

**RECONSTRUCCIÓN DE IMÁGENES  
MEDIANTE  
AUTÓMATAS CELULARES EVOLUTIVOS**

**Ing. Mauricio Paletta**

**Año 2000**

## VEREDICTO

**Quienes suscriben, Miembros del Jurado designado por el Consejo de la Facultad de Ingeniería de la Universidad Central de Venezuela, para examinar el Trabajo de Grado presentado por el Ingeniero Mauricio Paletta portador de la Cédula de Identidad V-6560965, bajo el título Reconstrucción de Imágenes mediante Autómatas Celulares Evolutivos a los fines de cumplir con el requisito legal para optar al grado de Magister Scientiarum en Computación Emergente, dejan constancia de lo siguiente:**

1.- Leído como fue dicho trabajo por cada uno de los Miembros del Jurado, éste fijó el día 9 del mes de Junio del 2000 a las 3:30 p.m., para que el autor lo defendiera en forma pública, lo que éste hizo en el aula No. 118 del Edificio de la Escuela Básica, mediante un resumen oral de su contenido, luego de lo cual respondió satisfactoriamente a las preguntas que le fueron formuladas por el Jurado; todo conforme a lo dispuesto en los Artículos 44, 49, 50, 51 y 52 del Reglamento de Estudios de Postgrado vigente.

2.- Finalizada la defensa pública del trabajo, el Jurado decidió aprobarlo por considerar, sin hacerse solidario de las ideas expuestas por el autor, que se ajusta a lo dispuesto y exigido en el Reglamento de Estudios de Postgrado.

Para dar este veredicto, el Jurado estimó que la obra examinada constituye un aporte original a la disciplina de Computación Emergente en cuanto presenta la aplicación de un programa evolutivo hibridado con autómatas celulares al problema de restauración de imágenes digitales. Esto constituye una generalización novedosa de los algoritmos evolutivos resultando en una metodología de amplia aplicación en la resolución de problemas complejos. Por esta razón el jurado por unanimidad le otorga **Mención Honorífica**.

En fe de lo cual se levanta la presente Acta, en Caracas a los 9 días del mes de Junio del 2000, dejándose también constancia de que conforme a lo dispuesto en la normativa jurídica vigente, actuó como Coordinador del Jurado el Tutor Profesor José Ali Moreno H.

**Prof. José M. Ramírez**

**Prof. José S. Rodríguez**

**Coordinador Prof. José Ali Moreno H.**

**UNIVERSIDAD CENTRAL DE VENEZUELA  
FACULTAD DE INGENIERÍA  
COORDINACIÓN DE ESTUDIOS DE POSTGRADO  
DEP. DE INVESTIGACIÓN DE OPERACIONES Y COMPUTACIÓN  
MAESTRÍA EN COMPUTACIÓN EMERGENTE**

---

**RECONSTRUCCIÓN DE IMÁGENES MEDIANTE  
AUTÓMATAS CELULARES EVOLUTIVOS**

**Trabajo de Grado  
presentado a la ilustre  
Universidad Central de  
Venezuela para optar al  
título de Magister  
Scientiarum en Computación  
Emergente**

---

Ing. Mauricio Paletta  
Dr. José Alí Moreno

## **RESUMEN**

Uno de los problemas que más ha llamado la atención en el campo de la investigación relacionada con el procesamiento digital de imágenes está relacionado con la reconstrucción, filtrado o tratamiento de ruido sobre imágenes distorsionadas para obtener las versiones originales de las mismas. Parte de la problemática está inmersa en lograr soluciones de software o hardware que permitan lidiar con dos medidas, por lo general inversamente proporcionales, como lo son el tiempo y la calidad de reconstrucción.

La motivación principal de este trabajo se centra en la presentación de una metodología de reconstrucción de imágenes basada en la dinámica de transición de autómatas celulares. La idea es aprovechar la facilidad de definición de los autómatas, aunado a la posibilidad de obtener soluciones de hardware, sumamente atractivo si se piensa en el tiempo de reconstrucción.

Se plantea el uso de un programa evolutivo para obtener el conjunto de reglas de transición del autómata y otra serie de criterios e ideas para definir y configurar debidamente los Autómatas Celulares Evolutivos para este problema particular. Se muestran diversos ensayos y resultados en problemas relacionados con imágenes en blanco y negro e imágenes en 256 tonos de grises.

## **CONTENIDO**

Lista de símbolos y abreviaturas.....	iv
Lista de tablas.....	v
Lista de figuras.....	vi
Lista de algoritmos.....	ix
<b>Fundamentos Teóricos</b>	
Una breve explicación sobre Autómatas Celulares.....	1
El problema de la Reconstrucción de Imágenes.....	6
Una breve explicación sobre Programación Evolutiva.....	7
Definición de Autómatas Celulares con Programación Evolutiva.....	10
<b>Autómatas Celulares Evolutivos para la Reconstrucción de Imágenes</b>	
<b>Elementos básicos del Autómata Celular.....</b>	<b>12</b>
Reglas de Transición del Autómata Celular.....	<b>13</b>
Metodología de Reconstrucción.....	22
Medida de Calidad de Reconstrucción.....	28
<b>Ensayos y Resultados</b>	
Introducción.....	30
Ensayos con imágenes blanco y negro.....	32
Ensayos con imágenes en 256 tonos de grises.....	49
Comentarios finales.....	59
<b>Conclusiones.....</b>	<b>60</b>
<b>Anexos</b>	
Anexo1. Detalles del desarrollo de software.....	62
Referencias Bibliográficas.....	63

## LISTA DE SÍMBOLOS Y ABREVIATURAS

$\leftarrow$	Operador de asignación en algoritmos
$\xi$	Funcional de error o medida de calidad
$\varphi$	Imagen
$\varphi^c$	Imagen contaminada
$\varphi^r$	Imagen reconstruida
$AC$	Autómata Celular
$D$	Descendencia de un programa evolutivo
$I$	Individuo de un programa evolutivo
$L$	Alfabeto de un autómata celular
$P$	Población de un programa evolutivo
$S$	Reticulado de un autómata celular
$s$	Celda o autómata del reticulado de un autómata celular
$T$	Conjunto de reglas de transición de un autómata celular
$t$	Medida discreta del tiempo
$V$	Vecindad de un autómata celular
$v$	Dimensión del vector de vecindad de un autómata celular

**LISTA DE TABLAS**

Tabla 1.	Dimensión del vector de vecindad en función del tipo de vecindad.....	15
Tabla 2.	Medidas de calidad sobre imágenes blanco y negro luego de aplicar Filtros de 30, 20 y 10 % de ruido.....	44
Tabla 3.	Medidas de calidad sobre imágenes blanco y negro luego de aplicar el Filtro 10 % de ruido 3 veces consecutivas.....	45
Tabla 4.	Medidas de calidad sobre imágenes blanco y negro luego de aplicar el Filtro 20 % de ruido 3 veces consecutivas.....	45

## LISTA DE FIGURAS

Figura 1.	Vecindades de von Neumann, Moore y Mixta.....	13
Figura 2.	Ejemplos de reglas de transición con vecindad von Neumann 1 y $L = \{ 0, 1 \}$ .....	15
Figura 3.	Representaciones vectoriales de vecindades de von Neumann, Moore y Mixta.....	16
Figura 4.	Ejemplo de la simetría de una regla de transición con vecindad von Neumann 1 y $L = \{ 0, 1 \}$ .....	17
Figura 5.	Ejemplo de tonos de gris.....	26
Figura 6.	Imágenes de referencia para ensayos con $L = 2$ .....	33
Figura 7.	Imagen A (blanco y negro) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (10 % ruido).....	34
Figura 8.	Imagen B (blanco y negro) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (10 % ruido).....	35
Figura 9.	Imagen C (blanco y negro) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (10 % ruido).....	36
Figura 10.	Imágenes A, B y C (blanco y negro) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (10 % ruido).....	37
Figura 11.	Imágenes A, B y C (blanco y negro) reconstruidas luego de aplicar autómatas celulares evolutivos (0 % ruido).....	38
Figura 12.	Imágenes A, B y C (blanco y negro) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (20 % ruido).....	39



Figura 13. Imágenes A, B y C (blanco y negro) reconstruidas (20 % ruido) luego de aplicar filtros de 20 y 10 % de ruido.....	40
Figura 14. Imágenes A, B y C (blanco y negro) reconstruidas (0 % ruido) luego de aplicar filtros de 20 y 10 % de ruido.....	40
Figura 15. Imágenes A, B y C (blanco y negro) original, contaminada y reconstruida luego de aplicar autómata celular evolutivo (30 % ruido).....	41
Figura 16. Imágenes A, B y C (blanco y negro) reconstruidas (30 % ruido) luego de aplicar filtros de 30, 20 y 10 % de ruido.....	42
Figura 17. Imágenes A, B y C (blanco y negro) reconstruidas (0 % ruido) luego de aplicar filtros de 30, 20 y 10 % de ruido.....	43
Figura 18. Imágenes A, B y C (blanco y negro) original, contaminada (40 %) y reconstruida luego de aplicar filtros de 30, 20 y 10 % de ruido.....	44
Figura 19. Imágenes 110 x 110 pixeles (blanco y negro) original, contaminada (30 %) y reconstruida luego de aplicar filtros de 30, 20 y 10 % de ruido.....	46
Figura 20. Imágenes 110 x 110 pixeles (blanco y negro) (sin contaminar) reconstruidas luego de aplicar filtros de 30, 20 y 10 % de ruido.....	47
Figura 21. Imágenes 110 x 110 pixeles (blanco y negro) (sin contaminar) reconstruidas luego de aplicar el filtros de 10 % de ruido 3 veces.....	47
Figura 22. Imágenes 300 x 300 pixeles (blanco y negro) original, contaminada (30 %) y reconstruida luego de aplicar filtros de 30, 20 y 10 % de ruido.....	48
Figura 23. Imágenes 300 x 300 pixeles (blanco y negro) (sin contaminar) reconstruidas luego de aplicar filtros de 30, 20 y 10 % de ruido.....	48

Figura 24. Imágenes de referencia para ensayos con $L = 256$ .....	50
Figura 25. Imagen A (256 tonos de grises) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (10 % ruido).....	51
Figura 26. Imagen A (256 tonos de grises) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (20 % ruido).....	52
Figura 27. Imagen A (256 tonos de grises) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (30 % ruido).....	52
Figura 28. Imagen B (256 tonos de grises) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (10 % ruido).....	52
Figura 29. Imagen B (256 tonos de grises) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (20 % ruido).....	53
Figura 30. Imagen B (256 tonos de grises) original, contaminada y reconstruida luego de aplicar autómatas celulares evolutivos (30 % ruido).....	53
Figura 31. Imágenes A y B (256 tonos de grises) original, contaminada y reconstruida luego de aplicar filtros de 30, 20 y 10 % de ruido.....	54
Figura 32. Imágenes A y B (256 tonos de grises) (sin contaminar) reconstruidas luego de aplicar filtros de 30, 20 y 10 % de ruido.....	54
Figura 33. Ensayo con filtro Gaussiano ( $\sigma = 1$ ) sobre Lena.....	55
Figura 34. Ensayo con filtro 30 % sobre Lena.....	56
Figura 35. Ensayo con imágenes de satélite con 30 % de ruido.....	58

**LISTA DE ALGORITMOS**

Algoritmo 1. Dinámica de un autómeta celular.....	5
Algoritmo 2. Dinámica de un programa evolutivo.....	10
Algoritmo 3. Evaluación de los individuos del autómeta celular evolutivo.....	20
Algoritmo 4. Metodología de reconstrucción para varios niveles de contaminación.....	25

## **FUNDAMENTOS TEÓRICOS**

### **Una breve explicación sobre Autómatas Celulares.**

Previo a explicar lo que es un *Autómata Celular*, es importante conocer las bases teóricas implícitas en un *Sistema Dinámico* [2] 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.

En muchos casos prácticos no es fácil identificar estados y/o reglas, por un lado los estados pueden ser difíciles de identificar como por ejemplo en ecología, asociados al clima o relativos el cerebro; en otros casos las reglas pueden ser muy complicadas (flujo plástico de metales o relativas el cerebro, por ejemplo) y en otras ocasiones, las reglas pueden constituirse en estados (un programa puede modificarse a sí mismo de manera que las reglas no son fijas).

La idea física que sugiere que un *Sistema* consiste de un objeto reconocible, en lenguaje matemático significa que se tiene una forma de suministrar información completa sobre el *Sistema*, información que al fin de cuenta constituye un estado.

Esto quiere decir que el concepto de estado está íntimamente ligado a las reglas que actúan sobre ellos.

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). Dependiendo si los estados y el tiempo son discretos o continuos, los *Sistemas Dinámicos* se pueden dividir en tres grandes categorías:

1. **Autómatas Celulares:** con estados y tiempo discretos.
2. **Mapas:** con estados representados por un conjunto continuo de números y tiempo discreto.
3. **Flujos:** con estados y tiempo continuo.

Los *Autómatas Celulares* [6][7] son un formalismo para el modelado de *Sistemas Dinámicos* donde el espacio, tiempo y estados son discretos. Está representado 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 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*.

Un *Autómata Celular* puede ser entonces completamente definido por cuatro componentes: un reticulado regular  $n$ -dimensional  $S$ , un alfabeto  $L$ , una vecindad  $V$  y un conjunto de reglas de transición  $T$ :

$$AC = \{S, L, V, T\} \quad (1)$$

El estado de un *Autómata Celular* en un tiempo  $t$  cualquiera, viene dado por el valor del conjunto de celdas  $s_i(t)$  de  $S$ . La regla de transición se puede expresar como una relación funcional entre el estado del sistema a tiempo  $t$  y el estado anterior (tiempo  $t-1$ ) que ocurre sincrónicamente para todas las celdas. Define una correspondencia entre una  $v$ -tupla de estados posibles a un sólo estado posible, donde  $v$  viene dado por la vecindad  $V$  del *Autómata Celular*:

$$\begin{aligned} T : L^v \rightarrow L &\Rightarrow (s_1, s_2, \dots, s_v) \in L^v \rightarrow T(s_1, s_2, \dots, s_v) \in L \\ s_i(t+1) &= T(s_1(t), \dots, s_v(t)) \quad \forall s_i \in S \end{aligned} \quad (2)$$

Por ejemplo, supongamos que se tiene un arreglo unidimensional de celdas o reticulado  $S$  de dimensión 1, cuyos posibles valores o alfabeto están en el conjunto  $\{0, 1\}$  y cuyos estados evolucionan con pasos discretos en el tiempo considerando

sólo los primeros vecinos de cada celda  $V$ , de acuerdo a las siguientes reglas de transición  $T$ :

- Si el estado es ... 1 1 1 ... evoluciona a ... 1 0 1 ...
- Si el estado es ... 1 0 1 ... evoluciona a ... 1 1 1 ...
- Si el estado es ... 0 1 0 ... evoluciona a ... 0 0 0 ...
- Si el estado es ... 1 1 0 ... evoluciona a ... 1 0 1 ...
- Si el estado es ... 0 0 0 ... evoluciona a ... 0 0 0 ...

