

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS

**DESARROLLO DE UN ALGORITMO ADAPTATIVO BASADO EN
AUTÓMATAS CELULARES PARA ELIMINAR RUIDO EN IMÁGENES
DIGITALES**

Karen Vanessa Angulo Sogamoso

Danilo Gustavo Gil Sierra

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS
FACULTAD DE INGENIERÍA
BOGOTÁ, D.C.
2019

Aprobada por la Facultad de Ingeniería
en cumplimiento de los requisitos
exigidos para otorgar el título de:
Ingeniero de sistemas.

Helbert Eduardo Espitia Cuchango Dr.
Director de la Tesis

Joaquín Javier Meza Álvarez. MsC

Universidad
Bogotá D.C., 2019

DEDICATORIA

A mis padres ya que ellos son un pilar fundamental en mi existencia, son la luz que alumbra mi camino, son mi soporte y mi fortaleza, gracias su amor y su esfuerzo he alcanzado tantas metas y objetivos de mi vida, a mis hermanos quienes me han animado e incentivado para salir adelante, a mis familiares que siempre estuvieron compartiendo sus enseñanzas y brindándome su apoyo, y a mis allegados quienes hicieron de la universidad una etapa única e inolvidable. Ellos siempre me dieron la mano cuando más lo necesitaba y cuando sentía que desfallecía. Danilo.

Agradezco infinitamente a la vida y a mis padres por guiarme en el camino, que me llevo a ser quien soy; sin sus consejos y ayuda, este logro no sería posible, ya que, en cada momento difícil que he atravesado, me han dado su apoyo incondicional para no desfallecer y no rendirme jamás. A mis hermanos y a mi familia por creer en mí y en mis capacidades, por brindarme su amor y motivación en todo momento para que siempre alcanzara y perseverara por mis ideales. A mis profesores, compañeros y amigos, por permitirme aprender de ellos en cada instante, por compartir su conocimiento conmigo sin esperar nada a cambio. A mi universidad por fomentar en mí, el espíritu investigador, por darme las herramientas para ser una persona íntegra que le aporte a esta sociedad y por impulsarme siempre hacia el futuro. Karen.

RECONOCIMIENTOS

Un agradecimiento especial para:

- El Ingeniero Helbert Eduardo Espitia Cuchango, quien fue la persona que nos acompañó e inspiró en parte de nuestra vida académica. Gracias a sus enseñanzas adquirimos un mayor gusto por la ciencia y obtuvimos la guía que necesitamos a lo largo de esta investigación. Siempre nos alentó para dar lo mejor de nosotros y nos mostró que los obstáculos son nuevas oportunidades, para iniciar y explorar nuevos caminos.
- El Ingeniero Marco Aurelio Álzate Monroy que, con su amor y pasión por la investigación y el conocimiento, nos dio la motivación, inspiración y el camino para iniciar con este trabajo.

Resumen

En este trabajo, se proponen la implementación de dos algoritmos basados en autómatas celulares que simulan dos técnicas del procesamiento de imágenes, la eliminación de ruido y la detección de bordes. Este proceso se desarrolló mediante la aplicación de dos características importantes, el vecindario de Moore y el comportamiento adaptativo del autómata celular. En la parte inicial, se dará una explicación de los temas fundamentales que hacen parte del alcance de este proyecto, posteriormente, se hará una revisión bibliográfica de las diversas investigaciones que se han llevado a cabo, tanto para la eliminación de ruido como para la identificación de contornos, teniendo en cuenta que solo se consideran como válidas, todas aquellas que utilicen los autómatas celulares, debido a que esto otorgara una base sólida para poder comparar los resultados que se presentaran en esta investigación. No obstante, se presentará el proceso evolutivo de los algoritmos durante el transcurso de la investigación. Igualmente, se mostrará el funcionamiento de estas técnicas de forma separada, y como se enlazan, para que finalmente, uno sea el complemento del otro. Finalmente se plasmarán los resultados obtenidos y la posibilidad existente de ampliar el desarrollo de los modelos planteados.

Palabras clave: autómata celular, adaptativo, borde, ruido, algoritmo, imágenes digitales, procesamiento.

Abstract

In this project, we propose the implementation of two algorithms based on cellular automata that simulate two image processing techniques, noise elimination and edge detection. This process is developed through the application of two important qualities, a Moore neighborhood and the adaptive behavior of the cellular automaton. In the initial part, an explanation of the fundamental issues that are part of the scope of this project will be given, subsequently; a bibliographic review of the various investigations that have been carried out will be presented, both for the elimination of noise and for the identification of contours, taking into account that they are only considered as valid, all those that use

cellular automata, because this gives a solid basis to compare the results presented in this research. However, the evolutionary process of the algorithms will be presented during the course of the investigation. Likewise, the operation of these techniques will be presented separately, and how they are linked, so that finally, one is the complement of the other. Finally, the results obtained and the existing possibility of expanding the development of the proposed models will be shown.

Keywords: cellular automaton, adaptive, edge, noise, algorithm, digital images, processing.

Índice general

Descripción del proyecto.....	18
Descripción del proyecto.....	19
1.1. Introducción.....	19
1.2. Justificación	20
1.3. Descripción del problema	21
1.4. Formulación del problema.....	22
1.5. Objetivo general y objetivos específicos.....	22
1.5.1. Objetivo general.....	23
1.5.2. Objetivos específicos	23
Antecedentes.....	24
2.1. Eliminación de ruido.....	24
2.2. Detección de bordes	27
Marco referencial.....	30
Marco Teórico	31
3.1. Introducción.....	31
3.2. Autómata Celular	31

3.3.	Algoritmo Adaptativo.....	35
3.4.	Ruido en imágenes.....	36
3.4.1.	Gaussiano.....	36
3.4.2.	Impulsional (sal y pimienta).....	36
3.4.3.	De disparo.....	37
3.5.	Filtros Digitales.....	37
3.6.	Detección de Bordes.....	38
3.6.1.	Método basado en la primera derivada (gradiente).....	39
3.6.2.	Método basado en la segunda derivada (laplaciano).....	39
3.6.3.	Método de Canny.....	40
3.7.	Funciones morfológicas.....	41
3.7.1.	Operaciones morfológicas.....	41
3.8.	Juego de la vida.....	43
	Desarrollo.....	45
	Desarrollo del proyecto.....	46
4.1.	Introducción.....	46
4.2.	Elementos incidentes en el desarrollo del algoritmo.....	48
4.2.1.	Ruido.....	48
4.2.2.	Autómatas celulares y comportamiento adaptativo.....	48
4.3.	Análisis de los algoritmos base e implementación.....	49
4.3.1.	Experimentos realizados.....	50
4.4.	Desarrollo del modelo.....	56
4.5.	Resultados Obtenidos.....	60
4.6.	Análisis de los resultados.....	63
4.7.	Análisis comparativo sobre la diferencia porcentual.....	73
	Desarrollo de un algoritmo eliminador de ruido y detector de bordes.....	76
5.1.	Introducción.....	76
5.2.	Características en las imágenes digitales.....	77
5.2.1.	Contornos.....	78

5.3.	Análisis de los acercamientos a la detección de bordes	78
5.3.1.	Experimentos realizados	79
5.4.	Desarrollo del modelo de bordes	90
5.5.	Unificación de los algoritmos.....	95
5.6.	Análisis de los resultados obtenidos	98
5.6.1.	Cualitativos.....	98
5.6.2.	Cuantitativos	101
	Aportes originales y discusión.....	106
6.1.	Aportes originales	106
6.2.	Discusión.....	108
	Conclusiones y expectativas	109
7.1.	Conclusiones	109
7.2.	Expectativas	110
	Productos académicos.....	112
8.1.	Realizados	112
8.2.	En proceso de aceptación.....	115
8.3.	En revisión	116
	Anexo 1.....	118
A.1.	Resultados del método propuesto para la eliminación de ruido impulsivo en imágenes a escala de grises.....	118
	Anexo 2	123
B.1.	Resultados del método propuesto para la eliminación de ruido y la detección de bordes en imágenes a escala de grises	123
	Bibliografía	130

Índice de figuras

1: Dimensiones de autómatas celulares.	32
2: Vecindad de Von Neumman.....	34
3: Vecindad extendida de Von Neumman (n=2)..	34
4: Vecindad de Moore.....	34
5: Vecindad extendida de Moore (n=2).....	35
6: Representación de un filtro digital.....	37
7: Métodos de la primera y segunda derivada para la detección de bordes.	40
8: Imagen base (20x16 pixeles).	41
9: Aplicación de erosión.	42
10: Aplicación de dilatación.....	42
11: Aplicación de apertura.	42
12: Aplicación de cierre.....	43
13: Soledad - Regla del juego de la vida.	44

14: Sobrepoblación – regla del juego de la vida.	44
15: Supervivencia – regla del juego de la vida.....	44
16: Nacimiento – regla del juego de la vida	44
17: Esquema del funcionamiento de los métodos propuestos.	47
18: Eliminación de ruido, intento número 1. Imagen original con ruido al 10%.	51
19: Eliminación de ruido, intento número 1. Imagen original con ruido al 50%.	51
20: Eliminación de ruido, intento número 1. Imagen original con ruido al 90%.	52
21: Eliminación de ruido, intento número 2. Imagen original con ruido al 10%.	53
22: Eliminación de ruido, intento número 2. Imagen original con ruido al 50%.	53
23: Eliminación de ruido, intento número 2. Imagen original con ruido al 90%.	54
24: Eliminación de ruido, intento número 3. Imagen original con ruido al 10%.	54
25: Eliminación de ruido, intento número 3. Imagen original con ruido al 50%.	55
26: Eliminación de ruido, intento número 3. Imagen original con ruido al 90%.	55
27: Eliminación de ruido con funciones morfológicas. Imagen original con ruido al 50%.	56
28: Aplicación del Algoritmo Propuesto sin ajustar las variables de entrada, imagen con 10% de ruido.	60
29: Aplicación del Algoritmo Propuesto sin ajustar las variables de entrada, imagen con 50% de ruido.....	61
30: Aplicación del Algoritmo Propuesto sin ajustar las variables de entrada, imagen con 90% de ruido.....	61
31: Algoritmo final propuesto, imagen con 10% de ruido.....	62
32: Algoritmo final propuesto, imagen con 50% de ruido.....	62
33: Algoritmo final propuesto, imagen con 90% de ruido.....	63

34: Esquema del índice SSIM.	64
35: Porcentaje de ruido vs. SSIM para la imagen del babuino de 512 x 512 pixeles.	66
36: Porcentaje de ruido vs. SSIM para la imagen del Lenna de 512 x 512 pixeles.	67
37: Porcentaje de ruido vs. SSIM para la imagen del Lenna de 256 x 256 pixeles.	68
38: Porcentaje de ruido vs. SSIM para la imagen del cameraman de 256 x 256 pixeles.	69
39: Resultados del algoritmo aplicado en la imagen del lenna (512 x 512 pixeles) con un ruido de 50%.	71
40: Resultados del algoritmo aplicado en la imagen del lenna (512 x 512 pixeles) con un ruido de 90%.	72
41: Aplicación del Algoritmo Propuesto basado en una imagen binarizada cameraman 512x512.	79
42: Aplicación del Algoritmo Propuesto basado en una imagen binarizada lenna 512x512.	80
43: Aplicación del Algoritmo Propuesto basado en una imagen binarizada babuino 512x512.	80
44: Aplicación del Algoritmo Propuesto basado en una imagen binarizada monedas 256x256.	81
45: Aplicación del Algoritmo Propuesto basado en una imagen binarizada cebra 256x256.	81
46: Aplicación del Algoritmo Propuesto basado en el juego de la vida, cameraman 512x512.	82
47: Aplicación del Algoritmo Propuesto basado en el juego de la vida, lenna 512x512. ..	82
48: Aplicación del Algoritmo Propuesto basado en el juego de la vida, babuino 512x512.	83
49: Aplicación del Algoritmo Propuesto basado en el juego de la vida, monedas 256x256.	83
50: Aplicación del Algoritmo Propuesto basado en el juego de la vida, cebra 256x256.	84

51: Aplicación del Algoritmo Propuesto basado 4 niveles, cameraman 512x512..	85
52: Aplicación del Algoritmo Propuesto basado 4 niveles, lenna 512x512.....	85
53: Aplicación del Algoritmo Propuesto basado 4 niveles, babuino 512x512.	86
54: Aplicación del Algoritmo Propuesto basado 4 niveles, monedas 256x256.	86
55: Aplicación del Algoritmo Propuesto basado 4 niveles, cebra 256x256	87
56: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, cameraman 512x512.	88
57: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, lenna 512x512.....	88
58: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, babuino 512x512.	89
59: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, monedas 256x256.....	89
60: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, cebra 256x256.....	90
61: Patrones de una vecindad de 3 x 3 que permanecen invariantes con una rotación de $\pm 90^\circ$, 180° y una simetría tipo espejo.	91
62: Reglas de transición para el vecindario de Moore.....	93
63: Detección de bordes con ruido al 5%, monedas.	99
64: Detección de bordes sin ruido, monedas.	100
65: Casos de prueba para la detección de bordes.	102
66: Grafica de los resultados de métricas utilizadas al comprar el Algoritmo Propuesto versus los métodos convencionales	104
67 – A.1: Resultados del algoritmo aplicado en la imagen del babuino (512 x 512 pixeles) con un ruido de 50%.	119
68 – A.2: Resultados del algoritmo aplicado en la imagen del babuino (512 x 512	

pixeles) con un ruido de 90%.....	120
69 – A.3: Resultados del algoritmo aplicado en la imagen del Cameraman (256 x 256 pixeles) con un ruido de 50%.	121
70 – A.4: Resultados del algoritmo aplicado en la imagen del Cameraman (256 x 256 pixeles) con un ruido de 90%.	122
71 – B.1: Detección de bordes con ruido al 5%, cebra.....	124
72 – B.2.: Detección de bordes sin ruido, cebra.....	125
73 – B.3: Detección de bordes con ruido al 5%, Lenna.....	126
74 – B.4: Detección de bordes sin ruido, Lenna	127

Índice de diagramas

1: Causalidad para la eliminación de ruido.	50
2: Diagrama de flujo para la eliminación de ruido con autómatas celulares..	59
3: Causalidad para la detección de bordes.	77
4: Diagrama de flujo para la detección de bordes con autómatas celulares.....	94
5: Diagrama de flujo para eliminación de ruido y la detección de bordes con autómatas celulares. Parte 1.	96
6: Diagrama de flujo para eliminación de ruido y la detección de bordes con autómatas celulares. Parte 2.....	97

Índice de tablas

1: Resultados SSIM y porcentaje de ruido para la imagen del babuino (512x512).....	65
2: Resultados SSIM y porcentaje de ruido para la imagen de Lenna (512x512).	67
3: Resultados SSIM y porcentaje de ruido para la imagen de Lenna (256x256).	68
4: Resultados SSIM y porcentaje de ruido para la imagen de cameraman (256x256). ...	69
5: Clasificación cualitativa del desempeño al eliminar ruido.	73
6: Comparación valores SSIM con imágenes que tiene ruido del 50%.	73
7: Comparación valores SSIM con imágenes que tiene ruido del 90%.	74
8: Resultados de las métricas aplicadas para las imágenes 1, 2, 3 y 4.....	104

Abreviaturas

Abreviatura	Término
<i>2D</i>	Dos dimensiones
<i>AC</i>	Autómata Celular
<i>CPU</i>	Unidad central de procesamiento
<i>GPU</i>	Unidad de procesamiento gráfico
<i>MAXERR</i>	Error Máximo
<i>MSE</i>	Error Cuadrático Medio
<i>L2RAT</i>	Relación de la Norma al Cuadrado
<i>LoG</i>	Operador laplaciano del gaussiano
<i>px</i>	Píxeles
<i>RGB</i>	Sigla en inglés de Red, Green, Blue, o en español Rojo, Verde y Azul»
<i>SSIM</i>	Índice de Similitud Estructural

Parte I

Descripción del proyecto

Capítulo 1

Descripción del proyecto

1.1. Introducción

El presente trabajo tiene como finalidad diseñar un algoritmo adaptativo basado en autómatas celulares para la detección, depuración de ruido impulsivo y localización de bordes en imágenes digitales con escala de grises. Gracias a las investigaciones realizadas en el ámbito de procesamiento de imágenes, una imagen puede tratarse como una matriz en la cual cada pixel es representado por una celda y, asimismo, es como esta configuración permite la simulación del comportamiento de un autómata celular en su totalidad.

Considerando las investigaciones realizadas y teniendo en cuenta el estado del arte, se pretende proponer un algoritmo adaptativo como alternativa de las metodologías para eliminación de ruido tradicionales y detección de contornos, el cual puede ser implementado en un entorno programable de modelado especialmente diseñado para simular las condiciones básicas para el desarrollo idóneo del mismo.

Del mismo modo, se proyecta plantear un algoritmo con la capacidad de detectar bordes teniendo como base los autómatas celulares y el comportamiento adaptativo. Las labores primordiales del algoritmo se enfocan en la localización de píxeles y la identificación, ya que el autómata de una manera eficaz y eficiente debe tener la capacidad de distinguir las variaciones en las intensidades y demarcar estos cambios en la imagen resultante.

Mediante la simulación del algoritmo en un sistema computacional considerando

diferentes casos, se espera que los resultados obtenidos muestren una mejora cualitativa y cuantitativamente en relación a otros algoritmos con características similares.

Finalmente, es importante recalcar que el uso de técnicas poco convencionales que aplican en conjunto algoritmos adaptativos y autómatas celulares, podrán tomarse como base para procesamiento de imágenes digitales al momento de eliminar ruido conservando las características esenciales de la imagen.

1.2. Justificación

La propuesta de implementación de este algoritmo surge de la necesidad presente en el procesamiento de imágenes a la hora de reducir los niveles de ruido que se genera en el momento de la adquisición de una imagen [Esqueda and Palafox, 2005]. Las técnicas clásicas que comúnmente son usadas en este proceso según varios estudios, aun no satisfacen completamente las expectativas al restaurar imágenes con elevados niveles ruido [Tourounis et al., 2018], en este sentido, la implementación de un algoritmo que cumpla o que tenga un acercamiento visualmente aceptable podría considerarse como un avance en este ámbito.

Con el paso tiempo, el uso de las imágenes digitales han tenido mayor impacto en la vida cotidiana, y su aplicación en diferentes campos ha generado una pauta para lograr avances considerables. Partiendo de lo anterior, la búsqueda de nuevas metodologías que combinan técnicas y herramientas se han convertido en un aspecto fundamental para los nuevos avances, debido a buenos los resultados obtenidos en las etapas experimentales. Es por esto que partiendo de una serie de algoritmos que en su esencia aplican el concepto de autómatas celulares se logra una mejoría en la calidad de las imágenes cuando estas se han visto afectadas por el ruido [Tourounis et al., 2018].

La comprensión, explicación y construcción de sistemas y soluciones ingenieriles capaces de combinar el concepto de un modelo matemático junto con técnicas básicas pueden llegar a generar un sistema dinámico que interactúa con su entorno y modifica su estado en pasos discretos para generar un resultado, es decir, la simulación de un autómata celular puede definirse como un pixel que modifica su valor de intensidad a partir de la información obtenida de sus pixeles contiguos y de acuerdo con un conjunto de parámetros previamente establecidos, toma una decisión para alterar su valor. Este proceso se repite hasta recorrer la totalidad de la imagen y obtener un resultado.

Por lo anterior, se busca realizar un sistema que combine filtrado en el dominio del espacio y el principio de funcionamiento de autómatas celulares con el objetivo de eliminar el ruido de sal y pimienta modificando los píxeles corrompidos para así mejorar

el proceso de filtrado.

Desde otra perspectiva dentro del ámbito de la investigación, la localización de contornos a pesar ser un concepto sencillo a simple vista, implica una labor compleja de establecer en la medida que no es fácil obtener los resultados deseados de manera óptima. Ya que, desde la perspectiva histórica, algunos algoritmos, métodos y metodologías establecidos, han marcado una pauta en este ámbito por los resultados obtenidos en sus investigaciones, pero normalmente tienen un alto costo computacional y una complejidad considerable [Rosin and Sun, 2013]. Además de lo anterior, la definición de bordes en una imagen, está ligada a la subjetividad del algoritmo, teniendo en cuenta que para algunos lo que puede ser considerado como un contorno, para otros no.

No obstante, la detección de bordes basada en autómatas celulares complementa la labor desempeñada previamente la eliminación de ruido impulsivo y genera un aporte valioso a la investigación. Es por esto que el resultado exitoso de la implementación de estos algoritmos podrá traer consigo grandes beneficios a diferentes áreas del conocimiento que en su práctica hagan uso de imágenes digitalizadas, ya que el planteamiento consiste en implementar un algoritmo basado en autómatas celulares que sea capaz de eliminar de manera sustancial el ruido y a su vez, distinga contornos en una imagen.

1.3. Descripción del problema

El procesamiento de imágenes digitales surge a partir de la necesidad de mejorar el aspecto de las mismas, y con ello hacer más evidentes determinados detalles [Malacara, 2002]. A raíz de esto fueron implementadas técnicas que permiten tratar las diferentes características que posee una imagen (p.e. brillo, nitidez, contraste, intensidad, ruido, entre otras), pero particularmente, surgieron métodos que permiten resaltar o suprimir, de forma selectiva, la información contenida en una imagen [Malacara, 2002].

En este sentido, uno de los defectos frecuentemente abarcados es el ruido, ya que, a través de los diferentes dispositivos con capacidad de capturar imágenes digitales, se pueden generar errores o interferencias al transmitir los bits de información. En consecuencia, teóricamente se han definido diferentes tipos de ruido, los cuales se clasifican de acuerdo a sus características. Debido a esta situación, para cada uno de ellos se han desarrollado métodos y técnicas capaces de tratarlos.

Dentro de las técnicas convencionales se encuentran los filtros en el dominio del espacio y en el dominio de la frecuencia (pasa altos, pasa bajos, media, mediana, etc.), los cuales frente a nuevas técnicas han mostrado resultados menores [Tourounis et al., 2018]

y, por tal razón, se ha intentado buscar alternativas con el ánimo de obtener resultados eficientes y visualmente estéticos. Por ende, la combinación de métodos, técnicas y diversos modelos matemáticos, han sido de utilidad para intentar alcanzar dichos objetivos.

Igualmente, la detección de bordes desempeña un papel importante dentro del procesamiento de imágenes, ya que a través de las metodologías existentes es posible identificar objetos, definir patrones o segmentar información dentro de las imágenes. Es por esto que, para realizar algunas precisiones, dentro del ámbito de este trabajo investigativo, un borde se define cuando existe una diferencia notable entre los niveles de intensidades de grises dentro de la imagen, en otras palabras, dicho cambio es reconocido como una frontera entre dos (2) regiones diferentes en una imagen. Es por esto que el concepto de autómata celular propone una alternativa viable para que, por medio de reglas pre-establecidas, una celda tenga la capacidad de modificar su estado de acuerdo a la manera como interactúe con sus vecinos. Es por esto, que resulta factible aplicar este concepto en el ámbito del procesamiento de imágenes junto con otras técnicas para potenciar los resultados esperados [Delorme, 1998].

Dado lo anterior, investigadores e interesados en el tema han implementado métodos, algoritmos, o diversas técnicas que combinan teorías básicas con autómatas celulares para eliminar el ruido y detectar bordes en imágenes. A pesar que se ha llegado a obtener resultados sobresalientes, no siempre se han sido satisfactorios a nivel visual, de allí se genera la necesidad de intentar mejorar las propuestas realizadas por diversos autores, procurando mejorar los resultados que estos presentan.

1.4. Formulación del problema

¿Qué características son relevantes a la hora de realizar un algoritmo adaptivo basado en autómatas celulares que sea capaz de reconocer las características propias de la imagen a la hora de eliminar el ruido presente en la misma?

¿Cómo eliminar diferentes tipos de ruido impulsivo en imágenes digitales a través de la implementación de un algoritmo que tenga inmerso los conceptos de adaptación y autómatas celulares?

1.5. Objetivo general y objetivos específicos

1.5.1. Objetivo general

Desarrollar un algoritmo basado en autómatas celulares que permita eliminar de ruido impulsivo en imágenes digitales.

1.5.2. Objetivos específicos

- Determinar el tipo de ruido en imágenes que puede ser eliminado, por medio de la implementación de un autómata celular con vecindad de Moore.
- Identificar entre la literatura científica actual una serie de algoritmos con capacidades semejantes a la planteada, los cuales permitan establecer un marco comparativo y un punto de referencia.
- Establecer casos de aplicación, donde se pueda evidenciar los resultados y la utilidad del Algoritmo Propuesto.
- Obtener un algoritmo que aplique el concepto de autómatas celulares fundamentado en el vecindario de Moore para filtrar los píxeles de una imagen digital.
- Establecer el nivel de efectividad y eficiencia del algoritmo mediante pruebas que permitan validar de forma cualitativa y cuantitativa los resultados obtenidos.

