes digitales a color.

1.2.2 Objetivos Específicos

- ✓ Describir en términos matemáticos una dinámica reversible de autómata celular bidimensional no uniforme y dependiente de una clave.
- ✓ Codificar en un lenguaje de programación la descripción de la dinámica de forma tal que el código resultante acepte como entrada imágenes digitales a color.
- ✓ Estimar la robustez de la dinámica como método de encriptamiento de imágenes digitales a color.

CAPÍTULO II: MARCO TEÓRICO

2.1 Criptografía

La *criptografía* (de los vocablos griegos *kriptos*: "ocultar" y *grafos*: "escribir", literalmente "escritura oculta") es el arte o ciencia de cifrar y descifrar información utilizando técnicas matemáticas que hacen posible el intercambio de mensajes de manera que sólo pueden ser leídos por las personas a quienes van dirigidos.

Con más precisión, cuando se habla de esta área de conocimiento como ciencia, se debería hablar de *criptología*, término que engloba tanto las técnicas de cifrado, la criptografía propiamente dicha, como las técnicas complementarias de *criptoanálisis*, que estudian los métodos que se utilizan para romper textos cifrados con objeto de recuperar la información original en ausencia de la clave.

La finalidad de la criptografía es, en primer lugar, garantizar el secreto en la comunicación entre dos entidades (personas, organizaciones, etc.) y, en segundo lugar, asegurar que la información que se envía es auténtica en un doble sentido, que el remitente sea realmente quien dice ser y que el contenido del mensaje enviado, habitualmente denominado *criptosistema*, no haya sido modificado en su tránsito.

2.1.1 Conceptos

En la jerga de la criptografía, la información original que debe protegerse se denomina texto plano. El cifrado es el proceso de convertir el texto plano en un galimatías irreconocible, denominado texto cifrado o

criptograma. Por lo general, la aplicación concreta del algoritmo de cifrado (también llamado cifra) se basa en la existencia de una clave: información secreta que adapta el algoritmo de cifrado para cada uso diferente. Las dos técnicas más básicas de cifrado en la criptografía clásica son la sustitución (que supone el cambio de significado de los elementos básicos del mensaje –las letras, los dígitos o los símbolos-) y la transposición (que supone una reordenación de las mismas); la gran mayoría de los cifrados clásicas son combinaciones de estas dos operaciones básicas. El descifrado es el proceso inverso que recupera el texto plano a partir del criptosistema y la clave. El protocolo criptográfico especifica los detalles de cómo se utilizan los algoritmos de cifrado y las claves (y otras operaciones primitivas) para conseguir el efecto deseado. El conjunto de protocolos, algoritmos de cifrado, procesos de gestión de claves y actuaciones de los usuarios, en su globalidad es lo que constituyen un criptosistema, que es con lo que el usuario final trabaja e interactúa.

Con frecuencia los procesos de cifrado y descifrado se encuentran en la literatura como encriptado y desencriptado, aunque ambos son neologismos todavía sin reconocimiento académico. Hay quien hace distinción entre “cifrado/descifrado” y “encriptado/desencriptado” según esté hablando de criptografía simétrica o asimétrica, pero la mayoría de los expertos en el mundo académico prefiere evitar ambos neologismos.

2.1.2 Criptografía Moderna

La criptografía moderna nace al mismo tiempo que las computadoras, durante la segunda guerra mundial, en un lugar llamado Bletchley Park. Allí un grupo de científicos entre los que se encontraba Alan Turing, trabajaban

en el proyecto “ULTRA” tratando de descifrar los mensajes enviados por el ejército alemán, cifrados con los más sofisticados ingenios de codificación ideados hasta ese entonces, la máquina “ENIGMA” y el cifrado Lorenz. Este grupo de científicos diseñó y utilizó el primer computador de la historia, denominado “COLOSSUS”, aunque esta información permaneció en secreto hasta mediados de los años setenta. Desde entonces hasta hoy ha habido un crecimiento espectacular de la tecnología criptográfica, si bien la mayor parte de estos avances se mantenían y se siguen manteniendo en secreto. Financiadas principalmente por la NSA (siglas en inglés de la Agencia Nacional de Seguridad de los EE. UU.), la mayor parte de las investigaciones hasta hace relativamente poco tiempo han sido tratadas como secretos militares (Ramio, 2006).

Esta dualidad civil-militar ha dado lugar a una curiosa doble historia de la criptografía, en la que los mismos algoritmos eran descubiertos, con pocos años de diferencia, por equipos de anónimos militares y posteriormente por matemáticos civiles, alcanzando únicamente estos últimos el reconocimiento público por sus trabajos. Sin embargo, en los últimos años, investigaciones serias llevadas a cabo en universidades de todo el mundo han logrado que la criptografía sea una ciencia al alcance de todos y que se convierta en la piedra angular de asuntos tan importante como el comercio electrónico, la telefonía móvil, o las nuevas plataformas de distribución de contenidos multimedia.

2.1.3 Criptosistema

Se define un criptosistema como una quíntupla (M,C,K,E,D), donde:

- M representa el conjunto de todos los mensajes sin cifrar (lo que se denomina texto plano).

- C representa el conjunto de todos los posibles mensajes cifrados o criptogramas.
- K representa el conjunto de claves que se pueden emplear en el criptosistema.
- E es el conjunto de transformaciones de cifrado o familia de funciones que se aplica a cada elemento de M para obtener un elemento de C. Existe una transformación diferente E_k para cada valor posible de la clave k.
- D es el conjunto de transformaciones de descifrado, análogo a E.

Todo criptosistema ha de cumplir la siguiente condición:

$$D_k(E_k(m)) = m$$

Es decir, que si un mensaje m, se cifrará empleando la clave k y luego se descifrará empleando la misma clave, se obtendría de nuevo el mensaje original m. Existen dos tipos fundamentales de criptosistemas:

- Criptosistemas simétricos o de clave privada. Son aquellos que emplean la misma clave k tanto para cifrar como para descifrar. Presentan el inconveniente de que para ser empleados en comunicaciones la clave k debe estar tanto en el emisor como en el receptor, lo cual lleva a preguntarse cómo transmitir la clave de forma segura.
- Criptosistemas asimétricos o de llave pública. Que emplean una doble clave (kp,kP). kp se conoce como clave privada y kP se conoce como clave pública. Una de ellas sirve para la transformación E de cifrado y la otra para la transformación D de

descifrado. En muchos casos son intercambiables, es decir, si se emplea una para cifrar la otra sirve para descifrar y viceversa. Estos criptosistemas deben asegurar además que el conocimiento de la clave publica kP, no permita calcular la clave privada kp ofreciendo un abanico superior de posibilidades, pudiendo emplearse para establecer comunicaciones seguras por canales inseguros puesto que únicamente viaja por el canal la clave pública.

En la práctica se emplea una combinación de estos dos tipos de criptosistemas, puesto que los segundos presentan el inconveniente de ser computacionalmente mucho más costosos que los primeros. En el mundo real se codifican los mensajes (largos) mediante algoritmos simétricos, que suelen ser muy eficientes y luego se hace uso de la criptografía asimétrica para codificar las claves simétricas (cortas).

El proporcionar bases firmes a la criptografía ha sido el planteamiento de investigación principal en las tres últimas décadas. El artículo pionero de W. Diffie y M. Hellman (Diffie y Hellman, 1976) se considera el iniciador de este planteamiento. Siguiendo esta tendencia, las dos actividades más importantes han sido:

1. La relacionada con las definiciones. En la que se han identificado, conceptualizado y especificado con rigor las tareas criptográficas que cubren los aspectos de seguridad, proporcionando una definición del cifrado seguro. La actividad relativa a las definiciones ha identificado conceptos que antes no se conocían, uno de los ejemplos es la introducción de las pruebas de conocimiento cero.

2. La relacionada con las construcciones. En la que se estudian y se diseñan los esquemas criptográficos que satisfacen las definiciones. Se trata de demostrar lo plausible de obtener objetivos ciertos. Los desafíos del diseñador no son sólo los atacantes sino también los matemáticos y los computadores, que con sus avances teóricos y de prestaciones, ponen en duda para un futuro la seguridad de métodos que se consideran hoy en día seguros.

Por su parte, el criptoanálisis, que comienza siendo un arte, es hoy en día una ciencia/tecnología que persigue descifrar las informaciones que se encuentran cifradas sin necesidad de conocer sus claves correspondientes.

2.1.4 Criptoanálisis

El criptoanálisis consiste en comprometer la seguridad de un criptosistema. Esto se puede hacer descifrando un mensaje sin conocer la clave, o bien obteniendo a partir de uno o más criptogramas la clave que ha sido empleada en su codificación. No se considera criptoanálisis el descubrimiento de un algoritmo secreto de cifrado; se supone, por el contrario, que los algoritmos siempre son conocidos.

En general el criptoanálisis se suele llevar a cabo estudiando grandes cantidades de pares mensaje-criptogramas generados con la misma clave. El mecanismo que se emplea para obtenerlos es indiferente, y puede ser resultado de escuchar un canal de comunicaciones, o de la posibilidad de que el objeto de nuestro ataque responda con un criptograma cuando se envíe un mensaje. Obviamente, cuanto mayor sea la cantidad de pares, más probabilidades de éxito tendrá el criptoanálisis.

Uno de los tipos de análisis más interesantes es el de texto plano escogido, que parte de que se conoce una serie de pares de textos planos y sus criptogramas correspondientes. Esta situación se suele dar cuando se tiene acceso al dispositivo de cifrado y éste permite efectuar operaciones, pero no permite leer su clave. El número de pares necesarios para obtener la clave desciende entonces significativamente. Cuando el sistema es débil, pueden ser suficientes unos cientos de mensajes para obtener información que permita deducir la clave empleada.

También se puede tratar de criptoanalizar un sistema aplicando el algoritmo de descifrado, con todas y cada una de las claves, a un mensaje codificado y comprobar cuáles de las salidas que se obtienen tienen sentido como posible texto plano. En general, todas las técnicas que buscan exhaustivamente por el espacio de claves K se denominan de fuerza bruta, y no suelen considerarse como auténticas técnicas de criptoanálisis, reservándose este término para aquellos mecanismos que explotan posibles debilidades intrínsecas en el algoritmo de cifrado. En general, se denomina ataque a cualquier técnica que permita recuperar un mensaje cifrado empleando menos esfuerzo computacional que el que se usaría por la fuerza bruta. Se da por supuesto que el espacio de claves para cualquier criptosistema digno de interés ha de ser suficientemente grande como para que los métodos basados en la fuerza bruta sean inviables. No obstante se tiene en cuenta que la capacidad de cálculo de las computadoras crece a gran velocidad, por lo que algoritmos que hace años eran resistentes a la fuerza bruta hoy pueden resultar inseguros, como es el caso del DES.