La dinámica de estos sistemas es estudiada asignando un patrón inicial, frecuentemente escogido al azar y permitiendo que el sistema evolucione hasta alcanzar algún criterio de finalización o convergencia, tal como se aprecia en el Algoritmo 1. Dependiendo de las reglas, estos sistemas exhiben un conjunto muy rico de comportamientos dinámicos:

- Patrones que no cambian en el tiempo.
- Secuencia de patrones que se repiten.
- Patrones a los cuales otros patrones evolucionan.
- Patrones casi idénticos evolucionan a patrones finales substancialmente diferentes<sup>1</sup>.

---

<sup>1</sup> Ver “El Juego de la Vida” de Jhon Conway

El hecho que la función de transición tome valores discretos sobre un conjunto finito, le da características de exactitud a la dinámica de los *Autómatas Celulares*. Todo esto permite representar y modelar *Sistemas Dinámicos* tan complejos como se desee.

La dimensión de  $S$ , el número de estados discretos que definen el alfabeto  $L$  y la vecindad  $V$  tienen una dependencia directa del problema que se quiere modelar. La principal complejidad de los *Autómatas Celulares* está en la definición de las reglas de transición que gobiernan su dinámica, esto aunado al hecho de la gran cantidad de posibilidades que existen y que se complica cada vez más a medida que  $L$  y  $V$  aumentan.

```

t ← 0
InicializarAutómata()
mientras (no condición de terminación) hacer
  para todo  $s_i(t) \in S$  hacer
     $s_i(t+1) \leftarrow \text{TransiciónCelda}(T, V(s_1(t)^i, s_2(t)^i, \dots, s_v(t)^i))$ 
  t ← t + 1
solución ← EstadoAutómata()

```

**Algoritmo 1.**  
Dinámica de un *Autómata Celular*

Uno de los aspectos más importantes que motivan el hacer modelos de *Sistemas Dinámicos* complejos con *Autómatas Celulares* es que su simplicidad en la definición facilita su implementación mediante dispositivos de hardware, sumamente atractivo para desarrollar la dinámica sincrónica del problema.



### **El problema de la Reconstrucción de Imágenes.**

La *Reconstrucción de Imágenes* [3] envuelve los conceptos matemáticos y algoritmos necesarios para generar una imagen a partir de un conjunto de data adquirido. El caso más típico es aquél en el cual el conjunto de data corresponde a una alteración ruidosa de la información que representa la imagen original que se quiere obtener. En este caso el proceso de reconstrucción se convierte en una técnica de eliminación de ruido o filtro, también llamado *Restauración de Imágenes*.

La restauración de imágenes degradadas es una rama del procesamiento digital estrechamente relacionado con los conceptos de segmentación y definición de bordes. Este tema es muy estudiado por su evidente importancia práctica e interés teórico.

Se han definido varios algoritmos que han tratado el tema de la *Reconstrucción de Imágenes* para dos y tres dimensiones. En forma general se pueden identificar tres métodos: por proyección, en los que se pueden mencionar los filtros de retroproyección; iterativos, básicamente representados por las técnicas algebraicas y los métodos de máxima entropía y según análisis de Fourier.

Los métodos más comunes que se han usado para implementar filtros en imágenes digitales tienen que ver con procesos de convolución o multiplicación de frecuencias empleando transformadas de Fourier [6]. La proyección tomográfica (Bracewell y Riddle, 1967) es una de las aplicaciones más estudiadas en el problema de *Reconstrucción de Imágenes*.

También se han aplicado algunas técnicas emergentes entre los cuales se pueden citar el trabajo de Nikos Drakos (1994) sobre reconstrucción de imágenes de proyección tomográfica haciendo uso de Recocido Simulado<sup>2</sup> y el trabajo de Luigi Bedini y Anna Tonazzini (1990) haciendo uso de Métodos de Máxima Entropía<sup>3</sup>.

### **Una breve explicación sobre Programación Evolutiva.**

Uno de los procesos más comunes y básico en la operación de una computadora es la *búsqueda*. Los métodos tradicionales de búsqueda basados en el cálculo, enumerativos y estocásticos son muy dependientes de la información sobre el dominio del problema y lo que es peor, un crecimiento en la dimensión del espacio de estados los torna muy ineficientes en tiempo de ejecución y calidad de respuesta.

Los seres vivos han utilizado una estrategia muy exitosa para resolver el complejo problema de búsqueda de la supervivencia de las especies. Se trata de la *Evolución Natural*. Consiste en un proceso de adaptación al ambiente cuya recompensa es la supervivencia. Es un proceso de selección acumulativo en el cual cada mejora en la calidad de un individuo va en miras a mejorar la calidad de la próxima generación de individuos. Basa su principio en dos procesos primarios:

---

<sup>2</sup> “PET (Positron Emission Tomography) Image Reconstruction using Simulated Annealing; Nikos Drakos; Computer Based Learning Unit; University of Leeds.

<sup>3</sup> “Neural Network used in Maximum Entropy Image Restoration”; Luigi Bedini & Anna Tonazzini.

- La selección natural, que determina los individuos aptos a participar en la próxima generación de individuos.
- La reproducción, que asegura la mezcla de rasgos de los padres previamente seleccionados para generar la descendencia.

Por otro lado, hay una serie de hechos que se sabe forman parte en el proceso de *Evolución Natural* y es importante mencionar:

- La evolución ocurre en el ámbito de los cromosomas.
- Los cromosomas codifican la estructura de los seres.
- La codificación y decodificación de la información genética son esenciales para la existencia.
- La selección natural relaciona los cromosomas con la calidad de ejecución del ser codificado.
- La evolución ocurre durante la reproducción.
- La mutación y combinación del material genético son los principales mecanismos de reproducción.
- Todo lo que se conoce sobre la especie está en los cromosomas de los individuos de la población actual (no hay memoria).

Varios investigadores entre los cuales se pueden citar a Rechemberg (1973), Fogel (1966), Glover (1977), Holland (1975) y Koza (1991), llegaron a la conclusión que el

proceso de *Evolución Natural* era poderoso en la resolución de problemas complejos, específicamente aquellos relacionados con búsqueda [1][5]. Estrategias evolutivas, programación evolutiva, algoritmos genéticos y programación genética son algunos de los nombres con los cuales se conoce a estas técnicas de búsqueda basadas en el proceso de *Evolución Natural*.

La programación evolutiva es una técnica emergente de búsqueda de buenas soluciones a problemas complejos. Muy utilizada en problemas de optimización. Enfoca el proceso de resolución a través de una dinámica evolutiva, emulando el proceso de selección de los más aptos y la reproducción de las especies. La dinámica evolutiva genera trayectorias en el espacio de estados de forma tal que se logra un acotamiento apreciable en el espacio de búsqueda, obviando trayectorias que no son productivas y dejando aquellas que mejor se adaptan al espacio de solución del problema tratado.

Dada una representación a la solución de un problema al cual llamamos *individuo* e iniciando un proceso con un conjunto finito de individuos que llamamos *población*, la idea es usar una estrategia estocástica de reproducción sobre los individuos de la población para crear una descendencia de forma tal de mejorar un funcional que mide que tan buena es la solución del problema representada por cada individuo. Cada paso de este proceso es una *evolución*; las operaciones que se aplican sobre los individuos de la población llevan el nombre de *operadores genéticos* y el funcional se conoce como *función de aptitud* o costo. El Algoritmo 2 presenta los detalles de este proceso.

Definiendo correctamente la función de aptitud y los operadores genéticos, por lo general cruce y mutación de individuos y configurando adecuadamente el tamaño de la población, la frecuencia de uso de los operadores y la selección de los individuos a operar, la programación evolutiva garantiza converger a soluciones aceptables del problema.

```

t ← 0
InicializarPoblación(P(t))
mientras (no condición de terminación) hacer
  EvaluarPoblación(P(t))
  mientras (no complete descendencia D(t)) hacer
    pi ← SeleccionarIndividuo(P(t))
    CrearDescendencia(D(t), pi)
  ReemplazarPoblación(P(t), D(t))
t ← t + 1
solución ← MejorIndividuo(P(t))

```

**Algoritmo 2.**  
Dinámica de un *Programa Evolutivo*

### **Definición de Autómatas Celulares con Programación Evolutiva.**

Desde hace ya unos 10 años atrás y debido a la complejidad existente en la definición de las reglas de actuación de los *Autómatas Celulares* para representar o modelar una gran gama de problemas, muchos investigadores han trabajado en la búsqueda de métodos y algoritmos para facilitar esta labor. Es prudente mencionar los trabajos realizados por el Grupo de Autómatas Celulares Evolutivos del Instituto de Santa Fe<sup>4</sup>.

<sup>4</sup> Como principales investigadores se pueden citar a Melanie Mitchell y James P. Crutchfield.

Bajo este contexto se pueden observar casos híbridos de técnicas emergentes que involucran a los Autómatas Celulares para tratar de obtener los esquemas de solución requeridos. Con Redes Neuronales<sup>5</sup> y Algoritmos Genéticos<sup>6</sup>, por ejemplo.

Una forma de llegar o *Evolucionar* a la definición de un *AC* es usando las bondades de la *Programación Evolutiva*. En este esquema, un *Individuo* representa un *AC* completamente definido para un caso particular. Si se plantea una adecuada representación del problema con un *AC* y una función que permita medir la calidad con la cual la dinámica del *AC* emerge a la solución, en teoría, el algoritmo implícito en un programa evolutivo debe tender a buscar la optimización de esta medida de calidad y por ende buenos resultados del problema.

La idea básica es esconder la complejidad inmersa en la definición de las reglas de transición del *AC* con un programa evolutivo. La clave para tener éxito al aplicar esta idea está en la forma de representar los conceptos propios de la programación evolutiva, principalmente en lo que se refiere a *Individuos* y *Operadores Genéticos* y la manera como éstos se asocien a la definición de un *AC*.

---

<sup>5</sup> “Evolving Cellular Automata Neural Systems”; Dong Wook Lee & Kwee Bo Sim.

<sup>6</sup> “Evolving Cellular Automata with Genetic Algorithms: A Review of Recent Work”; Melanie Mitchell, James P. Crutchfield & Rajashi Das.

## AUTÓMATAS CELULARES EVOLUTIVOS PARA LA RECONSTRUCCIÓN DE IMÁGENES

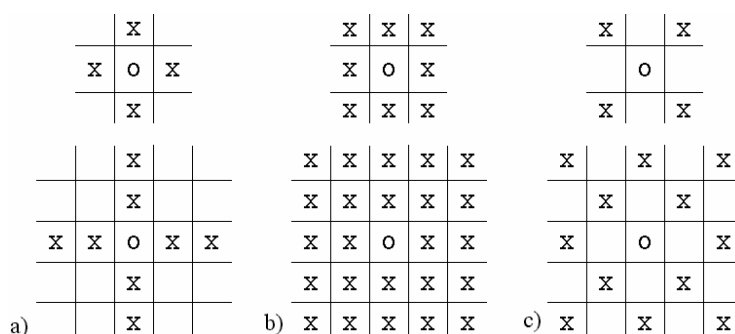
### **Elementos básicos del Autómata Celular.**

La motivación principal de este trabajo consiste en buscar una solución al problema de reconstrucción de imágenes haciendo uso de *Autómatas Celulares*. Para lograr esto, es necesario plantear un esquema de representación del autómata acorde a este problema de acuerdo a la definición dada en (1).

Una imagen se puede representar mediante un reticulado en el cual cada retícula o pixel contiene información sobre el color que corresponde a esa posición específica. Está claro la similitud en la forma como se representa una imagen y la forma como se define el conjunto de celdas  $S$  de un *Autómata Celular*.  $S$  debe ser un reticulado de 2 dimensiones con tantas celdas como pixeles tenga la imagen.

El alfabeto  $L$  se define según el tipo de imagen tratada. Por ejemplo,  $L = 2$  para imágenes blanco y negro;  $L = 16$  para imágenes a 16 colores;  $L = 256$  para imágenes en escalas de grises; etc. Imágenes con una mayor resolución de colores formadas por la 3-tupla de valores (*rojo, verde, azul*) donde *rojo, verde* y *azul* son valores discretos entre 0 y 255, se puede representar mediante la combinación de tres *Autómatas Celulares* con  $L = 256$ , cada uno de los cuales haría el papel de una componente o plano del color.

La vecindad  $V$  del *Autómata Celular* debe ser válida para un reticulado de 2 dimensiones. Para este trabajo se han seleccionado las vecindades más conocidas y usadas en el espacio  $n$ -dimensional con  $n > 1$ : *von Neumann* y *Moore*. Se ha incorporado además, el uso de una vecindad combinada al que hemos llamado *Mixta*. Para simplificar la definición de reglas sólo estamos considerando niveles de vecindad a 1 y 2 celdas. La figura 1 muestra una representación esquemática de las vecindades de *von Neumann* (parte a), *Moore* (parte b) y *Mixta* (parte c). El símbolo O corresponde a la celda en transición y X a las celdas vecinas. Las vecindades a un nivel se muestran en la parte superior y las de dos niveles en la parte inferior de la figura.



**Figura 1.**  
Vecindades: a) von Neumann; b) Moore; c) Mixta

### Reglas de Transición del Autómata Celular.

La última componente para completar la definición del *Autómata Celular* se refiere al conjunto de reglas de transición que permiten al autómata modelar un proceso de reconstrucción de imágenes. Si se mira el dominio de búsqueda de todas las combinaciones posibles en las cuales se puede decidir el valor siguiente de una celda



del autómata en función de su vecindad, se puede apreciar que se trata de un problema muy complejo de optimización combinatoria. Parte de los resultados a alcanzar en este trabajo es definir una metodología o estrategia para resolver este problema.

Nuestro planteamiento es usar un programa evolutivo que permita converger a este conjunto de reglas. En este modelo, un *Individuo*  $I_i$  representa la definición completa de un *Autómata Celular*. Para toda la población, el tamaño de  $S$  y los valores discretos de  $L$  se mantienen fijos en función del conjunto de imágenes que se usan como referencia para evaluar el comportamiento de cada individuo.

$$I_i = \{S, L, V_i, T_i\} \quad (3)$$

La vecindad  $V_i$  se inicializa al azar con alguno de los seis casos posibles (figura 1) y según una distribución de probabilidades que se puede ajustar previo al proceso de inicialización. Una vez que a la vecindad  $V_i$  del individuo  $I_i$  se le haya asignado una opción válida, permanece fija durante el proceso de evolución.