Capítulo 2

Antecedentes

A continuación, se presenta de manera resumida los estudios relacionados con el tema de este trabajo, cabe resaltar que, no se ha encontrado una investigación que contemple las mismas características del presente estudio.

2.1. Eliminación de ruido

El procesamiento de imágenes se remonta a cientos de años cuando filósofos y científicos como Alhazen, Aristóteles, Roger Bacon y Leonardo Da Vinci, entre otros, centraron sus esfuerzos en el estudio de la formación de imágenes en el ojo humano (óptica) [Nieto and Díaz, 2019]. Siendo este el primer paso para que años más tarde, investigadores y expertos en el tema hicieran uso de estos conceptos como base fundamental, para así lograr desencadenar una corriente investigativa enfocada al tratamiento de fenómenos ópticos que fueron útiles posteriormente para la creación de imágenes [Nieto and Díaz, 2019]. Años más tarde, gracias a la teoría desarrollada por el matemático Jean-Baptiste-Joseph Fourier (1768-1830), se pudo verificar que el fundamento del procesamiento de imágenes tanto para medios ópticos como para medios digitales, está basado en el teorema de Fourier, ya que la representación de la variación de la irradiancia o brillantez de una imagen puede ser vista como una suma de distribuciones senoidales de varias frecuencias [Malacara, 2002].

Actualmente, las imágenes desempeñan un papel importante en la percepción de la

realidad, debido a que el movimiento se puede interpretar como una secuencia de imágenes (fotogramas) que se visualizan en un segundo y a partir de estos fotogramas es posible identificar características relevantes de un objeto en particular o de su entorno en general. Por lo anterior, esta temática es de vital importancia en diversos ámbitos científicos tales como: análisis y diagnósticos médicos, exploración espacial, visión computacional, reconocimiento de imágenes, etc. [Diwakar et al., 2013], ya que con el uso de imágenes se han generado aportes significativos en estos campos y al mismo tiempo, se han desencadenado el desarrollo de técnicas que tienen como objeto mejorar el aspecto de la imagen y resaltar los detalles que se desean hacer más visibles de la misma [Nieto and Díaz, 2019], [Malacara, 2002].

Son múltiples las aplicaciones en el campo del procesamiento de imágenes que hacen uso del concepto de Autómata Celular (AC) el cual se remonta al desarrollo realizado por John von Neumann [Neumann, 1952], [Gómez, 2013], para solucionar eficientemente diversos problemas que se presentan en este campo. La simplicidad del diseño y la similitud inherente entre un AC y las técnicas de procesamiento de imágenes permiten que estos hayan pasado a ser considerados como una de las herramientas computacionales inspiradas en la biología innovadoras de últimos tiempos [Dalhoum et al., 2012].

El procesamiento de imágenes abarca una gran cantidad de técnicas que tienen diversos objetivos. Entre ellas se encuentra la eliminación de ruido, la cual busca que una imagen corrupta presente su mejor forma. Entendiendo como corrupta aquella imagen que ha sido afectada por algún factor externo y, en consecuencia, presenta ruido sobre ella imposibilitando una lectura correcta de la misma [Betancourt et al, 2003]. Los ruidos más comunes dentro del ámbito del procesamiento de imágenes son el ruido gaussiano y el impulsivo.

Como se mencionó anteriormente, las imágenes a menudo son afectadas por ruido, siendo el más común el ruido impulsivo, el cual puede ser causado por diferentes factores como errores o interferencias en los tramos de transmisión (es decir, al transmitir los bits de información), ubicaciones de memoria defectuosas en el hardware, errores asociados al equipo utilizado para la captación de imágenes o por el mal funcionamiento de los píxeles en los sensores de la cámara [Tourounis et al., 2018].

Una imagen al igual que cualquier tipo de dato puede llegar a revelar información valiosa en cualquier ámbito y la presencia de ruido puede afectar significativamente los resultados que se esperan de ella. Dentro de la clasificación del ruido impulsivo existe un caso especial y bastante conocido denominado “Sal y Pimienta”. Este puede ser distinguido con facilidad en una imagen contaminada a través de los píxeles que toman valores máximos o mínimos; es decir que normalmente aparecen como puntos negros y blancos sobre la imagen [Betancourt et al, 2003], [Tourounis et al., 2018].

Para la restauración de una imagen que ha sido afectada por ruido impulsivo, particularmente por el ruido de sal y pimienta, se han propuesto diversas técnicas que han mostrado su eficiencia y sus aportes a este campo, al generar variaciones o al formular nuevos tipos de filtros los cuales han sido perfeccionados continuamente, con el fin de obtener mejores resultados en menor tiempo. En la actualidad, el método más utilizado para la suprimir este tipo de ruido corresponde a filtrado no lineal denominado filtro de mediana, gracias a la efectividad que ha mostrado al ser aplicado sobre imágenes donde la presencia de ruido es de baja densidad [Tourtounis et al., 2018].

El filtro de mediana aplica una operación de mediana sobre cada pixel, indistintamente si este es ruidoso o no, lo que genera una leve destrucción de los detalles de la imagen (líneas, esquinas, bordes) [Pitas, 1990]. No obstante, al observar los problemas que aún se presentan en el filtrado básico del ruido de sal y pimienta, diversos investigadores han desarrollado nuevas técnicas y enfoques, para obtener mejoras en este tema.

Particularmente [Garnett et al., 2005] Garnett, Huegerich, Chui y He, desarrollaron un filtro por medio de valores estadísticos obtenidos de la cuantificación de las diferencias entre intensidades de pixeles respecto a sus vecinos más similares. Al aplicarlo sobre una imagen se pudo apreciar que este filtro se extiende de tal forma que es posible eliminar tanto ruido gaussiano como ruido impulsivo, los resultados fueron sobresalientes a nivel cuantitativo y cualitativo. Sin embargo, el filtro presenta algunas fallas cuando el nivel de ruido es demasiado alto, puesto que, si la mayoría de sus vecinos son pixeles con ruido, el valor estadístico calculado no es óptimo.

Por otro lado, Echeverri, Rudas, Toscano y Ballesteros por medio del uso de la interpolación a través de funciones de base radial, lograron generar un método para la eliminación de ruido impulsivo en imágenes a color [Echeverri et al., 2011]. Para este método compararon sus resultados con algoritmos clásicos no lineales como filtrado por mediana, media y “outlier”, logrando así demostrar que este algoritmo es más eficiente frente a imágenes con alto porcentaje de ruido. Mientras que con imágenes de baja densidad de ruido no fue tan efectivo. Al mismo tiempo, Esakkirajan, Veerakumar, Subramanyam y Prem propusieron un algoritmo de filtro asimétrico de mediana basado en decisión, que fue aplicado en la restauración de imágenes en escala de grises y a color que estaban afectadas por el ruido impulsivo [Esakkirajan et al., 2011]. A nivel cualitativo el algoritmo presentó mejores resultados que el Filtro de Mediana Estándar (Median Filter o MF), el Algoritmo Basado en Decisión (Decision Based Algorithm o DBA), el Algoritmo Basado en Decisión Modificado (Modified Decision Based Algorithm o MDBA) y el Filtro de Mediana con Cambio Progresivo (Progressive Switched Median Filter o PSMF).

Asimismo, Sahin, Uguz y Sahin proponen un algoritmo que combina los conceptos de un autómata celular 2D y lógica difusa para la restauración de imágenes digitales afectadas por ruido impulsivo [Sahin et al., 2014]. El algoritmo utiliza una regla de

transición difusa local que asigna un valor de membresía a la vecindad de píxeles corruptos y establece el siguiente valor de estado como un valor de píxel central, logrando eliminar el ruido hasta en un 90%. Aunque, el método de eliminación de ruido dividido en dos (2) fases muestra similitud con el anterior en el porcentaje de ruido eliminado en la imagen. Este algoritmo ha demostrado ser muy efectivo para imágenes con altas densidades de ruido, sin embargo, el tamaño de ventana al ser tan grande, aumenta considerablemente el tiempo de procesamiento [Chan et al., 2005]. Dicho algoritmo en la primera fase, aplica un filtro de mediana adaptable el cual identifica los píxeles que probablemente estén contaminados por ruido (llamados candidatos de ruido), y en la segunda fase se realiza la restauración de la imagen a través de un método de regularización especializado el cual es aplicado a los píxeles que han sido seleccionados como candidatos ruidosos.

También hay otro grupo de algoritmos que buscan eliminar del ruido presente en una imagen, a través de distintas técnicas, una de ellas se basa en la aplicación de lógica difusa con el fin de crear un filtro multi-paso para la eliminación de ruido impulsivo [Betancourt et al, 2003]. Otro trabajo realizado en [Cano and Díaz, 2013] consiste en la implementación de un algoritmo para eliminar ruido impulsivo en imágenes por medio de interpolación a través de funciones de base radial; también se realiza un análisis comparativo de tiempos de respuesta bajo arquitecturas CPU y GPU (Unidad Central de Procesamiento y Unidad de Procesamiento Gráfico). Se realizan pruebas de ruido impulsivo con diferentes dimensiones, con el fin de determinar los tiempos de respuesta. Los resultados son analizados considerando principalmente la eliminación completa del ruido observando que la arquitectura CPU presenta un mejor desempeño que la arquitectura GPU.

2.2. Detección de bordes

Con el paso de los años, diferentes técnicas para la detección de contornos se han planteado, las cuales abarcan diferentes características que puede presentar una imagen digital y emplean diferentes metodologías para exhibir soluciones eficientes. El objetivo de esta técnica contempla la identificación de los cambios de intensidad en una imagen (variación de la luminancia), es decir que, a través de una serie de operaciones matemáticas se busca detectar las discontinuidades entre los píxeles [Wongthanavas, 2011].

Dentro de los métodos que se enfocan en resolver esta temática, se destacan los trabajos realizados por Lawrence Roberts en 1963 y John Canny en 1986, que hicieron uso de métodos basados en el gradiente para estimar la magnitud y la dirección del mismo, esto

con el fin de identificar los píxeles que son considerados como bordes [Rosin and Sun, 2013], lo que quiere decir, que los bordes son detectados mediante las derivadas espaciales de la imagen que se calculan utilizando operadores de convolución. Cabe señalar, que además de los presentados anteriormente, existen los operadores de Sobel y Prewitt que también hacen uso de la derivada de primer orden.

Por otra parte, el operador Laplaciano de Gauss (o LoG) utiliza la segunda derivada de la imagen y dada esta cualidad, el filtro es sensible al ruido [Öztürk and Akdemir, 2015]. Al evaluar los resultados, que ofrece cada uno de estos operadores, se puede apreciar que a pesar que los resultados obtenidos son destacables (visualmente), todos estos presentan falencias en imágenes que contienen cierta cantidad de ruido y, en consecuencia, la calidad de la detección se ve afectada por la localización de bordes falsos [Xiao and Hui, 2010].

Específicamente, en el ámbito de los autómatas celulares, la mayor cantidad de trabajos, se han orientado a imágenes binarias, debido a que en este tipo de imágenes el potencial de este concepto se puede explotar de manera sencilla y eficiente. Aunque al aplicar esta técnica es posible que se ignore algunos bordes presentes en la imagen, debido a lo anterior, resulta bastante común que se integren diferentes técnicas que permiten a los investigadores avanzar más en el campo del procesamiento de imágenes digitales.

Una de las técnicas más simples fue propuesta por Wongthanavasú y Sadananda [Wongthanavasú and Sadananda, 2003], en donde su funcionamiento básicamente se limita a que el píxel en evaluación de un autómata con vecindario de von Neumann en una imagen binaria, cambia de valor (a 0 o a 1), si y solo si todo el vecindario tiene el mismo valor, y solo es necesaria una iteración para que el resultado sea obtenido, ya que con las demás iteraciones no se modifica la imagen. A pesar de su simpleza y efectividad, este algoritmo presenta falencias cuando las imágenes tienen grandes cantidades de ruido, ya que impide el funcionamiento adecuado del autómata.

Otro tipo de acercamientos se centraron en encontrar reglas de estado de transición óptimas, para esto es bastante común el uso de algoritmos genéticos, los cuales pueden presentar resultados considerablemente satisfactorios. Este es el caso de Sahota y compañía [Sahota et al., 1994], generaron tres (3) imágenes con y sin los bordes denominándolas imágenes de entrenamiento y a pesar de que no presentaron las reglas obtenidas, obtuvieron resultados destacables, por lo cual otros investigadores se orientaron hacia ese tipo de herramientas. Este fue el caso de Selvapeter y Hordijk, que de manera análoga [Selvapeter and Hordijk, 2013] usaron un método similar para entrenar autómatas celulares, e incluso imágenes semejantes de entrenamiento. La diferencia radica en que este algoritmo tiene la capacidad de realizar su trabajo en imágenes con diferentes niveles de ruido. A nivel técnico utilizaron un filtro de ruido basado en autómatas celulares, y posteriormente entrenaron otro autómata celular para la detección de bordes, y sus resultados son comparables con los métodos de Sobel, Prewitt y Canny a

pesar de que estos métodos trabajan directamente sobre la imagen con ruido y no es filtrado previamente.

Desde una perspectiva más eficiente, Batouche, Meshoul y Abbassene [Batouche et al., 2006] establecieron un algoritmo genérico basado en un autómata celular para la detección de bordes, que no requería una cantidad tan alta de patrones de entrenamiento (en un vecindario de Moore son necesarias aproximadamente $(2^9 = 512)$). Su funcionamiento en términos generales consistía en combinar patrones de acuerdo a un umbral de similitud lo cual redujo la cantidad de patrones de entrenamiento a 15, y posteriormente se giraba la imagen cada 90° aplicando las mismas reglas. Gracias a esto, el tiempo de maquina se redujo y los resultados, a pesar de que son un poco gruesos, presentan buena calidad.

Slatnia, Batouche y Melkemi [Slatnia et al., 2007], adoptaron una metodología similar a [Batouche et al., 2006], ya que asignaron patrones simétricos a una misma regla, pero no entrenaron un conjunto de reglas de la misma manera. En su lugar, entrenaron un algoritmo genético con una sola regla genérica, en la cual el pixel central cambia su estado solamente cuando su vecindario de Moore coincide con un patrón específico. Coincidentemente obtuvieron la misma regla que Wongthanavasú y Sadananda [Wongthanavasú and Sadananda, 2003] para este caso compararon los resultados con los de Canny, pero visualmente no son tan buenos.

Por otro lado, Yang, Ye y Wang [Yang et al., 2002], establecieron un algoritmo basado en autómatas celulares teniendo presentes los vecindarios tanto de Moore como de von Neumann, el cual puede definirse en dos pasos. El primer paso consiste en una iteración en la cual evalúa si tres (3) vecinos contiguos del píxel central tienen el valor uno (1) y los otros tres (3) vecinos contiguos simétricos con respecto al píxel central tienen el valor cero (0), entonces el píxel en evaluación mantiene su estado, de lo contrario lo cambia. En el segundo paso se realizan varias iteraciones basado en una regla en la cual el píxel en evaluación se considera borde, si exactamente dos (2) píxeles en el vecindario de von Neumann son bordes, y si los demás vecinos en un vecindario de Moore no son bordes.

Como los anteriores, existe un considerable número de trabajos en los cuales se abarca la temática de detección de contornos, los cuales abarcan esta situación desde diferentes perspectivas, teniendo presente que hoy en día existen diversos tipos de imágenes con escalas de intensidades variadas. Asimismo, computacionalmente es posible abarcar metodologías que antiguamente no se consideraban eficientes o eficaces debido a su complejidad y tiempos. Es común encontrar comparaciones entre resultados obtenidos versus los de Sobel, Prewitt y sobre todo Canny. El problema de estos radica en los resultados ya que, a pesar de los destacables avances por estos investigadores, no es coherente comparar los resultados con algoritmos o métodos que no cuentan con las mismas condiciones.

Parte II

Marco referencial

Capítulo 3

Marco Teórico

3.1. Introducción

Para desarrollar la idea de esta investigación, se tuvo claro un marco teórico que delimito el campo de acción conceptual, por ello los siguientes temas que se mencionan, están definidos en concordancia con las necesidades de este proyecto y la manera como se relacionan con la labor investigativa.

3.2. Autómata Celular

Los autómatas celulares constituyen una clase particular de sistemas dinámicos discretos [Neumann, 1952], los cuales a través de una definición sencilla pueden modelar sistemas de gran complejidad [Rueda and Torres, 2013] [Dhillon, 2012]. Estos pueden definirse como un conjunto de celdas con una dimensión D , las cuales pueden tomar la forma de $D = 1$, $D = 2$ o $D = 3$, como se muestra en la Figura 1, dependiendo de la utilidad que se le dé. Actualmente, el caso más utilizado tanto en las diversas aplicaciones de las ciencias investigativas como particularmente en el procesamiento de imágenes, es aquel que simula un espacio bi-dimensional.

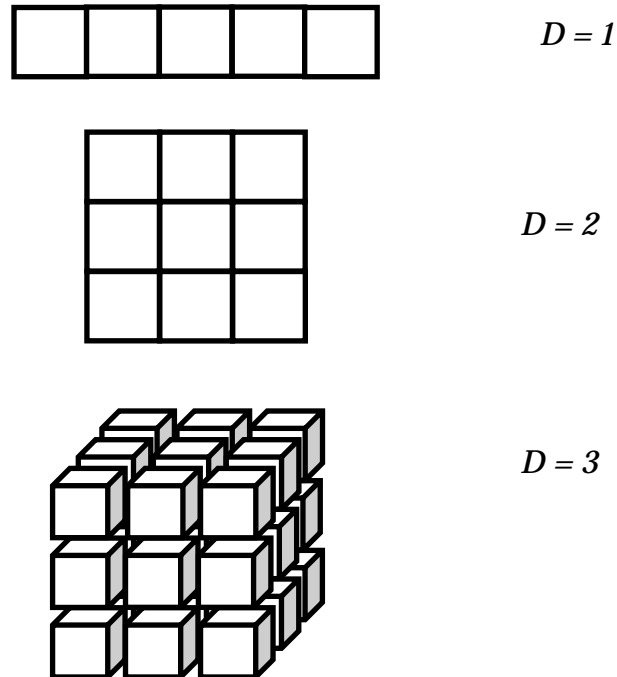


Figura 1: Dimensiones de autómatas celulares. **Fuente:** Elaboración propia.

Independientemente de las dimensiones elegidas, todo autómata celular, cuenta con una estructura básica determinada, que está compuesta por tres (3) partes; a) conjunto de celdas o enrejado, b) un conjunto de vecinos adyacentes o vecindario y c) Un grupo de reglas para transiciones locales [Pan et al., 2009].

- **Conjunto de celdas o enrejado**, el cual tiene un número finito de posibles estados o valores para cada una de las celdas, en el caso de imágenes binarias 1 o 0, y para imágenes en escala de grises de 0 a 255.
- **Un conjunto de vecinos adyacentes o vecindario**, como el vecindario de Von Neumann (cuatro (4) vecinos) o el vecindario de Moore (con ocho (8) vecinos).
- **Un grupo de reglas para transiciones locales** (funciones locales de transición, transición del estado de las celdas o tabla de transiciones de los estados de las celdas). En síntesis, dentro de los grupos de reglas existen dos (2) particularidades a destacar: i) la comunicación entre la celda central y sus vecinos es local, uniforme y síncrona y ii) la evolución global del sistema a través de un paso de tiempo discreto es determinista [Pan et al., 2009].

Teniendo en cuenta lo anterior, respecto al funcionamiento de un autómata celular, se puede afirmar que, el estado de la célula en evaluación, en el tiempo $t + 1$, será determinado por el estado actual de los vecinos circundantes y su estado actual, en otra palabras, este será actualizado de manera síncrona en intervalos de tiempo discretos [Shukla, 2016] [Neumann, 1952] [Popovici and Popovici, 2002]. E Al definirlo desde una perspectiva más formal, se puede decir que un autómata celular d -dimensional (o d -CA), puede ser visto como una 4-tupla $A = (Z^d, S, N, \delta)$, tal que,

Z^d : es un espacio d -dimensional de números enteros.

S : es un conjunto finito que corresponde a los estados de A .

$N = \left\{ \begin{matrix} n_i \\ n_j \end{matrix} = (x_{ij}, \dots, x_{dj}), j \in \{1, \dots, n\} \right\}$: es un sub-conjunto finito ordenado de Z^d , llamado vecindario de A , y

δ : corresponde a una función de transición local, o regla local de A tal que $\delta: S^{n+1} \rightarrow S$.

Por otro lado, la vecindad de un autómata celular determinista, se encuentra representada por celdas contiguas o circundantes respecto a una celda central, que al extrapolar el concepto a una imagen, se podría decir que estas están asociadas directamente con un pixel. Este conjunto de celdas tiene una formación en particular de acuerdo con la situación que se presente, ya que puede ser requerida mayor o menor información.

Esencialmente las vecindades pueden definirse en términos de simetría: vecindades simétricas y asimétricas. Las vecindades simétricas cuentan con una base (un numero entero), donde el valor asignado a esta corresponde al número de vecinos en cada dirección, aunque por defecto el valor de la base corresponde a 1, lo que quiere decir, que a partir de la celda central existe un vecino a la derecha, uno a la izquierda, uno arriba y uno abajo, para el caso en particular de la vecindad de Neumann. Por otro lado, las vecindades asimétricas como su nombre lo indica, pueden variar el número de celdas a partir de la celda central.

Dentro de las múltiples variaciones existentes en una vecindad, existen dos tipos de vecindario muy conocidos, los definidos por Neumann y por Moore. El primero de estos en su nivel más elemental presenta 4 vecinos a partir del pixel central como se muestra en la Figura 2.

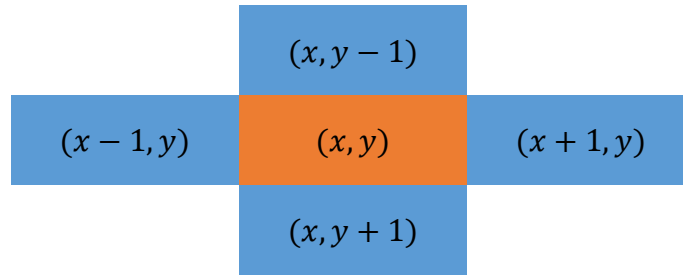


Figura 2: Vecindad de Von Neumann. **Fuente:** Elaboración propia.

En muchos de los casos este vecindario puede extenderse n veces, buscando así que mediante las reglas previamente establecidas pueda manejarse una cantidad mayor de información tal como se muestra en la Figura 3.

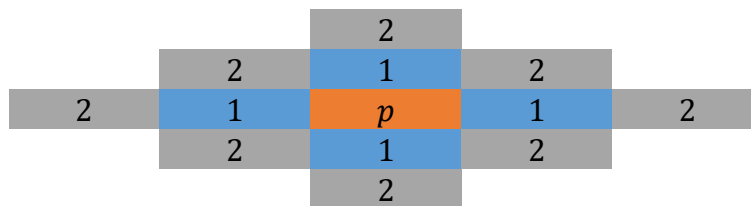


Figura 3: Vecindad extendida de Von Neumann ($n = 2$). **Fuente:** Elaboración propia.

En el caso de la vecindad de Moore en su forma esencial considera 8 vecinos a partir de la celda base, lo cual facilita en el manejo de bordes y esquinas; además la información que se puede gestionar con esta vecindad es mayor que la vecindad de Neumann, debido a que se consideran cuatro celdas adicionales, tal como se muestra en la Figura 4.

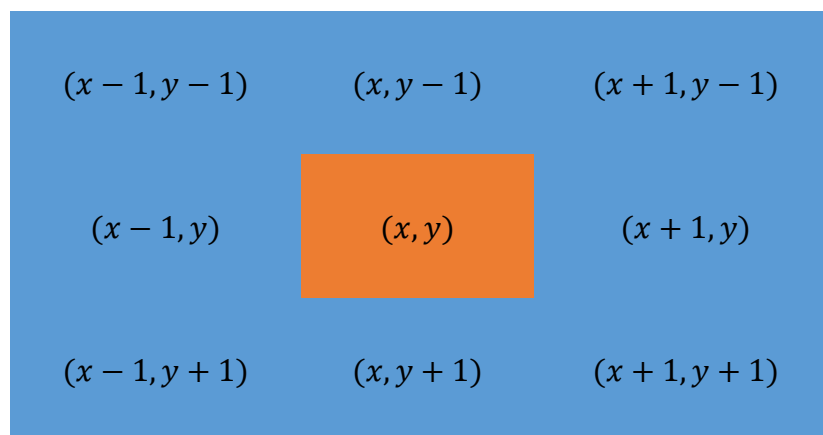


Figura 4: Vecindad de Moore. **Fuente:** Elaboración propia.

De la misma forma como ocurre con la configuración de Neumann, este vecindario también presenta versiones extendidas de n pixeles, las cuales obedecen a la fórmula $(2n + 1)^2 + 1$, un ejemplo de esta vecindad se puede apreciar en la Figura 5.