El DES (*Data Encryption Standard*) es un algoritmo de cifrado desarrollado por IBM, adoptado como estándar por el gobierno de los Estados Unidos en 1976, y cuyo uso se ha propagado ampliamente por todo el mundo (Stallings, 2011). El algoritmo fue controvertido al principio, con

algunos elementos de diseño clasificados y una longitud de clave relativamente corta. Posteriormente DES fue sometido a un intenso análisis académico y motivó el concepto moderno del cifrado por bloques y su criptoanálisis. Hoy en día, DES se considera inseguro para muchas aplicaciones debido principalmente a que el tamaño de clave de 56 bits es corto; las claves de DES se han roto en menos de 24 horas. Existen también resultados analíticos que demuestran debilidades teóricas en su cifrado.

Sin embargo, existen longitudes de clave para las que resultaría imposible a todas luces, empleando computación tradicional, aplicar un ataque por fuerza bruta. Por ejemplo, si se diseña una máquina capaz de recorrer todas las combinaciones que pueden tomar 256 bits, cuyo consumo fuera mínimo en cada cambio de estado (según las leyes de la termodinámica existe una cantidad mínima de energía necesaria para poder modificar el estado de un sistema físico), no habría energía suficiente en el universo para que pudiera completar sus trabajo.

Un par de métodos de criptoanálisis que han dado interesantes resultados son el análisis diferencial y el análisis lineal. El primero de ellos, partiendo de pares de mensajes con diferencias mínimas (usualmente un bit), estudia las variaciones que existen entre los mensajes cifrados correspondientes, tratando de identificar patrones comunes. El segundo emplea operaciones XOR entre algunos bits del texto plano y algunos bits del texto cifrado, obteniendo finalmente un único bit.

Otro tipo de análisis, esta vez para los algoritmos asimétricos, consistiría en tratar de deducir la clave privada a partir de la pública. Suelen ser técnicas analíticas que básicamente intentan resolver los problemas de elevado coste computacional en los que se apoyan estos criptosistemas: factorización, logaritmos discretos, etc. Mientras estos problemas genéricos

permanezcan sin solución eficiente, se podrá seguir confiando en estos algoritmos.

Como se puede apreciar, la gran variedad de sistemas criptográficos producen necesariamente gran variedad de técnicas de criptoanálisis, cada una de ellas adaptada a un algoritmo o familia de ellos. Con toda seguridad, cuando en el futuro aparezcan nuevos mecanismos de protección de la información surgirán con ellos nuevos métodos de criptoanálisis. De hecho, la investigación en este campo es tan importante como el desarrollo de algoritmos de criptográficos dado que permite detectar la presencia de fallos en un sistema. En resumen, existen diferentes formas de atacar un sistema criptográfico:

- Ataque por fuerza bruta. Si se tiene un criptograma mediante este método se probarán todas las claves posibles para obtener el texto plano. Si el conjunto de posibles claves es alto, este sistema es inviable. Normalmente a este tipo de ataques no se les suele considerar como una forma de criptoanálisis ya que no busca puntos débiles, únicamente utiliza todas las claves posibles.

- Ataque por texto plano escogido. Consiste en elegir varios textos planos y obtener sus criptogramas asociados. Esto implica tener acceso al dispositivo de cifrado, pero no a la clave.

- Ataque a partir de texto plano. El atacante tiene acceso a textos planos y a sus correspondientes criptogramas.

- Análisis por frecuencias. Este tipo de ataque es utilizado para romper sistemas criptográficos simétricos y se basa en estudiar la frecuencia con la que aparecen los distintos símbolos en un

lenguaje determinado y luego estudiar la frecuencia con la que aparecen en los criptogramas, y de esta manera establecer una relación y obtener el texto plano.

Estos aspectos básicos de la criptografía muestran las bases teóricas y las consideraciones técnicas fundamentales para la construcción de criptosistemas. Este cuerpo de conocimientos es usado en esta investigación para desarrollar una metodología de cifrado y descifrado de imágenes digitales, mediante autómatas celulares bidimensionales invertibles, resistente a los diferentes ataques criptográficos estudiados. A continuación se presentan los aspectos que relacionan a los criptosistemas con la teoría de la información y los autómatas celulares.

2.2 **Autómatas Celulares**

Antes de explicar lo que es un Autómata Celular (AC), es importante conocer las bases teóricas implícitas en un Sistema Dinámico (Devaney, 1989) y para ello hay que tener claro el concepto de Sistema. Para un físico, un Sistema es un objeto o conjunto de objetos reconocibles que pueden ser considerados como un todo, como por ejemplo una caja llena de átomos, un grupo de animales en su ambiente natural o equipos conectados en una red. Por su parte, un matemático diría usando una definición más precisa, que un Sistema es un conjunto de estados y un conjunto de reglas que actúan sobre esos estados.

Un Sistema Dinámico es un conjunto de estados con reglas que hacen que estos estados cambien en el tiempo. Se puede pensar en el tiempo como un conjunto creciente de números enteros (valores discretos) o números reales (valores continuos).

Los Autómatas Celulares (Toffoli y Margoulus, 1987) pueden ser estructurados para el modelado de Sistemas Dinámicos donde el espacio, tiempo y estados son discretos. Estos están representados por un conjunto *n-dimensional* de celdas organizadas geométricamente según la conveniencia del problema. Cada una de las celdas tiene un estado específico en un momento dado, valor que viene determinado según el alfabeto del Autómata Celular. La dinámica de cambio del autómata es bastante simple, a cada celda del espacio celular se le aplica un proceso de transición a partir de su estado actual y del estado de sus celdas vecinas para ese mismo instante, el número de celdas vecinas y la forma como ellas son seleccionadas es parte de la configuración del autómata y se conoce con el nombre de vecindad.

Desde el punto de vista de sistemas de cómputo se suelen considerar las siguientes correspondencias entre un modelo de cómputo tradicional y el de un autómata celular:

- *Las reglas de transición*. Corresponderán con el “programa” que en este caso no deriva de algoritmo alguno.
- *La dinámica del autómata celular*. Es comparable al proceso de ejecución temporal del programa.
- *La configuración o estado inicial*. Es la data inicial o de entrada del programa.
- *Los estados o configuraciones sucesivas*. Son las sucesivas etapas de cómputo intermedio.
- *El estado final*. Generalmente un atractor de la dinámica, sería el cómputo objetivo.

De manera más precisa un Autómata Celular se puede concebir como una tupla {G,V,Q,F} donde:

- G representa el espacio celular.
- V es alguna región local finita dentro del reticulado, definida por un patrón de vecindad a un sitio o nodo. Esta vecindad $\{i_1,i_2,\ldots,i_n\}$ de sitios tiene, en general, el mismo tamaño y estructura para todos los nodos del espacio celular.
- Q denota un alfabeto finito sobre el cual toman valores los estados de los sitios.
- F es la función de transición local que asocia la configuración de la vecindad con cada estado de los sitios.

2.2.1 Condiciones de Frontera

Por definición, un AC consta de un espacio celular infinito. Sin embargo, para fines prácticos (modelos de sistemas físicos, llevados a cabo en computadores de memoria finita), se requiere tomar ciertas consideraciones a la hora de implementar un AC en un sistema de cómputo. Es por ello que la definición original se modifica para dar cabida a espacios celulares finitos en los que las células del AC interactúen. Esto conlleva a la consideración extra de lo que debe suceder con aquellas células que se encuentren en los bordes del espacio celular. La implementación de una o varias consideraciones específicas es conocida como condición de frontera.

Dentro del ámbito de los AC, se pueden implementar numerosas condiciones de frontera, de acuerdo a lo que el problema real requiera para su modelado. Un AC puede exhibir las siguientes condiciones de frontera:

- *Frontera abierta*. Se considera que fuera del espacio celular residen células, todas con un valor fijo. En el caso particular del juego de la vida y otros AC con dos estados en su conjunto k, una frontera se dice fría si las células fuera de la frontera se consideran muertas y caliente si se consideran vivas.

- *Frontera periódica*. Se considera al espacio celular como si sus extremos se tocaran. En un espacio celular de dimensión 1, esto puede visualizarse en dos dimensiones como una circunferencia. En dimensión 2, el espacio celular podría visualizarse en tres dimensiones como un toroide.

- *Frontera reflectora*. Se considera que las células fuera del espacio celular reflejan los valores de aquellas dentro de la misma. Así, una célula que estuviera junto al borde del espacio celular (fuera de ella) tomaría como valor el de la célula que este junto al borde del espacio celular, dentro de ella.

- *Sin frontera*. Haciendo uso de implementaciones que hagan crecer dinámicamente el uso de memoria del espacio celular implementado, se puede asumir que cada vez que las células deben interactuar con células fuera del espacio celular, este se hace más grande para dar cabida a estas interacciones. Obviamente, existe un límite (impuesto por la memoria disponible) para esta condición. Es muy importante no confundir esta condición de frontera con la definición original de AC cuyo espacio celular es inicialmente infinito. En el caso de un AC sin frontera, el espacio celular comienza con un tamaño definido y finito, y conforme se requiera va creciendo en el tiempo, lo cual no lo hace necesariamente un modelo más cercano a la realidad, pues si se inicializara el espacio celular aleatoriamente, con esta condición sólo

se pueden inicializar las células dentro del espacio inicial finito, mientras que en el caso de la definición original, en teoría todas las células del espacio celular infinito deberían ser inicializadas.

2.2.2 Estado

Un autómata celular se construye por una serie de celdas, es decir un arreglo de elementos que se denominan células. Cada célula puede tener un número finito de valores, ya sea un valor entero, una letra o un color, lo que se quiera que represente cada una, a estos valores se les denomina estados, todas las células tienen el mismo número posible de estados.

2.2.3 Función de Transición

La función de transición F se define como:

$$F : Q^n \rightarrow Q$$

$$(s_1, s_2, ..., s_n) \in Q^n \rightarrow F(s_1, s_2, ..., s_n) \in Q$$

donde n es la cardinalidad de V, es decir el número de nodos |V|, del reticulado, contenidos en la vecindad.

De esta forma la función de transición define una correspondencia desde el producto cartesiano de los estados de las celdas pertenecientes a la vecindad a uno de ellos, esta correspondencia es claramente de muchos a uno. Así el alfabeto α de entrada para el autómata celular consiste en el conjunto de posibles configuraciones de las células de la vecindad:

$$\alpha = Q^n$$

La cardinalidad de α es igual a la cardinalidad de Q elevada a la n:

$$|\alpha| = |Q|^n = k^n$$

donde se denota por k la cardinalidad del alfabeto Q.

2.2.4 Autómatas Celulares Bidimensionales

Los autómatas celulares en dos dimensiones evolucionan en el plano cartesiano. Estos se encuentran determinados por dos tipos de vecindades fundamentales, la vecindad de Von Neumann y la vecindad de Moore:

- *Vecindad de Moore*. Está formada por una célula central y ocho células vecinas alrededor.
- *Vecindad de Von Neumann*. Suprime las células diagonales y conserva las células ortogonales con respecto a la vecindad de Moore.

Las vecindades de Moore y Von Neumann son ilustradas en la figura 2.1.

Figura 2.1. **Vecindades de Autómatas Celulares Bidimensionales**.
(Fuente: elaboración propia)

Es de notar que los autómatas celulares bidimensionales han sido utilizados en numerosas aplicaciones relacionadas con el estudio de sistemas dinámicos reales, tales como:

- Formación de estructuras cristalinas o de patrones específicos. Por ejemplo en reacciones químicas o propagación de fallas en materiales.
- Recurrencia de patrones como en la turbulencia de los fluidos.
- Variedad de sistemas autoreproductivos o evolutivos.
- Propagación de infecciones.
- Procesamiento de señales.
- Modelos económicos.