Luego de escoger  $V_i$ , el conjunto de reglas de transición  $T_i$  es inicializado convenientemente al azar de acuerdo al alfabeto  $L$  y al valor dado a  $V_i$ . Una regla de transición se puede ver como una  $(v+1)$ -tupla de valores discretos que pertenecen al alfabeto  $L$ :

$$T = \{ (c_1, c_2, \dots, c_v, c_{v+1}) \mid \forall c_i \in L \} \quad (4)$$

Los primeros  $v$  elementos pertenecen a la vecindad mientras que el elemento  $v+1$  corresponde a la conclusión de la regla, es decir, el nuevo valor en tiempo  $t+1$  que debe tomar la celda en el caso que la regla se satisfaga en tiempo  $t$ . El tamaño de  $v$  depende del tipo de vecindad. La tabla 1 presenta un resumen de estas cantidades.

Vecindad	Tamaño de $v$	Vecindad	Tamaño de $v$
von Neumann - 1	4	von Neumann - 2	8
Moore - 1	8	Moore - 2	24
Mixta - 1	4	Mixta - 2	12

**Tabla 1.**  
Dimensión de  $v$  en función del tipo de vecindad

La figura 2 muestra un ejemplo de un par de reglas de transición relativas a la vecindad *von Neumann* a un nivel sobre el alfabeto binario  $L = \{ 0, 1 \}$ .

1	0
1   1   1	0   0   0
1	0
1, 1, 1, 1 → 1	0, 0, 0, 0 → 0

**Figura 2.**  
Ejemplos de reglas de transición con vecindad *von Neumann* 1 y  $L = \{ 0, 1 \}$

El orden en el cual se construye el vector de elementos de la vecindad a partir de la representación bidimensional, sigue una secuencia según las manecillas del reloj empezando en el primer vecino inmediatamente superior. La figura 3 muestra los

índices o posiciones de los elementos en la representación vectorial que corresponde a cada tipo de vecindad en su reticulado bidimensional.

		1		
4		5		2
		3		

8		1		2
7		9		3
6		5		4

4			1
		5	
3			2

		5		
		1		
8		4		9
		2		6
		3		

23		24		9		10		11
22		8		1		2		12
21		7		25		3		13
20		6		5		4		14

11			12			5
		4			1	
10			13			6
		3			2	

a) 

		7		
--	--	---	--	--

b) 

19		18		17		16		15
----	--	----	--	----	--	----	--	----

c) 

9			8			7
---	--	--	---	--	--	---

**Figura 3.**

Representaciones vectoriales de vecindades: a) *von Neumann*; b) *Moore*; c) *Mixta*

Según (2) y (4), la satisfacción de una regla se establece cuando la  $\nu$ -tupla que se forma con los elementos vecinos de la celda en evaluación cumplan algún criterio de unificación con los primeros  $\nu$ -elementos de la regla:

$$s_i(t+1) = c_{\nu+1} \Leftrightarrow \exists (c_1, c_2, \dots, c_\nu, c_{\nu+1}) \in T / (c_1, c_2, \dots, c_\nu) \leftrightarrow (s_1, s_2, \dots, s_\nu) \quad (5)$$

La operación  $\leftrightarrow$  establece una relación de unificación entre las tuplas  $(c_1, c_2, \dots, c_\nu)$  de la regla y  $(s_1, s_2, \dots, s_\nu)$  de la celda a evaluar. La igualdad entre los elementos de las tuplas es el principal criterio de unificación. Otros criterios que dependen de la cardinalidad del alfabeto del autómata serán tratados más adelante en este documento.

El número de reglas de  $T$  es decidido al azar. Las reglas de un mismo conjunto deben ser definidas de forma tal de no ser redundantes ni inconsistentes. La redundancia se

da cuando dos reglas unifican entre sí y concluyen el mismo resultado. Se tiene una inconsistencia o aparece una contradicción cuando dos reglas unifican entre sí y concluyen resultados diferentes.

No es necesario que un conjunto de reglas satisfaga la propiedad de completitud sobre el dominio de posibles combinaciones. Si tenemos una tupla  $(s_1, s_2, \dots, s_v)$  que no unifica con ninguna de las reglas de un conjunto, en este caso se asume que el valor de la celda en evaluación no cambia para esta transición del autómata.

Dado que estamos tratando con un problema de procesamiento de imágenes y buscando reducir la cardinalidad de  $T$ , es conveniente evaluar cada regla de forma simétrica. Esto implica que para la evaluación de las reglas se deben considerar todos sus puntos de rotación como por ejemplo los cuatro casos posibles para *von Neumann* (ver figura 4). En los ejemplos de la figura 2 es claro que todas las simetrías están inmersas en la misma tupla de elementos. Este criterio de evaluación simétrica se debe tomar en cuenta en la comprobación de redundancia e inconsistencia de las reglas.

	1		1, 0, 0, 1 → 1
1	1	0	0, 0, 1, 1 → 1
			0, 1, 1, 0 → 1
	0		1, 1, 0, 0 → 1

**Figura 4.**

Ejemplo de la simetría de una regla de transición con vecindad *von Neumann* 1 y  $L = \{0, 1\}$

Otro aspecto que tomamos en cuenta para reducir el número de reglas y facilitar sus definiciones fue el de incorporar el concepto de *Valor Genérico* como parte de los primeros  $v$  elementos que representan la vecindad. Se identifica con el valor discreto ‘\*’ e indica que para esa posición correspondiente en la vecindad se puede tomar cualquiera de los posibles valores válidos de  $L$ . El uso de estos *Valores Genéricos* está condicionado por una probabilidad y se presenta como un parámetro que debe ser ajustado previo a la creación de los *Individuos*.

En resumen, el conjunto de reglas de transición se debe definir respetando los siguientes criterios:

- El conjunto no puede ser vacío.
- No hay redundancia entre las reglas de un conjunto.
- Las reglas de un conjunto son consistentes entre sí.
- El conjunto no necesariamente debe ser completo.
- Toda regla del conjunto se evalúa de forma simétrica.
- Se permite el uso de *Valores Genéricos* de acuerdo a una probabilidad dada.

Con relación a las operaciones genéticas de cruce y mutación sobre la *Población* de *Individuos*, se definieron los siguientes criterios:

- Cruce simple del conjunto de reglas para *Individuos* con la misma *Vecindad*.

- Cruce al azar del conjunto de reglas para *Individuos* con la misma *Vecindad*.
- Agregación de una regla al conjunto  $T$  de un *Individuo*.
- Eliminación de una regla del conjunto  $T$  de un *Individuo*.
- Modificación de la conclusión de una de las reglas del conjunto  $T$  de un *Individuo*.

Para la selección de los padres se hace uso de un algoritmo de rueda de la ruleta [1]. Una porción de los mejores *Individuos* se mantiene para la próxima generación (Gap generacional) y el peor *Individuo* es reemplazado por una nueva configuración tomada al azar.

Cada *Individuo* de la *Población* debe ser evaluado (ver “**Evaluar Población** ( $P(t)$ )” en Algoritmo 2). La evaluación en este contexto se refiere a ejecutar la dinámica del *Autómata Celular* definido en el *Individuo* sobre un conjunto de imágenes de referencia del mismo tipo e igual tamaño previamente conformado. Estas imágenes son contaminadas con cierto nivel de ruido blanco o uniformemente distribuido y las imágenes contaminadas resultantes constituyen la data de inicialización del autómata. Por cada imagen contaminada del conjunto de referencia se debe ejecutar una dinámica del autómata. Al concluir se tiene un conjunto de *Autómatas Celulares* en su estado final cada uno de los cuales con una tentativa imagen reconstruida o filtrada de su imagen correspondiente. La evaluación final del *Individuo* es una suerte de

promedio de la calidad de reconstrucción del autómata sobre todas las imágenes de referencia. El Algoritmo 3 resume lo expuesto.

para todos los *Individuos*  $I_i$  de  $P(t)$  hacer  
 $\xi \leftarrow 0$   
 para todas las *Imágenes de Referencia*  $\varphi_i$  hacer  
     **InicializarIndividuo**( $I_i, \varphi_i$ )  
     **TransiciónAutómata**( $I_i$ )  
      $\xi \leftarrow \xi + \mathbf{MedidaCalidad}(I_i)$   
**EvaluaciónIndividuo**( $I_i$ )  $\leftarrow \xi / \text{NúmeroImágenes}$

**Algoritmo 3.**  
 Evaluación de los Individuos

Para medir la calidad con la cual cada *Individuo* o *Autómata Celular* limpia el ruido de las imágenes se requiere un funcional de error. Usamos diferencia de Hamming para  $L = 2$  y error cuadrático para  $L > 2$  con expresiones (6) y (7) respectivamente que se muestran a continuación, siendo  $s_i^o$  el valor de la  $i$ -ésima celda del autómata que representa la imagen original y  $s_i^r$  el valor de la  $i$ -ésima celda del autómata que representa la imagen reconstruida.

$$\xi = \sum_{i=1}^S |s_i^o - s_i^r| \quad (6)$$

$$\xi = \frac{1}{2} \sum_{i=1}^S (s_i^o - s_i^r)^2 \quad (7)$$

Es claro que para realizar el cálculo con estas expresiones se requieren las imágenes originales no contaminadas. Esto genera cierta inquietud ya que se supone que las imágenes originales no están disponibles a la hora de usar este paradigma en aplicaciones reales. En realidad estos funcionales sólo se requieren para definir completamente los autómatas para lo cual se tiene total información sobre las imágenes de referencia usadas para la definición. La única limitante es no poder suministrar una medida de calidad en tiempo de ejecución.

Para evitar que las reglas de transición alteren o tengan efectos negativos sobre las imágenes originales en su intento de disminuir el error o mejorar la calidad de la reconstrucción, hemos incorporado en la evaluación del *Individuo* un factor de castigo sobre perturbaciones de la imagen original. De esta forma este paradigma evolutivo busca configuraciones de *Autómatas Celulares* que disminuyan un error entre una imagen reconstruida y su imagen original, tratando de no alterar la imagen original. Una manera de evaluar esta condición es observar y analizar el efecto de aplicar la transición del autómata sobre la imagen sin contaminar.

La medida de calidad  $f(I_i)$  que se aplica a cada *Individuo*  $I_i$  del *Programa Evolutivo* para una *Población* de tamaño  $N$  (ver “**Evaluación Individuo**( $I_i$ )” en Algoritmo 3), es una expresión (8) que busca maximizar el inverso (minimización) de una suerte de error promedio que se obtiene al evaluar el Autómata Celular que representa el Individuo, a cada una de las imágenes de referencia ocurrentes para un total de  $NI$  imágenes.



$$f(I_i) = \left( \frac{\sum_{i=1}^N \xi_i}{NI} \right)^{-1} \quad (8)$$

El criterio o condición de parada del programa evolutivo evaluado (ver “condición de terminación” en Algoritmo 2) es dado por un número finito de iteraciones con posibilidad de continuar si se desea. El mantener un porcentaje de los mejores *Individuos* de la *Población* para la siguiente descendencia garantiza mantener ó mejorar la solución alcanzada. El reemplazar el peor *Individuo* por una nueva definición ayuda a no quedar estancado en mínimos locales de la dinámica impuesta por la evolución.

### **Metodología de Reconstrucción.**

Otro factor a resolver tiene que ver con el criterio a usar para detener la dinámica del *Autómata Celular*. Un primer planteamiento se centró en usar un funcional para medir la calidad de la imagen reconstruida. La dinámica se tendría que detener cuando el error dejara de disminuir. Esta hipótesis es bastante rígida y tiene una dependencia muy directa con la expresión de calidad que se vaya a usar. Nótese que los funcionales utilizados en tiempo de definición, la diferencia de Hamming y el error cuadrático, no nos sirven para este fin ya que requiere de la información implícita en las imágenes originales que no disponemos en tiempo de ejecución.

Para resolver el aspecto relacionado con la condición de terminación de la dinámica del *Autómata Celular* se plantean las siguientes dos hipótesis:

1. El proceso de reconstrucción de una imagen se puede lograr al aplicar una sola transición de varios autómatas configurados de forma diferente, en lugar de aplicar varias transiciones de un mismo autómata.
2. A diferentes niveles de contaminación de las imágenes se evoluciona a configuraciones diferentes de autómatas. Las imágenes reconstruidas para niveles altos de contaminación tienen semejanza con las imágenes contaminadas con niveles más bajos.

La hipótesis 1 garantiza una simplificación del problema tanto en el ámbito del software como a un posible desarrollo futuro de hardware. Es importante recordar que los algoritmos evolutivos se pueden adecuar convenientemente al problema. En nuestro caso, se desea buscar los *Individuos* de la *Población* que en solo una transición de las reglas se obtenga las mejores reconstrucciones. Por otro lado, la distribución de píxeles puede ser tan diversa entre una imagen y otra que puede dar lugar a la evolución de autómatas diferentes. En este contexto se presentan dos casos posibles:

1. Cuando los autómatas están definidos con el mismo tipo de vecindad, los conjuntos de reglas de transición se pueden fusionar de forma adecuada para concebir un solo autómata.

2. Cuando los autómatas están definidos con tipos de vecindad diferente, la solución final resulta al aplicar en serie una transición de cada autómata.

Con respecto a la hipótesis 2, está claro que en una imagen contaminada a un nivel bajo de 10% por ejemplo, se aprecian muchos puntos aislados (la mayoría o todos sus vecinos son de otro color) que lo que se ve en un alto nivel de ruido mayor como 30%. Este efecto de vecindades tiene una influencia muy directa sobre las reglas de transición a definir, lo que implica que no se puede esperar que un mismo conjunto de reglas pueda ser usado para filtrar el ruido sobre imágenes contaminadas con estos niveles diferentes de ruido. Más aún, se observa que el resultado de aplicar la transición del autómata sobre una imagen con un porcentaje alto de contaminación, puede representar la misma imagen pero esta vez con porcentajes menores de ruido.

La hipótesis 2 nos lleva a la definición de una metodología de reconstrucción que implica la transición sucesiva de varios autómatas desde aquellos que fueron configurados para los niveles mayores de ruido hasta aquellos que se definieron sobre los menores niveles de ruido.