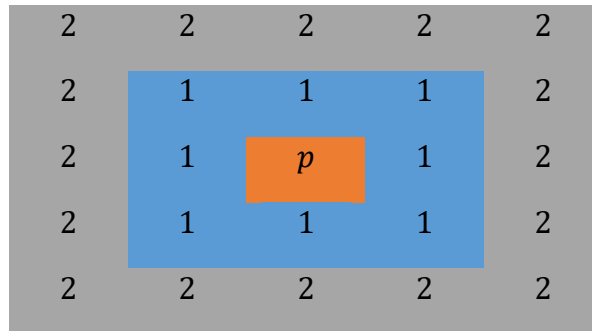


Figura 5: Vecindad extendida de Moore ($n = 2$). **Fuente:** Elaboración propia.

Para determinar las ecuaciones que definen un el vecindario de von Neumann, Se hace uso de una cuadrícula de tamaño $l \times l$ bidimensional para definir el conjunto de celdas que rodean a la celda en evaluación (x_0, y_0) . El vecindario de von Neumann de rango r está definido por ecuación (1):

$$N_{x_0, y_0} = \{(x, y) : |x - x_0| + |y - y_0| \leq r\} \quad (1)$$

Al igual que en un vecindario de Neumann, este vecindario utiliza una cuadrícula de tamaño $l \times l$ y el rango r del mismo está definido por la siguiente ecuación (2):

$$N_{x_0, y_0} = \{(x, y) : |x - x_0| \leq r, |y - y_0| \leq r\} \quad (2)$$

3.3. Algoritmo Adaptativo

El concepto de adaptabilidad en los algoritmos evolutivos consiste en un tema clave en el campo de la ingeniería y de las ciencias en general [BenHamida and Schoenauer, 2000]. Debido a lo anterior, la idea de desarrollar un algoritmo que sea capaz de ajustar sus propios parámetros de forma libre resulta conveniente [Montillet et al., 2016],

[BenHamida and Schoenauer, 2000].

Entonces, en un sentido más formal, un algoritmo adaptativo se puede definir como un conjunto de instrucciones que realizar una función determinada y tiene la capacidad de adaptarse a cambios que se producen en el entorno o las circunstancias [BenHamida and Schoenauer, 2000], [Montillet et al., 2016], [Chaikalis and Riera-Palou, 2009], [Pérez and Nakato, 2009].

Los algoritmos adaptativos pueden ajustar inteligentemente sus actividades a la luz de las circunstancias cambiantes para lograr el mejor resultado posible [Benavides et al., 2009]. Es decir, que este cambia su comportamiento en función de la información disponible al momento de su ejecución y puede o no cambiar su comportamiento si detecta la necesidad de hacerlo.

3.4. Ruido en imágenes

Se dice que, todas las imágenes tienen cierta cantidad de ruido presente en las mismas, el cual puede provenir de la cámara o el medio de transmisión de la señal. Generalmente el ruido se manifiesta como píxeles aislados que toman un nivel de gris diferente al de sus vecinos [Esqueda and Palafox, 2005]. Dado lo anterior, el ruido puede clasificarse en los siguientes tipos:

3.4.1. Gaussiano

Este tipo de ruido produce pequeñas variaciones en la imagen en donde todos los valores de los píxeles aumentan o disminuyen levemente su intensidad [Esqueda and Palafox, 2005]; usualmente se debe a diferentes ganancias en la cámara, ruido en los digitalizadores, perturbaciones en la transmisión, etc. Se considera que el valor final del píxel sería el ideal más una cantidad correspondiente al error que puede describirse como una variable aleatoria gaussiana [Esqueda and Palafox, 2005].

3.4.2. Impulsional (sal y pimienta)

El valor que toma el píxel no tiene relación con el valor ideal, por lo contrario, se relaciona con el valor del ruido que toma valores muy altos o bajos (puntos blancos y/o negros) causados por una saturación del sensor o por un valor mínimo captado, si se ha perdido la señal en ese punto [Esqueda and Palafox, 2005]. Este también está presente al

trabajar con objetos a altas temperaturas, ya que las cámaras tienen una ganancia en el infrarrojo que no es detectable por el ojo humano; por ello las partes más calientes de un objeto pueden llegar a saturar un píxel.

3.4.3. De disparo

También llamado ruido fotónico, consiste en cambios que se generan por la detección de fotones para un determinado nivel de exposición. Este fenómeno ocurre por la naturaleza cuántica de la luz, y obedece una ley de distribución de Poisson y en consecuencia, el efecto que causa en la imagen es similar al ruido Gaussiano [Márquez, 2013]. Asimismo, se suele presentar de manera más constante cuando los niveles de iluminación son muy bajos.

3.5. Filtros Digitales

Los filtros digitales se pueden definir como el proceso computacional o algoritmo mediante el cual, una señal digital (secuencia de muestras) es transformada en otra secuencia de salida [González and Woods, 2006]. Es decir, que corresponde a un proceso que toma una sucesión de números (señal de entrada) y la transforma en otra (señal de salida) con el objetivo de cumplir con determinadas especificaciones. Un filtro digital puede ser representado mediante el diagrama de la Figura 6.

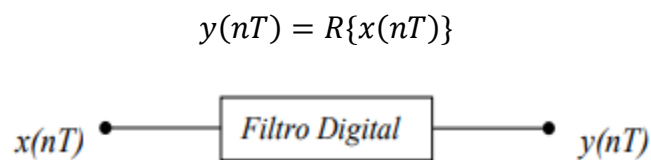


Figura 6: Representación de un filtro digital. **Fuente:** Basado en [Friedrich, 2002].

Donde $x(nT)$ es la secuencia de entrada (la excitación del filtro) y $y(nT)$ es la respuesta del filtro ante la excitación $x(nT)$. Considerando lo anterior, una imagen se puede filtrar en el dominio del espacio, trabajando directamente sobre los píxeles de la imagen, o en el dominio de la frecuencia, donde las operaciones se llevan a cabo mediante la transformada de Fourier de la imagen [García and Viteri, 2009]. Para el proceso de filtrado en el dominio del espacio se debe relacionar un conjunto o vecindario de píxeles cercanos con el píxel objetivo, con el fin de obtener información útil que le permita al filtro operar sobre el píxel

especifico en el cual se está realizando el proceso y así, obtener mejoras o datos útiles que pueden ser posteriormente utilizados sobre ella; esto aplica para todos y cada uno de los píxeles propios de la imagen [Castillo et al., 2013]. Dichos filtros se pueden categorizar en filtros lineales (filtros basados en Kernels o máscaras de convolución) y filtros no lineales.

Considerando lo anterior, actualmente dentro de las técnicas o filtros comunes, existen una gran variedad de ellas que buscan restaurar de forma eficiente una imagen corrompida por el ruido de sal y pimienta, el cual se define mediante la función densidad de probabilidad que se presenta a continuación:

$$p(z) = \begin{cases} P_a & \text{para } z = a \\ P_b & \text{para } z = b \\ 0 & \text{para otro } z \end{cases} \quad (3)$$

En la ecuación (3) si $b > a$, intensidad de b se verá reflejada como un punto blanco sobre la imagen y a como un punto negro. Hasta ahora, para el filtrado de imágenes, la técnica no línea denominada filtro de mediana ha sido una de las comunes para eliminar este tipo de ruido debido a la simplicidad y facilidad de implementación a la hora de suavizar imágenes (es decir, reducir la cantidad de variaciones de intensidad entre píxeles vecinos) [Tourounis et al., 2018]. Sin embargo, debido a que este asigna a cada pixel el valor de la mediana local (muestras alrededor de cada valor de la señal $f(x) = y_{med}$), se genera pérdida de los detalles de la imagen (puntos y líneas finas), redondeo de las esquinas de los objetos y desplazamiento de los bordes; además, se destaca la poca efectividad que tiene este filtro al eliminar ruido en altas densidades [Tourounis et al., 2018], [Márquez, 2013].

3.6. Detección de Bordes

En imágenes a escala de grises, la detección de bordes puede realizarse al generar una identificación de los cambios en la intensidad de luz sobre el número de píxeles en evaluación [Wongthanavas, 2011]. Es decir, si se tienen una serie lineal de píxeles con intensidades de: 255, 248, 252, 76, 73, 79, se esperaría un borde o discontinuidad entre los píxeles con intensidades de 252 y 76. Dentro de algunos textos esos píxeles son denominados como “puntos de borde” y estos pueden ser de utilidad al momento realizar el análisis de la imagen.

Dada la importancia de este proceso de análisis digital en las imágenes, a través de los años ha surgido diversos métodos básicos para la detección de bordes, entre los más

destacados se pueden mencionar los siguientes:

3.6.1. Método basado en la primera derivada (gradiente)

En este método la intensidad tiene mayor magnitud a la del umbral predefinido de la imagen. Se buscan los picos más grandes de la imagen.

El gradiente de una imagen en un punto indica la variación máxima de la función en ese punto [Chacón et al., 2015]. Este se encuentra definido por:

$$\nabla f(x, y) = [G_x \ G_y] = \left[\frac{\delta f}{\delta x} \ \frac{\delta f}{\delta y} \right] \quad (4)$$

Donde, $G_x = \frac{\Delta f}{\Delta x}$ y $G_y = \frac{\Delta f}{\Delta y}$

3.6.2. Método basado en la segunda derivada (laplaciano)

La intensidad tiene cruce por cero (0). A diferencia del método de la primera derivada, este no requiere de un valor de umbral para tener una aproximación más efectiva. Se buscan los cambios de signo en donde se encuentra un borde [Chacón et al., 2015]. Su formulación matemática está descrita por:

$$\nabla^2 f(x, y) = \left[\frac{\delta^2 f}{\delta x^2} \ \frac{\delta^2 f}{\delta y^2} \right] \quad (5)$$

En la Figura 7 se presenta una relación del comportamiento de estos métodos cuando se detectan variaciones en la intensidad.

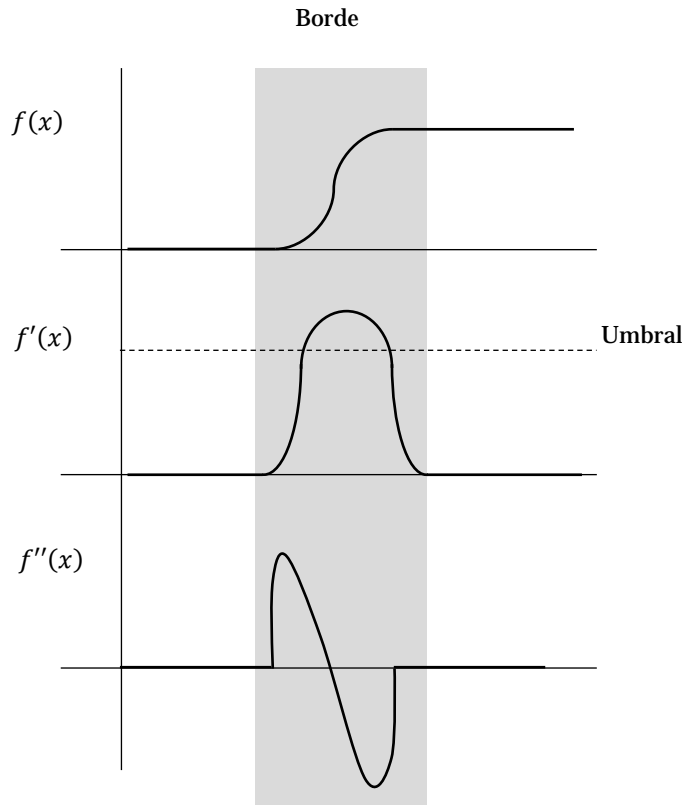


Figura 7: Métodos de la primera y segunda derivada para la detección de bordes. **Fuente:** Elaboración propia.

3.6.3. Método de Canny

Este algoritmo para la identificación de contornos fue desarrollado en 1986 por John F. Canny, presenta mejores resultados que los métodos previamente explicados, aunque la complejidad computacional es más alta [Chacón et al., 2015]. Este algoritmo se basa en tres (3) criterios básicos:

- **Detección:** Evita la eliminación de bordes relevantes, así como también la generación de falsos bordes que pueden perjudicar el resultado final.
- **Localización:** Estipula que la distancia entre la posición real del borde y la generada, debe ser reducida.
- **Respuesta única:** Expresa que el algoritmo debe retornar un pixel de borde por cada pixel verdadero, es decir que no deben existir grupos de pixeles donde solo debe existir uno.

3.7. Funciones morfológicas

Estos métodos y técnicas, son comúnmente usados para la identificación y/o extracción de componentes dentro de la misma, como etapa de post-procesamiento de una imagen digital, porque normalmente en las labores de segmentación de una imagen, se suelen presentar imperfecciones, bien sea por la aparición de píxeles mal clasificados o por regiones incorrectamente delimitadas [Platero, 2009].

Desde la perspectiva esencial, esta es una técnica de procesamiento no lineal que se basa en la teoría de conjuntos para realizar la forma de los objetos de una imagen. Además, su funcionamiento se basa en un elemento estructural que se utiliza de acuerdo con la necesidad [Platero, 2009]. El elemento estructural define la forma y tamaño de la vecindad que será utilizado en las operaciones morfológicas, las más relevantes son: cuadrado, rombo, círculo, línea recta, octágono y rectángulo [Tosina, 2008].

3.7.1. Operaciones morfológicas

Actualmente existen varias operaciones las cuales pueden ser combinadas, con el ánimo de realizar una labor específica de acuerdo con las necesidades que se tengan [Tosina, 2008]. En esta investigación, se utilizaron principalmente 4 de ellas, las cuales son las siguientes.

Nota: para la explicación de las siguientes operaciones, se utilizará como base una imagen de 20×16 píxeles (ver Figura 8), y las operaciones se llevaran a cabo con un elemento estructural cuadrado de 2 píxeles.



Figura 8: Imagen base (20x16 píxeles). **Fuente:** Elaboración propia.

- **Erosión:** elimina píxeles del contorno de la imagen.



Figura 9: Aplicación de erosión. **Fuente:** Elaboración propia.

- **Dilatación:** se agregan píxeles al contorno de la imagen.



Figura 10: Aplicación de dilatación. **Fuente:** Elaboración propia.

- **Apertura:** Consiste en realizar una dilatación y luego una erosión utilizando el mismo elemento estructural. Sirve para eliminar pequeñas imperfecciones y suavizar bordes.



Figura 11: Aplicación de apertura. **Fuente:** Elaboración propia.

- **Cierre:** Se ejecuta primero una erosión y posteriormente una dilatación con el mismo elemento estructural. Se utiliza para rellenar píxeles contiguos y definir contornos.



Figura 12: Aplicación de cierre. **Fuente:** Elaboración propia.

3.8. Juego de la vida

Este autómata celular fue desarrollado por el matemático Británico John Horton Conway en 1970. Desde el ámbito científico, ha servido como pauta para el desarrollo de un gran número de investigaciones, debido al potencial y complejidad que desarrolla de acuerdo con un grupo de reglas sencillas [Lahoz-Beltra, 2010].

Dentro de lo que contempla este trabajo, este autómata celular fue útil ya que permitió reducir el ruido que se generaba al intentar identificar los contornos de las imágenes. Del mismo modo, la lógica que contiene el algoritmo, se consideran las celdas vivas (encendidas) como borde, y las celdas muertas (apagadas) no son bordes [Lahoz-Beltra, 2010].

Originalmente, se establecieron tres (3) reglas para su funcionamiento. No obstante, para términos de este trabajo se realizó un ajuste a las mismas con el fin de mejorar los resultados obtenidos con las iteraciones del autómata celular. Las reglas originales son las siguientes:

- Una celda muere, si tiene solo un (1) vecino vivo (aislamiento) o más de tres (3) vivos a su alrededor (sobrepoblación).
- Una celda se mantiene viva, si tiene dos (2) o tres (3) vecinos vivos (supervivencia).
- Una celda nace, si tiene exactamente tres (3) vecinos vivos (nacimiento).

En particular, para la investigación se utilizaron una adaptación de las reglas originales definidas por Conway, estas se presentan a continuación:

1. **Soledad:** el pixel deja de considerarse borde cuando tiene uno (1) o dos (2) vecinos vivos.

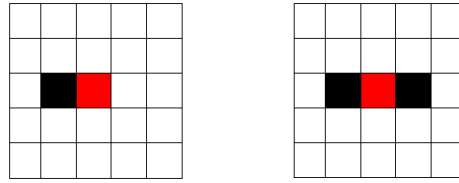


Figura 13: Soledad - Regla del juego de la vida. **Fuente:** Elaboración propia.

2. **Sobrepoblación:** un pixel de borde muere cuando tiene ocho (8) vecinos vivos.

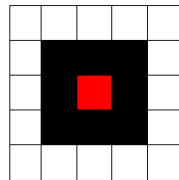


Figura 14: Sobrepoblación – regla del juego de la vida. **Fuente:** Elaboración propia.

3. **Supervivencia:** Un pixel considerado borde se mantiene así, si sus tiene tres (3), cuatro (4), seis (6) y siete (7) vecinos vivos

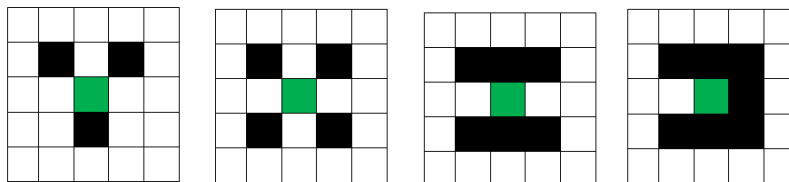


Figura 15: Supervivencia – regla del juego de la vida. **Fuente:** Elaboración propia.

4. **Nacimiento:** Un pixel que no es borde, nace cuando tiene exactamente cinco (5) vecinos vivos

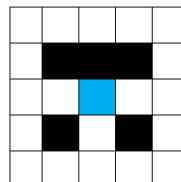


Figura 16: Nacimiento – regla del juego de la vida . **Fuente:** Elaboración propia.

Parte III

Desarrollo

Capítulo 4

Desarrollo del proyecto

4.1. Introducción

El desarrollo del presente proyecto involucró diferentes pasos que permitieron la construcción y simulación de los algoritmos. Estas etapas serán explicadas en este capítulo, con el fin de mostrar la evolución de estas a través de la investigación. Inicialmente se realizó un análisis del comportamiento del ruido impulsivo y de las diferentes metodologías que existen para su tratamiento, luego de identificar que los autómatas celulares son una de las opciones adecuada, se realizó una revisión de trabajos que contemplaran el procesamiento de imágenes desde este campo. Posteriormente, se realizaron los primeros experimentos y acercamientos para identificar los aspectos esenciales, en donde se pudieran realizar aportes significativos en los resultados. Durante esta misma etapa, se revisaron otras metodologías que, al combinarlas con los autómatas celulares, permitieran alcanzar de una manera más eficiente los objetivos del estudio. Sin embargo, la mejora propuesta resultó de la necesidad propia del autómata de obtener más información de su entorno cuando los resultados que ofrece no satisfacen las expectativas. Por lo cual se analiza el comportamiento adaptativo de un autómata y se realiza su implementación, donde efectivamente se pueden notar mejoras significativas. Después, se ajustan los detalles necesarios y se presentan los resultados esperados.

Con la primera fase ya finalizada, se considera la propuesta de aportar un valor agregado a la investigación, con lo cual surge la idea de adicionar un algoritmo para la detección de bordes. Por lo cual fue necesario realizar un proceso similar al de la

eliminación de ruido, donde se realizó la revisión de trabajos previos, experimentos iniciales, refinamiento de detalles y presentación de resultados. Cabe resaltar que al igual que el algoritmo de eliminación de ruido, la detección de bordes también presenta un comportamiento adaptativo de acuerdo a las necesidades que se presenten como se presentará en el siguiente capítulo.

En la Figura 17, se muestra un vistazo general de las características que contempla esta investigación.

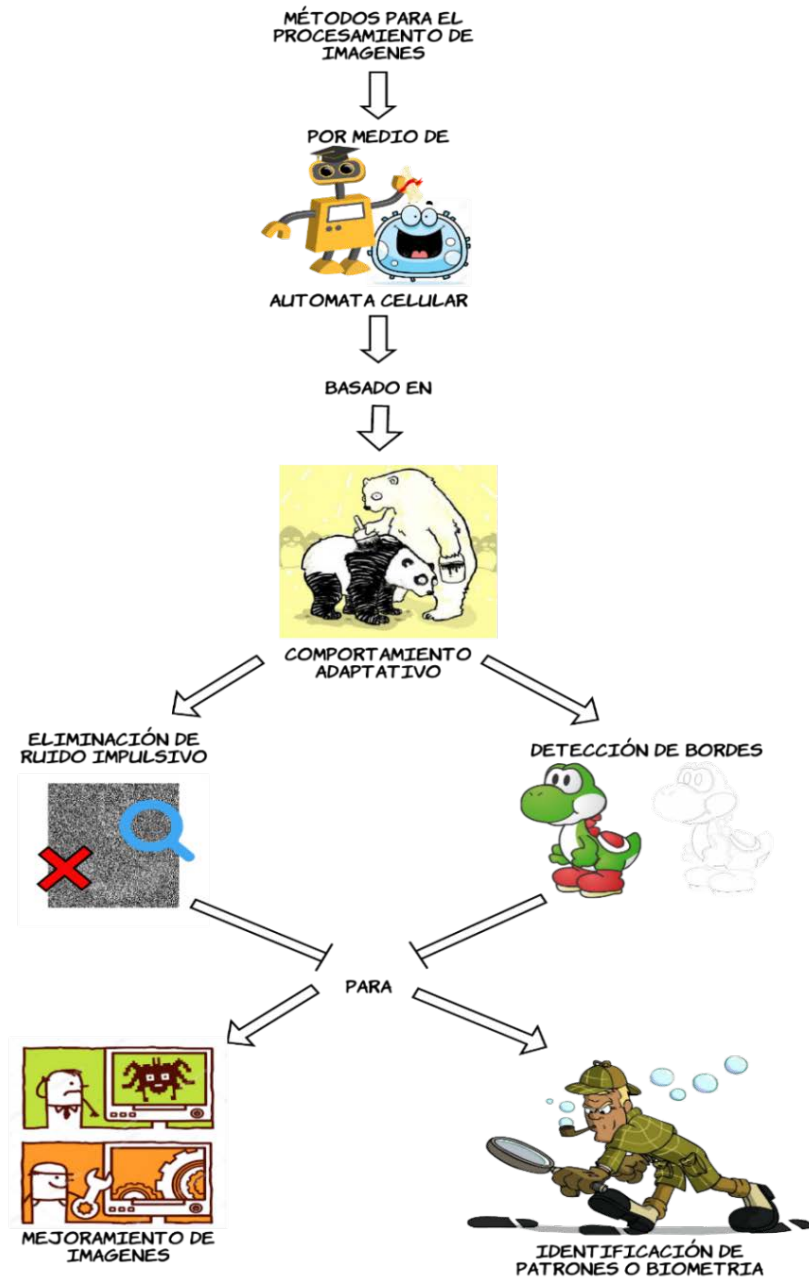


Figura 17: Esquema del funcionamiento de los métodos propuestos. Fuente. Elaboración propia.

4.2. Elementos incidentes en el desarrollo del algoritmo

4.2.1. Ruido

Dentro del área del procesamiento de imágenes, existen diferentes tipos de ruido que puede afectar una imagen digital (impulsional, gaussiano, de disparo, luminancia, crominancia, etc.), por motivo, fue necesario definir el tipo de ruido que se trató durante la investigación. Puesto que hasta el momento no existe algún algoritmo o metodología que permita el tratamiento de ruido sin importar su tipo.

Es por esto, que las investigaciones realizadas por otros autores repercutieron favorablemente en esta investigación puesto que la comparación de resultados otorga una base sólida para determinar la eficiencia y eficacia de los algoritmos presentes en este trabajo. Teniendo en cuenta que existen diferentes variables al momento de reducir o eliminar el ruido en imágenes, fue necesario definir la escala de intensidades de la imagen sobre la cual se iba a trabajar (grises, RGB, binario, etc), ya que se deben considerar diversos aspectos de acuerdo con la escala, es importante mencionar que no todos los tipos de ruido presentan la misma repercusión en la imagen. En consecuencia, fue seleccionado el ruido impulsivo de tipo sal y pimienta, por su presencia frecuente y habitual en imágenes digitales y porque es una temática que ha sido comúnmente abordada en diferentes investigaciones. En conjunto con el ruido impulsivo, la aplicación en imágenes a escala de grises, ha sido base de muchas investigaciones, por su fácil implementación e identificación de resultados.

4.2.2. Autómatas celulares y comportamiento adaptativo

La organización de la información en una imagen digital puede esquematizarse como una matriz, como se ha mencionado anteriormente, donde cada pixel representa un espacio en la matriz y los colores se representan con un valor dentro de la misma, es decir, que cada valor dentro de la matriz representa intensidades dentro de la imagen, es por esto que, un autómata celular resulta ser una de las técnicas más compatibles para el procesamiento de imágenes. Adicionalmente, estos modelos matemáticos tienen la capacidad de realizar considerables actividades con una codificación relativamente sencilla y de bajo costo computacional.