La complejidad implícita en esta clase de dispositivos es tan grande que su estudio sistemático es muy deficiente haciendo que exista poca literatura que defina una generalización, entre otras cosas esto se debe a que las herramientas que se emplean en una dimensión no son prácticas para aplicarse en dos y tres dimensiones.

En este trabajo se estudia un caso específico de aplicación de los autómatas celulares bidimensionales invertibles, debido a que por sus propiedades desordenan de manera compleja el estado inicial del espacio celular y lo restablecen al seguir la dinámica inversa, lo que resulta útil para construir un método de cifrado.

2.2.5 Autómatas Celulares Invertibles

Un autómata celular invertible es aquel para el cual, dada su regla dinámica directa, existe una nueva regla inversa cuya aplicación provoca que el autómata recorra las configuraciones, que conforman su trayectoria, en sentido inverso. Es claro que esto es posible sólo si el sistema definido por la regla directa también es determinista para la regla inversa, es decir, que para cada configuración del autómata celular existe una y sólo una configuración precedente.

La propiedad importante que poseen los autómatas celulares invertibles es que la información total sobre la configuración inicial se conserva en todo momento. En tal sentido habrán tantas “constantes de movimiento” (observables que se conservan) como celdas en el sistema. La mayoría de estas constantes de movimiento serán de poco interés ya que en general sólo interesan los rasgos de las celdas activas. En cualquier caso un autómata celular invertible posee el mayor número posible de observables que se conservan, lo que lo convierte en una herramienta plausible para el cifrado de información debido a que ella podría ser desordenada de manera compleja por reglas invertibles y luego ser recuperada efectivamente invirtiendo la dinámica.

2.2.5.1 Vecindad de Margolus

Este tipo especial de vecindad para autómatas celulares bidimensionales fue introducida por Norman Margolus (Toffoli y Margoulus, 1987) y resulta muy útil en el modelado de sistemas físicos. Para

implementar la vecindad de Margolus se define un *autómata celular particionado* de la siguiente manera:

- El arreglo de células (rejilla) se particiona en un conjunto finito de partes disjuntas dispuestas uniformemente y que se denominan bloques.
- Se especifica una regla de bloque que sirve para actualizar un bloque completo en términos de su configuración actual.
- La regla no actualiza una sola celda de bloque, lo hace para todas las células del bloque.
- Los bloques no se superponen de tal forma que no hay intercambio de información entre bloques adyacentes.
- La partición se cambia de una iteración a la otra de manera de permitir el intercambio de información entre los bloques.

Para implementar la vecindad de Margolus se emplea un esquema de partición bien simple que se muestra en la figura 2.2.

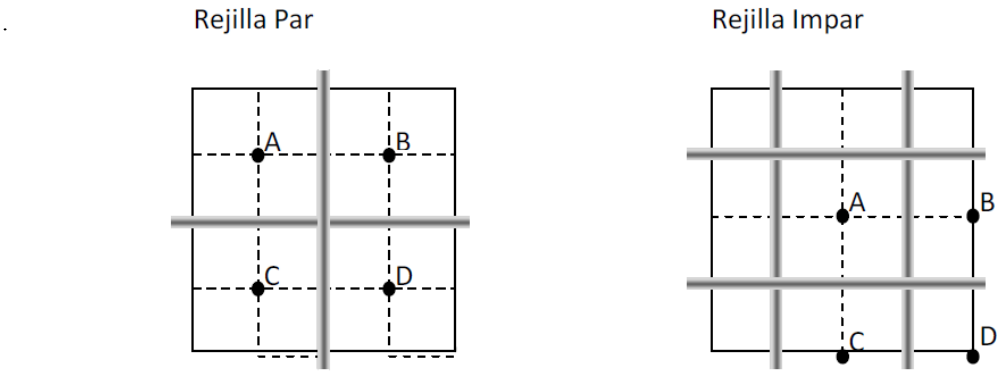

Figura 2.2. **Particionamiento del Espacio Celular**. (Fuente: elaboración propia)

En general, se supone que las diferentes particiones empleadas en cada paso de actualización por un autómata celular particionado deben ser finitas en número y deben emplearse (actualizarse) cíclicamente de tal forma de conservar la uniformidad en el espacio-tiempo. En particular en el caso del particionamiento de la vecindad de Margolus se requieren dos pasos de actualización en la regla de bloque (uno aplicado sobre la rejilla par y el otro sobre la rejilla impar) esto implica que un paso de tiempo corresponde a dos iteraciones del autómata celular. Dependiendo de cuál rejilla se esté considerando (par o impar), cambia de punto la vecindad y por ende la regla de bloque se aplica en cada caso a una vecindad diferente.

En los autómatas celulares particionados las reglas se aplican al bloque completo, por ende la operación representada por la regla tendrá tantas salidas como entradas. Con esto es posible asegurar que la regla no pierda información (pueden reconstruirse las entradas a partir de las salidas) dando la posibilidad de programar la invertibilidad.

En cada actualización ninguna de las entradas se comparte con entradas de bloques adyacentes. En esta situación, las reglas determinan el flujo temporal de información de los bloques con control total sobre los mismos. De manera que si la regla individual es invertible, también lo será el proceso global. Por el contrario si hay pérdida de información en un bloque, ninguno de los bloques vecinos podrá compensar esta perdida, por lo que el proceso será no-invertible. Así, una regla en la partición de Margolus será invertible si y solo si esta establece una correspondencia uno a uno entre las configuraciones “viejas” y “nuevas” de cada bloque. Para invertir la dinámica de un autómata celular particionado en el caso del movimiento de partículas, simplemente se aplica la regla en el orden inverso:

- *Orden directo* .- Rejilla par $\Rightarrow$ Regla $\Rightarrow$ Rejilla impar $\Rightarrow$ Regla

- *Orden inverso*.- Rejilla impar $\Rightarrow$ Regla $\Rightarrow$ Rejilla par $\Rightarrow$ Regla

2.2.5.2 Reglas Invertibles

La parte central del estudio de un autómata celular invertible es su regla de evolución, ya que ésta indica como se comporta el sistema a través del tiempo, lo interesante es que esta regla de evolución es de influencia local, es decir, sólo afecta las vecindades que forman parte de la configuración, sin embargo induce un mapeo global que es invertible. Desde esta perspectiva se puede tomar la evolución como una función que esencialmente transforma elementos de un conjunto a otro, donde tales elementos son las configuraciones globales que puede tener el autómata celular y la función el mapeo global producto de la regla de evolución. Las funciones se pueden clasificar por el mapeo que producen, obteniéndose de esta forma las siguientes definiciones (Stone, 1973). Una función $f : \mathrm{X} \to \mathrm{Y}$ se dice:

- I*nyectiva* o mapeo uno a uno si a cada elemento de X le corresponde un elemento distinto de Y.
- *Sobreyectiva* o mapeo sobre si a todo y $\in$ Y le corresponde al menos una x $\in$ X.
- *Biyectiva* o mapeo que es *uno a uno* y *sobre* a la vez. Si un mapeo $f : \mathrm{X} \to \mathrm{Y}$ es biyectivo, entonces $f^{-1} : \mathrm{Y} \to \mathrm{X}$ también define una función biyectiva.

Un ejemplo del uso de la vecindad de Margolus es el modelaje de un gas de partículas en el cual las partículas se mueven a velocidad constante y uniforme sin interacción. En este caso las células pueden acceder dos estados, cero (ausencia de partícula) y uno (célula ocupada por una partícula), siendo los bloques de tamaño 2x2. Para simular el movimiento de una partícula se aplica una regla bloque muy simple que consiste en

intercambiar estados en la diagonal, intercambiando la posición de la rejilla en cada iteración, como se muestra en la figura 2.3.

$$\begin{bmatrix} a_{00} & a_{01} \\ a_{10} & a_{11} \end{bmatrix} \Leftrightarrow \begin{bmatrix} a_{11} & a_{10} \\ a_{01} & a_{00} \end{bmatrix}$$

***Figura 2.3*. Regla de Intercambio de la Diagonal**. (Fuente: elaboración propia)

Las reglas modifican todas las celdas ocupadas en el bloque. En este estudio se emplean bloques de 2x2 y nos concentramos en reglas que realizan desplazamientos de los contenidos de las celdas. De esta forma tras la aplicación de la regla, el bloque contiene las mismas unidades de información, pero distribuidas de forma diferente.

CAPÍTULO III: DESCRIPCIÓN Y CODIFICACIÓN DE LA DINÁMICA

3.1 Descripción de la Dinámica

El desarrollo de un nuevo esquema de encriptamiento requiere tomar un conjunto de decisiones de diseño y la formulación de nuevas aproximaciones teóricas que sustenten los mecanismos que permitirán el funcionamiento del esquema.

Del planteamiento del problema se sigue que el esquema propuesto de encriptamiento simétrico para imágenes digitales a color debe estar basado en una dinámica reversible de autómata celular bidimensional no homogéneo y variable en el tiempo.

3.1.1 Vecindad de Margolus y reversibilidad

La adopción de la vecindad de Margolus facilita la definición de funciones de transición reversibles que actúan sobre espacios celulares bidimensionales. Con este tipo de vecindad el espacio celular bidimensional se divide en bloques. En el esquema propuesto cada bloque *B* está formado por cuatro celdas b_{00}, b_{01}, b_{10} y b_{11} que están dispuestas de forma tal que en la iteración *i* conforman una matriz B^i de 2x2 celdas

$$B^i = \begin{bmatrix} b_{00}^i & b_{01}^i \\ b_{10}^i & b_{11}^i \end{bmatrix}$$

Si definimos una función biyectiva

$$F:\{00,01,10,11\}\rightarrow\{00,01,10,11\}$$

de forma tal que

$$B^{i+1}=\begin{bmatrix} b^{i}_{F(00)} & b^{i}_{F(01)} \\ b^{i}_{F(10)} & b^{i}_{F(11)} \end{bmatrix}=\begin{bmatrix} b^{i+1}_{00} & b^{i+1}_{01} \\ b^{i+1}_{10} & b^{i+1}_{11} \end{bmatrix}$$

entonces la reversibilidad está garantizada por la relación

$$B^{i}=\begin{bmatrix} b^{i+1}_{F^{-1}(00)} & b^{i+1}_{F^{-1}(01)} \\ b^{i+1}_{F^{-1}(10)} & b^{i+1}_{F^{-1}(11)} \end{bmatrix}=\begin{bmatrix} b^{i}_{00} & b^{i}_{01} \\ b^{i}_{10} & b^{i}_{11} \end{bmatrix}$$

donde la existencia de F^{-1} está asegurada por la biyectividad de *F*.

3.1.2 Nomenclatura de Reglas

Para designar las reglas construidas a partir de las funciones biyectivas $F:\{00,01,10,11\}\rightarrow\{00,01,10,11\}$, se propone una nomenclatura similar a la creada por Wolfran para autómatas binarios unidimensionales de radio 1.