Si se define una tabla de niveles de contaminación y por cada nivel se tiene un filtro representado con uno o más autómatas debidamente configurados, el proceso de reconstrucción está expresado en el Algoritmo 4.

$\varphi^c \leftarrow$  imagen contaminada dada  
 para todos los niveles de contaminación (mas alto al mas bajo) hacer  
 $AC \leftarrow$  filtro de nivel de contaminación actual  
 $\varphi^r \leftarrow$  **TransiciónAutómata**( $AC, \varphi^c$ )  
 $\varphi^c \leftarrow \varphi^r$   
 retornar imagen reconstruida  $\varphi^r$

**Algoritmo 4.**

Metodología de Reconstrucción para varios niveles de contaminación

Otro concepto que ayuda a acelerar el proceso de convergencia del programa evolutivo y reduce el número de reglas para los autómatas de alfabetos mayores que 2 para las imágenes en tonos de grises o a colores, es el de *Reglas Genéricas*. La idea es buscar un patrón de comportamiento o relación en la regla que permita asociar el valor de la celda en evaluación (conclusión de la regla) con el valor de las celdas vecinas (los  $v$  primeros elementos de la regla). Esta relación es luego usada como criterio válido de unificación para aceptar la regla en la transición del autómata. Esta opción se presenta como un parámetro de configuración del *Autómata Celular*. A continuación presentamos un ejemplo:

para la siguiente regla (vecindad  $v = 8$ ) de un conjunto  $T$  cualquiera:

$$*, 8, 8, 4, 4, *, 4, 4 \rightarrow 4$$

los siguientes vectores en evaluación unifican con ella:

$$*, 4, 4, 2, 2, *, 2, 2 \rightarrow 2$$

$$*, 10, 10, 5, 5, *, 5, 5 \rightarrow 5$$

Los valores discretos iguales a  $-1$  representan *Valores Genéricos*.

Un caso particular ocurre con las imágenes en tonos de gris. Para estas imágenes se tiene un alfabeto  $L = 256$  donde las 3-tupla de valores (*rojo*, *verde*, *azul*) cumplen la relación  $rojo = verde = azul$  para valores discretos entre 0 y 255. En la figura 5 se puede apreciar un bloque gris que a simple vista pareciera ser de la misma tonalidad pero realmente hay 8 bloques de igual tamaño con tonos grises diferentes que van de izquierda a derecha desde el nivel (128, 128, 128) hasta el nivel (135, 135, 135).



**Figura 5.**  
Ejemplo de tonos de gris

Con el fin de aprovechar esta característica particular de las imágenes en tonos de gris, hemos definido un parámetro llamado *Holgura de Similitud* que permite agrupar diferentes valores discretos del alfabeto en un valor equivalente al promedio del grupo. Esto obviamente reduce el alfabeto del autómata y por ende el número de reglas de transición.

La *Holgura de Similitud* es otro parámetro cuyo valor determina el ancho del grupo a considerar. Este concepto también es aplicable a las imágenes en color al tratar por separado cada componente (*rojo, verde, azul*) con un autómata particular.

A continuación un resumen sobre la configuración del programa evolutivo para la reconstrucción de imágenes.

Sobre los *Individuos*:

- Definen completamente un *Autómata Celular*.
- El reticulado es bidimensional con tantas celdas como pixeles tenga la imagen.
- La cardinalidad del alfabeto está asociada a la cantidad de colores que puede tener un pixel.
- Los tipos de vecindad son von Neumann, Moore y Mixta a 1 y 2 niveles.
- El conjunto de reglas de transición no puede ser vacío.
- No hay redundancia entre las reglas de un conjunto.
- Las reglas de un conjunto son consistentes entre sí.
- El conjunto no necesariamente debe ser completo.
- Toda regla del conjunto se evalúa de forma simétrica.
- Se permite el uso de *Valores Genéricos* de acuerdo a una probabilidad dada.
- Para la unificación de las reglas se pueden aplicar los conceptos de *Reglas Genéricas y Holgura de Similitud*.

Sobre las *Operaciones Genéticas*:

- Cruce simple del conjunto de reglas para *Individuos* con la misma *Vecindad*.
- Cruce al azar del conjunto de reglas para *Individuos* con la misma *Vecindad*.
- Agregación de una regla al conjunto *T* de un *Individuo*.
- Eliminación de una regla del conjunto *T* de un *Individuo*.
- Modificación de la conclusión de una de las reglas del conjunto *T* de un *Individuo*.

Sobre los parámetros del *Programa Evolutivo*:

- Mantener un porcentaje de la descendencia para la siguiente generación.
- Se reemplaza el peor *Individuo* de la *Población* por uno nuevo creado al azar.
- Algoritmo de rueda de la ruleta para selección de padres.
- Evaluación de *Individuos*: una transición del autómata.
- Funciones de calidad: diferencia de Hamming ó error cuadrático mas penalización por alteración de imágenes originales.

### **Medida de Calidad de Reconstrucción.**

Con miras de cuantificar en tiempo de ensayo, que tan buenas son las imágenes reconstruidas luego de pasar la secuencia de filtros, es importante definir y calcular

una medida de la calidad de las mismas. Lo más simple es cuantificar el porcentaje de la imagen reconstruida  $\delta(\varphi_i)$  que no difiere de la imagen original. Para ello basta contar el número de píxeles  $\Gamma$  que permanecen iguales entre las dos imágenes y obtener una proporción en función del número total de píxeles  $T$ . Para el cálculo es importante considerar el uso de la *Holgura de Similitud*  $\mu$  utilizada en la definición de los filtros. La expresión correspondiente es dada en (9).

$$\delta(\varphi_i) = \Gamma * 100 / T; \quad \Gamma = \sum_{i=1}^S \theta(|s_i^o - s_i^r|, \mu); \quad \theta(x, y) = \begin{cases} 1, & x \leq y \\ 0, & x > y \end{cases} \quad (9)$$



## **ENSAYOS Y RESULTADOS**

### **Introducción.**

Los experimentos que se detallan a continuación se han clasificado según el tipo de imágenes usadas como referencia:

1. Ensayos con imágenes blanco y negro.
2. Ensayos con imágenes en 256 tonos de grises.

En ambos casos se ha seguido una metodología de trabajo que contempla los siguientes pasos:

1. Seleccionar las imágenes de referencia para la definición de los filtros.
2. Escoger valores iniciales para los parámetros de ajuste que controlan la evolución del *Programa Evolutivo* y permiten definir los *Autómatas Celulares*.
3. Ejecutar el *Programa Evolutivo* utilizando cada una de las imágenes de referencia por separado para analizar los resultados obtenidos. Esto se hace para niveles de ruido de 10, 20 y 30 % de contaminación.
4. Ejecutar el *Programa Evolutivo* utilizando el conjunto completo de imágenes de referencia con niveles de ruido de 10, 20 y 30 % de contaminación de forma tal de obtener los filtros correspondientes.
5. Analizar el orden de aplicación de los filtros obtenidos de forma tal de definir la metodología de reconstrucción de las imágenes contaminadas.

6. Aplicar los filtros obtenidos a las imágenes originales sin contaminar para analizar el efecto causado.
7. Aplicar los filtros a otras imágenes diferentes de las de referencia, de forma tal de analizar el comportamiento genérico de los mismos.

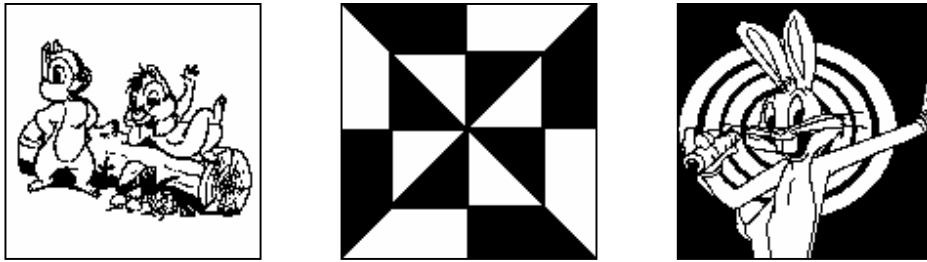
Con respecto a los parámetros de ajuste tanto para la definición de los *Autómatas Celulares* como para el control de ejecución del *Programa Evolutivo*, es conveniente mencionar los efectos que se tienen cuando se modifican sus valores:

1. La distribución de probabilidades de ocurrencia de *Vecindades* permite configurar el tipo de *Vecindad* que se desea predomine en las soluciones del *Programa Evolutivo*. Para las vecindades más simples, aquellas con menos elementos en sus vectores, se definen menos cantidades de reglas y esto trae como consecuencia mayor rapidez en la transición del *Autómata Celular*.
2. La probabilidad de uso de los *Valores Genéricos* en la definición de las reglas permite disminuir el número de reglas para cada autómata. Se recomienda usar valores que van desde 0.05 hasta 0.20.
3. El uso de *Reglas Genéricas* y *Holgura de Similitud* permiten manejar convenientemente alfabetos grandes de valores discretos. Reducen el número de reglas de transición pero tienen efectos dañinos sobre las imágenes al aplicar los *Autómatas Celulares*. Es recomendable su uso para el trato de imágenes en 256 tonos de grises.

4. El tamaño de la *Población* tiene repercusión directa sobre los tiempos de ejecución del *Programa Evolutivo*. Un número pequeño puede ser insuficiente para converger a una solución adecuada y requerir muchas iteraciones del algoritmo, mientras que un valor grande afecta el tiempo de ejecución de cada iteración del algoritmo evolutivo. Se recomienda valores entre 20 y 50 *Individuos*.
5. El porcentaje Gap permite que el proceso evolutivo mantenga soluciones iguales o mejores durante sus iteraciones. Cantidades grandes pueden provocar que la dinámica caiga en los no deseables mínimos locales. Es recomendable usar valores que van del 10 al 20 % del tamaño de la *Población*.
6. Las probabilidades de aplicación de las operaciones de cruce y mutación determinan la frecuencia de uso de las operaciones genéticas para obtener la descendencia durante cada iteración del proceso evolutivo. Se recomienda dar mayor importancia a los cruces entre padres ya que una de las hipótesis de la programación evolutiva tiene que ver con la obtención de descendientes producto de la mezcla de características de dos padres de la *Población* actual.

### **Ensayos con imágenes blanco y negro.**

Para estos ensayos se usaron las imágenes que se muestran en la figura 6. Se trata de imágenes en blanco y negro (con valores discretos iguales a 0 y 1 respectivamente) de 130 x 130 píxeles. La que está más a la izquierda o Imagen A tiene mayor predominio del color blanco en contraste con la Imagen C que es la que está más a la derecha y en la cual abunda el color negro, mientras que la imagen ubicada en el medio o Imagen B contiene igual proporción de blanco y negro.



**Figura 6.**  
Imágenes de referencia para ensayos con  $L = 2$

Para la asignación inicial de los *Individuos* de la *Población* se utilizó la siguiente distribución de probabilidades de ocurrencia de *Vecindades* (nótese que se da mayor importancia a las reglas más simples) y considerando una probabilidad de uso de *Valores Genéricos* igual a 0.15:

- $p_1 = 0.26$  (*von Neumann* nivel 1);
- $p_2 = 0.18$  (*von Neumann* nivel 2);
- $p_3 = 0.18$  (*Moore* nivel 1);
- $p_4 = 0.10$  (*Moore* nivel 2);
- $p_5 = 0.18$  (*Mixta* nivel 1);
- $p_6 = 0.10$  (*Mixta* nivel 2);

Para los parámetros del *programa evolutivo* fueron considerados los siguientes valores:

- Tamaño de la población: 20;
- Porcentaje Gap: 10 %;

- Probabilidad de aplicar operaciones de cruce: 0.85;
- Probabilidad de aplicar operaciones de mutación: 0.15.

Utilizando solamente la Imagen A como referencia y aplicando un nivel de contaminación de 10 %, el *Autómata Celular Evolutivo* resultante luego de 50 evoluciones está definido según la *Vecindad de Moore* nivel 1 con la siguiente regla:

$$0, *, 1, 0, 0, 0, 0, 0 \rightarrow 0$$

La figura 7 muestra la calidad de reconstrucción luego de aplicar la transición del autómata (97.57 % de similitud con la imagen original). En la parte izquierda se repite la imagen original, en la parte central aparece la imagen contaminada y la imagen reconstruida en la parte derecha. Nótese la tendencia a limpiar, según la conclusión de la regla, puntos dispersos o aislados que se identifican por estar rodeados de blancos o valores discretos iguales a 0.



**Figura 7.**  
Imagen A original, contaminada y reconstruida (97.57 %) luego de aplicar *Autómata Celular Evolutivo* (10 % ruido)

Al usar la imagen B, se llegó al siguiente conjunto de reglas, nuevamente según la *Vecindad de Moore* nivel 1:

$*, 1, 1, 1, 1, 1, 1, 1 \rightarrow 1$

$0, 0, 0, 1, 0, 1, 0, 0 \rightarrow 0$

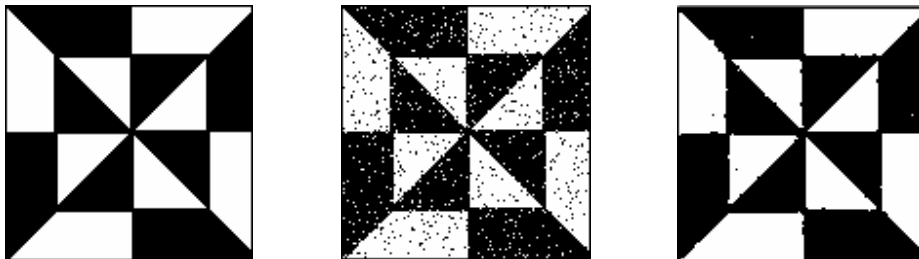
$1, 0, 0, 1, 0, 0, 1, 0 \rightarrow 0$

$0, *, 1, 0, 0, 0, 0, 0 \rightarrow 0$

$0, 0, 1, 0, 0, 0, 1, 0 \rightarrow 0$

$1, 1, 1, 1, 0, 0, 0, 1 \rightarrow 1$

$0, 1, 0, 1, 1, *, 1, 1 \rightarrow 1$



**Figura 8.**  
Imagen B original, contaminada y reconstruida (99.38 %) luego de aplicar *Autómata Celular Evolutivo* (10 % ruido)

En el conjunto de reglas se puede apreciar una parte para limpiar el ruido en los sectores blancos (reglas 2, 3, 4 y 5) y otra parte para limpiar el ruido en los sectores negros (reglas 1, 6 y 7). Nótese también que la regla 4 coincide con la única regla del conjunto del caso A lo que implica que hay congruencia en la solución a la cual

converge el *Autómata Celular Evolutivo*. La calidad de la reconstrucción se puede ver en la figura 8 (99.38 %).