Por otro lado, dentro de los tipos de autómata celular el vecindario de Moore fue considerado como la opción adecuada para atender los requerimientos que se fueran

presentando al recorrer la imagen, considerando que el vecindario de Von Neumann contempla menor información para la toma de decisiones. Dentro de las primeras experimentaciones, los resultados con el vecindario de Moore fueron más satisfactorios que los obtenidos con la vecindad de Von Neumann.

A medida que fueron realizando experimentos, en diferentes con diferentes variables de entrada, la información con la que contaba el autómata, era insuficiente y, en consecuencia, los resultados no eran eficientes. En virtud de lo anterior, surgió la idea de otorgarle al autómata un comportamiento adaptativo, donde la vecindad de Moore se expandiera cuando considera que no contaba con suficiente información para tomar una decisión, tanto para eliminar ruido como para definir bordes.

4.3. Análisis de los algoritmos base e implementación

Los primeros acercamientos que se llevaron a cabo en la investigación, consistieron en el conocimiento básico de los autómatas celulares, y en la replicación de los algoritmos propuestos por diferentes autores. Cabe resaltar, que la replicación de otros trabajos cuenta con un margen de error, debido a que comúnmente los autores no presentan explícitamente el algoritmo resultante o las reglas propuestas o información vital para la replicación del algoritmo, lo cual hizo necesario tomar como base descripciones generales, modelos matemáticos, diagramas de flujo y pseudocódigos.

En el Diagrama 1 se puede ver claramente, mediante un diagrama de causalidad el desarrollo del algoritmo para la eliminación de ruido en imágenes a escala de grises. En este se ilustra la dinámica del modelo presentado, para simplificar la comprensión y posterior liberación del mismo.

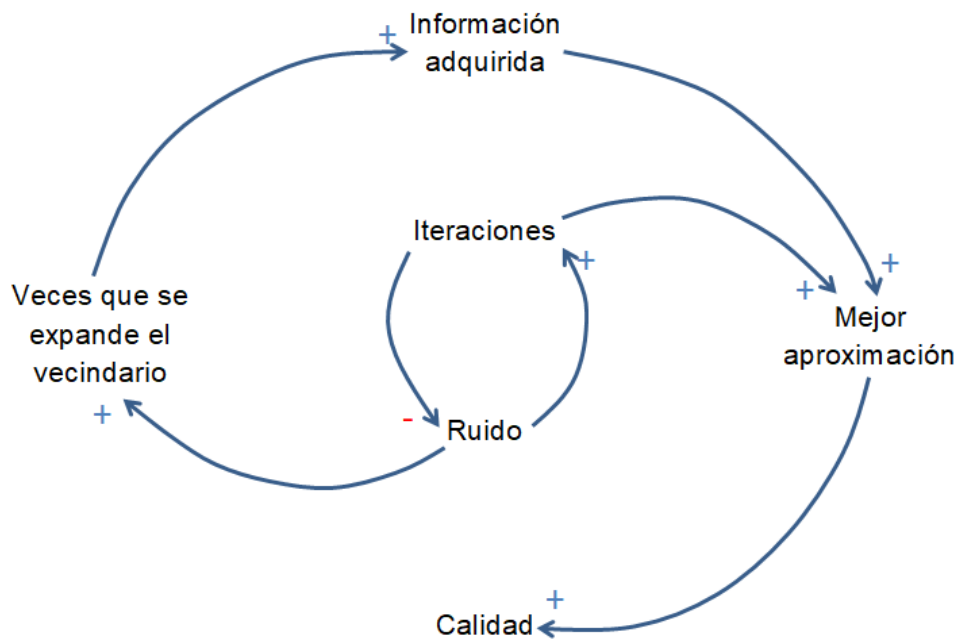


Diagrama 1: Causalidad para la eliminación de ruido. **Fuente:** Elaboración propia.

Es por ello, que en algunos casos no fue posible plantear algoritmos que fueran consistentes con los resultados presentados por los autores. Aunque, por otro lado, fue posible conformar una base comparativa sólida con aquellos trabajos que al momento de ser replicados sus resultados se asemejaba a los presentados en sus trabajos investigativos.

4.3.1. Experimentos realizados

El primer acercamiento que se dio en la investigación, exceptuando el de la etapa de aprendizaje donde se evaluó el comportamiento de los autómatas y sus características básicas, se basó en el concepto de la moda aritmética, en el cual el autómata al detectar un valor máximo o mínimo de intensidad, registraba el valor que más veces se repetía dentro del vecindario circundante al pixel en evaluación y, posteriormente, lo asignaba al pixel en evaluación. Este procedimiento se realizaba iteradamente con el fin de suavizar los bordes existentes y mejorar visualmente los resultados.

No obstante, el mayor problema de este algoritmo era que inicialmente el algoritmo consideraba los pixeles ruidosos dentro del cálculo, y, en consecuencia, la posibilidad de asignar pixeles corruptos al pixel en evaluación incrementaba proporcionalmente de acuerdo al nivel de ruido.

A continuación, en las Figuras 18, 19 y 20 se presentan las imágenes resultantes al aplicar la implementación del algoritmo descrito anteriormente, en porcentajes de ruido 10%, 50% y 90%, respectivamente. La parte (a) de la figura es la imagen con ruido y la parte (b) es el resultado de la imagen.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 18: Eliminación de ruido, intento número 1. (a) Imagen original con ruido al 10%. (b) Imagen resultante.
Fuente. Elaboración propia.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 19: Eliminación de ruido, intento número 1. (a) Imagen original con ruido al 50%. (b) Imagen resultante.
Fuente. Elaboración propia.

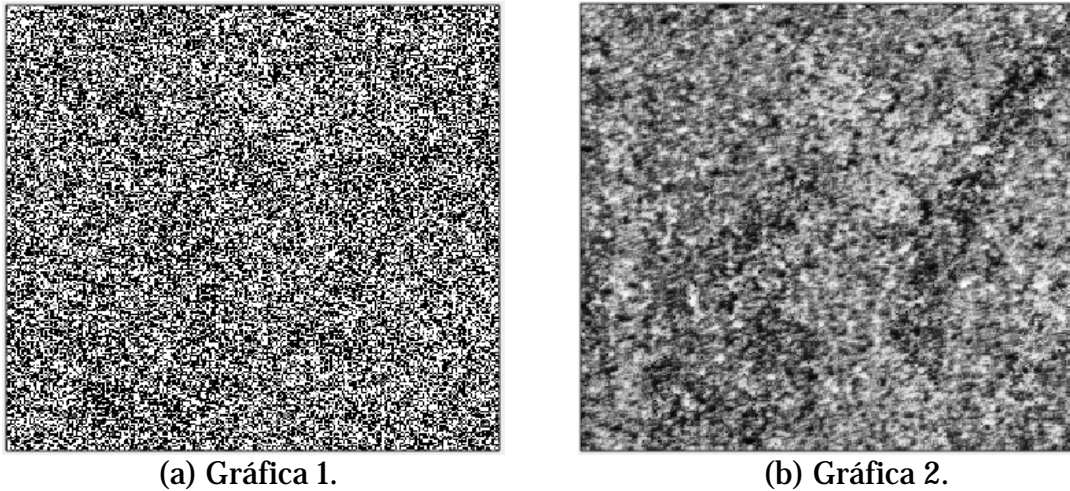


Figura 20: Eliminación de ruido, intento número 1. (a) Imagen original con ruido al 90%. (b) Imagen resultante.
Fuente. Elaboración propia.

Luego de realizar diversos ajustes y experimentar continuamente, se decidió modificar la parte esencial del autómata, ya que en ningún momento los resultados esperados fueron los deseados. En la siguiente prueba, se consideró modificar la esencia del algoritmo para que el autómata no tuviera como base la moda aritmética, en su lugar, el autómata celular hiciera uso del concepto de la mediana, ya que varios autores consultados, lo utilizaban como fundamento para la eliminación de ruido impulsivo, e incluso diferentes softwares actuales utilizan esta metodología para el filtrado de imágenes digitales.

Para alcanzar este objetivo se le confirió al autómata la capacidad de ordenar los elementos, esta función se logró acomodando los vecinos no ruidosos del pixel en evaluación (máximo ocho (8) elementos) en un arreglo y posteriormente, realizando el recorrido del mismo con el algoritmo de ordenamiento de burbuja (Bubble sort, en inglés), el cual al recorrer el arreglo compara e intercambia las posiciones de los elementos por pares hasta acomodarlos en su respectivo orden. Al tener el arreglo debidamente organizado, el valor asignado el pixel en evaluación fue aquel que hubiese quedado en la posición central, en caso de que el tamaño del arreglo fuera impar, o en el caso de que fuera par, era asignado el menor de los dos (2) valores centrales.

Al terminar la ejecución del algoritmo, fue evidente notar que esta labor incrementaba la complejidad computacional, ya que en el peor caso se tendrían que hacer sesenta y cuatro (64) comparaciones e intercambios $\theta(n^2)$, por tal razón, esta posibilidad fue rechazada. Los resultados obtenidos se muestran en las Figuras 21, 22 y 23.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 21: Eliminación de ruido, intento número 2. (a) Imagen original con ruido al 10%. (b) Imagen resultante.
Fuente. Elaboración propia.

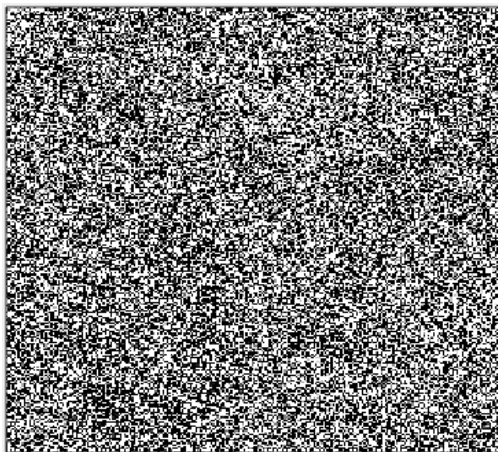


(a) Gráfica 1.



(b) Gráfica 2.

Figura 22: Eliminación de ruido, intento número 2. (a) Imagen original con ruido al 50%. (b) Imagen resultante.
Fuente. Elaboración propia.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 23: Eliminación de ruido, intento número 2. (a) Imagen original con ruido al 90%. (b) Imagen resultante.
Fuente. Elaboración propia.

Después de realizar y experimentar en variaciones del algoritmo inicial, la investigación se encaminó hasta tomar como base el algoritmo propuesto en el artículo “Salt-n-pepper noise filtering using Cellular Automata” [Tourounis et al., 2018]. En este algoritmo, se reemplaza el pixel corrupto por el valor promedio obtenido por medio del valor de intensidad de sus vecinos, descartando aquellos que al igual que el pixel central son considerados como ruido, ya que representan los valores máximos y mínimos en la escala de intensidad (0 o 255). Las imágenes después de aplicado el algoritmo se presentan a continuación:



(a) Gráfica 1.



(b) Gráfica 2.

Figura 24: Eliminación de ruido, intento número 3. (a) Imagen original con ruido al 10%. (b) Imagen resultante.
Fuente. Elaboración propia.

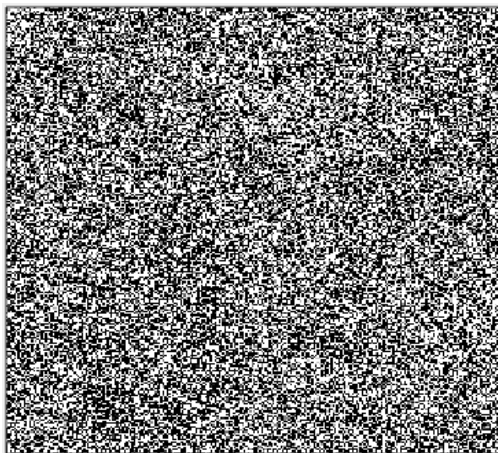


(a) Gráfica 1.



(b) Gráfica 2.

Figura 25: Eliminación de ruido, intento número 3. (a) Imagen original con ruido al 50%. (b) Imagen resultante.
Fuente. Elaboración propia.



(a) Gráfica 1.



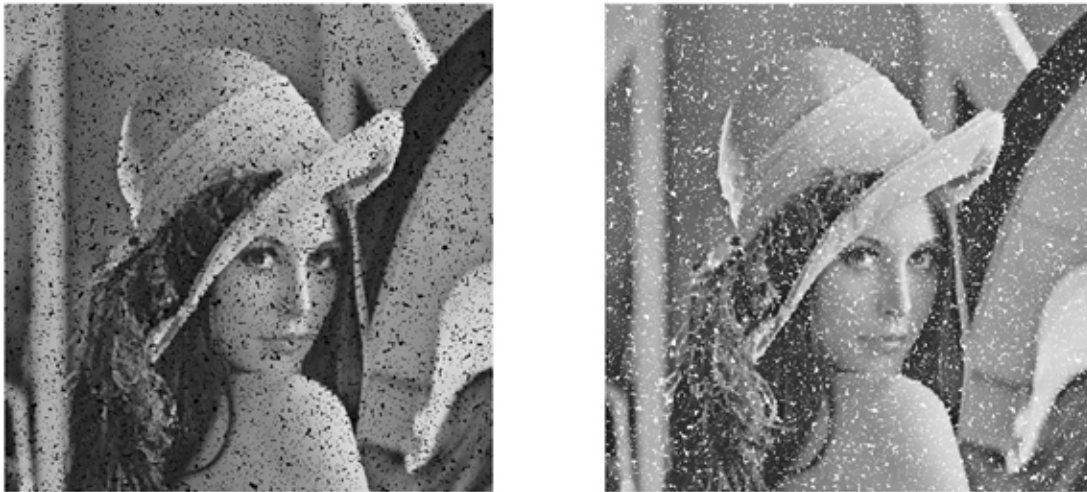
(b) Gráfica 2.

Figura 26: Eliminación de ruido, intento número 3. (a) Imagen original con ruido al 90%. (b) Imagen resultante.
Elaboración propia.

Como se puede evidenciar en las Figuras 24, 25 y 26, el autómata remueve la totalidad del ruido. A pesar de que la calidad de la imagen se ve gravemente afectada debido a las reglas establecidas para el funcionamiento del autómata.

Desde otro punto de vista, se propone la combinación de técnicas para el filtrado de ruido, ya que es bastante común que los diferentes investigadores utilicen diversas metodologías que permitan obtener mejor calidad en la imagen. De tal forma que una opción viable para llevar a cabo esta labor, fue mediante el uso de funciones morfológicas

(ver sección 3.7) puesto que son bastante utilizadas para la depuración de ruido por su facilidad de implementación. Sin embargo, la adaptación de esta técnica no resulto adecuada para la investigación, ya que las pruebas realizadas mostraron resultados poco convenientes. Como los mostrados en la Figura 27, donde las operaciones morfológicas no eliminan adecuadamente el ruido.



(a) Gráfica 1.

(b) Gráfica 2.

Figura 27: Eliminación de ruido con funciones morfológicas. (a) Imagen original con ruido al 50%. (b) Imagen resultante. **Fuente.** Elaboración propia.

4.4. Desarrollo del modelo

Desde un sentido formal, en este trabajo se propone un método basado en autómatas celulares con el cual se busca remover ruido impulsivo en imágenes digitales representadas en escala de gris, teniendo como referencia el método propuesto por Tourtounis, Mitianaudis y Sirakaulis en [Tourtounis et al., 2018], el cual recorre la imagen píxel a píxel y haciendo uso del concepto de vecindad propuesto por Moore, genera una matriz de 3×3 que tiene como centro el píxel en evaluación y dependiendo del estado del mismo y de sus vecinos, toma una decisión la cual puede o no modificarlo.

Una de imagen de dos dimensiones (2D), se puede representar como una matriz de $m \times n$, donde cada posición corresponde a un píxel dentro de la imagen y los valores asignados a dichas posiciones hacen referencia a la intensidad de la imagen en ese punto en particular. Asimismo, se considera que cada celda del autómata celular es un píxel dentro de la imagen; por lo tanto, las dimensiones del autómata celular es $n = 2$ y el número de sus vecinos es igual a 8 lo cual indica que tanto el ancho y el largo son iguales,

es decir $l_1 = l_2$ En el caso de las condiciones de frontera $C(i_0, j_0)$, se extienden la primera y última fila y columna, de tal forma que cuando el autómata se posicione en los límites de la imagen no se afecte el resultado final del proceso.

Teniendo en cuenta los conceptos previamente revisados, a través de la definición de vecindad de Moore (M) de un autómata celular (denominado N) en la celda $C(i_0, j_0)$, entonces el AC se puede definir como una vecindad de tamaño fijo de 3×3 , que es usado en toda imagen, el cual se encuentra descrito por la siguiente ecuación:

$$N(i_0, j_0)^M = \{(i, j): |i - i_0| \leq r, |j - j_0| \leq r\} \quad (6)$$

Donde r , es el rango y es igual a uno (1), es de señalar que para una imagen en escala de grises cada píxel puede tomar valores discretos entre 0 y 255, los cuales son empleados para establecer los valores de estado del autómata celular; de esta forma se tiene que el valor mínimo y máximo de los estados son $estadoMin = 0$ y $estadoMax = 255$.

En general, la regla 2D local para el autómata celular propuesto corresponde a:

$$C^{t+1}(i, j) = \begin{cases} C^t(i, j), & \text{si } 0 < C^t(i, j) < 255 \\ Cn^t(i, j), & \text{si } C^t(i, j) = 0 \text{ ó } C^t(i, j) = 255 \end{cases} \quad (7)$$

Como se muestra en la ecuación (7), el valor $C^{t+1}(i, j)$ de la celda del autómata celular en la posición $C^t(i, j)$ puede tomar dos (2) valores dependiendo de su estado, para el nuevo valor se realiza un cálculo con base a una sub-regla que se describe como se muestra a continuación:

$$Cn^t(i, j) = \begin{cases} C(2n + 1)2 - 1, & \text{si } \forall C^t(i, j) \in N, C^t(i, j) \neq 0 \text{ ó } C^t(i, j) \neq 255 \text{ y } N = \emptyset \\ \text{promedio}(C^t(i, j)), & \text{si } \forall C^t(i, j) \in N, \exists C^t(i, j) \neq 0 \text{ ó } \exists C^t(i, j) \neq 255 \end{cases} \quad (8)$$

Teniendo presente la propuesta del autómata celular y el enfoque de la aplicación del mismo, el cual únicamente actúa sobre píxeles que son considerados como ruidosos, se crea una regla para el autómata celular, la cual logra la localización de píxeles que son determinados como ruidosos o no, tal como se describió en las ecuaciones (6) y (7).

Como se ha mencionado, la imagen al ser afectada por el ruido de tipo sal y pimienta toma valores máximos o mínimos discretos en cada píxel afectado. Esto quiere decir que

al aplicar la ecuación (6) solo se verán afectados aquellos píxeles donde su valor es máximo (255 o blanco) o mínimo (0 o negro).

Al ser considerado un píxel como ruidoso, se aplica la sub-regla descrita en la ecuación (8), en el caso donde el vecindario de Moore solo tenga valores máximos o mínimos en su estructura, se adaptará y se extenderá, tal como se aparecía en Figura 5 (vecindario extendido de Moore) y el estado de la celda será el promedio con la información adicional excluyendo los valores máximos y mínimos. De lo contrario, se eliminarán todos los valores 0 y 255 y se aplicará un promedio sobre los restantes para obtener el nuevo valor de la celda.

Por otro lado, cuando el estado de la celda del autómata celular no sea igual a ninguno de los valores considerados como umbrales, esta celda no será considerada como una celda ruidosa y, en consecuencia, mantendrá su valor inicial.

Todo el autómata celular es evaluado un número finito de veces dependiendo del valor del ruido aplicado, es decir que si el nivel de ruido es igual a n , el autómata celular iterará $\left(\frac{n}{10}\right) + 1$ veces.

De forma general, en el Diagrama 2, se presenta de forma gráfica la funcionalidad del Algoritmo Propuesto; donde existen cuatro (4) regiones relevantes. En la primera región, se verifica que la imagen se encuentre en escala de grises, en el caso de que no sea así, se realiza el proceso de conversión a esta escala. En la segunda región, el autómata evalúa si el píxel en evaluación es ruidoso o no, en el caso afirmativo, se plantea otra condición en la cual se verifica el número de vecinos ruidosos, teniendo en cuenta que, si estos superan en cinco (5) el número, se ejecutara la ampliación de vecindario. Adicionalmente, se realiza el cálculo de la media aritmética en el caso donde el píxel en evaluación sea considerado como ruidoso. En la tercera región, se corrobora que el autómata haya recorrido la totalidad de la imagen pixel por pixel.

Finalmente, en la última región se hace realiza una verificación con el fin de identificar si aún existen píxeles corruptos, ya que en el caso de ser cierto, se ejecuta una nueva iteración hasta cumplir $\left(\frac{n}{10}\right) + 1$ veces, donde n , es el nivel de ruido.

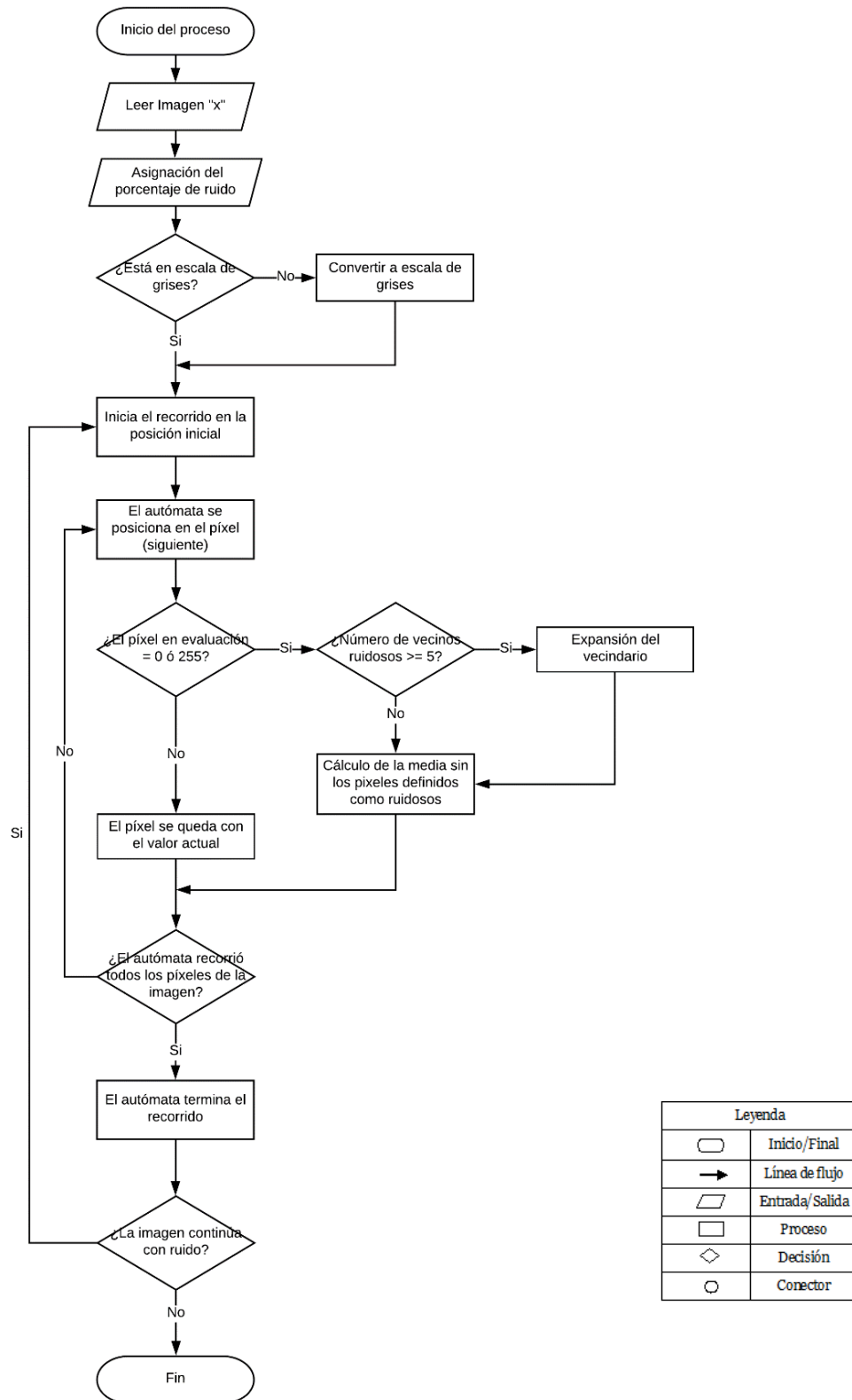


Diagrama 2: Diagrama de flujo para la eliminación de ruido con autómatas celulares. **Fuente:** Elaboración propia.

4.5. Resultados Obtenidos

Luego de realizar diferentes experimentos con diversos ajustes, los resultados que se obtuvieron no fueron satisfactorios con altos niveles de ruido (el mayor reto de la eliminación de ruido), a causa de que el autómata cuenta menor información para determinar el valor adecuado del pixel en evaluación. Por esta razón, surgió la propuesta de expandir el vecindario de Moore con el fin de que el autómata contara con mayor información a la hora realizar su labor.