Dada una función biyectiva $F:\{00,01,10,11\}\rightarrow\{00,01,10,11\}$, la concatenación de *F(11) , F(10), F(01)* y *F(00)* produce la cadena binaria de ocho bits *F(11)F(10)F(01)F(00),* cuyo valor numérico expresado en el sistema decimal

resulta una descripción precisa y conveniente de la regla asociada a la función biyectiva *F*.

Por ejemplo, la regla de intercambio de estados en la diagonal, que se ilustra en la figura 3.1, está asociada a la función biyectiva *f(00)=11, f(01)=10, f(10)=01, f(11)=00*. Siguiendo la nomenclatura propuesta, está regla está representada por el número 27, dado que este es el valor numérico decimal de la cadena de dígitos binarios *f(11)f(10)f(01)f(00)=00011011*.

$$\begin{bmatrix} a_{00}^{i} & a_{01}^{i} \\ a_{10}^{i} & a_{11}^{i} \end{bmatrix} \Leftrightarrow \begin{bmatrix} a_{11}^{i+1} & a_{10}^{i+1} \\ a_{01}^{i+1} & a_{00}^{i+1} \end{bmatrix}$$

***Figura 3.1*. Regla 27 de intercambio de la diagonal**. (Fuente: elaboración propia)

3.1.3 Vecindad de Margolus Ampliada

En este trabajo se propone un mecanismo que modifica la vecindad de Margulos, al introducir en esta dinámica de autómata celular bidimensional, el parámetro *distancia intercelular: DI*. Este parámetro toma valores enteros positivos y particiona el espacio celular C creando un conjunto de células activas C_A y un conjunto de células inactivas C_I. Durante la aplicación de una regla sólo las células activas conforman vecindades de Margolus y son transformadas. Las células inactivas permanecen inalteradas.

Dado un espacio celular C de mxn células, dispuestas como m filas y n columnas, al definir el espacio intercelular DI se obtienen:

$C_A = \{ c_{ij} \in C / i = k (DI+1)$ y $j = k´(DI+1) ; k, k´ \in N , k \leq m, k´ \leq n\}$

$C_I = C - C_A$

Todas las células del espacio celular C conservan sus posiciones, pero, con la introducción de la distancia intercelular *DI*, al momento de conformar las vecindades de Margolus, si bien es cierto que se siguen los principios originales también lo es que sólo conformarán bloques las células activas que dispongan de células vecinas activas disponibles. Las células inactivas no son tomadas en cuenta.

Si se define *DI* = 0, se obtiene el esquema de vecindad de Margolus original.

La incorporación de este mecanismo permite aplicar las reglas locales originales a grupos de células que se encuentran distantes en el espacio celular original (*DI*=0). De esta forma se incrementa el poder de difusión de las reglas locales, al permitir que una sola aplicación de una regla local intercambie información entre regiones distantes en el espacio celular original.

En la figura 3.2 se ilustra la formación de las vecindades de Margolus para DI=1 y DI=2 en sus iteraciones par e impar. En esta figura se presenta un espacio celular original de 14X14 células (para DI=0). La parte superior izquierda de la figura 3.2 ilustra la iteración par con DI=1. Las células activas se representan en blanco, mientras que las células inactivas se presentan sombreadas. Se destaca en esta imagen que no todas las células activas conforman bloques. En la iteración par con DI=1, las células activas pertenecientes a la última fila (de células activas) y las células activas pertenecientes a la última columna (de células activas) no conforman bloques por no tener células activas vecinas disponibles y se mantienen

inalteradas durante la iteración. Situaciones similares ocurren en los otros casos presentados en la figura 3.2.

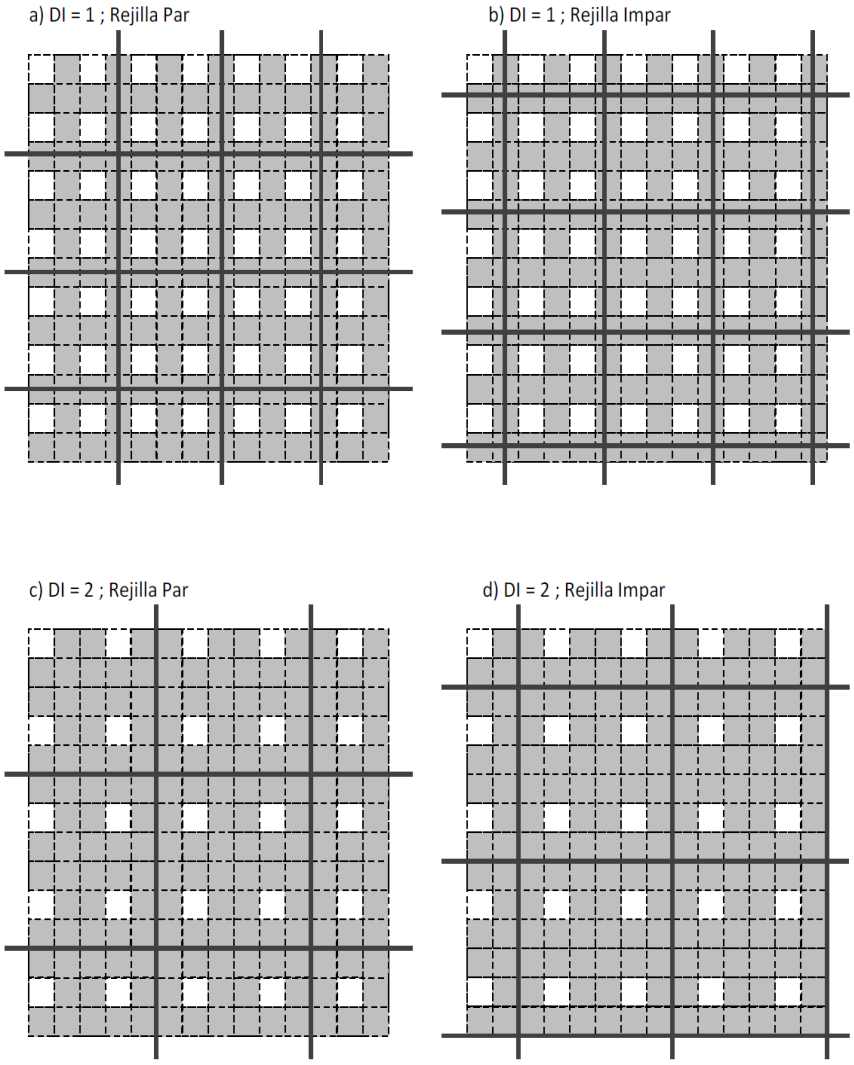

***Figura 3.2*. Vecindades de Margolus para DI=1 y DI=2.** (Fuente: Elaboracion propia)

3.1.4 Selección de Reglas

Existen 24 funciones biyectivas $F:\{00,01,10,11\}\rightarrow\{00,01,10,11\}$ que pueden ser consideradas a la hora de armar una combinación de reglas que ocasionen un desorden reversible sobre la imagen representada en la configuración inicial de un espacio celular bidimensional.

Por su capacidad para producir desorden fueron seleccionadas las reglas 57 y 27 para dirigir el proceso de encriptamiento.

3.1.4.1 Regla 57

La regla 57 corresponde a la función biyectiva de {00,01,10,11} en {00,01,10,11} definida por:

f(00) = 01

f(01) = 10

f(10) = 11

f(11) = 00

La figura 3.3 ilustra la aplicación de la regla 57. Tras la aplicación de esta regla todas las células del bloque han cambiado su posición y su vecindad dentro del bloque.

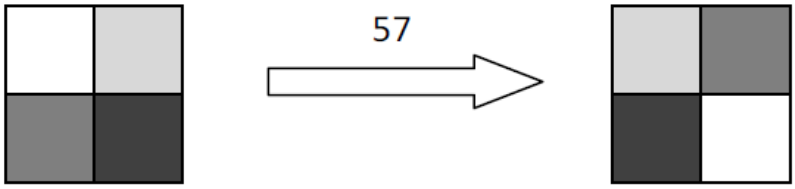

***Figura 3.3*. Regla 57.** (Fuente: Elaboracion propia)

3.1.4.2 Regla 27

La regla 27 corresponde a la función biyectiva de {00,01,10,11} en {00,01,10,11} definida por:

f(00) = 11

f(01) = 10

f(10) = 01

f(11) = 00

La figura 3.4 ilustra la aplicación de la regla 27. Esta regla efectúa el intercambio de las diagonales dentro del bloque.

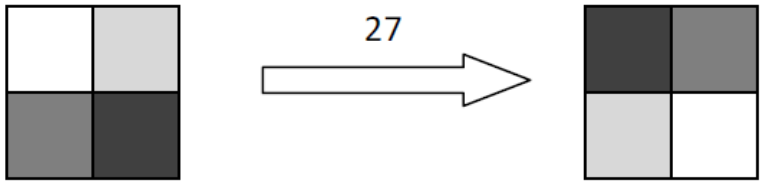

***Figura 3.4*. Regla 27**. (Fuente: Elaboracion propia)

La dinámica de encriptamiento estará fundamentada en la aplicación alternada de las reglas 57 y 27 durante las iteraciones par (rejilla par) e impar (rejilla impar) respectivamente.

Para complementar la acción de las reglas 57 y 27, se seleccionaron las reglas: 180, 225,108 y 198.

3.1.4.3 Reglas 180 y 225

Las regla 180 y 225 corresponden a las funciones biyectivas de {00,01,10,11} en {00,01,10,11} definidas respectivamente por:

f(00) = 00 g(00) = 01

f(01) = 01 y g(01) = 00

f(10) = 11 g(10) = 10

f(11) = 10 g(11) = 11

La aplicación alternada de las reglas 180 y 225 durante las iteraciones par (rejilla par) e impar (rejilla impar) respectivamente, producen un corrimiento (circular) dentro de las filas impares del espacio celular, mientras las filas pares permanecen inalteradas. La figura 3.5 ilustra el efecto producido por la aplicación alternada de las reglas 180 y 225.

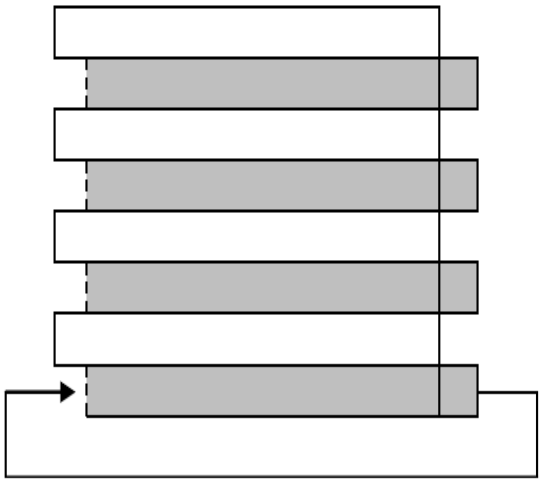

***Figura 3.5*. Efecto de la aplicación de las reglas 180 y 225.** (Fuente: Elaboración propia)

3.1.4.4 Reglas 108 y 198

Las regla 108 y 198 corresponden a las funciones biyectivas de {00,01,10,11} en {00,01,10,11} definidas respectivamente por:

f(00) = 00 g(00) = 10

f(01) = 11 y g(01) = 01

f(10) = 10 g(10) = 00

f(11) = 01 g(11) = 11

La aplicación alternada de las reglas 108 y 198 durante las iteraciones par (rejilla par) e impar (rejilla impar) respectivamente, producen un corrimiento hacia abajo (circular) dentro de las columnas impares del espacio celular, mientras las columnas pares permanecen inalteradas. La figura 3.6 ilustra el efecto producido por la aplicación alternada de las reglas 108 y 198.