Por último, usando solamente la Imagen C como referencia y repitiendo la tendencia de seleccionar la *Vecindad Moore* nivel 1 en la solución, se alcanzó el conjunto de reglas que se muestra a continuación. El aspecto más importante a resaltar es la regla 2 que aparece en la solución de los casos A y B anteriores. En la figura 9 se puede apreciar la calidad de la imagen reconstruida (97.73 %).

1, 1, 1, 0, 1, 1, 1, 1 → 1

0, \*, 1, 0, 0, 0, 0, 0 → 0



**Figura 9.** Imagen C original, contaminada y reconstruida (97.73 %) luego de aplicar *Autómata Celular Evolutivo* (10 % ruido)

Al usar las tres imágenes de referencia juntas y con un nivel de ruido de 10 %, se converge al siguiente conjunto de reglas, esta vez bajo el criterio de *Vecindad Von Neumann* nivel 2:

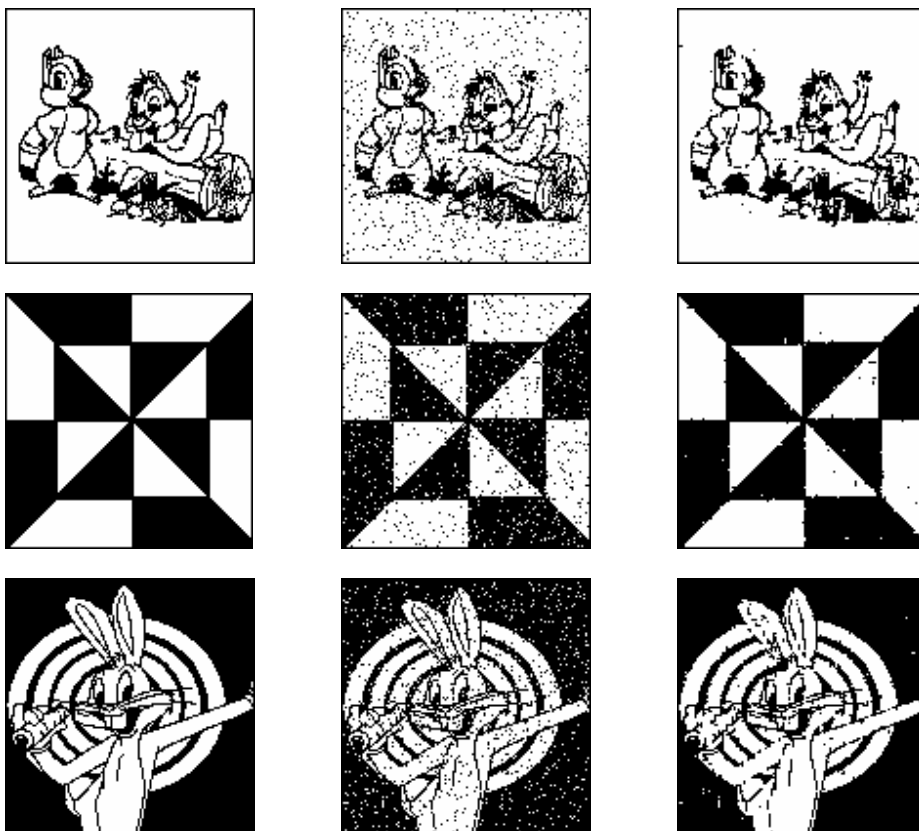
1, \*, 1, 1, 1, 1, 1, 0 → 1

0, 0, 0, 0, 1, 1, 1, 0 → 0

0, 0, 0, 0, \*, 0, 0, 0 → 0

0, 1, 0, 0, 0, 0, 0, 0 → 0

0, 1, 1, 1, 1, 1, 1, 1 → 1



**Figura 10.**

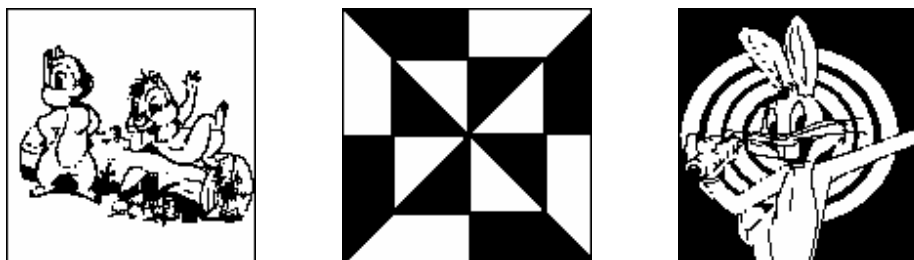
Imágenes A, B y C original, contaminada y reconstruida (97.43, 99.41 y 97.44 %) luego de aplicar *Autómata Celular Evolutivo* (10 % ruido)

Aquí es donde tratamos con la mayor dificultad para el programa evolutivo:  
converger a una solución que busque buenas reconstrucciones de todas las imágenes,



sin perjudicar a una de ellas al tratar de favorecer a otra. Esta es la explicación de por qué el cambio en el tipo de vecindad. La figura 10 muestra la calidad de las tres imágenes reconstruidas con un nivel de contaminación del 10 % (97.43, 99.41 y 97.44 % respectivamente). Nótese que las medidas de calidad no difieren en gran cantidad sobre los resultados que se obtienen al trabajar con las imágenes de forma individual.

La figura 11 presenta los tres resultados luego de propagar el *Autómata Celular* sobre las mismas imágenes originales sin contaminar. Las medidas de calidad fueron 98.86, 99.98 y 98.96 % respectivamente. De esta forma podemos observar el comportamiento de la función de calidad aplicada por *el Programa Evolutivo* para tratar de no alterar las imágenes originales.

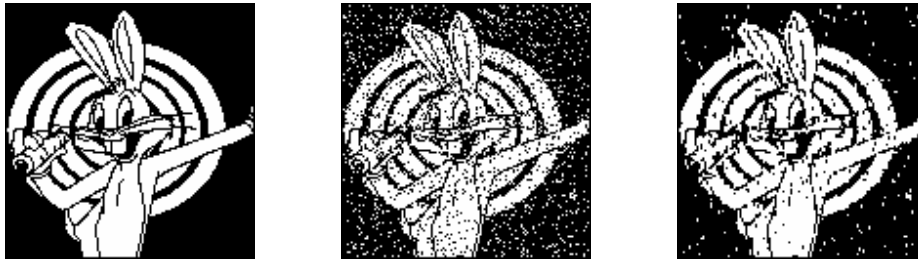


**Figura 11.**  
Imágenes A, B y C reconstruidas (98.86, 99.98 y 98.96 %) luego de aplicar *Autómata Celular* (0 % ruido)

Al aumentar el nivel de ruido a 20 % y usando las tres imágenes de referencia juntas, el conjunto de reglas, basado en la *Vecindad Von Neuman* nivel 1, es el siguiente:

0, 0, 0, 0 → 0

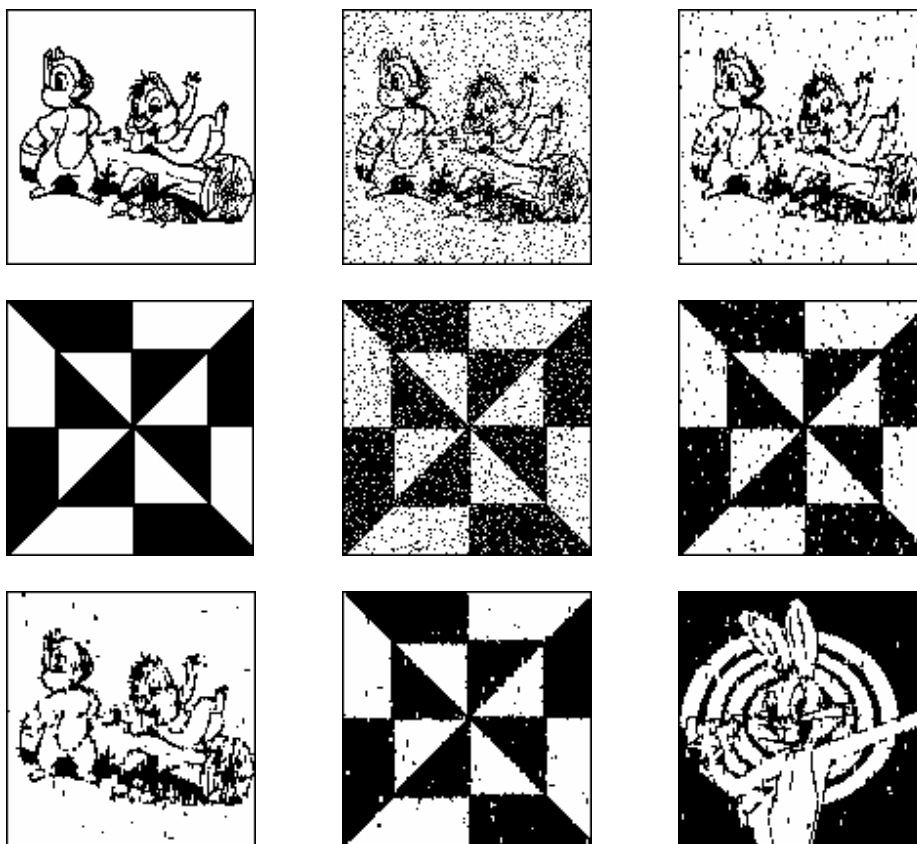
1, 1, 1, 1, → 1



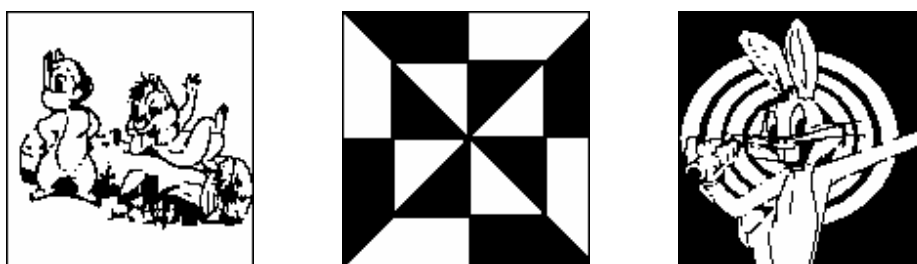
**Figura 12.** Imágenes A, B y C original, contaminada y reconstruida (94.45, 96.71 y 94.27 %) luego de aplicar *Autómata Celular Evolutivo* (20 % ruido)

En este caso la tendencia es a limpiar puntos negros sobre espacios blancos (regla 1) y puntos blancos sobre espacios negros (regla 2). La figura 12 presenta las imágenes de la solución (94.45, 96.71 y 94.27 %). En esta situación podemos aplicar las dos hipótesis mencionadas en este documento cuando se describió en detalle el *Autómata Celular Evolutivo*. Nótese que las imágenes reconstruidas en la figura 12 tienen cierta semejanza con las imágenes contaminadas de la Figura 10.

Al aplicar el filtro de 10 % de ruido a las reconstrucciones obtenidas sobre 20 % es razonable esperar una mejora en la calidad de la solución obtenida. La figura 13 muestra estos resultados, con medidas de calidad o porcentajes de similitud con imágenes originales respectivas de 95.44, 98.58 y 95.18 %, las cuales permiten observar una ligera mejora.



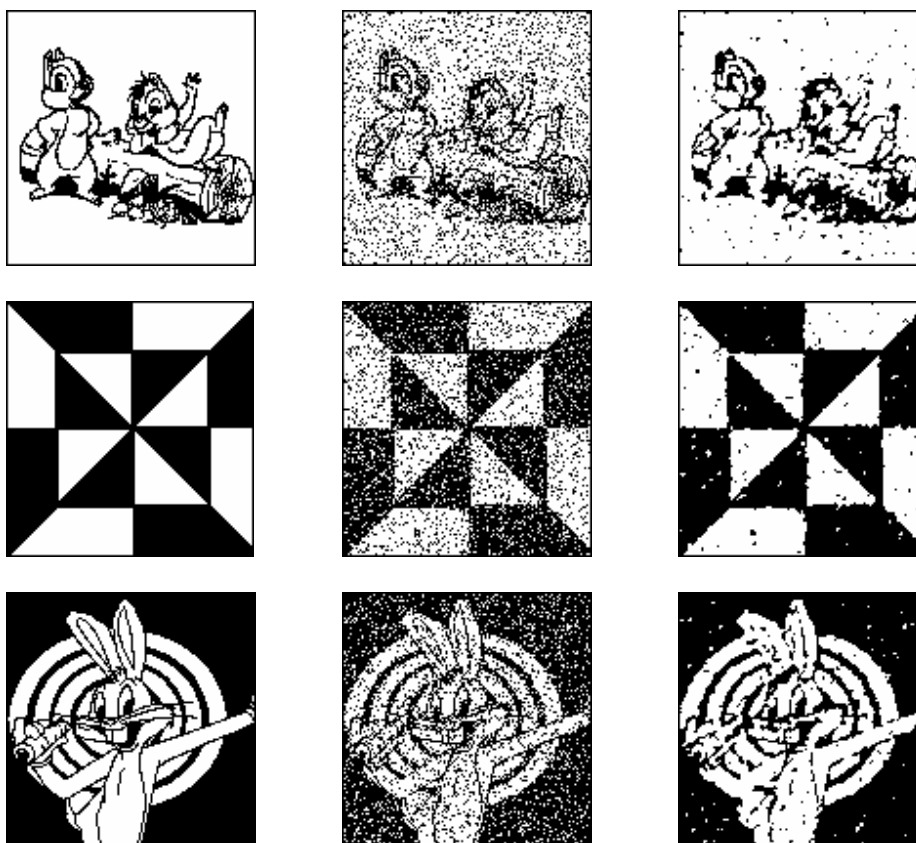
**Figura 13.**  
Imágenes A, B y C reconstruidas (20 % ruido) (95.44, 98.58 y 95.18 %) luego de aplicar Filtros de 20 y 10 % de ruido



**Figura 14.**  
Imágenes A, B y C reconstruidas (0 % ruido) (98.46, 99.98 y 98.47 %) luego de aplicar Filtros de 20 y 10 % de ruido

En la figura 14 se pueden ver las imágenes reconstruidas luego de aplicar en serie los dos filtros (20 y 10 %) sobre las imágenes originales (98.46, 99.98 y 98.47 %).

Nótese que las medidas de calidad presentan una diferencia casi nula con respecto a las obtenidas cuando se pasó sólo el filtro de 20 %.