Sin embargo, no siempre es necesario que el vecindario se expanda, dado que la información necesaria para obtener un buen resultado, puede provenir de los vecinos adyacentes. Además, si el autómata siempre tuviera el vecindario expandido, podría perjudicar la calidad de la imagen.

Por consiguiente, se decidió experimentar con la idea de un autómata que expandiera su vecindario cuando considere que sea necesario. Es decir, cuando el autómata detecta un pixel corrupto y analiza la cantidad de vecinos ruidosos, puede expandir su vecindario si la mayoría de vecinos no le aportan la información necesaria para corregir el pixel en evolución.

Dentro de los acercamientos más efectivos que resultaron hasta este punto de la investigación estaban todos donde el autómata se basaba en la media aritmética. De modo que resultaba adecuado asignarle el comportamiento adaptativo a este tipo de autómata.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 28: Aplicación del Algoritmo Propuesto sin ajustar las variables de entrada, imagen con 10% de ruido. (a) Imagen original con ruido al 10%. (b) Imagen resultante. **Fuente.** Elaboración propia.

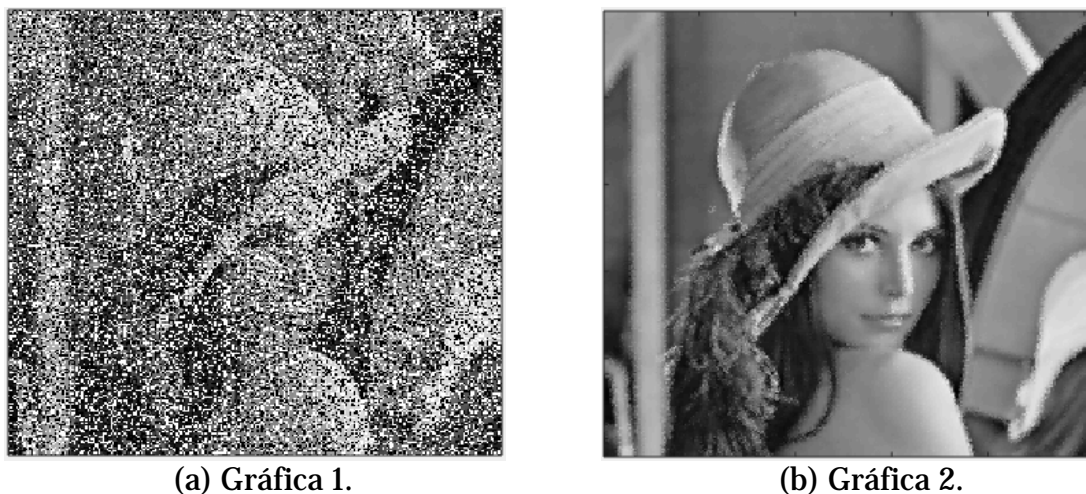


Figura 29: Aplicación del Algoritmo Propuesto sin ajustar las variables de entrada, imagen con 50% de ruido. (a) Imagen original con ruido al 50%. (b) Imagen resultante. **Fuente.** Elaboración propia.

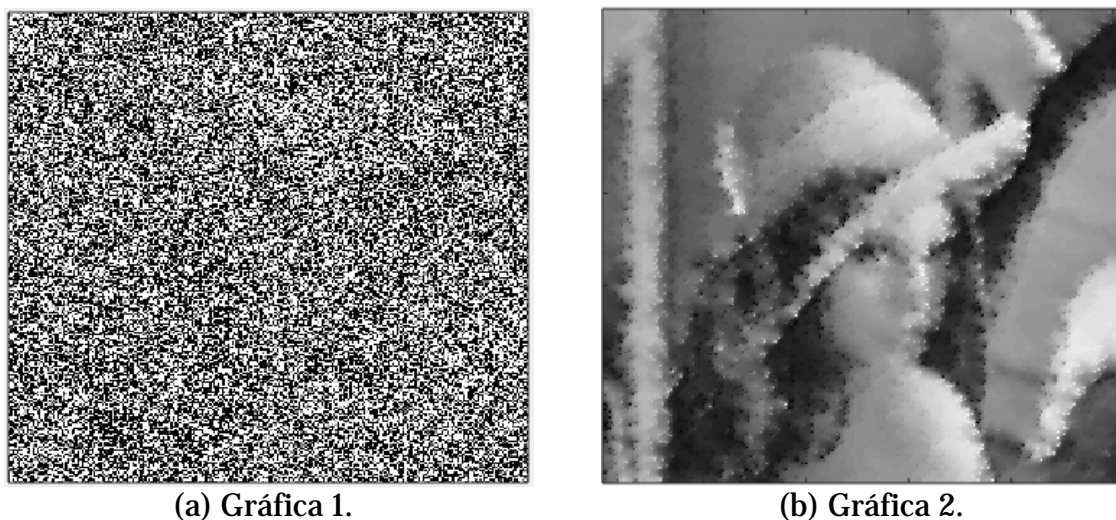


Figura 30: Aplicación del Algoritmo Propuesto sin ajustar las variables de entrada, imagen con 90% de ruido. (a) Imagen original con ruido al 90%. (b) Imagen resultante. **Fuente.** Elaboración propia.

Como se puede apreciar en las Figuras 28, 29 y 30, la calidad de la imagen mejora sustancialmente con el algoritmo adaptativo, sobre todo en los resultados donde el ruido supera el nivel del 50%.

Por otra parte, se realizaron ajustes a los parámetros del algoritmo con el fin de obtener mejores resultados y que la disposición del pixel en evaluación abarcara diferentes

características antes de tomar una decisión, como por ejemplo el análisis de los vecinos que más predominan dentro del conjunto y proporcionarle una mayor relevancia, o la cantidad de iteraciones necesarias que se deben realizar antes de alcanzar un buen resultado, ya que en diferentes ocasiones se afectó el resultado al realizar excesivas iteraciones para mejorar la calidad. En las Figuras 31, 32 y 33 se presentan los resultados obtenidos.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 31: Algoritmo final propuesto, imagen con 10% de ruido. (a) Imagen original con ruido al 10%. (b) Imagen resultante. **Fuente.** Elaboración propia.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 32: Algoritmo final propuesto, imagen con 50% de ruido. (a) Imagen original con ruido al 50%. (b) Imagen resultante. **Fuente.** Elaboración propia.

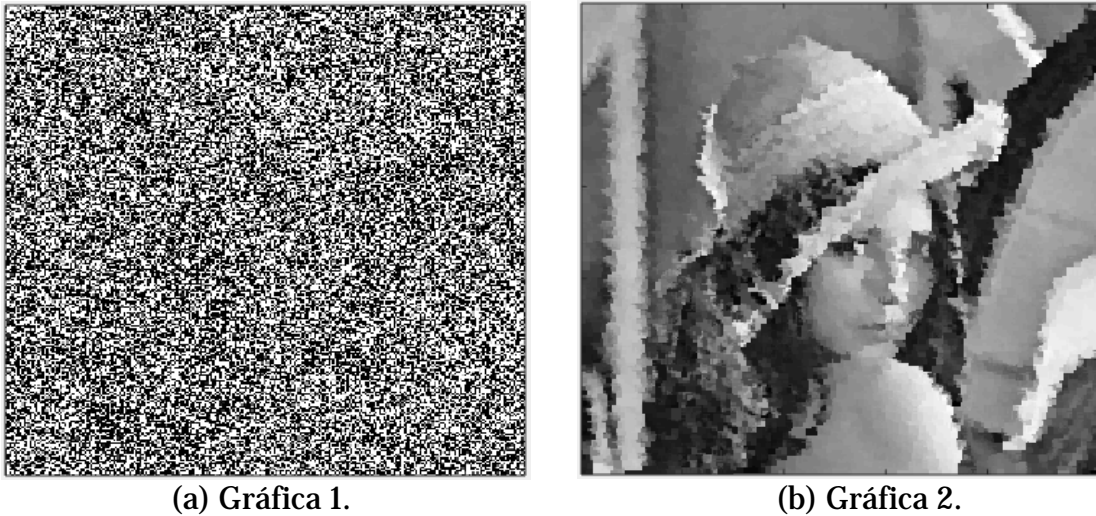


Figura 33: Algoritmo final propuesto, imagen con 90% de ruido. (a) Imagen original con ruido al 90%. (b) Imagen resultante. **Fuente.** Elaboración propia.

Como se puede evidenciar, en la Figura 33, la imagen con 90% de ruido no cuenta con mucho suavizado, pero se pueden identificar de una forma más fácil las características principales de la imagen, a diferencia de las figuras presentadas de los otros algoritmos con el mismo porcentaje de píxeles corruptos.

4.6. Análisis de los resultados

Para el análisis de los resultados obtenidos en la sección 4.5, se empleó el Índice de Similitud Estructural (Structural Similarity Index o SSIM), el cual es un indicador de la calidad de una imagen digital. Es usado para medir la similitud de dos (2) imágenes, para determinar qué tan diferente es la imagen original frente a la distorsionada. Para esto se calculan tres (3) índices que comparan los términos de luminancia (l), contraste (c) y estructura (s) de un par de imágenes. La combinación de los tres (3) términos corresponde al SSIM mediante la ecuación (9).

$$SSIM(x, y) = [l(x, y)]^\alpha \cdot [c(x, y)]^\beta \cdot [s(x, y)]^\gamma \quad (9)$$

Dónde:

$$l(x, y) = \frac{2\mu_x 2\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad (10)$$

$$c(x, y) = \frac{2\sigma_x \sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \quad (11)$$

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x \sigma_y + C_3} \quad (12)$$

En estas expresiones μ denota el valor medio, σ la desviación típica y σ_{xy} la covarianza. De la misma forma las variables x , y representan a las imágenes original y distorsionada respectivamente. Las constantes C_i se aplican para evitar la división por cero (0). Los parámetros α , β y γ tienen por defecto el valor de 1, por lo cual al multiplicar los tres (3) índices se obtiene la ecuación (13):

$$lSSIM(x, y) = \frac{(2\mu_x 2\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (13)$$

En la Figura 34 se muestra el esquema para el cálculo del índice SSIM, el cual se realiza de manera automática.

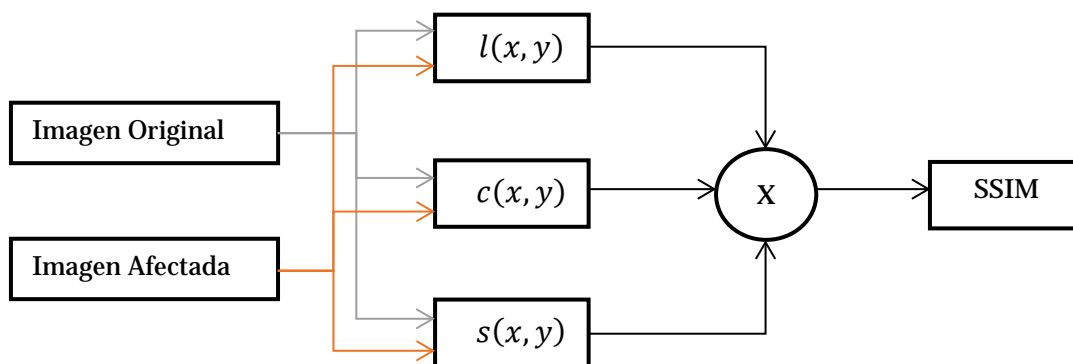


Figura 34: Esquema del índice SSIM. **Fuente.** Elaboración propia.

Teniendo en cuenta lo anterior y con el ánimo de realizar una comparación, se tomaron

como referencia los algoritmos descritos en los trabajos de Abu, Al-Dhamari, Ortega y Alfonseca (Algoritmo 1), Tourtounis, Mitianoudis, y Sirakoulis (Algoritmo 2), Sahin, Uguz y Sahin (Algoritmo 3) y Garnett, Huegerich, Chui y He (Algoritmo 4), los cuales también implementan un algoritmo basado en autómatas celulares para eliminar el ruido en imágenes en escala de grises.

Para evaluar cuantitativamente el desempeño de los algoritmos, se realizó una comparación de las imágenes originales versus las imágenes resultantes de cada uno de los algoritmos, y mediante el índice SSIM se logró establecer una serie de valores con rango entre 0 y 1, donde de cero (0) equivale a la pérdida total de similitud y uno (1) es semejanza absoluta. Los valores son presentados en las Tablas de la 1, 2, 3 y 4.

Babuino (512x512)					
% Ruido	Algoritmo 1	Algoritmo 2	Algoritmo 3	Algoritmo 4	Algoritmo Propuesto
10	0,9265	0,9767	0,9736	0,9717	0,9772
20	0,7973	0,9450	0,9427	0,9176	0,9529
30	0,6567	0,8948	0,9086	0,8196	0,9237
40	0,5287	0,8102	0,8664	0,6896	0,8886
50	0,4180	0,6968	0,8185	0,5503	0,8467
60	0,3161	0,5574	0,7545	0,4118	0,7927
70	0,2333	0,4082	0,6778	0,2883	0,7232
80	0,1560	0,2715	0,5584	0,1817	0,6266
90	0,0908	0,1469	0,3870	0,0958	0,4740

Tabla 1: Resultados SSIM y porcentaje de ruido para la imagen del babuino (512x512).

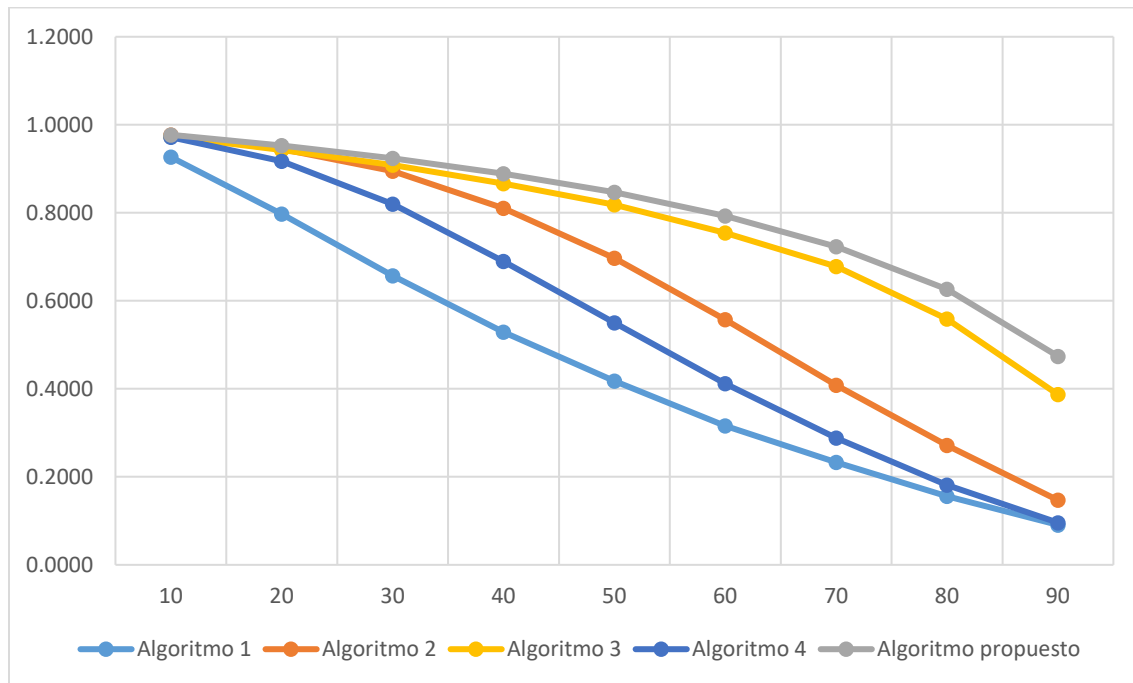


Figura 35: Porcentaje de ruido vs. SSIM para la imagen del babuino de 512 x 512 pixeles. **Fuente.** Elaboración propia.

Como se puede observar en la Figura 35, los Algoritmos 1, 2 y 4 tienen una tendencia a 0, en la medida que el porcentaje de ruido incrementa, lo cual indica que estos algoritmos presentan dificultades al eliminar el ruido cuando este es muy alto. Sin embargo, el Algoritmo Propuesto se afecta en menor medida cuando el nivel de ruido incrementa, aunque también se observa, que después de un nivel superior al 50% de ruido la pendiente se hace más pronunciada, lo cual representa una desmejora en la calidad de la imagen.

Al efectuar el mismo ejercicio con una imagen distinta que a su vez, tenía dimensiones entre sí, los resultados se evidencian en la Tabla 2.

Lenna (512x512)					
% Ruido	Algoritmo 1	Algoritmo 2	Algoritmo 3	Algoritmo 4	Algoritmo Propuesto
10	0,9532	0,9875	0,9877	0,9789	0,9889
20	0,7766	0,9615	0,9737	0,9037	0,9765
30	0,5507	0,9007	0,9573	0,7654	0,9632
40	0,3641	0,7830	0,9380	0,5905	0,9469
50	0,2468	0,6325	0,9128	0,4250	0,9270
60	0,1657	0,4693	0,8798	0,2978	0,8998
70	0,1132	0,3214	0,8228	0,1980	0,8634
80	0,0772	0,2024	0,7102	0,1249	0,8057
90	0,0467	0,1104	0,5088	0,0705	0,7063

Tabla 2: Resultados SSIM y porcentaje de ruido para la imagen de Lenna (512x512).

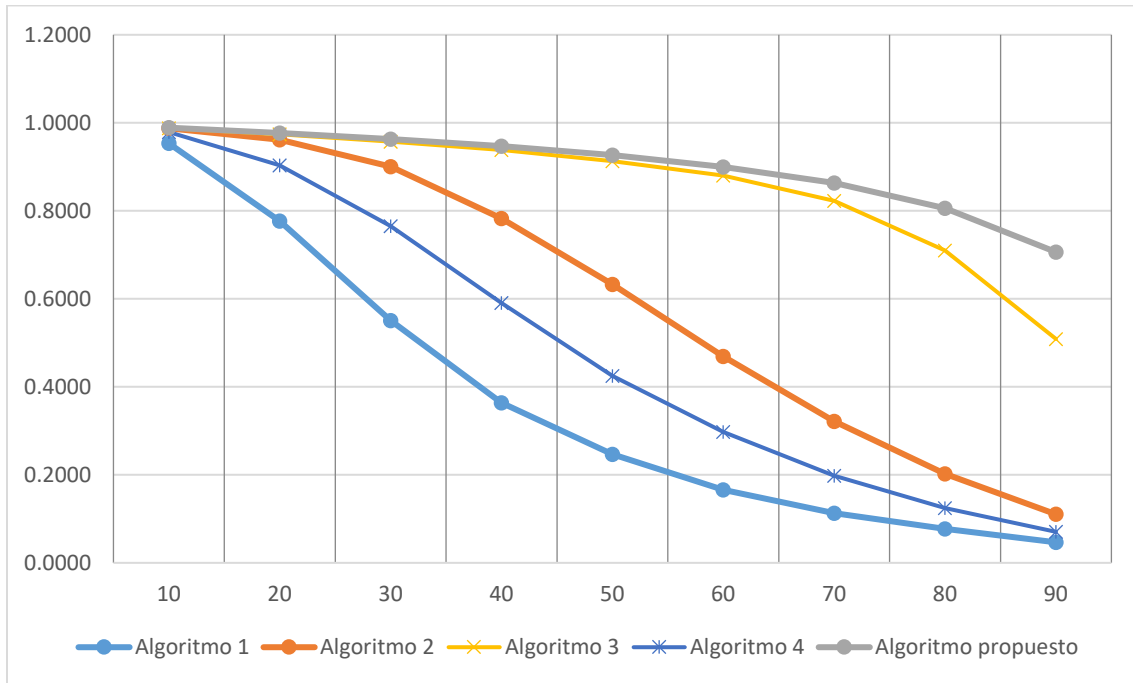


Figura 36: Porcentaje de ruido vs. SSIM para la imagen del Lenna de 512 x 512 pixeles. **Fuente.** Elaboración propia.

Lenna (256x256)					
% Ruido	Algoritmo 1	Algoritmo 2	Algoritmo 3	Algoritmo 4	Algoritmo Propuesto
10	0,9614	0,9875	0,9888	0,9812	0,9897
20	0,8084	0,9642	0,9753	0,9123	0,9784
30	0,6007	0,9062	0,9578	0,7827	0,9659
40	0,4351	0,8000	0,9361	0,6389	0,9499
50	0,3040	0,6671	0,9076	0,4906	0,9318
60	0,2162	0,5252	0,8708	0,3564	0,9010
70	0,1512	0,3828	0,8057	0,2421	0,8592
80	0,1020	0,2501	0,6972	0,1571	0,7943
90	0,0610	0,1417	0,5020	0,0888	0,6628

Tabla 3: Resultados SSIM y porcentaje de ruido para la imagen de Lenna (256x256).

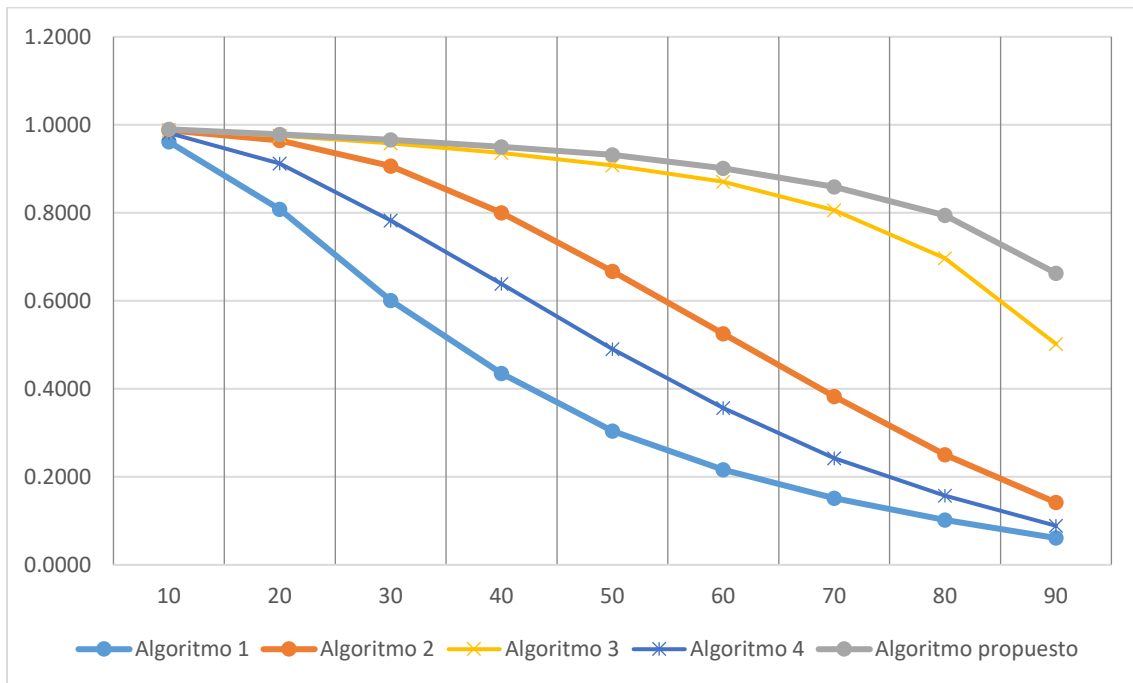


Figura 37: Porcentaje de ruido vs. SSIM para la imagen del Lenna de 256 x 256 píxeles. **Fuente.** Elaboración propia.

La simulación expuesta en las Tabla 1, 2, 3 y en Figura 35, 36 y 37, demostró que la eliminación de ruido es más efectiva en imágenes de mayores dimensiones, puesto que al ejecutar el algoritmo sobre la misma imagen (p.e. Lenna), se obtuvieron mejores resultados, esto se debe a que el autómata celular cuenta con mayor información para apoyarse y realizar mejores aproximaciones en los diferentes niveles de intensidad de la imagen.

Finalmente, para corroborar la hipótesis anteriormente expuesta, se tomó una imagen de 256 x 256 píxeles y los resultados fueron los siguientes:

Cameraman (256x256)					
% Ruido	Algoritmo 1	Algoritmo 2	Algoritmo 3	Algoritmo 4	Algoritmo Propuesto
10	0,9438	0,9767	0,9756	0,9676	0,9782
20	0,7748	0,9391	0,9525	0,8849	0,9582
30	0,5580	0,8741	0,9263	0,7452	0,9314
40	0,3994	0,7412	0,8974	0,5795	0,9062
50	0,2762	0,5867	0,8597	0,4355	0,8753
60	0,2087	0,4383	0,8086	0,3100	0,8337
70	0,1426	0,2990	0,7292	0,2155	0,7799
80	0,0965	0,1994	0,5386	0,1434	0,7100
90	0,0533	0,1129	0,2971	0,0746	0,6197

Tabla 4: Resultados SSIM y porcentaje de ruido para la imagen de cameraman (256x256).