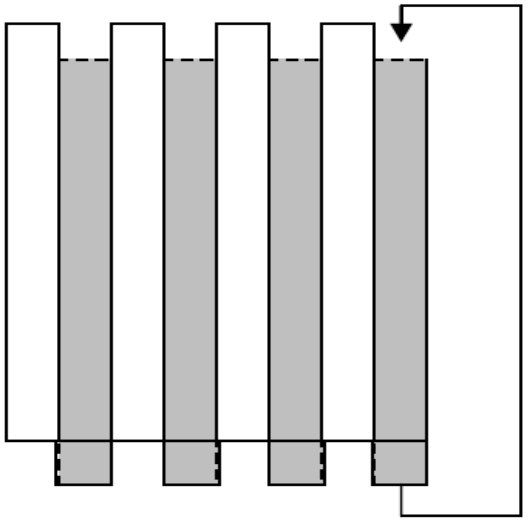

***Figura 3.6*. Efecto de la aplicación de las reglas 108 y 198.** (Fuente: Elaboración propia)

3.1.5 Cicloe

Al aplicar la dinámica de encriptamiento sobre un espacio celular de dimensión nxm, un cicloe(x) consiste en la aplicación de la secuencia de reglas 180, 225, 57, 27, 108 y 198 para cada uno los valores de (distancia intercelular) DI:0,1,2,…,maxDI; con maxDI = (max(n,m) – 4) / 3. El cicloe tiene una parámetro x, que toma valores en el conjunto {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15} y que indica cuantas veces debe aplicarse la combinación de reglas 57,27. Las combinaciones de regla 180, 225 y 108,198 se aplican sólo una vez cada una. La tabla 3.1 detalla la forma en que estas reglas son aplicadas durante un cicloe(x) para un valor particular de DI.

Tabla 3.1. Aplicación de las reglas en un cicloe(x)

Regla	Iteración	Aplicaciones
180	Par	1
225	Impar	
57	Par	x
27	Impar	
108	Par	1
198	Impar	

3.1.6 La Clave

En la dinámica de encriptamiento se usa una clave K = $k_0k_1k_{2...}k_{15}$ de 16 dígitos hexadecimales (o 64 bits). Los primeros 10 dígitos $k_0k_1k_{2...}k_9$ son usados, uno por uno, como parámetros de cicloe(k_i). Los dígitos $k_{10}k_{11}k_{12}k_{13}$ son utilizados como semilla de un algoritmo que produce una secuencia de mxn números pseudoaleatorios que son dispuestos sobre un espacio celular M de dimensión mxn que es sometido al procesamiento determinado por cicloe(k_{14}) y cicloe(k_{15}). La configuración del espacio celular M obtenida por este procesamiento es usada como máscara que se superpone mediante la operación XOR a la configuración del espacio celular original previamente procesado.

El algoritmo utilizado para la generación de los números pseudoaleatorios que conforman la máscara está descrito en (Weiss, 1993) donde además se implementa mediante el código que se muestra en la figura 3.7.

```
unsigned int seed;              /* global variable*/
#define a 16807                 /* 7^5     */
#define m 2147483647            /*2^31 -1 */
#define q 127773                /* m/a */
#define r 2836                  /* m%a */
double
random( void )
{
        int tmp_seed;
        tmp_seed = a * ( seed % q) – r *(seed/q);
        if( tmp_seed >= 0 )
                seed = tmp_seed;
        else
                seed = tmp_seed + m;
        return (((double) seed  ) / m);
}
```

***Figura 3.7*. Código generador de números pseudoaletorios.** (Weiss, 1993)

3.1.7 Dinámica de Encriptamiento

Ahora que se conocen todas las piezas que conforman la dinámica de encriptamiento, es posible ensamblar una descripción paso a paso. Estos pasos se listan a continuación:

- Paso 0: La imagen, de mxn píxeles, a encriptar es leída y sus píxeles son dispuestos sobre un espacio celular C de dimensión mxn, conservando las posiciones relativas de los píxeles dentro de la imagen. Adicionalmente se suministra una clave de encriptamiento $K = k_0k_1k_{2...}k_{15}$.

- Paso 1: El espacio celular C es transformado por la aplicación de la secuencia ciclo(k_i); i = 0,1,2,…,9.

- Paso 2: Tomando como semilla $k_{10}k_{11}k_{12}k_{13}$, se genera una secuencia pseudoaleatoria de mxn píxeles que son dispuestos sobre un espacio celular M de dimensión mxn. Adicionalmente el espacio celular M es transformado por la aplicación de la secuencia ciclo(k_i); i = 14,15.

- Paso 3: C = C XOR M.

Esta secuencia de pasos, que describen el funcionamiento de la dinámica de encriptamiento, es esquematizada en el diagrama que se presenta en la figura la figura 3.8.

En la figura 3.8 se usa como ejemplo la (muy conocida) imagen “Lena”. Se aprecia en la figura la difusión que sobre la imagen causa la aplicación de las reglas seleccionadas con diferentes valores de Distancia Intercelular.

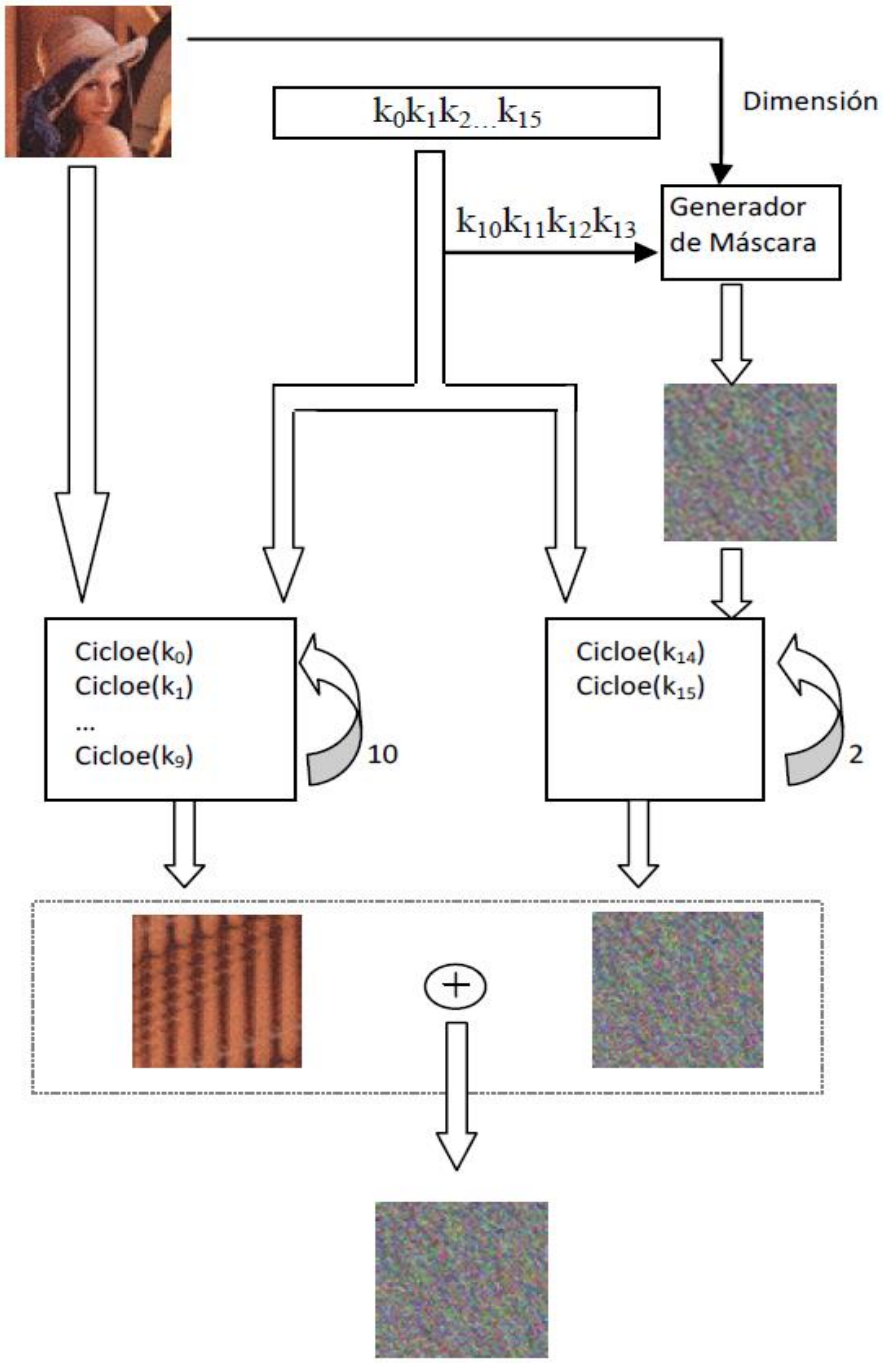

***Figura 3.8*. Diagrama del proceso de encriptamiento**. (Fuente: Elaboración propia)

3.1.8 Dinámica de Desencriptamiento

Para realizar el proceso de desencriptamiento se aplican las mismas transformaciones mencionadas anteriormente, pero en orden inverso. Como transformación inversa al cicloe del proceso de encriptamiento, en el proceso de encriptamiento se usa el ciclod.

3.1.9 Ciclod

Al aplicar la dinámica de desencriptamiento sobre un espacio celular de dimensión nxm, un ciclod(x) consiste en la aplicación de la secuencia de reglas 198, 108, 27, 57, 225 y 180 para cada uno los valores de (distancia intercelular) DI: maxDI,…,2,1,0; con maxDI = (max(n,m) – 4) / 3. El ciclod tiene una parámetro x, que toma valores en el conjunto {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15} y que indica cuantas veces debe aplicarse la combinación de reglas 27,57. Las combinaciones de regla 198, 108 y 225,180 se aplican sólo una vez cada una. La tabla 3.2 detalla la forma en que estas reglas son aplicadas durante un ciclod(x) para un valor particular de DI.