**Figura 15.** Imágenes A, B y C original, contaminada y reconstruida (92.41, 97.25 y 92.03 %) luego de aplicar *Autómata Celular Evolutivo* (30 % ruido)

Haciendo ahora un aumento de 30 % de contaminación, la evolución llegó al siguiente conjunto de reglas (*Moore* nivel 1). Es interesante notar que para los tres niveles de ruido (10, 20 y 30 %) se obtuvieron filtros con criterios diferentes de *Vecindad*.

1, 1, \*, 1, 1, 1, 0, 1 → 1

1, 0, 1, 1, 1, 1, 0, 1 → 1

0, 0, 0, 1, 0, 1, 0, 0 → 0

0, 0, 0, \*, 0, 0, 1, 0 → 0

0, 0, 1, 1, 1, 1, 1, 1 → 1

1, 1, \*, 0, 0, 0, 0, 0 → 0

0, 1, 1, 1, 1, 1, 0, 1 → 1

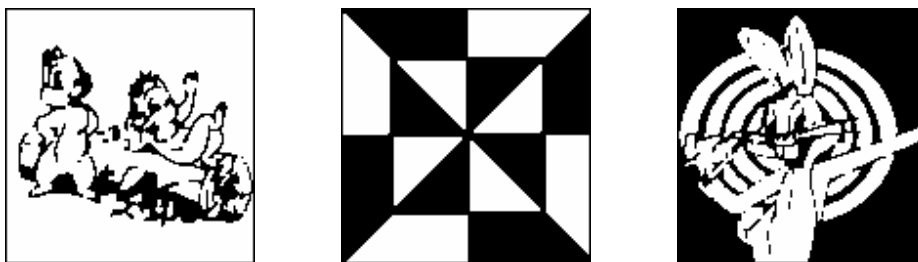
La figura 15 presenta las imágenes a las que se llegó con el ensayo anterior obteniendo medidas de calidad de 92.41, 97.25 y 92.03 % respectivamente. En la figura 16 es posible observar las imágenes reconstruidas luego de aplicar en secuencia y en orden los filtros de 30, 20 y 10 % de ruido (92.57, 98.25 y 92.30 %) y la figura 17 muestra las imágenes reconstruidas luego de aplicar las imágenes originales sin contaminación a estos tres filtros (95.95, 99.92 y 96,38 %).



**Figura 16.**

Imágenes A, B y C reconstruidas (30 % ruido) (92.57, 98.25 y 92.30 %) luego de aplicar Filtros de 30, 20 y 10 % de ruido

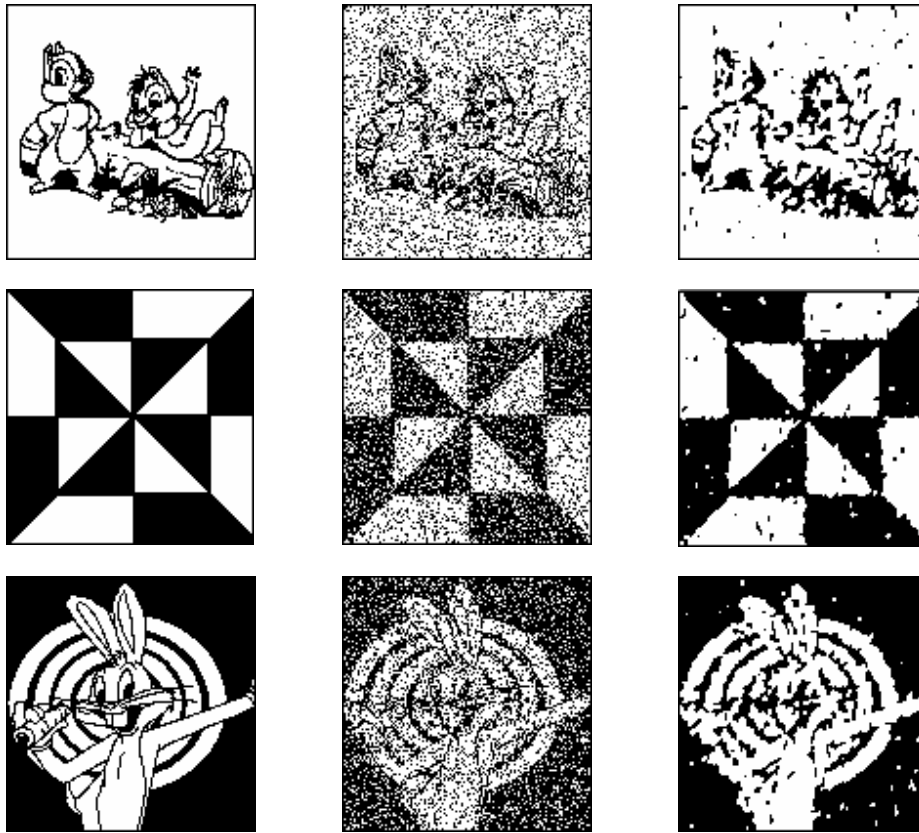
Usando la misma metodología de reconstrucción de aplicar en orden y consecutivamente los tres filtros de 30, 20 y 10 % obtenidos con las tres imágenes de referencia, se realizó el ensayo sobre las imágenes A, B y C con 40 % de contaminación. La figura 18 presenta la solución gráfica obtenida con medidas de 91.36, 96.56 y 91.05 % respectivamente.



**Figura 17.**  
Imágenes A, B y C reconstruidas (0 % ruido) (95.95, 99.92 y 96,38 %) luego de aplicar Filtros de 30, 20 y 10 % de ruido

Cabe destacar que a medida que se van aplicando los filtros se observa un aumento en la medida de calidad. Nótese la fuerte tendencia a eliminar la granularidad que la contaminación produce en las imágenes y el tratar de llegar en las imágenes reconstruidas, a las formas o elementos principales que definen las imágenes originales.

La tabla 2 presenta los resultados obtenidos en cuanto a los porcentajes de similitud de las imágenes de referencia, luego de ser contaminadas con los diferentes niveles de ruido y pasar los filtros de 30, 20 y 10 % en forma secuencial.



**Figura 18.**

Imágenes A, B y C original, contaminada (40 %) y reconstruida (91.36, 96.56 y 91.05 %) luego de aplicar Filtros de 30, 20 y 10 % de ruido

Imagen	0 %	10 %	20 %	30 %	40 %
A	95.95	95.21	93.95	92.57	91.36
B	99.92	99.46	99.04	98.25	96.56
C	96.38	95.09	94.33	92.30	91.05

**Tabla 2.**

Medidas de calidad luego de aplicar Filtros de 30, 20 y 10 % de ruido

Hay que notar que para los niveles bajos de contaminación (10 % por ejemplo), la aplicación de filtros para niveles mayores de ruido (20 y 30 %) perjudica la calidad de reconstrucción (basta comparar los datos de la tabla 2 con aquellos presentados en la figura 10). Bajo este criterio es razonable pensar lo que ocurre si en lugar de aplicar

los filtros de 30, 20 y 10 % en ese orden, se aplica repetidamente el filtro para 10 % de contaminación, por ejemplo. Se realizó un ensayo con las tres imágenes contaminadas a varios niveles de ruido aplicando de forma consecutiva tres veces el filtro de 10 %. Se obtuvieron los resultados presentados en la tabla 3.

<b>Imagen</b>	<b>0 %</b>	<b>10 %</b>	<b>20 %</b>	<b>30 %</b>	<b>40 %</b>
A	98.41	97.15	95.65	93.59	89.64
B	99.98	99.43	98.62	97.25	93.69
C	98.49	97.46	95.22	92.93	89.76

**Tabla 3.**  
Medidas de calidad luego de aplicar Filtro de 10 % de ruido 3 veces consecutivas

Comparando los resultados de las tablas 2 y 3, se observa que este nuevo criterio de aplicar los filtros daña menos a las imágenes originales y logra mejor calidad en niveles bajos de contaminación. Se realizó el mismo experimento pero esta vez con el filtro de 20 %. Los resultados están en la tabla 4.

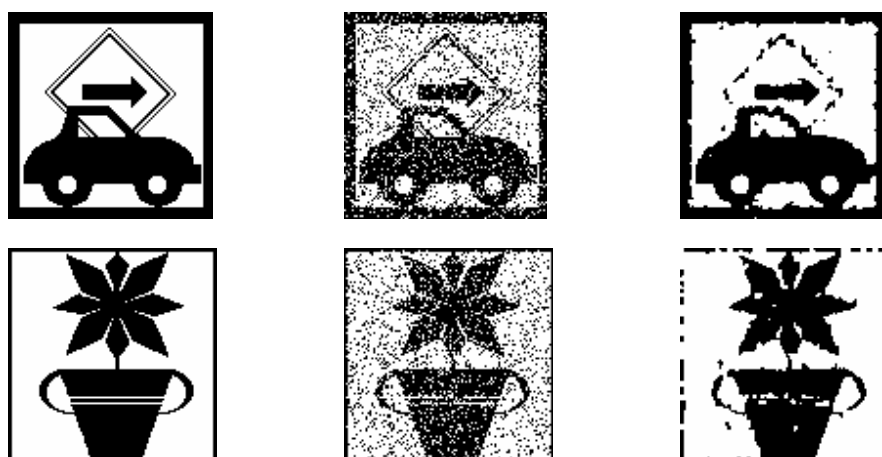
<b>Imagen</b>	<b>0 %</b>	<b>10 %</b>	<b>20 %</b>	<b>30 %</b>	<b>40 %</b>
A	98.86	97.33	94.66	91.36	87.09
B	99.98	99.14	97.31	94.49	90.89
C	98.82	97.02	94.67	91.33	87.09

**Tabla 4.**  
Medidas de calidad luego de aplicar Filtro de 20 % de ruido 3 veces consecutivas

En este caso se aprecia una leve mejora en las imágenes sin contaminar pero peores resultados con el aumento del nivel de ruido.



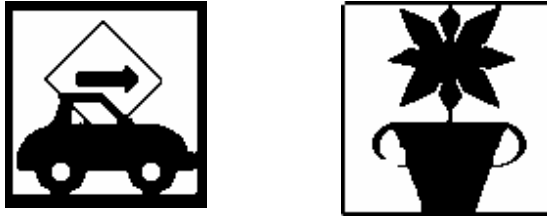
Por otro lado, es importante evaluar el *Autómata Celular* sobre imágenes diferentes a las de referencia utilizadas en el proceso de definición. Esto permite medir la capacidad del modelo frente a cualquier imagen. Bajo este sentido se realizaron los ensayos que se muestran en las figuras siguientes. La figura 19 presenta dos imágenes de 110 x 110 píxeles que fueron contaminadas con 30 % de ruido y al que se le aplicó la metodología de reconstrucción antes descrita (92.69 y 92.86 %). La figura 20 muestra los resultados luego de aplicar los filtros a las imágenes sin contaminar (95.17 y 94.74 %).



**Figura 19.**

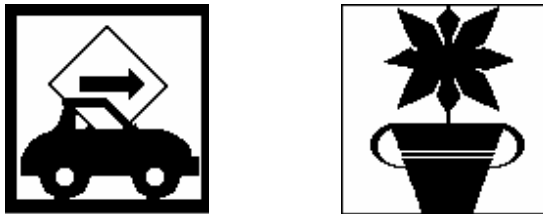
Imágenes 110 x 100 píxeles original, contaminada (30 %) y reconstruida (92.69 y 92.86 %) luego de aplicar Filtros de 30, 20 y 10 % de ruido

Cuando a estas imágenes con 30 % de ruido se les aplicó tres veces el filtro de 10 % se obtuvieron las medidas 93.92 y 95.19 % respectivamente. Con las imágenes sin contaminar se obtuvieron las medidas 99.07 y 99.72 % respectivamente. Como se puede observar, con este experimento las medidas son mucho mejores, sobre todo en lo que respecta a las imágenes originales (figura 21).



**Figura 20.**

Imágenes 110 x 100 píxeles (sin contaminar) reconstruidas (95.17 y 94.74 %) luego de aplicar Filtros de 30, 20 y 10 % de ruido



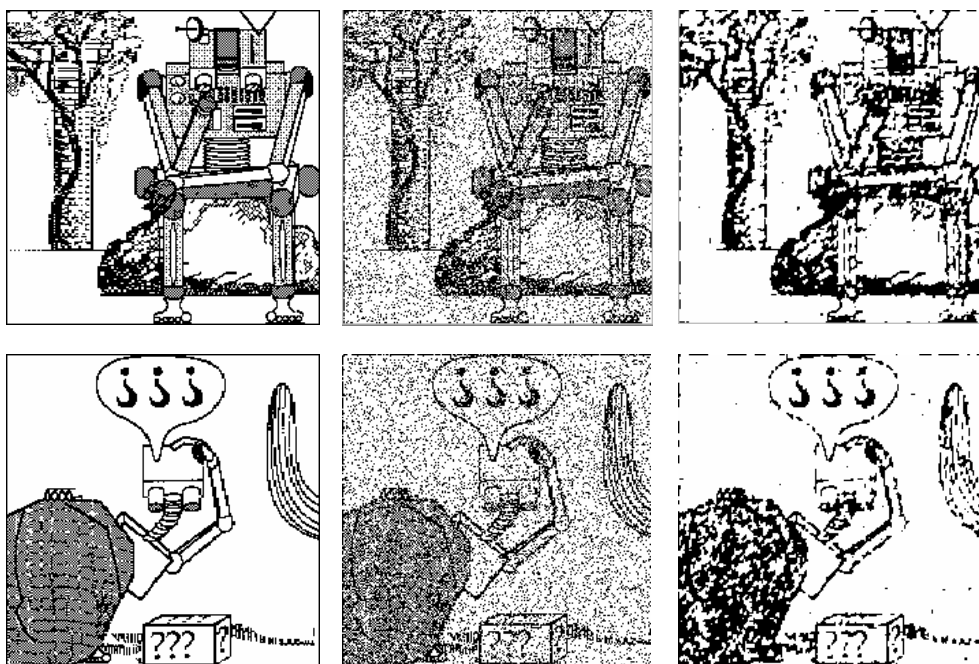
**Figura 21.**

Imágenes 110 x 100 píxeles (sin contaminar) reconstruidas (99.07 y 99.72 %) luego de aplicar el Filtro de 10 % de ruido 3 veces

La figura 22 muestra otros dos casos de 300 x 300 píxeles. Nótese que en estas imágenes, sobre todo la primera, hay mucha granularidad o puntos aislados, lo que empeora la calidad de reconstrucción (90.47 y 91.94 %). Esto se puede comprobar mejor en la figura 23 al ver el daño que los filtros le hacen a las imágenes originales sin contaminar (93.50 y 94.29 %).