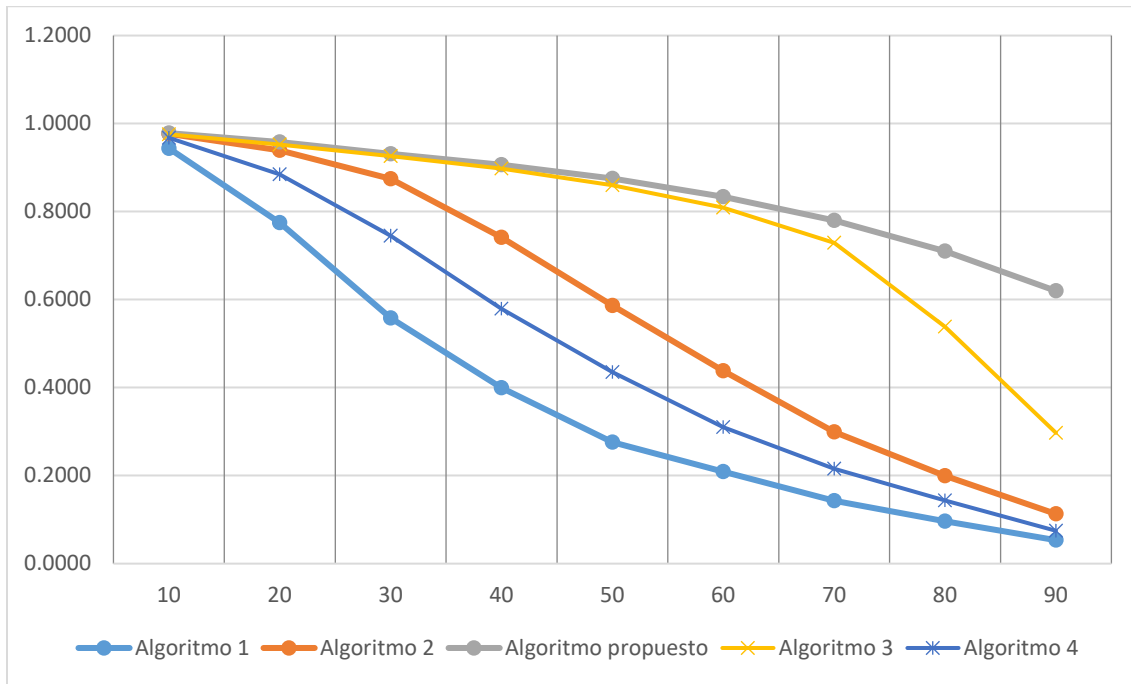
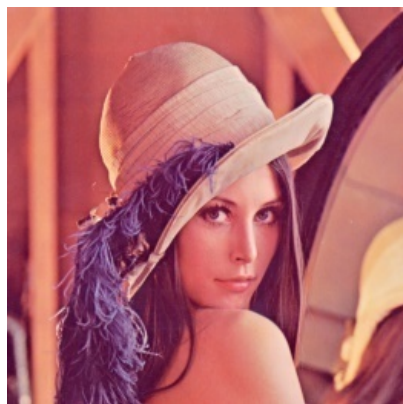


Figura 38: Porcentaje de ruido vs. SSIM para la imagen del cameraman de 256 x 256 píxeles. **Fuente.** Elaboración propia.

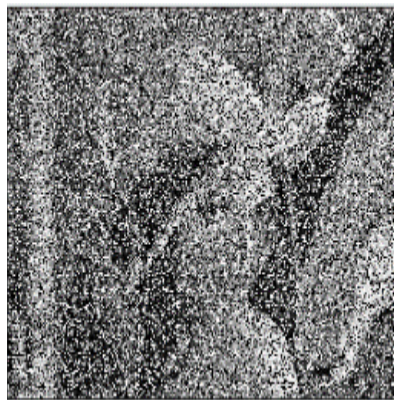
Los resultados expuestos en las Tablas 1, 2, 3 y 4 muestran que Algoritmo Propuesto presenta un mejor desempeño respecto a los que tomados como referencia en los

diferentes niveles de ruido considerados, ya que en ningún punto de comparación el Algoritmo Propuesto fue superado. Es de resaltar que la diferencia con 10% de ruido es mucho menor que la diferencia cuando el porcentaje de ruido supera un nivel de 50%, lo cual se puede apreciar en las Figuras 35, 36, 37 y 38.

Del mismo modo, en las Figuras 39 y 40, se encuentran los resultados gráficos luego de la aplicación de los diferentes algoritmos. Los cuales fueron aplicados sobre dos (2) imágenes con dimensiones de 512×512 pixeles que corresponden a la imagen de babuino (ver Figura A.1., Grafica (a)) y a la imagen de Lenna (ver Figura 39, Grafica (a)). Asimismo, estos fueron aplicados en imágenes con dimensiones de 256×256 pixeles, para este caso Lenna y Cameraman (ver Figura A.3., Grafica (a)).



(a) Gráfica 1.



(b) Gráfica 2.



(c) Gráfica 3.
Algoritmo 1



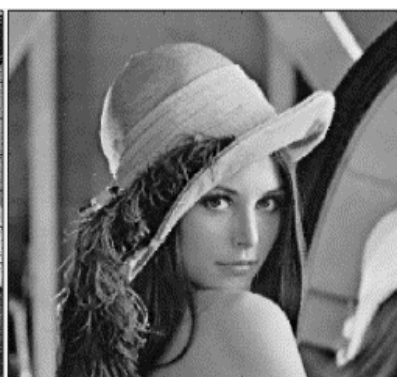
(d) Gráfica 4.
Algoritmo 2



(e) Gráfica 5.
Algoritmo 3



(f) Gráfica 6.
Algoritmo 4

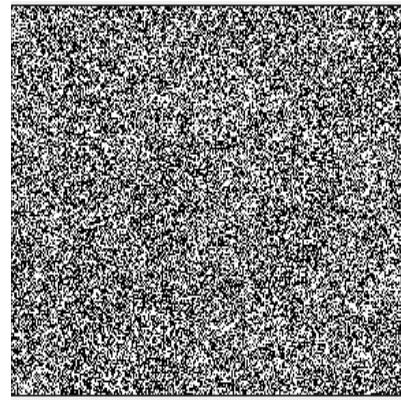


(g) Gráfica 7.
Algoritmo Propuesto

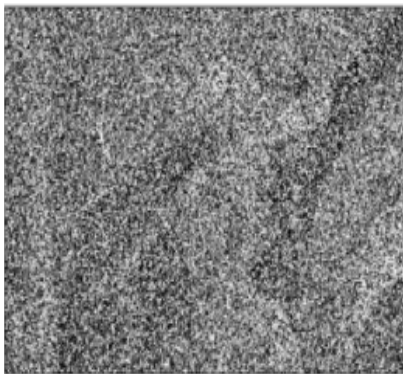
Figura 39: Resultados del algoritmo aplicado en la imagen del lenna (512 x 512 pixeles) con un ruido de 50%. (a) Imagen original. (b) Imagen en escala de grises con el ruido 50%. (c) Imagen resultante del Algoritmo 2. (d) Imagen resultante del Algoritmo 1. (e) Imagen resultante del Algoritmo Propuesto. **Fuente.** Elaboración propia.



(a) Gráfica 1.



(b) Gráfica 2.



(c) Gráfica 3.
Algoritmo 1



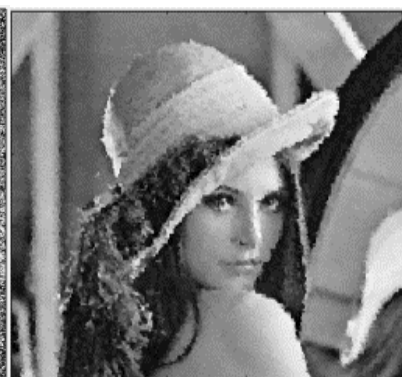
(d) Gráfica 4.
Algoritmo 2



(e) Gráfica 5.
Algoritmo 3



(f) Gráfica 6.
Algoritmo 4



(g) Gráfica 7.
Algoritmo Propuesto

Figura 40: Resultados del algoritmo aplicado en la imagen del lenna (512 x 512 pixeles) con un ruido de 90%. (a) Imagen original. (b) Imagen en escala de grises con el ruido 90%. (c) Imagen resultante del Algoritmo 2. (d) Imagen resultante del Algoritmo 1. (e) Imagen resultante del Algoritmo Propuesto. **Fuente.** Elaboración propia.

4.7. Análisis comparativo sobre la diferencia porcentual

Mediante una comparación porcentual a partir de los resultados de la sección 4.6., se pudo obtener la diferencia porcentual entre el Algoritmo Propuesto frente a cada uno de los algoritmos en comparación (ver Tablas 6 y 7). El cálculo de la diferencia porcentual es producto de la siguiente operación matemática:

$$\text{Diferencia porcentual} = (\text{SSIM Alg. Prop.} - \text{SSIM Alg. Comp.}) \times 100 \quad (14)$$

Los resultados del anterior cálculo, pueden ser clasificados según su eficiencia y desempeño al restaurar una imagen que contiene diversos niveles de ruido, estableciendo como punto de comparación el Algoritmo Propuesto.

Rango (%)	Clasificación
0 – 35	Malo
36 – 70	Aceptable
71 – 90	Bueno
91 – 100	Excelente

Tabla 5: Clasificación cualitativa del desempeño al eliminar ruido.

	SSIM al 50% de Ruido					Diferencia porcentual (Algoritmo Propuesto)			
	Algoritmo 1	Algoritmo 2	Algoritmo 3	Algoritmo 4	Algoritmo Propuesto	Algoritmo 1	Algoritmo 2	Algoritmo 3	Algoritmo 4
Imagen 1	0,4180	0,6968	0,8185	0,5503	0,8467	43%	15%	3%	30%
Imagen 2	0,2468	0,6325	0,9128	0,4250	0,9270	68%	29%	1%	50%
Imagen 3	0,3040	0,6671	0,9076	0,4906	0,9318	60%	52%	2%	44%
Imagen 4	0,2762	0,5867	0,8597	0,4355	0,8753	60%	29%	2%	44%

Tabla 6: Comparación valores SSIM con imágenes que tiene ruido del 50%.

Después de analizar los datos presentados en la Tabla 6, es posible observar que, al comparar el Algoritmo Propuesto frente al Algoritmo 1, se evidencia la mayor diferencia

porcentual, ya que esta oscila entre un 43% y un 68% de mejoría. Lo cual indica que, este algoritmo puede considerarse como el de peor rendimiento al ejecutar los objetivos propuestos, esto por los fallos considerables en el proceso de restauración de los píxeles corruptos dentro de la simulación. Por otra parte, teniendo como punto de comparación el Algoritmo 2, el porcentaje de mejora se encuentra entre un 15% y un 52%, lo que representa una mejora significativa respecto al Algoritmo 1, pero aun así los resultados que se muestran, no son buenos en comparación con el Algoritmo Propuesto.

En la siguiente comparación que se propuso, se puede notar que los resultados obtenidos por el Algoritmo 3 y el Algoritmo Propuesto son bastante semejantes. En estos se puede evidenciar claramente que, desde un análisis netamente cualitativo, las imágenes son bastante similares con excepción de unos cuantos píxeles que el Algoritmo 3 no clasifica correctamente, aun así, esto representa una diferencia máxima de 3% y una mínima de 1%, lo cual lleva a concluir que este algoritmo es el segundo que mejores resultados presento en la simulación después del Algoritmo Propuesto.

Finalmente, el Algoritmo 4, demuestra que sus resultados son en promedio, inferiores respecto a los datos resultantes del Algoritmo 2 y superiores al Algoritmo 1. En síntesis, el Algoritmo Propuesto manifiesta mejores resultados en imágenes con 50% de ruido, frente a sus competidores, los cuales se sitúan porcentualmente en un rango entre 1% y 68% de mejoría.

	SSIM al 90% de Ruido					Diferencia porcentual (Algoritmo Propuesto)			
	Algoritmo 1	Algoritmo 2	Algoritmo 3	Algoritmo 4	Algoritmo Propuesto	Algoritmo 1	Algoritmo 2	Algoritmo 3	Algoritmo 4
Imagen 1	0,0908	0,1469	0,3870	0,0958	0,4740	38%	33%	9%	38%
Imagen 2	0,0467	0,1104	0,5088	0,0705	0,7063	66%	60%	20%	64%
Imagen 3	0,0610	0,1417	0,5020	0,0888	0,6628	60%	52%	16%	57%
Imagen 4	0,0533	0,1129	0,2971	0,0746	0,6197	57%	51%	32%	55%

Tabla 7: Comparación valores SSIM con imágenes que tiene ruido del 90%.

Este punto de comparación desde la perspectiva cualitativa es el que representa una brecha más grande entre los resultados de los diferentes algoritmos (Figura 35, 36, 37 y 38), ya que, normalmente, este nivel de ruido representa el mayor reto para los investigadores especializados en este campo, pues el esfuerzo que debe realizar el autómatas celular tanto para eliminar tal volumen de ruido como para aproximar los valores de intensidad de los píxeles corruptos a su estado original, es significativo.

De acuerdo con la Tabla 7, la comparación hecha con el Algoritmo 1, presenta una variación entre el 38% y el 66% frente al Algoritmo Propuesto, lo cual vuelve a ubicar a

este algoritmo como uno con peores resultados presenta. Para el caso del Algoritmo 2, este ofrece unos resultados que no difieren mucho porcentualmente de los presentados por el Algoritmo 1 aunque el rango de diferencia calculado con el Algoritmo Propuesto se ubica entre un 33% y un 60%.

Nuevamente, el Algoritmo 3 demuestra que es el mejor entre los diferentes puntos de comparación. No obstante, cabe resaltar que los resultados obtenidos con este nivel de ruido (90%) desmejoran entre un 9% y un 32%, lo cual es significativamente inferior a los resultados señalados en la comparación con un nivel de ruido del 50%. Es decir, que a pesar del buen rendimiento presentado en niveles inferiores al 60%, este algoritmo tiene fallas con niveles superiores al 80% de ruido.

Para terminar, el Algoritmo 4 presenta en sus resultados un comportamiento similar al presentado por los Algoritmos 1 y 2, ya que la diferencia porcentual frente al Algoritmo Propuesto se ubica en un rango entre 38% y 64%.

Teniendo en cuenta lo anterior, se evidencia claramente que el Algoritmo Propuesto cuenta con la capacidad de responder de forma más eficiente a la dificultad de clasificar debidamente los pixeles ruidosos. Su comportamiento adaptativo ofrece una solución robusta para situaciones en las que la imagen presenta alto niveles de ruido y comparativamente, muestra una mejoría general entre 9% y 66%.

Capítulo 5

Desarrollo de un algoritmo eliminador de ruido y detector de bordes

5.1. Introducción

Durante la fase de consulta y exploración, fue notorio que la mayoría de autores mantienen un único enfoque de acuerdo con la técnica de procesamiento de imágenes que aborden, esto es independientemente de las metodologías que usen para el desarrollo. Sin embargo, luego de interactuar constantemente con los autómatas celulares, la idea de agrupar dos (2) perspectivas dentro de una misma investigación, marcó la pauta para contemplar una perspectiva más amplia. No obstante, estos autómatas debían prepararse de manera independiente, es decir; primero desarrollarlos cada uno dentro de una sola herramienta de procesamiento de imágenes y luego de obtener los resultados esperados, unificarlos dentro de un mismo algoritmo, para posteriormente obtener el modelo final.

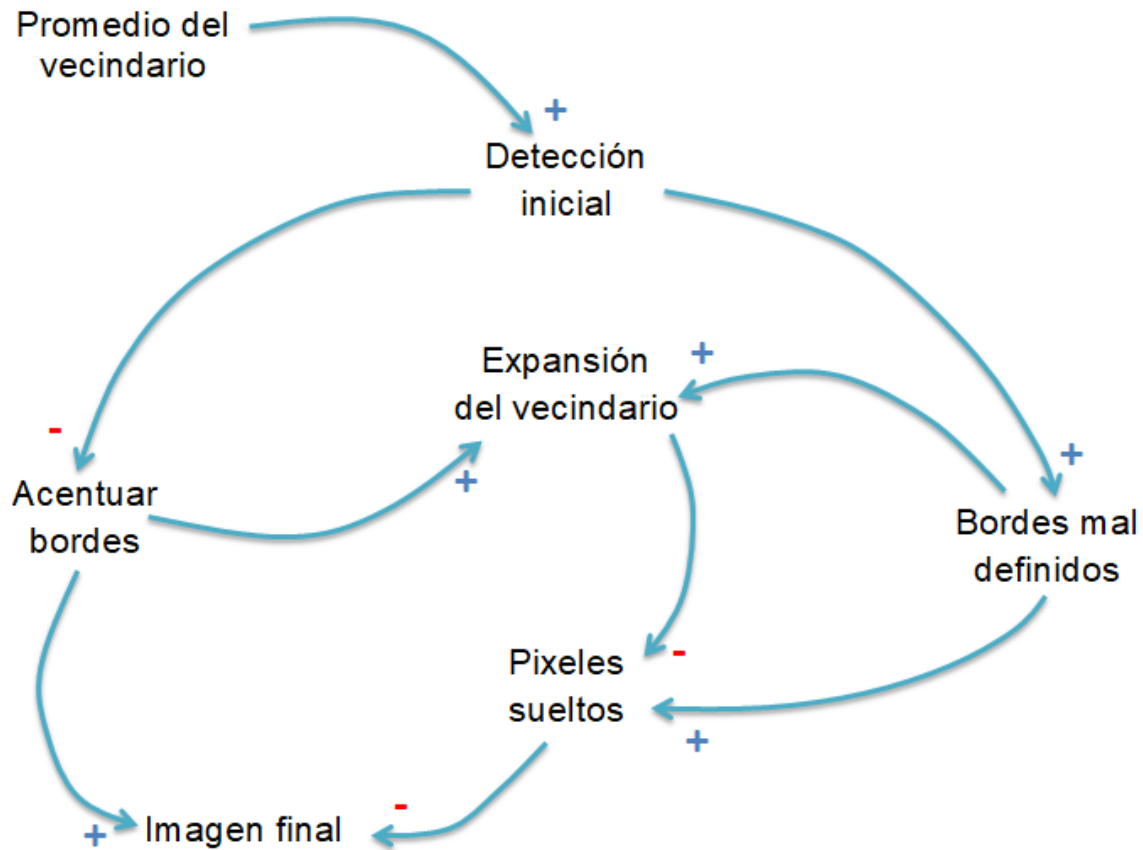


Diagrama 3: Causalidad para la detección de bordes. **Fuente:** Elaboración propia.

En el Diagrama 3 se puede evidenciar, mediante un diagrama de causalidad el desarrollo del algoritmo para la detección de bordes. En este se ilustra la dinámica del modelo presentado, para simplificar la comprensión y posterior liberación del mismo.

5.2. Características en las imágenes digitales

En esta sección, se presenta una de las características de las imágenes que es estudiada constantemente, gracias a que el uso de esta información combinada con distintas técnicas del procesamiento de imágenes, provee una cantidad de información que puede ser útil y gran aprovechamiento en diferentes áreas de conocimiento, pese que hasta el momento no se haya establecido una base de comparación robusta que garantice la objetividad de los resultados.

5.2.1. Contornos

En el campo del procesamiento de imágenes, la detección de bordes es una aplicación relevante y bastante utilizada, gracias a que es considerada como una operación de bajo nivel de procesamiento [Bhardwaj and Mittal, 2012]. Es decir, que un contorno es localizado en la primera operación del algoritmo sobre la imagen, al examinar cada pixel y determinar si este cuenta o no con las características propias de un borde. Así pues, el enfoque principal de esta herramienta es focalizar e identificar discontinuidades nítidas presentes en la imagen [Diwakar et al., 2013]. Dichas discontinuidades, son definidas, por algunos autores, como aquellos cambios abruptos en la intensidad de los pixeles y son la característica principal para encontrar los límites de los objetos que se encuentran inmersas en una imagen. A partir de este concepto surge la necesidad de denotar un borde como aquello que limita y segmenta las distintas regiones u objeto dentro de la imagen en evaluación [Diwakar et al., 2013]. Convirtiendo esto en el primer paso de muchos de los algoritmos de visión de computadora actuales (reconocimiento facial basado en bordes, detección de obstáculos basada en bordes, reconocimiento de objetivos basado en bordes, compresión de imágenes, entre otros).

El reto principal de este aspecto en la investigación, se centra en realizar diferentes tareas dentro de una imagen en escala de grises. Dado que algunos investigadores, utilizan técnicas de transformación de la para facilitar la labor de la detección de bordes, por ejemplo, la binarización, donde la imagen toma solo dos (2) intensidades (blanco y negro) y el algoritmo simplemente identifica las variaciones de intensidad y define el borde. Es por este hecho que se examinaron diferentes métodos para hacer de la labor de la detección de bordes algo más que una simple transformación de las intensidades a unos (1's) y ceros (0's), lo cual puede llevar a generar distintas posibilidades para la simulación de este proceso. Dentro las posibilidades más relevantes se encuentran la transformación a cuatro niveles y escala de grises.

5.3. Análisis de los acercamientos a la detección de bordes

Desde el inicio de la investigación se contemplaron dos (2) vertientes con gran potencial para generar impacto en el ámbito del procesamiento de imágenes, por un lado, se encontraba la eliminación de ruido y por otro la detección de bordes. Las cuales, al contemplar en retrospectiva se consideran posibles tareas que los autómatas celulares

pueden llevar a cabo de una manera eficaz y eficiente sobre una imagen digital, pero claramente estas tienen objetivos finales totalmente diferentes.

5.3.1. Experimentos realizados

Los primeros experimentos realizados, se efectuaron sobre imágenes binarizadas (convertidas a blanco y negro), ya que esto facilitaba la labor de identificar las diferencias de intensidad entre píxeles, y así poder definir los bordes de la imagen. En este proceso se realizó la binarización calculando la suma de todos los valores de los píxeles de la imagen en escala de grises y luego, se realizó el cálculo del valor del umbral dividiendo la suma calculada en el paso anterior por el número total de píxeles. Como último paso, sobre todos los píxeles de la imagen en escala de grises y se asignó el valor de uno (1) a la posición actual, si el valor de nivel de gris es mayor o igual al valor de umbral, de lo contrario asigna cero (0).



(a) Gráfica 1.



(b) Gráfica 2.

Figura 41: Aplicación del Algoritmo Propuesto basado en una imagen binarizada cameraman 512x512 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

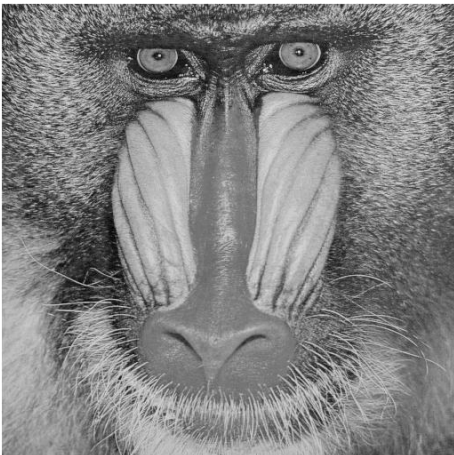


(a) Gráfica 1.



(b) Gráfica 2.