Tabla 3.2. Aplicación de las reglas en un ciclod(x)(1/2)

Regla	Iteración	Aplicaciones
198	Impar	1
108	Par	
27	Impar	X

Tabla 3.2. Aplicación de las reglas en un ciclod(x)(2/2)

Regla	Iteración	Aplicaciones
57	Par	x
225	Impar	1
180	Par	

3.1.10 La Clave en el Proceso de Desencriptamiento

En la dinámica de desencriptamiento se usa la misma clave K = $k_0k_1k_{2...}k_{15}$ de 16 dígitos hexadecimales. Los dígitos $k_{10}k_{11}k_{12}k_{13}$ son utilizados como semilla del algoritmo que produce una secuencia de mxn números pseudoaleatorios que son dispuestos sobre un espacio celular M de dimensión mxn que es sometido al procesamiento determinado por cicloe(k_{14}) y cicloe(k_{15}). La configuración del espacio celular M obtenida por este procesamiento es usada como máscara que se superpone mediante la operación XOR a la configuración del espacio celular de entrada C. Finalmente el espacio celular C´=C XOR M es sometido a la secuencia de transformaciones ciclod(k_i); i=9,8,7,…,0. Es decir los dígitos $k_9k_8k_{7...}k_0$ son usados, uno por uno, como parámetros de ciclod(k_i).

Las mismas operaciones fundamentales usadas anteriormente para realizar el cifrado, son ahora utilizadas para realizar el proceso de descifrado.

La secuencia de pasos, que describen el funcionamiento de la dinámica de desencriptamiento, es esquematizada en el diagrama que se presenta en la figura 3.9.

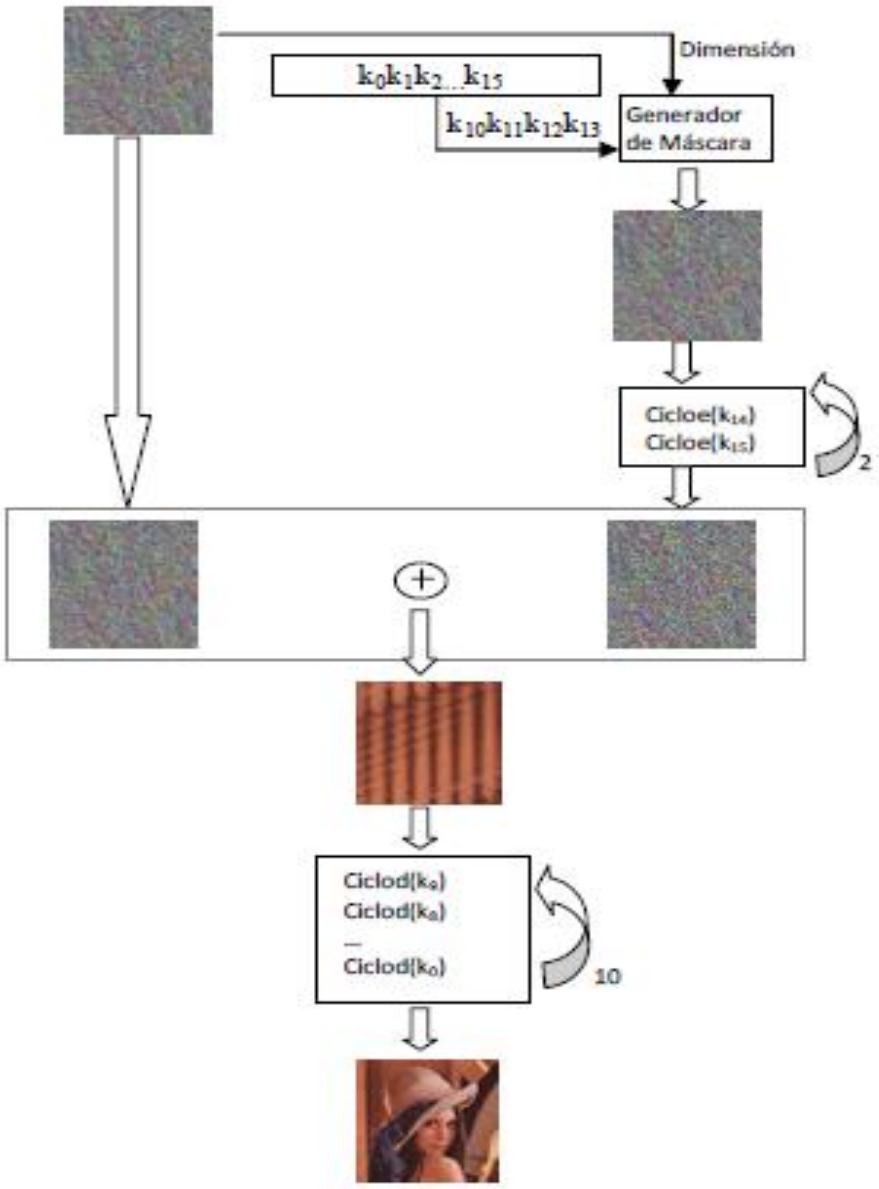

***Figura 3.9*. Diagrama del proceso de encriptamiento**. (Fuente: Elaboración propia)

3.2 Codificación de la Dinámica

3.2.1 Conceptos Básicos

Una imagen digital, tal como es vista por un ser humano, es un arreglo rectangular de puntos coloreados y muy proximos entre si. Idealmente los puntos son tan pequenos y cercanos que el ojo humano no puede distringuirlos individualmente, por lo que lucen como un continuo.

Tales puntos individuales reciben el nombre de pixeles. Los pixeles que constituyen una imagen son almacenados y transportados en archivos. De donde son extraidos para desplegar la imagen en una pantalla u hoja para el consumo humano. Existen numerosos formatos para el almacenamiento de pixeles en un archivo.

Un pixel es representado por cuatro bytes de 8 bits sin signo. Tres de estos bytes representan los colores rojo (*red*), verde (*green*) y azul (*blue*). El cuarto byte, referido como el byte alfa, representa la transparencia del pixel.

Cada uno de los bytes de color puede contener 256 valores diferentes. El valor particular del byte indica la proporcion del color correspondiete que es agregado a la mezcla para crear el color final correspondiente al pixel. El byte alfa tambien puede tomar 256 valores diferentes, si el valor es cero el pixel se hace completamente transparente; si el valor es 255 el pixel se torna completamente opaco. En medio de esots dos valores se despliega un espectro de transparencias.

Para el procesamiento de una imagen, es usual representarla como un arreglo tridimensional, donde cada pixel, representado por un arreglo de 4 posiciones, es ubicado en un arreglo bidimensional con las dimensiones de la imagen.

3.2.2 Aspectos de Codificación

Para la codificacion de la dinamica se selecciono el lenguaje de programacion Java y se hizo uso de un marco de referencia para el procesamiento de imágenes con Java, descrito en (Baldwin, 2004). Este marco de referencia organiza la construccion de un programa para el procesamiento de imágenes basandose en el desarrollo de dos progamas denominados **ImgMod02** y **ProgramTest** y una interface denominada **ImgIntfc02**. El programa denominado **ImgMod02** ofrece un entorno de trabajo para procesar imágenes, en tanto que el programa **ProgramTest** implementa la interface **ImgIntfc02** y realiza el procesamiento que se desea ejecutar. La interface **ImgMod02** declara un unico metodo denominado **processimage** que debe ser definido por las clases que la implementan.

Adaptandose a este esquema de trabajo se desarrollo el codigo correspondiente a la dinamica de encriptamiento-desencriptamiento descrita anteriormente como el metodo **processimage** del programa **ProgramTest**.

Cuando un usuario ejecuta el programa **ImgMod02**, el programa instancia un objeto de la clase de programas procesadores de imágenes e invoca el metodo **processimage** de ese objeto. Un arreglo tridimensional que contiene los pixeles de la imagen a procesar se pasa al metodo. Luego del procesamiento el metodo **processimage** retorna un arreglo tridimensional que contiene los pixeles de la imagen procesada.

El programa **ImgMod02** se encarga de la lectura de la imagen desde el archivo de entrada, su conversion en un arreglo tridimensional y su despliegue en pantalla. Complementariamente tambien se encarga de la escritura de la imagen resultante, expresada en un arreglo tridimensional, en el archivo de salida y su despliegue en pantalla.

CAPÍTULO IV: RESULTADOS

4.1 Indicadores

La calidad del cifrado puede ser estimada usando diversos indicadores. En este trabajo se utilizan tres indicadores de calidad del cifrado: los histogramas de frecuencia, la entropía y el coeficiente de correlación. Para el cálculo de estos indicadores se seleccionaron cinco imágenes: Lena, Dolly, Dog, Cyborg y Paisaje. Las cinco imágenes seleccionadas son mostradas en la figura 4.1.

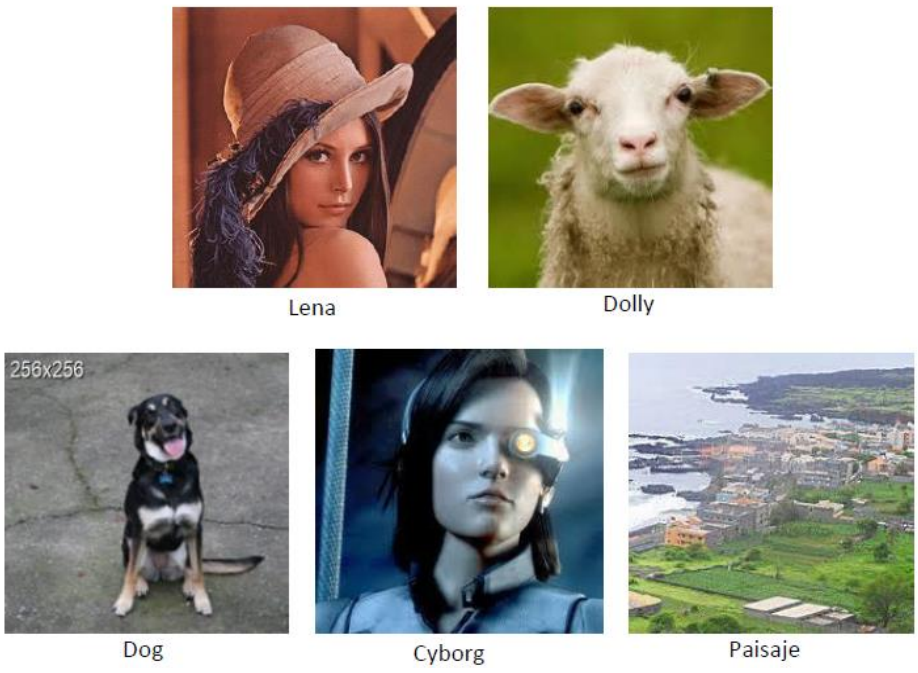

***Figura 4.1*. Imágenes de prueba**. (Pixabay,2013)

4.1.1 Histograma de Frecuencias

Dentro de una imagen cada píxel expresa una combinación de tres colores: rojo, verde y azul; combinación referida como rgb por los correspondientes nombres en inglés de estos colores. Cada uno de estos colores se manifiesta, en la totalidad del píxel, al asumir un valor entero en el rango 0…255, dado que el byte asociado a cada color puede estar en una de 256 configuraciones posibles. Una característica distintiva que ofrece mucha información sobre una imagen es la distribución particular de configuraciones de cada color (rojo, verde y azul).

Dada una imagen digital, para cada color, su histograma de frecuencias se construye representando en el eje horizontal las 256 posibles configuraciones y en el eje vertical el número de ocurrencias de cada configuración en la imagen.