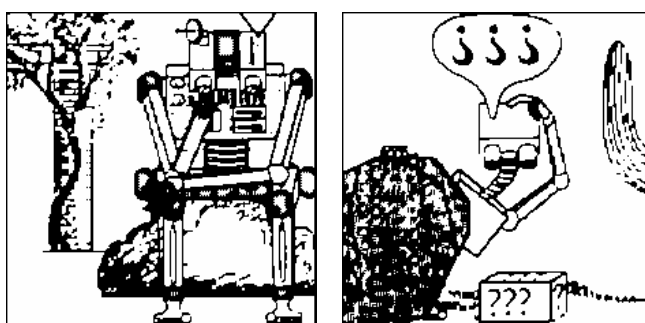
Una explicación razonable de este hecho es que las imágenes de referencia usadas durante el proceso de definición del *Autómata Celular* no presentan esta característica de granularidad. Una de las ventajas del uso de programación evolutiva es la

capacidad de adaptación sobre problemas típicos como este. Basta con realizar nuevas definiciones de *Autómatas Celulares* para tratar con este tipo de imágenes.



**Figura 22.**

Imágenes 300 x 300 píxeles original, contaminada (30 %) y reconstruida (90.47 y 91.94 %) luego de aplicar Filtros de 30, 20 y 10 % de ruido



**Figura 23.**

Imágenes 300 x 300 píxeles (sin contaminar) reconstruidas (93.50 y 94.29 %) luego de aplicar Filtros de 30, 20 y 10 % de ruido

Haciendo el experimento con el filtro de 10 % se obtuvieron las medidas de 91.84 y 92.92 % para la contaminación de 30 % y 96.78 y 96.92 % para las imágenes sin contaminar. Nuevamente con mejoras sobre la aplicación de los tres filtros diferentes.

Analizando todos estos resultados debemos concluir que la selección y orden de aplicación de los filtros es otro factor a tomar en cuenta para definir una adecuada metodología de reconstrucción haciendo uso de *Autómatas Celulares Evolutivos*, por lo menos en lo que a imágenes blanco y negro se refiere.

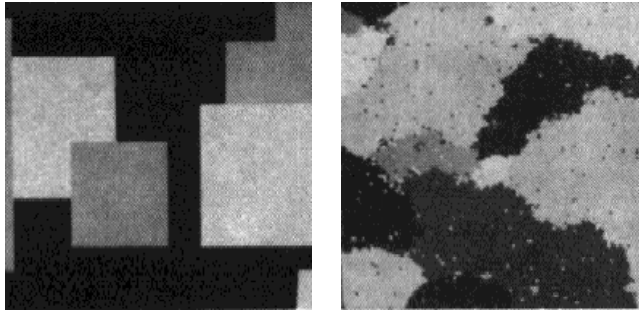
### **Ensayos con imágenes en 256 tonos de grises.**

Este tipo de problema es mucho más complicado que el caso blanco y negro. En primer lugar se aumenta el alfabeto del autómata a 256 valores discretos. Por otro lado se tiene el problema de la calidad visual de las imágenes originales que se han de reconstruir, considerando parámetros como brillo, sombra, luz, etc.

Para estos ensayos se usaron las imágenes que se muestran en la figura 24 [3]. Se trata de imágenes de tamaño 210 x 210 píxeles. La que está a la izquierda (Imagen A) tiene predominio de figuras geométricamente regulares y la de la derecha (Imagen B) se basa principalmente en manchas o figuras irregulares.

Para la asignación inicial de los *Individuos* de la *Población* se utilizó la misma distribución de probabilidades de ocurrencia de *Vecindades* que el caso anterior y considerando también una probabilidad de uso de *Valores Genéricos* igual a 0.15.

Además se hace uso del concepto de *Reglas Genéricas* y dada la naturaleza de las imágenes se aplica una holgura de similitud de 15 niveles discretos.



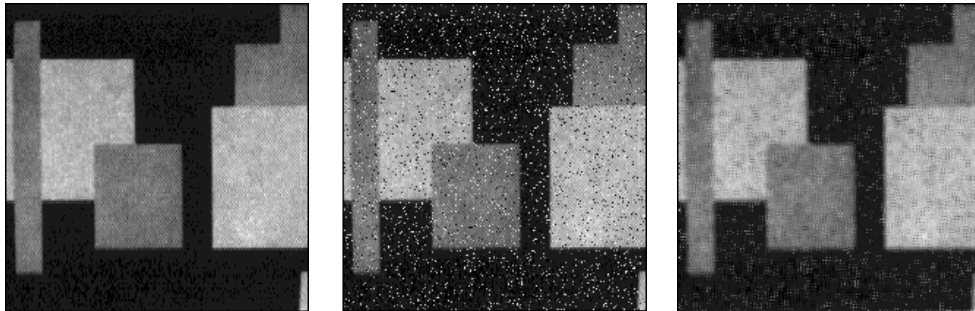
**Figura 24.**  
Imágenes de referencia para ensayos con  $L = 256$

Para los parámetros del *Programa Evolutivo* fueron considerados los siguientes valores:

- Tamaño de la población: 20;
- Porcentaje Gap: 10 %;
- Probabilidad de aplicar operaciones de cruce: 0.85;
- Probabilidad de aplicar operaciones de mutación: 0.15.

En vista del uso de los conceptos de *Reglas Genéricas* y *Holgura de Similitud* para la definición de las reglas de transición del autómata, en los resultados que aparecen a continuación no se presenta el conjunto de reglas obtenidos debido a lo difícil de su interpretación.

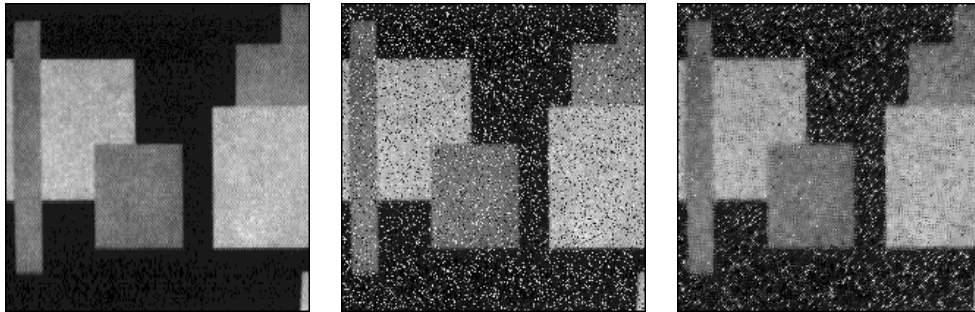
Como primer ensayo usamos sólo la imagen A como referencia y una contaminación de ruido blanco de 10 %. En la figura 25 se presenta, de izquierda a derecha, la imagen original, contaminada y reconstruida de este caso. La medida de calidad obtenida fue de 63.85 %.



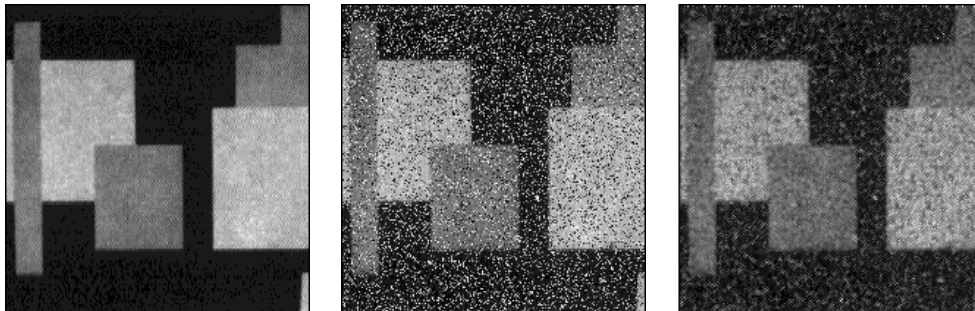
**Figura 25.**  
Imagen A original, contaminada y reconstruida (63.85 %) luego de aplicar *Autómata Celular Evolutivo* (10 % ruido)

Las figuras 26 y 27 muestran los resultados de los ensayos con la misma imagen pero esta vez con 20 y 30 % de contaminación. Para estos casos las medidas de calidad obtenidas fueron de 56.58 y 42.04 % respectivamente. En los tres ensayos se puede apreciar una ligera tendencia a reducir la granularidad observada en las imágenes contaminadas. Se puede ver también en la imagen reconstruida la pérdida de brillo en comparación con la imagen original.

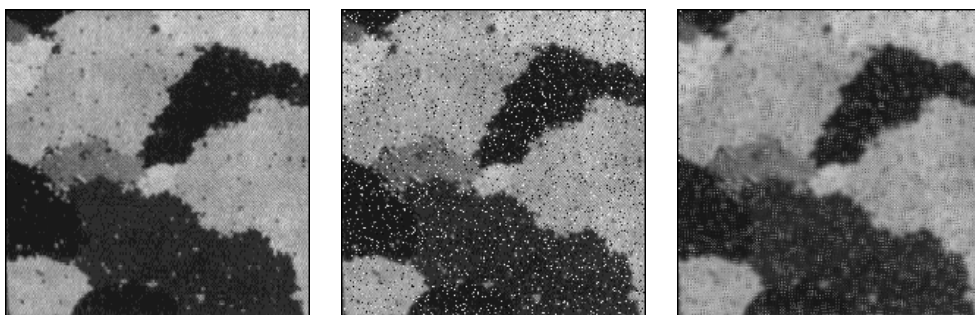
Las siguientes tres figuras, 28, 29 y 30, presentan el ensayo con la imagen B para 10, 20 y 30 % de ruido respectivamente. Se mantiene la tendencia a limpiar granularidad y la pérdida de brillo. Las medidas de calidad correspondientes son 63.91, 57.65 y 36.49 %.



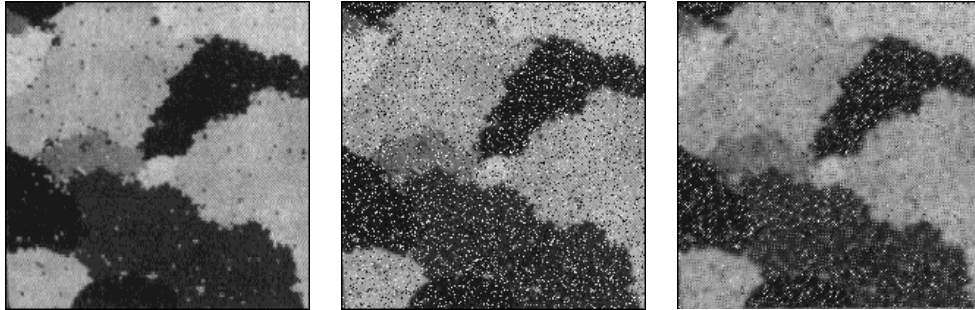
**Figura 26.**  
 Imagen A original, contaminada y reconstruida (56.58) luego de aplicar *Autómata Celular Evolutivo* (20 % ruido)



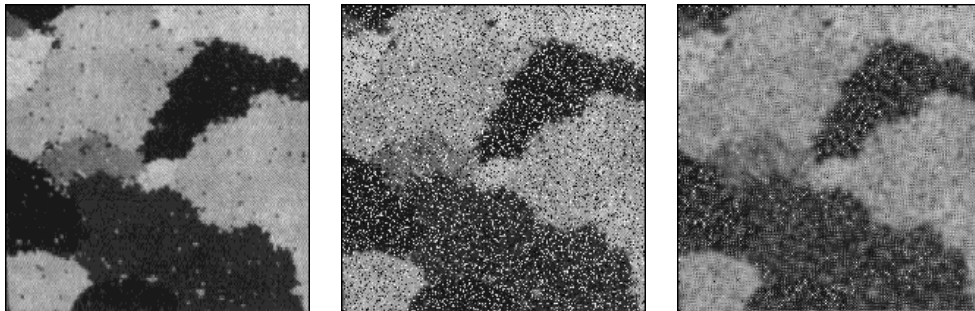
**Figura 27.**  
 Imagen A original, contaminada y reconstruida (42.04) luego de aplicar *Autómata Celular Evolutivo* (30 % ruido)



**Figura 28.**  
 Imagen B original, contaminada y reconstruida (63.91 %) luego de aplicar *Autómata Celular Evolutivo* (10 % ruido)



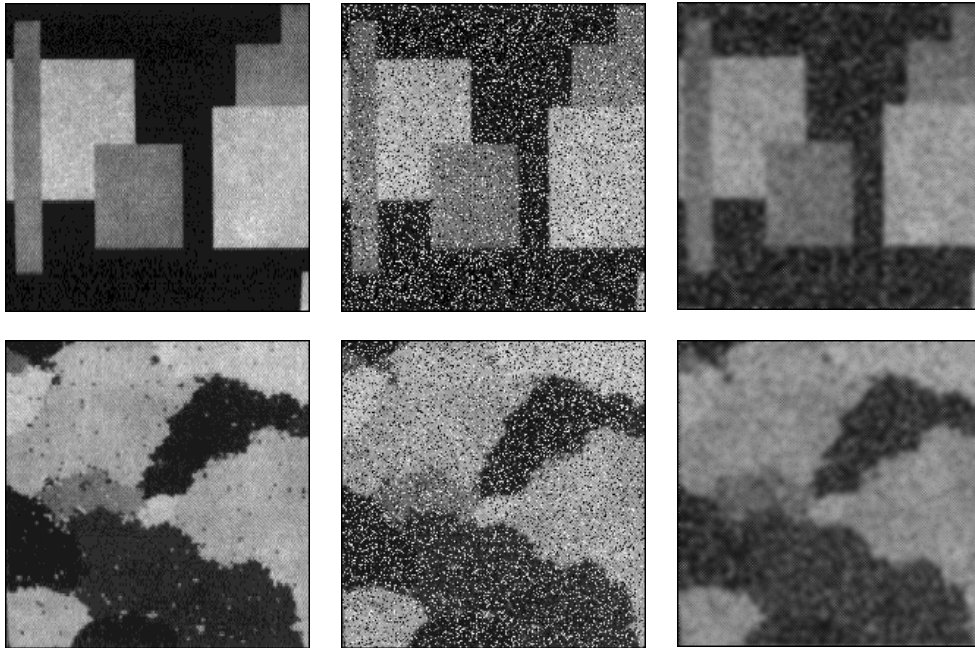
**Figura 29.**  
Imagen B original, contaminada y reconstruida (57.65 %) luego de aplicar *Autómata Celular Evolutivo* (20 % ruido)



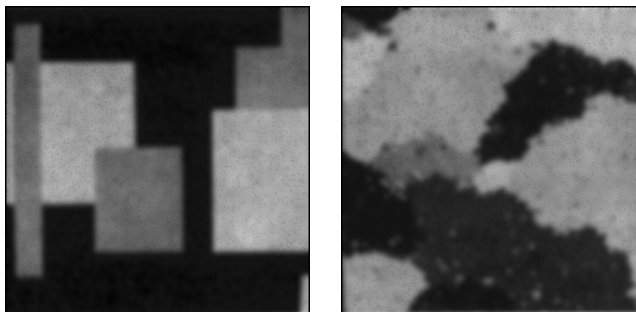
**Figura 30.**  
Imagen B original, contaminada y reconstruida (36.49 %) luego de aplicar *Autómata Celular Evolutivo* (30 % ruido)

Al usar ambas imágenes como referencia para la definición de los filtros se alcanzan soluciones muy similares a los casos individuales. Con una contaminación de 30 % de ruido y luego de aplicar en orden los filtros obtenidos en los casos de 30, 20 y 10 % se llegó a las soluciones mostradas en la figura 31 con medidas de calidad de 33.02 y 34.21 % respectivamente. La figura 32 presenta las imágenes originales sin contaminar luego de pasar por los filtros. En este caso la calidad obtenida es de 64.48 y 59.10 % respectivamente.