Figura 42: Aplicación del Algoritmo Propuesto basado en una imagen binarizada lenna 512x512 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 43: Aplicación del Algoritmo Propuesto basado en una imagen binarizada babuino 512x512 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

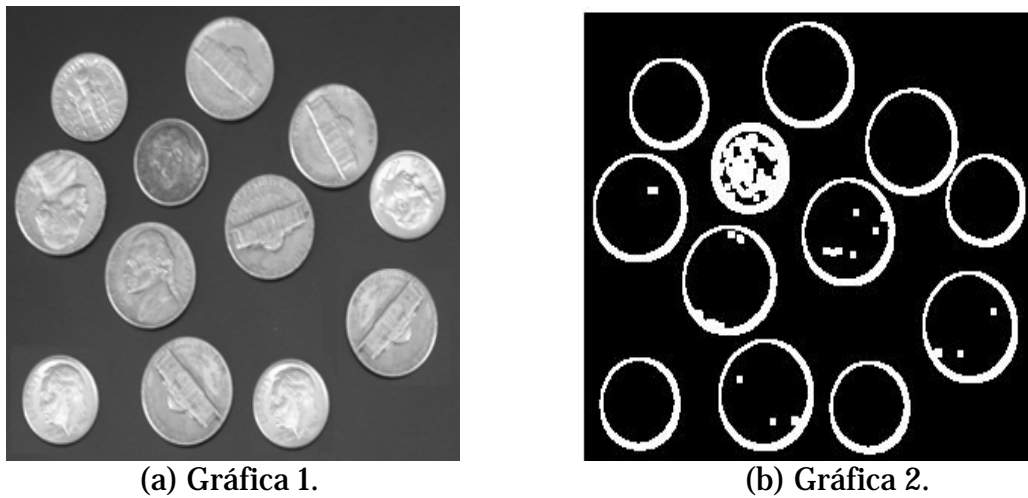


Figura 44: Aplicación del Algoritmo Propuesto basado en una imagen binarizada monedas 256x256 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

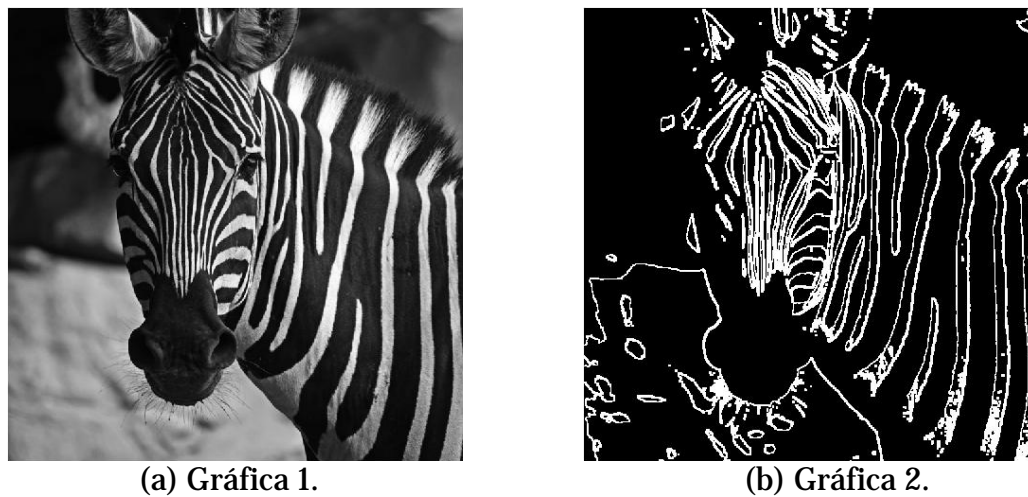


Figura 45: Aplicación del Algoritmo Propuesto basado en una imagen binarizada cebra 256x256 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

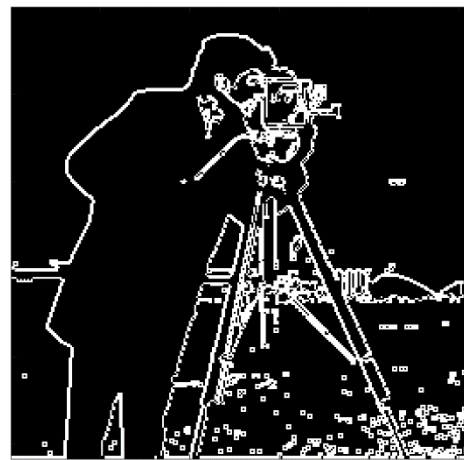
De acuerdo con las Figuras 41, 42, 43, 44 y 45, se puede apreciar que las siluetas de los objetos presentes en la escena fueron identificadas, sin embargo, no se puede considerar como un buen acercamiento, ya que se pierden muchas características relevantes de la imagen durante el proceso de binarización y de igual manera, vale la pena destacar que también se resaltaron otras cualidades poco relevantes de la imagen.

Posteriormente, se considera que el autómata realizara tanto la labor de binarizar la imagen como la de identificar los bordes. Además, dentro del algoritmo, se establecieron

parámetros similares a los planteados por John Conway en el juego de la vida, tales como a) si una célula tiene menos de dos (2) vecinos, muere – soledad, b) si una célula tiene más de ocho (8) vecinos, muere - sobre la población, c) si una célula tiene tres (3), cuatro (4), seis (6) o siete (7) vecinos vivos, continúa viviendo – felicidad y d) si una célula tiene exactamente cinco (5) vecinos, cobra vida – reproducción. Al ejecutar el algoritmo se obtuvieron los siguientes resultados expresado en las Figuras 46, 47, 48, 49 y 50. En donde se aprecia que existe una mejora en la calidad de los resultados.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 46: Aplicación del Algoritmo Propuesto basado en el juego de la vida, cameraman 512x512 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

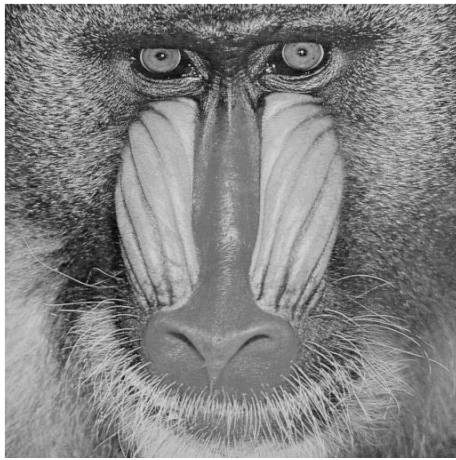


(a) Gráfica 1.



(b) Gráfica 2.

Figura 47: Aplicación del Algoritmo Propuesto basado en el juego de la vida, lenna 512x512 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.



(a) Gráfica 1.

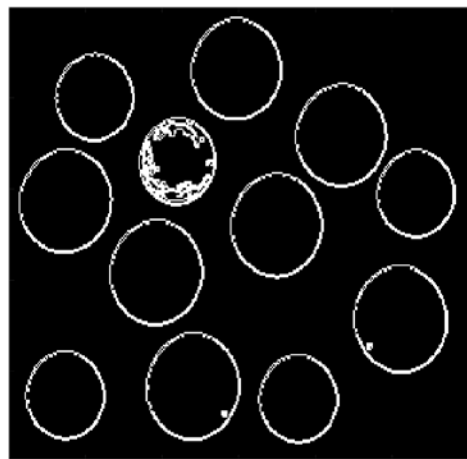


(b) Gráfica 2.

Figura 48: Aplicación del Algoritmo Propuesto basado en el juego de la vida, babuino 512x512 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 49: Aplicación del Algoritmo Propuesto basado en el juego de la vida, monedas 256x256 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

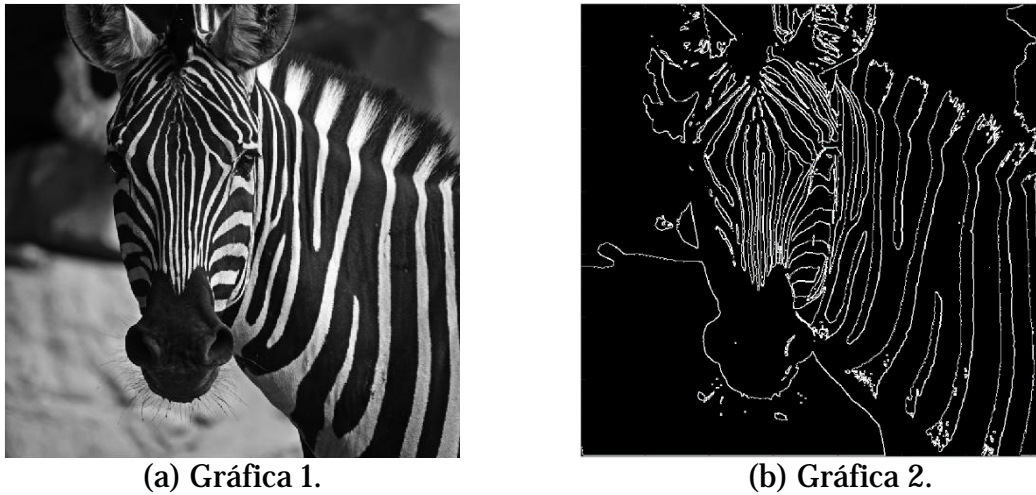


Figura 50: Aplicación del Algoritmo Propuesto basado en el juego de la vida, cebra 256x256 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

Teniendo en cuenta que, los resultados mejoraron significativamente, aún se presentan diferentes fallos en la identificación de detalles que deberían tener contorno, pese a eso, esta técnica aportó información que sería utilizada más adelante, para el refinamiento de detalles y la obtención de bordes más consistentes.

En el artículo “Cellular Automata for Medical Image Processing” [Wongthanavas, 2011], una imagen que está en escala de grises no requiere obligatoriamente pasar por un proceso de binarización. Desde una perspectiva general, el algoritmo funciona clasificando cada píxeles en uno de los cuatro (4) grupos de intensidad, seguidamente el autómatas consulta información proveniente de sus vecinos con el fin de saber si el píxel en evaluación se debe considerar como borde o no, ya que identifica los saltos entre los niveles de intensidad. Las imágenes resultantes de este proceso, se pueden apreciar en las Figuras 51, 52, 53, 54 y 55.



(a) Gráfica 1.

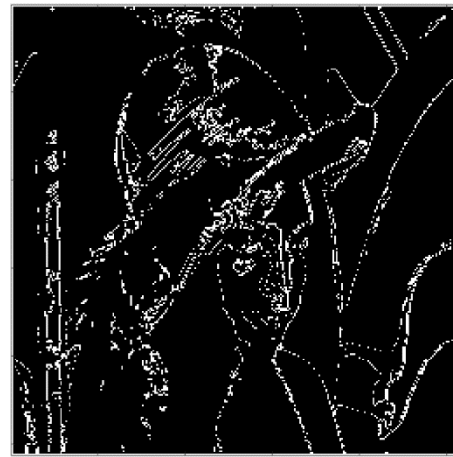


(b) Gráfica 2.

Figura 51: Aplicación del Algoritmo Propuesto basado 4 niveles, cameraman 512x512 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

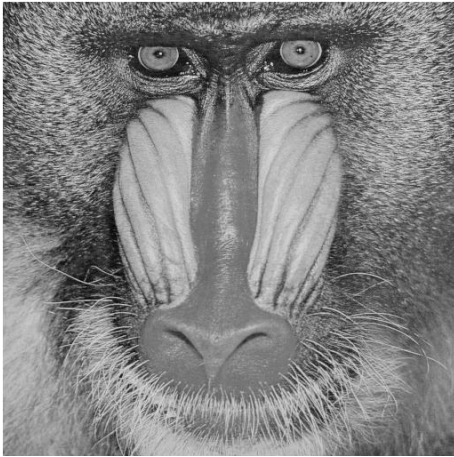


(a) Gráfica 1.



(b) Gráfica 2.

Figura 52: Aplicación del Algoritmo Propuesto basado 4 niveles, lena 512x512 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.



(a) Gráfica 1.

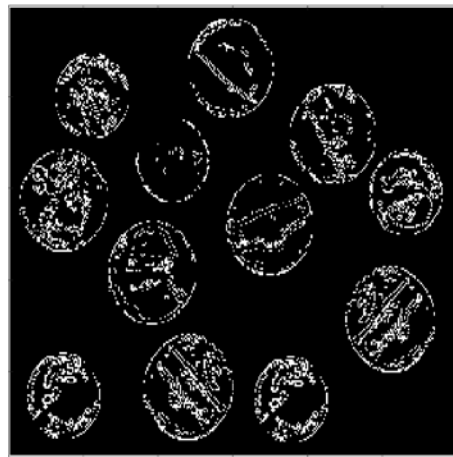


(b) Gráfica 2.

Figura 53: Aplicación del Algoritmo Propuesto basado 4 niveles, babuino 512x512 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.



(a) Gráfica 1.



(b) Gráfica 2.

Figura 54: Aplicación del Algoritmo Propuesto basado 4 niveles, monedas 256x256 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

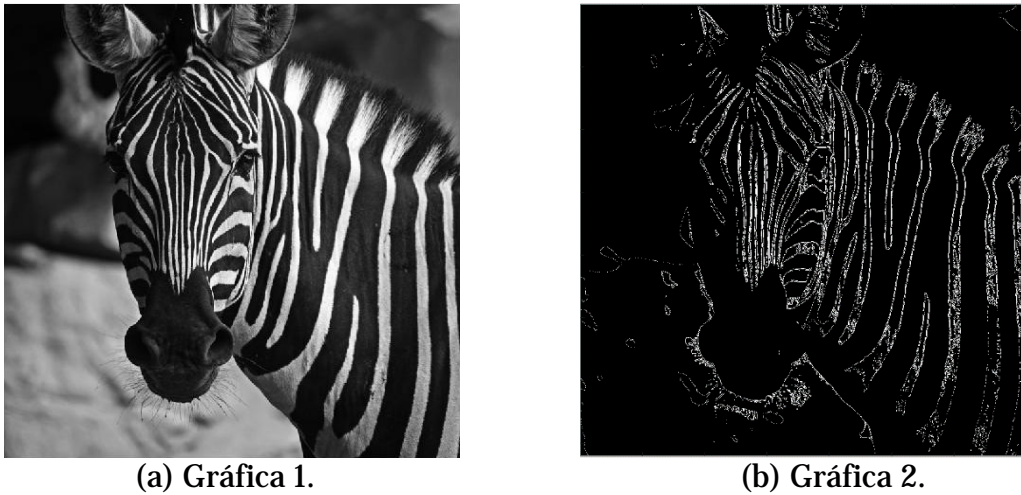


Figura 55: Aplicación del Algoritmo Propuesto basado 4 niveles, cebra 256x256 px. (a) Imagen original. (b) Imagen resultante. **Fuente.** Elaboración propia.

Si bien el autómata ofrece una solución aceptable, el mayor problema radica en la distribución los cuatro (4) niveles, ya que estos deben ser dinámicos de acuerdo con la imagen asignada, dado que, si se estandarizan estos niveles, los resultados dependerán de la imagen y no del funcionamiento propio del algoritmo.

Al realizar una evaluación visual de lo esperado frente a lo mostrado, el algoritmo presenta fallos en la definición de los bordes, puesto que en varias situaciones creaba bordes falsos, ya que consideraba algunos detalles en la imagen que no deberían ser resaltados.

Teniendo como guía y apoyo los distintos experimentos realizados, se decidió trabajar con imágenes en escala de grises, igualando así las condiciones del algoritmo de eliminación de ruido. No obstante, esto representaba un mayor reto, puesto que se tenía que plantear una manera de clasificar los 256 niveles de intensidad en grupos, de forma tal que el autómata celular identificara los bordes de los detalles relevantes de la imagen, considerando que, al no realizar de manera eficiente esta tarea, se generaría ruido en el resultado.

Con base en lo anterior, surgió la idea de combinar metodologías que permitieran principalmente omitir los píxeles que en el resultado se consideraran como ruido y conservar los contornos relevantes de la imagen. Así pues, se realizaron experimentos con dos (2) posibilidades, en ambas, inicialmente el autómata realiza una iteración para identificar los bordes y en la segunda pasada, en uno de los casos se utilizan las reglas planteadas en el juego de la vida y en otro la moda aritmética. Los resultados se presentan en las Figuras 56, 57, 58, 59 y 60.



(a) Gráfica 1.

(b) Gráfica 2.

(c) Gráfica 3.

Figura 56: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, cameraman 512x512 px. (a) Imagen original. (b) Imagen resultante 2da iteración juego de la vida. (c) Imagen resultante 2da iteración moda aritmética. **Fuente.** Elaboración propia.



(a) Gráfica 1.

(b) Gráfica 2.

(c) Gráfica 3.

Figura 57: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, lenna 512x512 px. (a) Imagen original. (b) Imagen resultante 2da iteración juego de la vida. (c) Imagen resultante 2da iteración moda aritmética. **Fuente.** Elaboración propia.

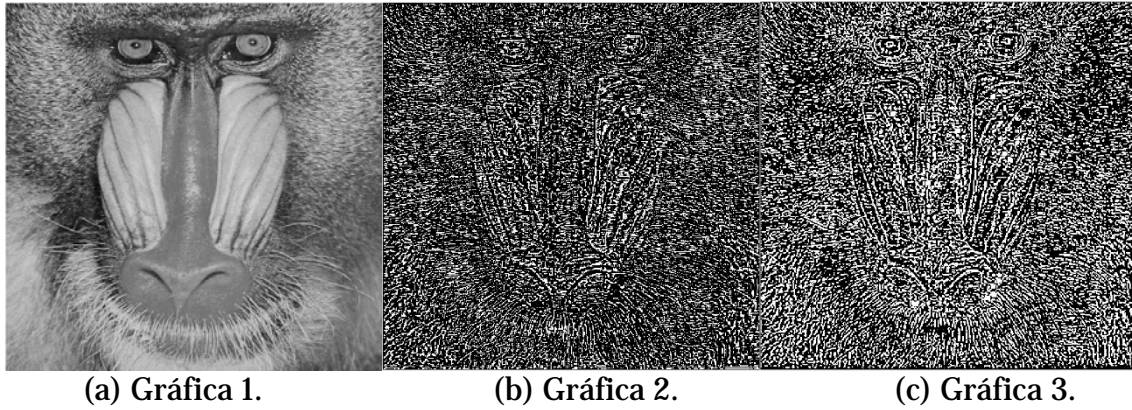


Figura 58: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, babuino 512x512 px. (a) Imagen original. (b) Imagen resultante 2da iteración juego de la vida. (c) Imagen resultante 2da iteración moda aritmética. **Fuente.** Elaboración propia.

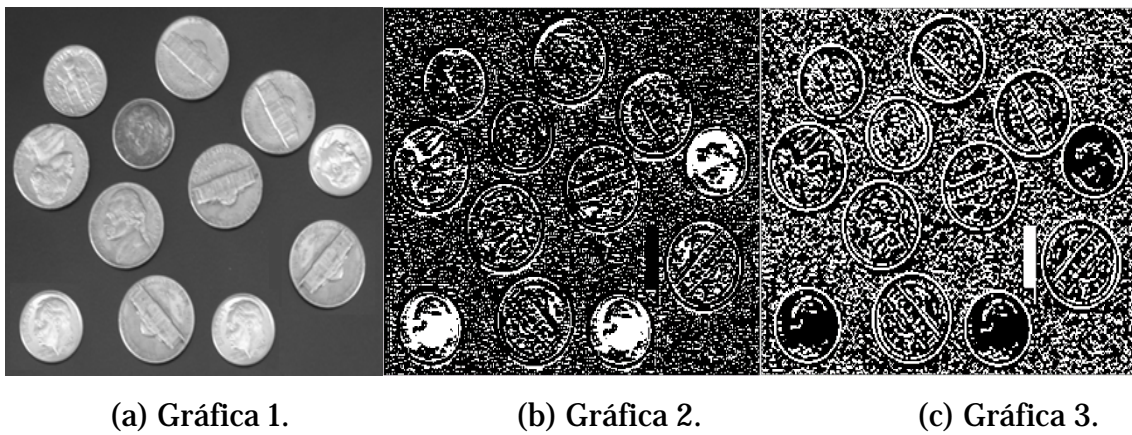


Figura 59: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, monedas 256x256 px. (a) Imagen original. (b) Imagen resultante 2da iteración juego de la vida. (c) Imagen resultante 2da iteración moda aritmética. **Fuente.** Elaboración propia.

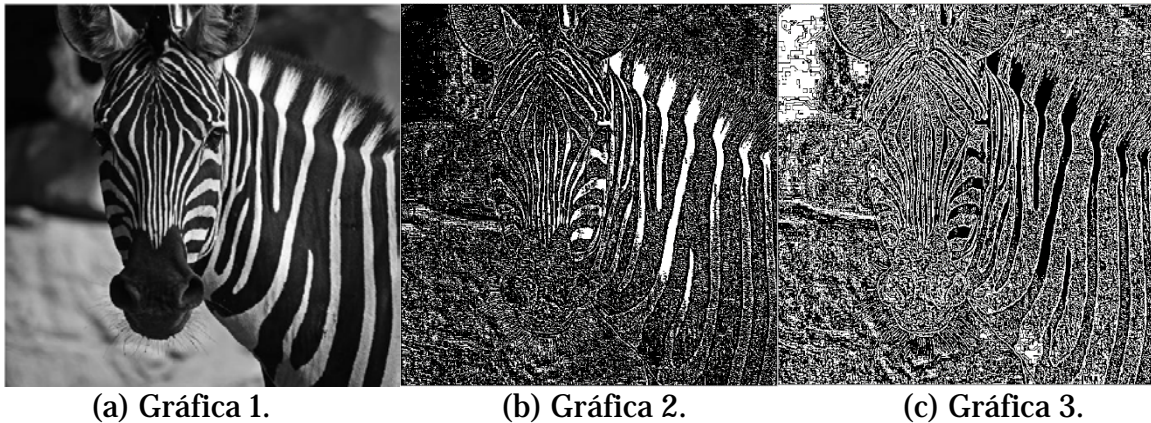


Figura 60: Aplicación del algoritmo doble iteración, 1era iteración con autómata celular, cebra 256x256 px. (a) Imagen original. (b) Imagen resultante 2da iteración juego de la vida. (c) Imagen resultante 2da iteración moda aritmética. **Fuente.** Elaboración propia.

Al revisar los resultados se presenta una mejoría en la delimitación de los bordes, también se identifica bastante información innecesaria que puede considerarse como ruido. Aun así, esto proporcionó un prototipo para establecer el algoritmo mejorado que se presentara en la siguiente sección.

5.4. Desarrollo del modelo de bordes

Todos los experimentos puestos en marcha se realizaron con imágenes de entrada en escala de grises y un vecindario inicialmente determinado con un tamaño 3×3 , es decir, vecindario de Moore, el cual recorre toda la imagen píxel a píxel y determina si el píxel en evaluación, de acuerdo con la configuración que tengan sus vecinos (reglas de transición) es definido como borde. Al ser identificado como borde este cambia su valor actual (por uno (1) o cero (0)) y el autómata avanza a la siguiente posición. Los valores de entrada (iniciales), son proporcionados por la imagen y teniendo esos datos, el autómata puede empezar a iterar sobre la imagen. Para los límites de valor fijo se aplican las condiciones en las que las reglas de transición solo se pueden ser designadas a celdas que no son de límite.

Al trabajar con un vecindario de 3×3 con celdas (píxeles) que pueden tomar valores de intensidad en un rango entre 0 y 255, existen una cantidad considerable de posibles patrones o reglas que puede tener el vecindario (256^8 , es decir, aproximadamente 18 trillones, sin considerar el valor del píxel central). Sin embargo, se pueden reducir considerablemente aplicando el concepto de simetría rotacional. Donde se establece que,

una figura plana tiene simetría rotacional cuando se puede encontrar un centro (llamado centro de rotación) de forma que si se gira la figura completa en un cierto ángulo (mayor o igual a 0° y menor que 360°), la figura rotada coincide con la figura original. De acuerdo con Costa, 2009], el número de veces que se puede hacer coincidir la imagen rotada con figura original se llama orden de la rotación y cada punto que le corresponden otro punto (se llama "punto rotado" o "imagen")

A	B	A
B		B
A	B	A

A	B	C
D		D
C	B	A

A	B	A
C		C
D	B	D

Figura 61: Patrones de una vecindad de 3×3 que permanecen invariantes con una rotación de $\pm 90^\circ$, 180° y una simetría tipo espejo. **Fuente:** Basado en [Rosin and Sun, 2011].

Teniendo en cuenta lo anterior, y aplicando el concepto sobre un vecindario de Moore, es posible afirmar que una misma regla puede ser descartada si al girar la imagen el patrón continúa siendo el mismo, reduciendo así el número de reglas en aproximadamente cinco (5) veces su tamaño [Rosin and Sun, 2011] [Rosin, 2006]. Con el fin de identificar el número de patrones distintos resultantes luego de la eliminación de pares simétricos equivalentes, es posible aplicar el método de enumeración descrito por Pólya-Burnside [Robert and Tesman, 2011], para todo G que sea un conjunto de permutaciones de un conjunto A , de modo que el número de clases de equivalencia es:

$$N = \frac{1}{|G|} \sum_{g \in G} |Fix(g)| \tag{15}$$

Donde, $Fix(g)$ es el número de elementos de A que son invariantes (dícese, de aquella configuración que al rotarla sigue presentado la misma simetría. Es decir, no cambia) en g (ver Figura 61). Al obtener el valor del número de reglas distintas N , es posible definir cuál sería el número de N en términos del número de intensidades posibles n , así como se presenta en la ecuación (16) de acuerdo con [Rosin and Sun, 2011]:

$$N = \frac{n^8 + 2n^2 + n^4 + 4n^5}{8} \quad (16)$$

A partir de lo anterior, el numerado está expresado como la suma de la rotación idéntica (p.e. 0°), dos (2) rotaciones de $\pm 90^\circ$, una (1) rotación simple de 180° y cuatro (4) rotaciones tipo espejo (a través de una línea de reflexión horizontal, vertical y diagonal) [Rosin and Sun, 2011]. Dado que a pesar de que al realizar este procedimiento, las posibles reglas ascienden a los $N > 2 \times 10^{18}$, en esa investigación se realizó un procesamiento en dos (2) pasadas para simplificar el número de operaciones, reglas y el gasto computacional que se puede generar por dicho resultado.