Los histogramas de frecuencia ofrecen una gran cantidad de información sobre una imagen; por esta razón, un algoritmo de encriptamiento debe producir imágenes cifradas que al ser sometidas al escrutinio del histograma de frecuencias no revelen información sobre la imagen plana. La intención es pasar de histogramas con distribuciones muy particulares en la imagen plana a histogramas con distribuciones parejas en la imagen cifrada.

En la imagen 4.2 se contrastan las distribuciones de configuraciones de color rojo, verde y azul para las imágenes plana y cifrada de Lena. Los diagramas de la izquierda corresponden a la imagen plana y los diagramas de la derecha a la imagen cifrada.

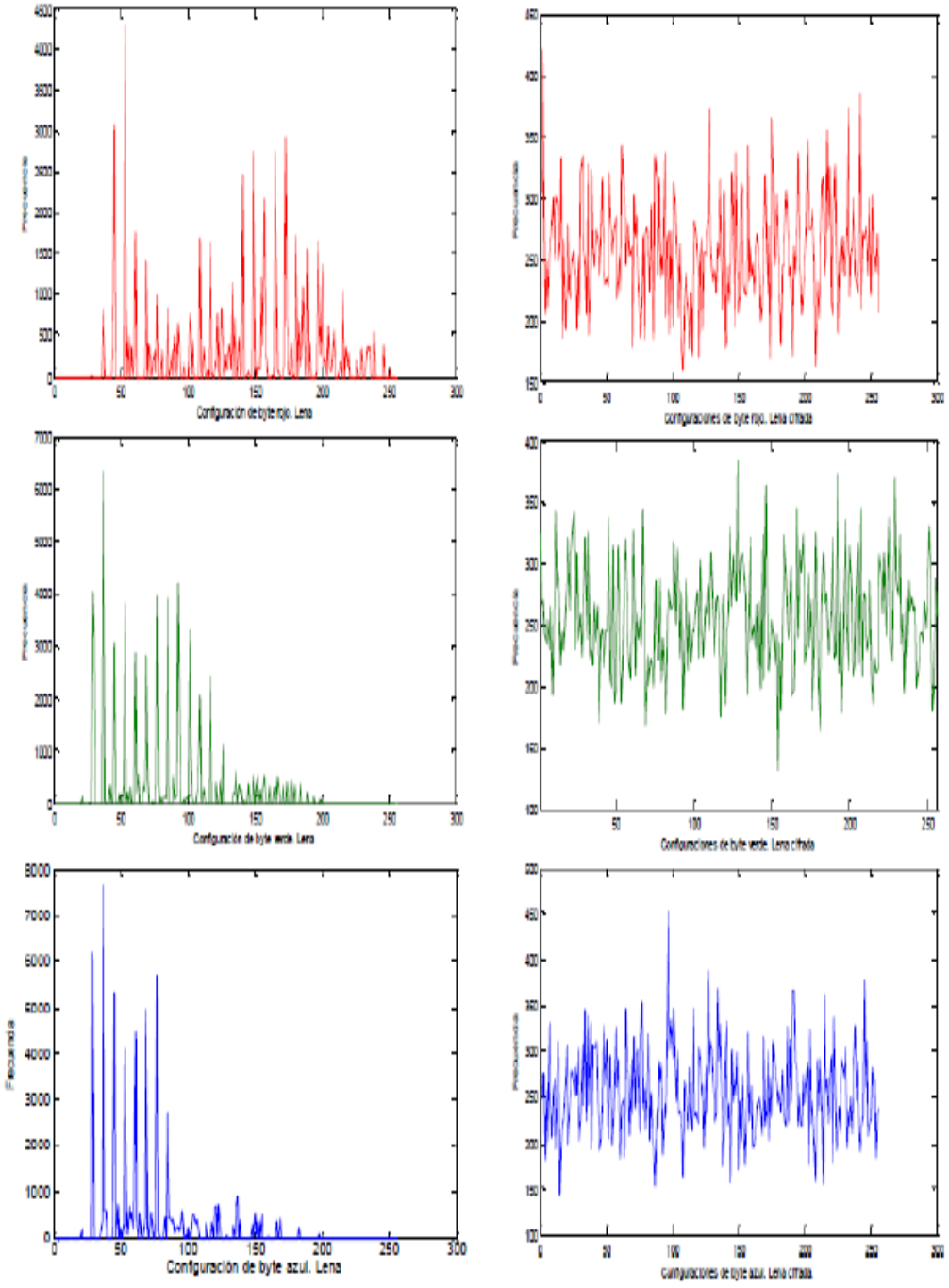

***Figura 4.2*. Frecuencias de configuraciones rgb, para Lena.** original (izquierda) y cifrada (derecha). (Fuente: Elaboración propia)

En todas las imágenes estudiadas se produjo un cambio similar en las características del histograma de frecuencias para cada color, al pasar de la imagen plana a la imagen cifrada. Consistentemente el método de encriptamiento propuesto genera imágenes cifradas cuyos histogramas no presentan ni picos ni valles pronunciados, es decir los histogramas resultan bastante planos, con lo cual se oculta información sobre la imagen plana.

4.1.2 Entropía de Información

Una de las ecuaciones más conocidas de la Teoría de la Información es la de Entropía de la Información H(S), la cual se expresa de la siguiente forma:

$$H(S) = \sum_{i=0}^{2^N - 1} P(s_i) \log\left(\frac{1}{P(s_i)}\right)$$

donde N es el número de bits que conforman a cada bloque de información y $P(s_i)$ es la probabilidad de ocurrencia de la configuración s_i. Dado que en las imágenes estudiadas, se utilizan 8 bits para representar la configuración de un color, dentro de un píxel, el tamaño del bloque de información N es 8. Así, para las imágenes estudiadas, el valor ideal para la entropía de las imágenes cifradas es 8. Tal valor de entropía indica la presencia en cantidades uniformes de las distintas configuraciones posibles s_i, i: 0…255. Este valor de entropía corresponde a una imagen generada completamente al azar. De esta forma la información de la imagen plana permanece oculta en la imagen cifrada. Al aplicar el método de encriptamiento propuesto sobre las cinco imágenes de prueba se obtuvieron imágenes cifradas con entropía superior a 7,98. Estos resultados son presentados en la figura 4.3.

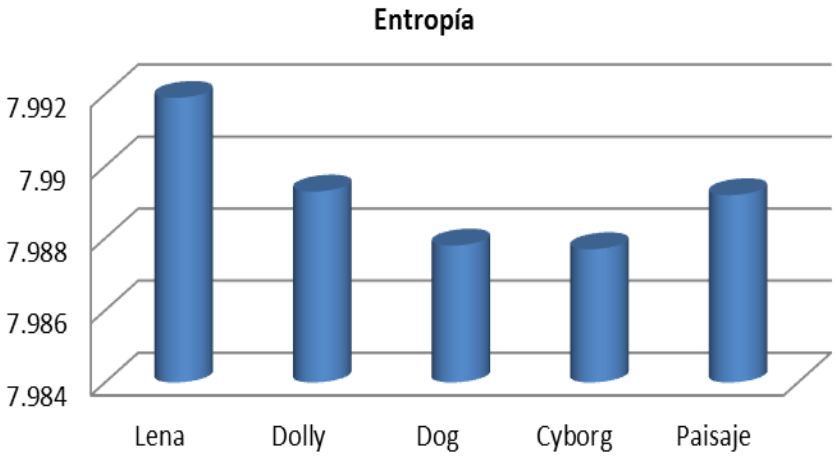

***Figura 4.3*. Entropía de imágenes encriptadas**. (Fuente: Elaboracion propia)

4.1.3 Coeficiente de Correlación

En esta sección, se calcularon los coeficientes de correlación horizontal, vertical y diagonal para píxeles adyacentes en la horizontal, la vertical y la diagonal respectivamente. Para alcanzar este fin se seleccionaron aleatoriamente 3000 pares de píxeles adyacentes para cada una de las tres direcciones estudiadas.

Para el cálculo de los coeficientes de correlación r_{xy} , se emplearon las siguientes ecuaciones, con M=3000 (tamaño de la muestra):

$$E(x) = \frac{1}{M}\sum_{i=1}^{M} x_i \qquad ; \qquad D(x) = \frac{1}{M}\sum_{i=1}^{M} (x_i - E(x))^2$$

$$\mathrm{cov}(x,y) = \frac{1}{M}\sum_{i=1}^{M} (x_i - E(x))(y_i - E(y))$$

$$r_{xy} = \frac{\mathrm{cov}(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}$$

Se espera que las imágenes planas presenten coeficientes de correlación altos (cercanos a 1) en todas las direcciones. Por el contrario, un buen método de encriptamiento debe reducir estos coeficientes de correlación en las imágenes cifradas. La figura 4.4 muestra los coeficientes de correlación horizontal, vertical y diagonal para cada una de las imágenes de prueba.

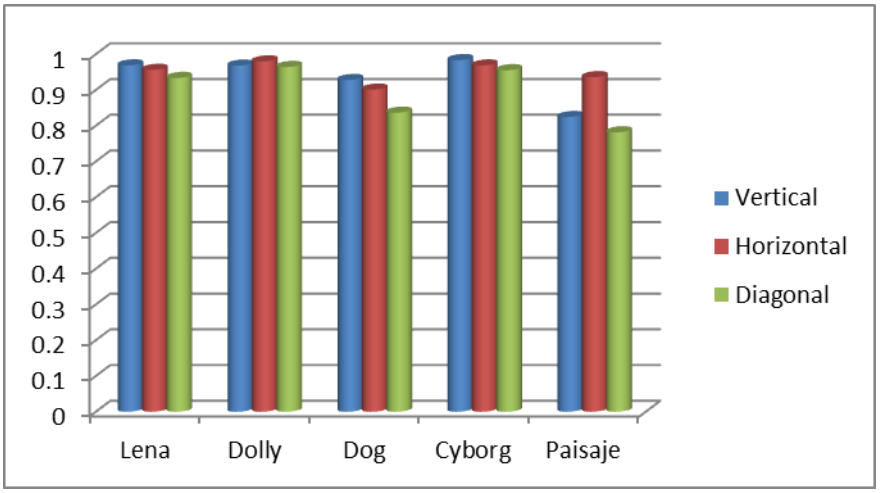

***Figura 4.4*. Coeficientes de correlación de imágenes originales**. (Fuente: Elaboración propia)

Se aprecia en la figura 4.4 que todas las imágenes de prueba presentan un alto coeficiente de correlación para píxeles vecinos en todas las direcciones.

La figura 4.5 muestra estos mismos coeficientes para las correspondientes imágenes cifradas por el método propuesto.