**Figura 31.**  
Imágenes A y B original, contaminada (30 %) y reconstruida (33.02 y 34.21 %) luego de aplicar Filtros de 30, 20 y 10 % de ruido



**Figura 32.**  
Imágenes A y B (sin contaminar) reconstruidas (64.48 y 59.10 %) luego de aplicar Filtros de 30, 20 y 10 % de ruido

Al aplicar tres veces el filtro de 10 % se obtienen las medidas 43.90 y 41.71 para un 30 % de contaminación y 67.16 y 65.10 % de similitud para las imágenes sin contaminar. En estos valores se observan ligeras mejoras con relación al otro caso.

A fin de realizar análisis comparativos con otro tipo de filtros, la figura 33 muestra ensayos hechos por Yi Ma y John Koode de la Universidad de Berkeley al usar un filtro gaussiano sobre la imagen de Lena<sup>7</sup>. De izquierda a derecha y de arriba hacia abajo se observa la imagen original de 205 x 205 pixeles, la imagen contaminada al 30 %, la imagen reconstruida luego de aplicar el filtro (sigma = 1) y la imagen reconstruida luego de aplicar el filtro a la imagen original sin contaminar.



**Figura 33.**  
Ensayo con filtro Gaussiano (sigma = 1) sobre Lena

La figura 34 presenta los resultados luego de aplicar el filtro de 30 % con el *Autómata Celular Evolutivo* sobre la misma imagen contaminada de Lena. En la izquierda se encuentra la imagen reconstruida con una medida de calidad de 46.67 % y en la

---

<sup>7</sup> Imagen de referencia usada por muchos investigadores en el área de Restauración de Imágenes.

derecha está la imagen obtenida luego de aplicar el filtro a la imagen original sin contaminar (82.33 % de calidad). Luego de varios ensayos se observó que el mejor resultado se alcanza al aplicar sólo el filtro de 30 %.



**Figura 34.**  
Ensayo con filtro 30 % sobre Lena

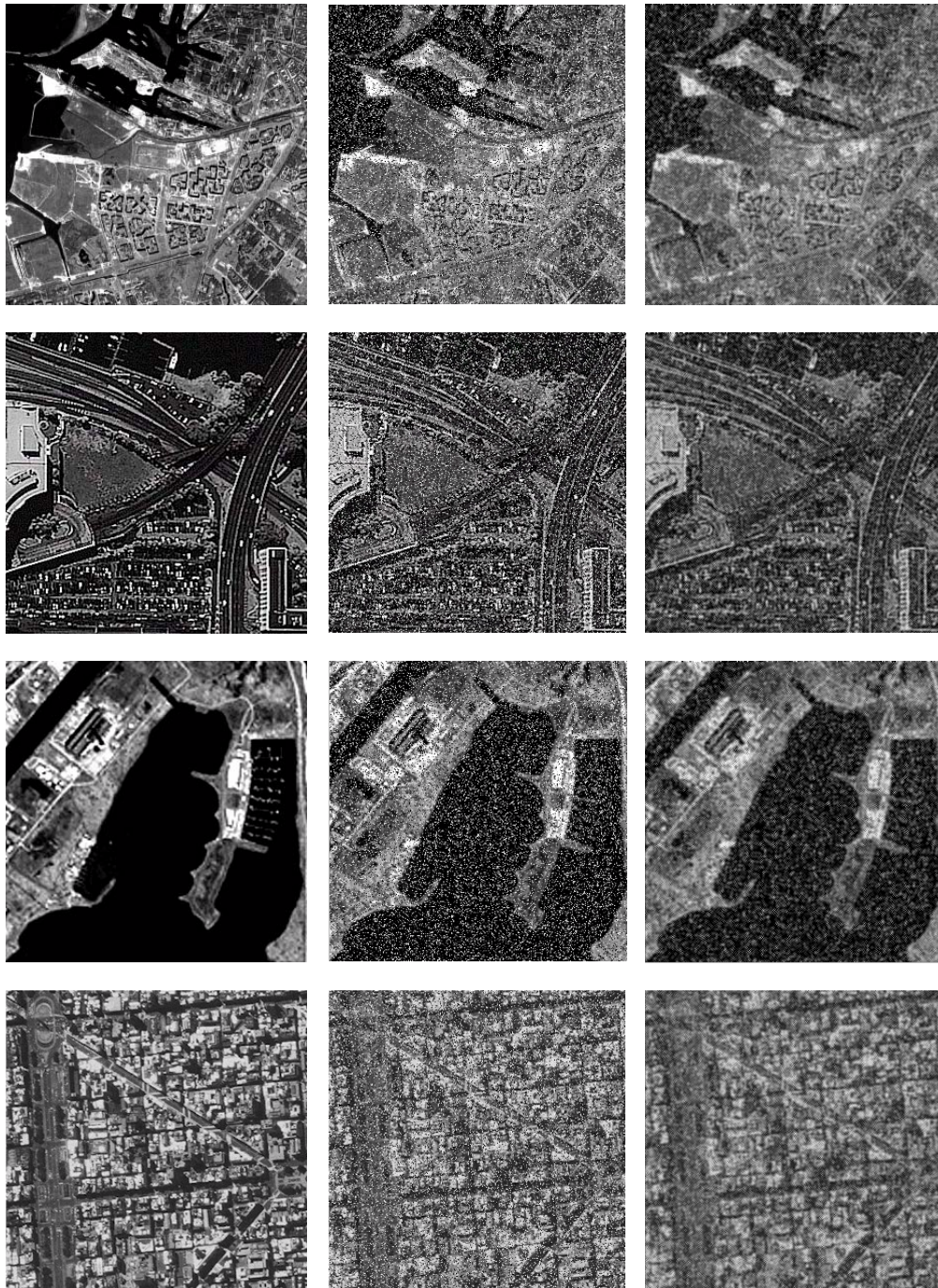
Si bien la comparación visual de ambos resultados está a favor del filtro gaussiano, no hay que dejar pasar por alto las características que por definición tienen los *Autómatas Celulares*, en particular la de poseer una dinámica sincrónica de procesamiento paralelo que los convierte en dispositivos rápidos de cómputo y su facilidad de ser implementados en hardware.

La mayoría de los métodos de restauración existentes son algoritmos complejos, en muchos casos muy lentos de ejecutar, difíciles de llevar a una solución vía hardware. La metodología de reconstrucción basada en *Autómatas Celulares Evolutivos* representa una alternativa que se perfila adecuada para buscar soluciones rápidas de restauración por hardware.

Los conceptos de *Reglas Genéricas* y *Holgura de Similitud* facilitan la definición de las *Reglas de Transición* de los autómatas, el precio a pagar está en la calidad de la imagen reconstruida, en particular en los criterios relacionados con el brillo y la luz, no así en lo que se refiere a la identificación de las formas y objetos de la imagen. Los filtros definidos con *Autómatas Celulares* tal como se plantean en este trabajo, pueden no ser buenas alternativas con imágenes relativas a fotos como la de Lena, pero es atractivo ensayar con casos donde lo importante puede ser el reconocimiento de formas, como por ejemplo imágenes derivadas de satélites.

La figura 35 presenta 4 ensayos con fotos tomadas de un satélite con las imágenes original, contaminada y reconstruida, vistas de izquierda a derecha para cada uno de los casos. En el orden de arriba hacia abajo se encuentra una toma de Sant Petersburgo con una calidad obtenida de 37.74 %. Luego se observa una toma de la ciudad de Washington con calidad de 26.08 %. En tercer lugar se tiene una foto de la ciudad de Toronto y una calidad de 41.30 % y por último una toma de Buenos Aires con una calidad de 38.76 %.

Estas imágenes son de 300 x 300 píxeles y fueron contaminadas con un factor de 30 % de ruido. Para los ensayos se aplicó el filtro de *Autómata Celular* obtenido para 30 %. Nuevamente se observa la misma tendencia a eliminar granularidad con pérdida de brillo.



**Figura 35.**  
Ensayo con imágenes de satélites con 30 % de ruido

### **Comentarios finales.**

Los resultados aquí presentados representan experimentos preliminares que motivan el continuar con esta línea de investigación. Hay que recordar que se está trabajando con una técnica de resolución de problemas emergente y por ende hay toda una serie de parámetros y consideraciones que se pueden tomar en cuenta para mejorar o adaptar la metodología a casos específicos. En particular, es posible hacer ajustes sobre los siguientes puntos:

- Parámetros propios del *Programa Evolutivo*.
- Criterios de convergencia o número de iteraciones del *Programa Evolutivo*.
- Parámetros de definición de las reglas de transición del *Autómata Celular*.
- Diferentes niveles de contaminación.
- Imágenes de referencia para la definición de los autómatas.
- Selección y orden de aplicación de los filtros.

Por otro lado, hay toda una serie de investigaciones abiertas que se pueden derivar de este trabajo:

- Nuevos funcionales para medir la calidad de reconstrucción.
- Manejo de otro tipo de imágenes (en colores por ejemplo).
- Otros criterios para definir reglas de transición de autómatas.
- Nuevos operadores genéticos adaptados a este problema.

## **CONCLUSIONES**

Se ha definido un paradigma basado en *Programación Evolutiva* capaz de hallar la dinámica de transición de un *Autómata Celular* conociendo la información inicial y los resultados deseados. Se ha dado el nombre de *Autómata Celular Evolutivo* a esta metodología de resolución de problemas.

Se ha probado esta metodología en el problema de *Reconstrucción o Restauración de Imágenes*. Para ello se han definido criterios como el de *Valor Genérico*, *Holgura de Similitud* y *Reglas Genéricas*, para ayudar a los *Autómatas Celulares* a lograr soluciones adecuadas de este problema.

La metodología de reconstrucción definida no es cerrada. Los filtros obtenidos y presentados en este trabajo no son definitivos. La posibilidad de seleccionar las imágenes de referencia y contar con varios parámetros de ajuste sobre la definición de los autómatas y sobre el programa evolutivo o simplemente ejecutar más iteraciones del algoritmo evolutivo, conllevan a la creciente mejoría de los resultados.

Los resultados preliminares obtenidos en este trabajo han mostrado que la metodología de reconstrucción se comporta bien en el trato con

imágenes en blanco y negro, pero se requiere de mayor fineza en el ajuste de parámetros para el caso de imágenes en tonos de grises.

La idea de definir una técnica híbrida que fusione las bondades de la *Programación Evolutiva* y los *Autómatas Celulares* ha servido para solventar con una técnica la deficiencia de la otra. La *Programación Evolutiva* resuelve el difícil problema de definir el conjunto de reglas de transición del autómata y los *Autómatas Celulares* ofrecen su facilidad de implementación por software y hardware y su característica de procesamiento paralelo que los convierte en dispositivos rápidos de resolución de problemas.

La definición y formalización de nuevos funcionales para medir la calidad de reconstrucción, el manejo de otro tipo de imágenes, el estudio de otros criterios para definir reglas de transición de autómatas y la obtención de otros operadores genéticos adaptados a este problema, son algunos temas abiertos de investigación relativos a esta metodología de reconstrucción, que motivan el seguir trabajando con esta temática.



## REFERENCIAS BIBLIOGRÁFICAS

- [1] Davis, Lawrence; *Handbook of Genetic Algorithms*; Van Nostrand Reinhold, 1991.
- [2] Devaney, R. L.; *An Introduction to Chaotic Dynamical Systems*; 1989.
- [3] Geman, Stuart and Geman, Donald; *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*; IEEE Transactions on Pattern Analysis and Machine Intelligence; PAMI-6: 721 - 741.
- [4] Lefe, Olu; *Data Compression and Encryption Using Cellular Automata Transforms*; Artificial Intelligence Applications; Vol. 10, No. 6, 1998.
- [5] Michalewicz, Zbigniew; *Genetic Algorithms + Data Structures = Evolution Programs*; Springer-Verlag; 1994.
- [6] Preston, Kendall and Duff, Michael; *Modern Cellular Automata: Theory and Applications*; Plenum Press; NY, 1984.
- [7] Toffoli, Tommaso and Margolus, Norman; *Cellular Automata Machines*; The MIT Press; 1989.
- [8] Wainer, Gabriel A.; *Simulación de sistemas de eventos discretos usando Autómatas Celulares*; Departamento de Computación; Facultad de Ciencias Exactas y Naturales; Universidad de Buenos Aires; 1996.

## **ANEXO 1. DETALLES DEL DESARROLLO DE SOFTWARE**

Los desarrollos de software realizados para implementar el *Autómata Celular Evolutivo* y obtener los resultados mostrados en este trabajo fueron realizados con técnicas de Programación Orientada a Objetos haciendo uso del lenguaje de programación C++ y bajo la plataforma de desarrollo Borland Builder 3<sup>8</sup>.

Se usaron arquitecturas de hardware Pentium III<sup>9</sup> 500 Mhz; 128 Mb de RAM y ambiente operativo Windows NT<sup>10</sup>.

Gracias a la plataforma de desarrollo y el ambiente operativo, la aplicación está totalmente concebida con una interfaz gráfica amigable lo que facilita el tratamiento con imágenes.

---

<sup>8</sup> Copyright © 1998 Borland International, Inc; <http://www.borland.com>.

<sup>9</sup> Producto registrado de Intel Corporation.

<sup>10</sup> Copyright © Microsoft Corporation.