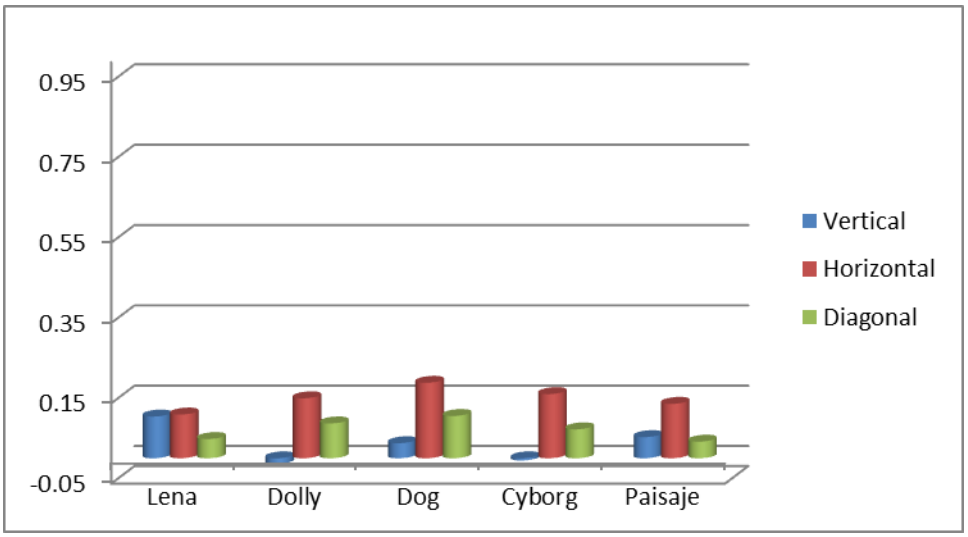

***Figura 4.5*. Coeficientes de correlación de imágenes cifradas**. (Fuente: Elaboración propia)

La figura 4.5 hace evidente la consistente reducción de los coeficientes de correlación en las imágenes cifradas. En los experimentos realizados los coeficientes de correlación se mantuvieron por debajo de 0,19 para todas las direcciones en todas las imágenes cifradas.

CONCLUSIONES

- ✓ La notación utilizada y los mecanismos creados para describir la dinámica, evidencian que la teoría de autómatas celulares ofrece un marco de referencia apropiado para el diseño de algoritmos de encriptamiento.
- ✓ La descripción formal del algoritmo de encriptamiento facilita el proceso de codificación en un lenguaje de programación.
- ✓ Los resultados obtenidos demuestran que la dinámica propuesta produce un cifrado alta calidad.
- ✓ La incorporación del parámetro *Distancia Intercelular* propicia un proceso de difusión de naturaleza no lineal que favorece la calidad de la dinámica.

RECOMENDACIONES

- ✓ Se recomienda la construcción de una aplicación de calidad comercial que implemente la dinámica de encriptamiento propuesta.
- ✓ Se recomienda explorar variantes de la dinámica propuesta en la cual el cifrado se realice por bloques de tamaño fijo.
- ✓ Se recomienda utilizar la notación y los mecanismos desarrollados en este trabajo para explorar nuevas dinámicas de encriptamiento.

BIBLIOGRAFÍA

Baldwin, R. (2004). Processing Image Pixeles using Java, Getting Started. Disponible: http://www.developer.com [Consulta: 2013, Enero 15]

Devaney, R. (1989). *An Introduction to Chaotic Dynamical Systems.* Addison-Wesley.

Guan, P. (1987). Cellular Automaton Public-key Cryptosystem. *Complex Systems.* 1, 51-57.

Meier, W. y Staffelbach, O. (1991). Analysis of Pseudorandom Sequences Generated by Cellular Automata. *Advances in Criptology: Eurocrypt'91,* Volume 547 of *Lecture Notes in Computer Science (*Springer-Verlag*)* 186-199.

Nandi, S., Kar, B. y Chaudhuri, P. (1994). Theory and Applications of Cellular Automata in Cryptography. *IEEE Transactions Computers,* 43(12), 1346-1357.

Pixabay. (2013). *Imágenes Digitales*. Disponible: http://www.pixabay.com [Consulta: 2014, Enero 23]

Ramio, J. (2006). *Seguridad Informática* [Libro en Linea].UPM. Disponible: http://www.upm.es [Consulta: 2013, Febrero 20]

Stallings, W. (2011). *Cryptography and Network Security: Principles and Practice.* New Jersey, USA: Prentice Hall.

Stone, H. (1973). *Discrete Mathematics.* Stanford University .CCS Editorial.

Toffoli, T. y Margolus, N. (1987). *Cellular Automata Machines.* Cambridge Massachussets: The MIT Press.

Tomassini, M. y Perrenoud, M. (2000). Stream Ciphers with One and Two-Dimensional Cellular Automata. *Parallel Problem Solving from Nature-PPSN VI. LNCS1927 (Springer) 722-731.*

Weiss, M. (1993). *Data Structures and Algorithm Analysis in C.* Redwood City, California. The Benjamin/Cummings Publishing Company.

Whitfield, D. y Hellman, M. (1976). New Direction in Criptography. *IEEE Transactions on Information Theory.* IT-22.

Wolfram, S. (1986a). Criptography with cellular automata. *Advances in Crryptology: Crypto'85 Proceedings, Lectura Notes in Computer Science,* 218 (Springer-Verlag) 429-432.

Wolfram, S. (1986b). Random Sequence Generation by Cellular Automata. *Advances in Applied Mathematics,* 7 123-169.

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

TÍTULO	DINAMICA REVERSIBLE DE AUTOMATAS CELULARES BIDIMENSIONALES PARA EL CIFRADO DE IMÁGENES DIGITALES A COLOR
SUBTÍTULO	

AUTOR (ES):

APELLIDOS Y NOMBRES	**CÓDIGO CVLAC / E MAIL**
Bastardo ., José	CVLAC: 6.890.832 EMAIL: josebastardo@gmail.com

PALABRAS O FRASES CLAVES:

Cifrado, Criptosistema, Autómata Celular, Dinámica.

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

ÁREA	SUB ÁREA
TRABAJO DE ASCENSO	PROFESOR TITULAR

RESUMEN (ABSTRACT):

La transferencia de la data de una imagen digital al espacio celular de un autómata celular bidimensional hace posible realizar un procesamiento sobre la imagen original mediante la aplicación reiterada de la regla del autómata sobre todas las vecindades del espacio celular. La regla del autómata y la disposición de las vecindades determinan una dinámica que hace que el espacio celular pase por diversas configuraciones, recorriendo una trayectoria que en su estado inicial refleja la imagen digital original y en su estado final contiene la imagen digital procesada. En una dinámica de autómata celular reversible, asociada a la regla, existe una regla inversa que permite recorrer la trayectoria anterior en sentido contrario. En este trabajo se hizo uso de la simplicidad de las reglas de autómatas celulares reversibles y de su capacidad para generar desorden o desinformación, sobre la configuración inicial del espacio celular, para desarrollar una dinámica reversible que permite el cifrado de imágenes digitales. La dinámica desarrollada propone un esquema de disposición de las vecindades que modifica el número de células que conforman vecindades, en el espacio celular. Se construyó una descripción formal de la dinámica propuesta, esta descripción se codificó en el lenguaje de programación Java y se realizaron diversas pruebas que demuestran la calidad del cifrado ofrecido.

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

CONTRIBUIDORES:

APELLIDOS Y NOMBRES	ROL / CÓDIGO CVLAC / E_MAIL				
	ROL	CA	AS	TU	JU
	CVLAC:				
	E_MAIL				
	E_MAIL				
	ROL	CA	AS	TU	JU(X)
	CVLAC:				
	E_MAIL				
	E_MAIL				
	ROL	CA	AS	TU	JU(X)
	CVLAC:				
	E_MAIL				
	E_MAIL				

FECHA DE DISCUSIÓN Y APROBACIÓN:

2014	**02**	
AÑO	**MES**	**DÍA**

LENGUAJE. SPA

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

ARCHIVO (S):

CARACTERES EN LOS NOMBRES DE LOS ARCHIVOS: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z. a b c d e f g h i j k l m n o p q r s t u v w x y z. 0 1 2 3 4 5 6 7 8 9.

NOMBRE DE ARCHIVO	TIPO MIME
TESIS.Dinamica reversible de automatas celulares bidimensionales para el cifrado de imagenes digitales.doc	**Aplicación/msword**

ALCANCE

ESPACIAL: (OPCIONAL)

TEMPORAL: **(**OPCIONAL)

TÍTULO O GRADO ASOCIADO CON EL TRABAJO:

TRABAJO DE ASCENSO A LA CATEGORÍA DE PROFESOR TITULAR

NIVEL ASOCIADO CON EL TRABAJO:

TRABAJO DE ASCENSO

ÁREA DE ESTUDIO:

DEPARTAMENTO DE COMPUTACIÓN Y SISTEMAS

INSTITUCIÓN:

UNIVERSIDAD DE ORIENTE NÚCLEO DE ANZOÁTEGUI

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO:

UNIVERSIDAD DE ORIENTE
CONSEJO UNIVERSITARIO
RECTORADO

CU Nº 0975

Cumaná, 04 AGO 2009

Ciudadano
Prof. JESÚS MARTÍNEZ YÉPEZ
Vicerrector Académico
Universidad de Oriente
Su Despacho

Estimado Profesor Martínez:

Cumplo en notificarle que el Consejo Universitario, en Reunión Ordinaria celebrada en Centro de Convenciones de Cantaura, los días 28 y 29 de julio de 2009, conoció el punto de agenda **"SOLICITUD DE AUTORIZACIÓN PARA PUBLICAR TODA LA PRODUCCIÓN INTELECTUAL DE LA UNIVERSIDAD DE ORIENTE EN EL REPOSITORIO INSTITUCIONAL DE LA UDO, SEGÚN VRAC Nº 696/2009"**.

Leído el oficio SIBI – 139/2009 de fecha 09-07-2009, suscrita por el Dr. Abul K. Bashirullah, Director de Bibliotecas, este Cuerpo Colegiado decidió, por unanimidad, autorizar la publicación de toda la producción intelectual de la Universidad de Oriente en el Repositorio en cuestión.

Comunicación que hago a usted a los fines consiguientes.

UNIVERSIDAD DE ORIENTE
SISTEMA DE BIBLIOTECA
RECIBIDO POR
FECHA 5/8/09 HORA 3:30

Cordialmente,

JUAN A. BOLAÑOS CURIEL
Secretario

REPÚBLICA BOLIVARIANA DE VENEZUELA * UNIVERSIDAD DE ORIENTE * SECRETARÍA * CONSEJO UNIVERSITARIO

C.C: Rectora, Vicerrectora Administrativa, Decanos de los Núcleos, Coordinador General de Administración, Director de Personal, Dirección de Finanzas, Dirección de Presupuesto, Contraloría Interna, Consultoría Jurídica, Director de Bibliotecas, Dirección de Publicaciones, Dirección de Computación, Coordinación de Teleinformática, Coordinación General de Postgrado.

JABC/ YGC/ maruja

Apartado Correos 094 / Telfs: 4008042 - 4008044 / 8008045 Telefax: 4008043 / Cumaná - Venezuela

METADATOS PARA TRABAJOS DE GRADO, TESIS Y ASCENSO

DERECHOS

De acuerdo al artículo 41 del Reglamento de Trabajo de Grado:

"Los trabajos son propiedad exclusiva de la Universidad de Oriente, y solo podrán ser utilizados para otros fines con el consentimiento expreso del Consejo de Núcleo respectivo, quien deberá participarlo previamente al Consejo Universitario para su autorización"

AUTOR

Prof. Bastardo., José L.

_____	_____	_____
JURADO	**JURADO**	**JURADO**