En la primera pasada, se calcula el valor promedio del vecindario excluyendo el píxel central. Este valor, sirve como base para obtener la variación existente entre el valor de intensidad de píxel en evaluación ($C(i, j)$) frente al promedio del vecindario:

$$Variacion = |Promedio - C(i, j)| \quad (17)$$

Los datos obtenidos anteriormente, son los pilares para determinar si un píxel es considerado como borde o no. En general, esta regla está definida así:

$$C^{t+1}(i, j) = \begin{cases} \text{fondo, si } z = C(i, j) > \text{promedio} \\ \text{fondo, si } z = \text{variacion} < 4 \\ \text{borde, para otro } z \end{cases} \quad (18)$$

Como se muestra en la ecuación anterior, $C^{t+1}(i, j)$ es el valor que será asignado a la celda en evaluación en el tiempo $t + 1$, y esta puede tomar tres (3) valores dependiendo de su estado actual $C(i, j)$, al concluir este proceso se da fin a la primer pasada. Los valores de fondo o borde están directamente relacionados con cero (0) (para este caso fondo) o uno (1) (para este caso borde), lo cual nos lleva a pensar en una imagen binaria. Es decir, que el conjunto de posibles reglas se restringió a $2^8 = 256$, no obstante, al aplicar el concepto de similaridad y eliminar las simetrías, estas se reducen a $N = 51$ [Rosin and Sun, 2011] [Rosin, 2006]. Teniendo en cuenta, el número reducido de reglas y que los bordes sobre la imagen deben presentar continuidad para ser denotados como uno, se

propusieron once (11) reglas que, gracias a la extensión del vecindario, permiten determinar, al tener mayor cantidad de información, si estos realmente son bordes o no.



Figura 62: Reglas de transición para el vecindario de Moore. **Fuente:** Elaboración propia.

La Figura 62, muestra las reglas de transición utilizadas por el autómata celular sobre la imagen, en cada una de las diferentes iteraciones realizadas. Cabe anotar, que para los vecindarios que carecían de información por estar en las esquinas o en las fronteras de la imagen, la primera y última fila y columna fueron duplicadas. El Diagrama 4 esquematiza el proceso del algoritmo.

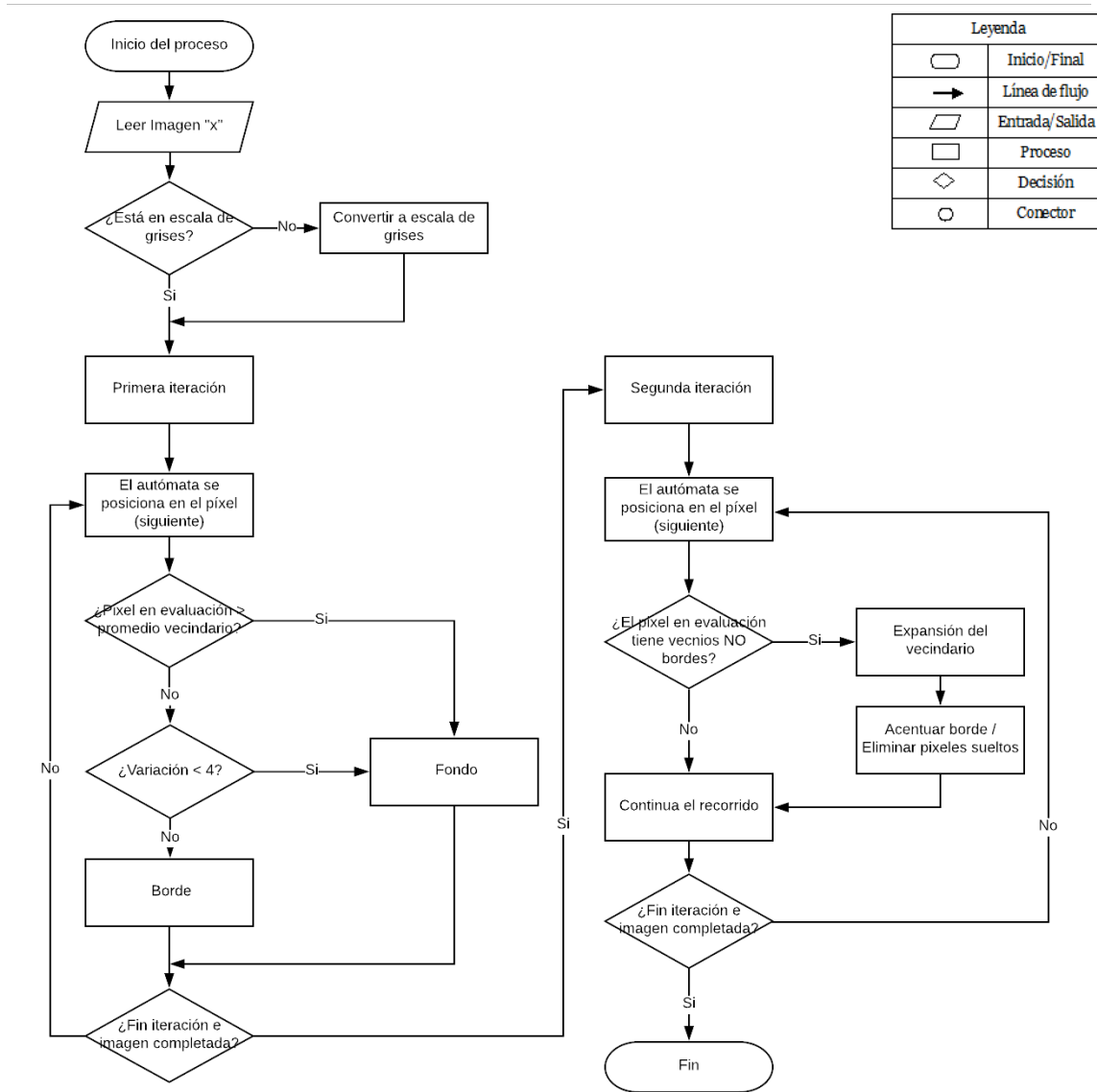


Diagrama 4: Diagrama de flujo para la detección de bordes con autómatas celulares. **Fuente:** Elaboración propia.

El algoritmo se divide básicamente en cinco regiones. La primera región verifica que la imagen se encuentre en escala de grises para poder efectuar las siguientes operaciones. La siguiente región contempla el promedio de intensidad del vecindario, lo cual determina si el píxel en evaluación es borde o no, al igual que la variación del mismo, la cual es inferior a cuatro (4). Posteriormente se evalúa si la imagen fue recorrida en su totalidad, en caso de ser falso el autómata pasa al siguiente píxel de la imagen, pero si el recorrido finaliza se inicia con la segunda iteración.

Luego de que inicia la segunda iteración, la cuarta región analiza el vecindario del pixel en evaluación, en este punto se observa el comportamiento adaptativo, el cual sirve para eliminar pixeles sueltos y afinar los bordes identificados en la primera iteración. Finalmente, en la quinta región se verifica si el autómata inspeccionó todos los pixeles de la imagen o si debe continuar con el recorrido.

Los resultados de este modelo serán presentados en secciones siguientes junto con la aplicación del algoritmo que remueve ruido en imágenes con las mismas características presentada en esta sección. Lo cual abre la posibilidad de generar un modelo más robusto y compuesto que no sólo evalúe una característica dentro de la imagen, sino que al mismo tiempo detecte imperfecciones (ruido) y las elimine de manera efectiva.

5.5. Unificación de los algoritmos

Como se ha mencionado anteriormente, el objetivo principal de esta investigación fue diseñar un algoritmo adaptativo capaz de eliminar el ruido (tanto en altos como en bajos niveles) en imágenes a escala de grises. Al lograr culminar satisfactoriamente este propósito y en el transcurso de la investigación fue posible notar que el desarrollo del mismo podría ser extendido con facilidad al campo de la detección de bordes, y en ese momento, se propuso no solo desarrollar un algoritmo que eliminara ruido de forma efectiva, sino que al mismo tiempo detectara los contornos presentes en una imagen.

Teniendo en cuenta, los desarrollos de los algoritmos descritos en las secciones 4.4 y 5.4., se generó un nuevo algoritmo que pudiera contener las dos (2) características de los modelos primigenios sin afectar la esencia de ninguno de los dos. El siguiente diagrama esquematiza gráficamente la propuesta descrita anteriormente.

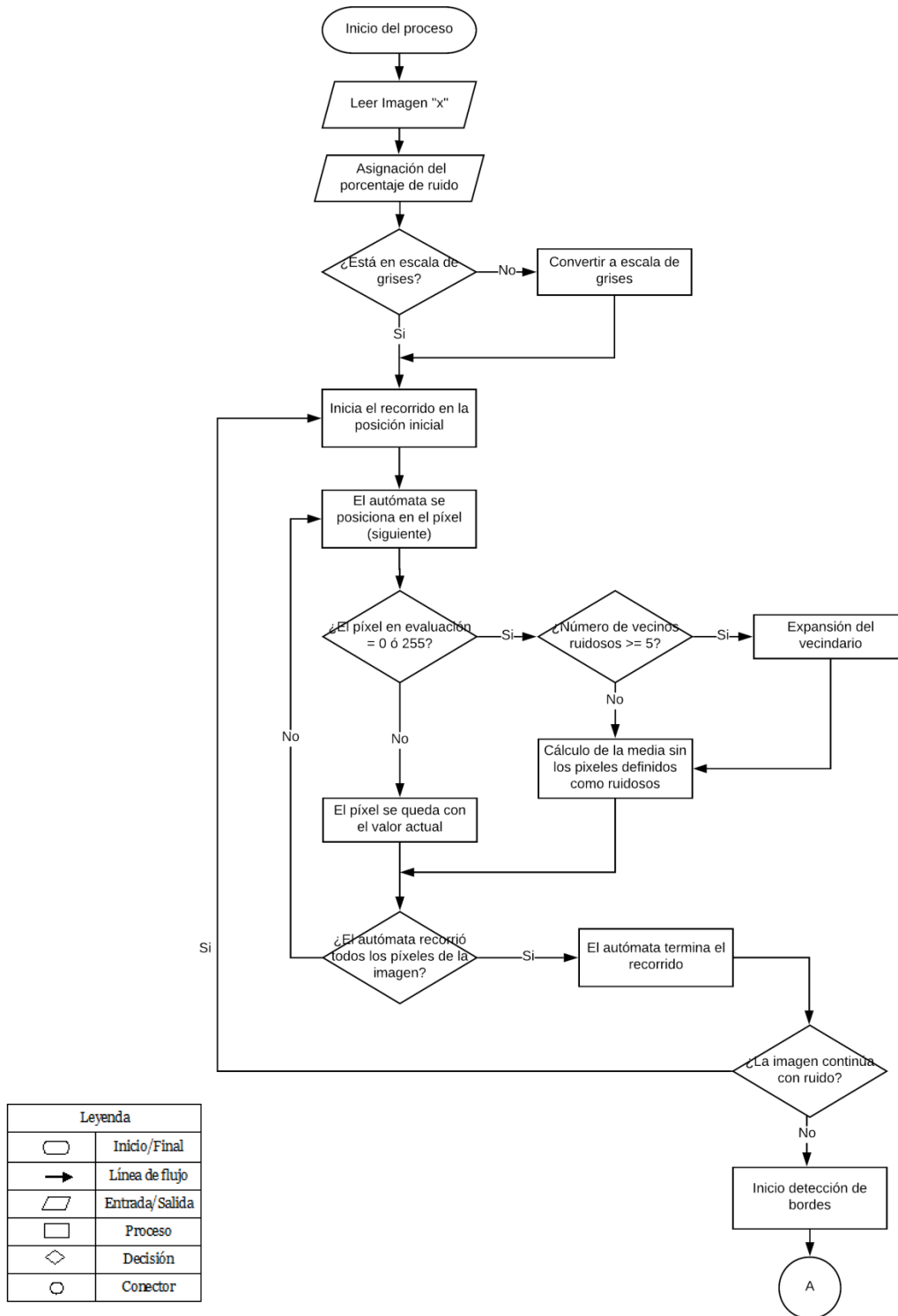


Diagrama 5: Diagrama de flujo para eliminación de ruido y la detección de bordes con autómatas celulares. Parte 1. **Fuente:** Elaboración propia.

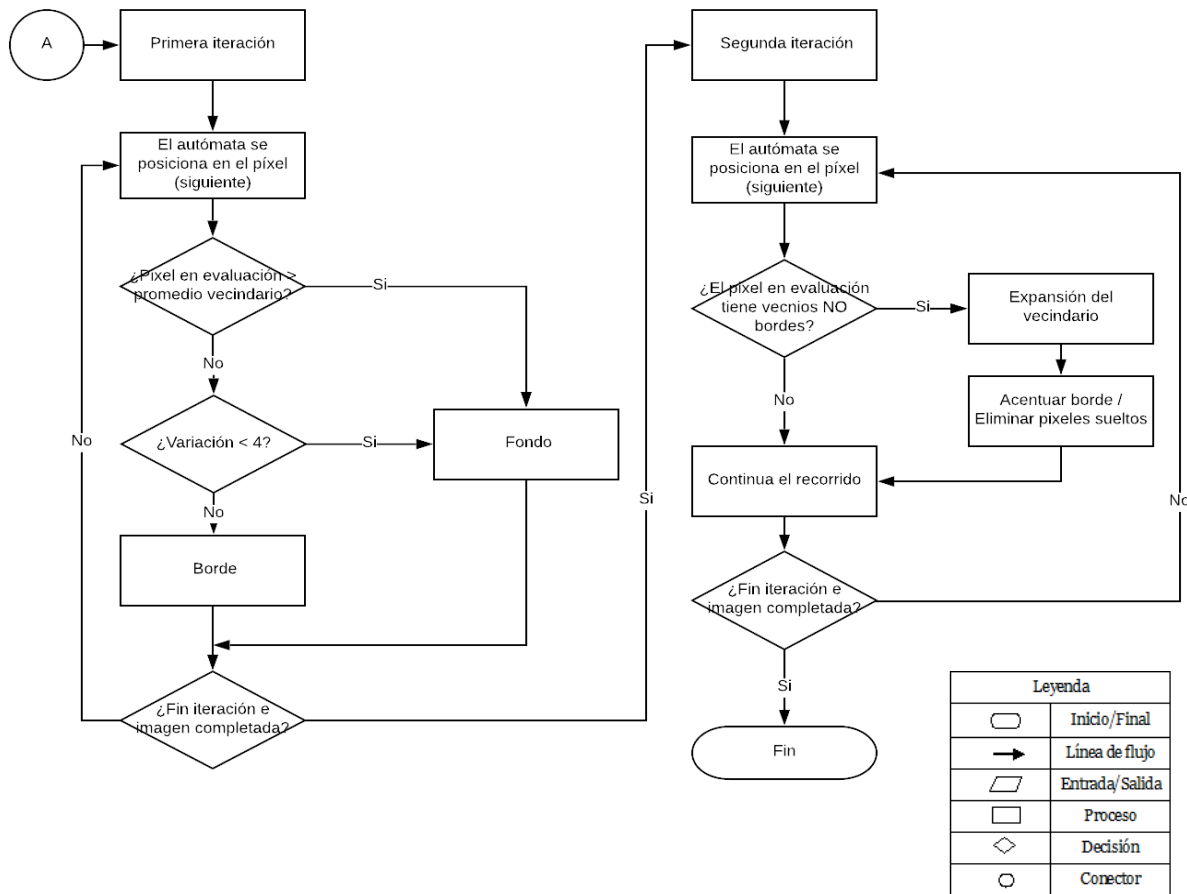


Diagrama 6: Diagrama de flujo para eliminación de ruido y la detección de bordes con autómatas celulares. Parte 2. **Fuente:** Elaboración propia.

Los Diagramas 5 y 6, muestran cómo se realiza un trabajo en paralelo con los dos (2) algoritmos (explicados detalladamente de forma individual en las secciones 4.4 y 5.4 respectivamente), donde en primer lugar, la imagen sometida al filtro de ruido, con el fin de suprimir las imperfecciones (ruido) y al obtener una imagen libre de variaciones aleatorias y deterioros evidentes, esta continua con el proceso de detección de contornos, en donde se obtiene como resultado una imagen que únicamente contiene la información de los bordes de cada uno de los objetos presentes en la misma.

Teniendo en cuenta que el modelo propuesto está compuesto por dos técnicas diferentes pero a las vez complementarias como lo son, la eliminación de ruido y la detección de bordes, esta sección aborda dos ámbitos desde los cuales se puede realizar un observación detallada de las virtudes de cada algoritmo, estos son los análisis

cuantitativo y cualitativo; que permite crear una base adecuada al momento de confrontar los resultados, ya que al establecer un marco comparativo que manifieste la mayor imparcialidad posible, cobran mayor relevancia las características que utilizan los diferentes algoritmos para lograr su objetivo.

Lo anterior se logra gracias a que esta comparación abarca diferentes parámetros tales como, la comparación uno a uno de los píxeles (resultante y original), continuidad en los trazos, inexistencia de bordes falsos, bordes definidos, etc., en donde cada algoritmo puede llegar a presentar su mejor rendimiento tanto en la parte gráfica como numérica. Además, se utilizan las métricas típicas de la detección de bordes, lo cual implica que los datos arrojados sean semejantes a los expresados por otros autores.

5.6. Análisis de los resultados obtenidos

Considerando que el modelo propuesto cuenta con dos (2) características tan singulares en lo referente a su funcionamiento interno, pero tan diferentes en cuanto a los objetivos que se desean alcanzar, y teniendo en cuenta la premisa que se expone en secciones anteriores, la cual especifica que los algoritmos para la detección de ruido están directamente ligados a la subjetividad del desarrollador.

Esta sección se divide en dos (2) apartados para mostrar tanto la lógica cuantitativa como cualitativa de los resultados obtenidos.

5.6.1. Cualitativos

Los resultados cualitativos de esta sección, están orientados a mostrar gráficamente las facultades del desarrollo propuesto. En este sentido, se plantea inicialmente un escenario que permita presentar las funcionalidades del autómata celular y de todo el algoritmo, por este motivo las imágenes que se ilustran a continuación fueron sometidas a una transformación inicial para que se pudiera simular el ruido impulsivo de sal y pimienta en un nivel de 5% inmerso en ellas

Cabe destacar que los algoritmos con los cuales se compara el Algoritmo Propuesto, no cuentan con una herramienta que les permita depurar el ruido, sin embargo, sirve como punto de comparación para establecer unas condiciones similares y así mismo, observar su comportamiento.



(a) Gráfica 1.

(b) Gráfica 2.

(c) Gráfica 3.



(d) Gráfica 4.

(e) Gráfica 5.

(f) Gráfica 6.

Figura 63: Detección de bordes con ruido al 5%, monedas (a) Imagen base. (b) Imagen resultante algoritmo de Canny. (c) Imagen resultante algoritmo de Prewitt. (d) Imagen resultante algoritmo de Sobel. (e) Imagen resultante algoritmo Laplaciano de Gauss. (f) Imagen resultante algoritmo de propuesto. **Fuente.** Elaboración propia.



(a) Gráfica 1.

(b) Gráfica 2.

(c) Gráfica 3.



(d) Gráfica 4.

(e) Gráfica 5.

(f) Gráfica 6.

Figura 64: Detección de bordes sin ruido, monedas (a) Imagen base. (b) Imagen resultante algoritmo de Canny. (c) Imagen resultante algoritmo de Prewitt. (d) Imagen resultante algoritmo de Sobel. (e) Imagen resultante algoritmo Laplaciano de Gauss. (f) Imagen resultante algoritmo de propuesto. **Fuente.** Elaboración propia.

Como se puede evidenciar en las Figuras 63 y 64, el Algoritmo Propuesto en este proyecto, tiene la capacidad de obtener resultados visualmente aceptables pese al ruido que tenga la imagen, esto se debe a las diferentes cualidades que se le adicionaron al autómata celular, lo cual le permitió este abarcar dos (2) aspectos del procesamiento de imágenes, bien sea de manera independiente o conjunta.

En la Figura 64, considerando únicamente las imágenes sin ruido, se puede destacar que las imágenes resultantes del Algoritmo Propuesto identifican más detalles que los presentados por los otros algoritmos o en su defecto, estos detectan bordes que pueden considerarse como innecesarios dentro de la imagen.

Finalmente, es notorio que los bordes generados por el Algoritmo Propuesto son en algunos puntos más gruesos que los demás. No obstante, esto es el resultado de las diferentes operaciones que realiza el autómata celular para procurar en todo momento identificar la información relevante de la imagen y así, realizar las labores pertinentes, como por ejemplo no dejar pixeles sueltos, identificar bordes sin importar su forma, no crear bordes incorrectos o no duplicar bordes.

5.6.2. Cuantitativos

Para el cálculo de los resultados cualitativos se utilizaron cuatro (4) de imágenes que prueba que pueden ser considerados como caso de laboratorio o ideales, vistos desde una perspectiva científica, ya que estas cuentan con líneas (horizontales, verticales y diagonales) y curvas claramente definidas, lo cual facilita la comparación del algoritmo frente a métodos habituales (Canny, Sobel, Prewitt y LoG) y genera una cierta coherencia, frente a las condiciones expuestas. En la Figura 65, se exponen las figuras que fueron utilizadas para este proceso.

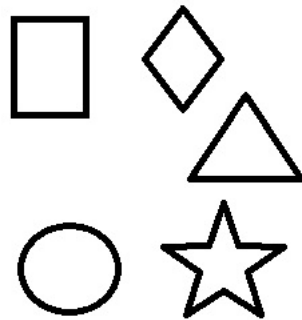


Imagen 1
(a) Gráfica 1.



Imagen 2
(b) Gráfica 2.

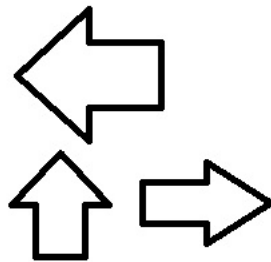


Imagen 3
(c) Gráfica 3.

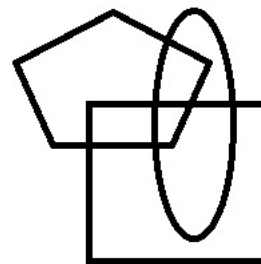


Imagen 4
(d) Gráfica 4.

Figura 65: Casos de prueba para la detección de bordes. **Fuente.** Elaboración propia.

No obstante, no sólo es necesario establecer unos casos de prueba para medir la efectividad del Algoritmo Propuesto, adicionalmente resulta conveniente establecer una serie de métricas que sirvan como marco para realizar una comparación aproximada, recordando que hasta el momento no existe un algoritmo, técnica o metodología que establezca una base fundamental para la detección de bordes.

Sin embargo, estos casos de prueba (o laboratorio) sirven como base para efectuar una comparación objetiva. Para llevar a cabo esta comparación, se hizo uso de tres (3) métricas de comparación, los cuales corresponden al Error Cuadrático Medio (por sus siglas en inglés, MSE), Error Máximo (en este contexto, MAXERR) y finalmente, la Relación de la Norma al Cuadrado (en este contexto, L2RAT).

Con el fin de entender las métricas, se hará una breve explicación de las mismas a continuación:

El Error Cuadrático Medio (MSE o Mean Square Error en inglés), es un estimador estadístico que es el encargado de medir el promedio de errores al cuadrado, es decir, la diferencia entre el valor esperado del estimador y el valor real del parámetro que se quiere estimar. El cálculo de este valor se da así:

$$MSE = \frac{\|X - Y\|^2}{N} \quad (19)$$

Donde, X , es la Imagen base, Y , es la imagen que se desea comparar y N , es el número de elementos de la imagen (píxeles). En segundo lugar, el Error Máximo o MAXERR, corresponde a la máxima desviación al cuadrado absoluta de la imagen que se desea comparar y finalmente, la Relación de la Norma al Cuadrado o L2RAT, está definida como la relación de la norma al cuadrado de la imagen que se desea comparar respecto a la imagen base.

Cabe resaltar que los valores obtenidos tanto del MSE es mejor en la medida que se aproxima o tiende a cero (0), mientras que los valores de L2RAT y MAXERR demuestra mejoría siempre que este se aproxime a uno (1).

Teniendo claro el objetivo de cada una de estas métricas, se procedió a efectuar la comparación de las imágenes resultantes a aplicar los métodos de Canny, Sobel, Prewitt y LoG frente a las arrojadas por el Algoritmo Propuesto. Los datos presentados en la Tabla 8 muestran los resultados de estos cálculos.

		Canny	Prewitt	Sobel	LoG
Imagen 1	MSE	0,05450	0,04550	0,04520	0,07680
	MAXERR	1,00000	1,00000	1,00000	1,00000
	L2RAT	1,13640	1,40290	1,39820	1,34130
Imagen 2	MSE	0,03710	0,03290	0,03320	0,05790
	MAXERR	1,00000	1,00000	1,00000	1,00000
	L2RAT	1,16040	1,44760	1,44300	1,40800
Imagen 3	MSE	0,04190	0,02790	0,02830	0,05370
	MAXERR	1,00000	1,00000	1,00000	1,00000
	L2RAT	1,07550	1,30900	1,29630	1,28390
Imagen 4	MSE	0,06110	0,04450	0,04480	0,07700
	MAXERR	1,00000	1,00000	1,00000	1,00000
	L2RAT	1,14860	1,23480	1,22290	1,21220

Tabla 8: Resultados de las métricas aplicadas para las imágenes 1, 2, 3 y 4.

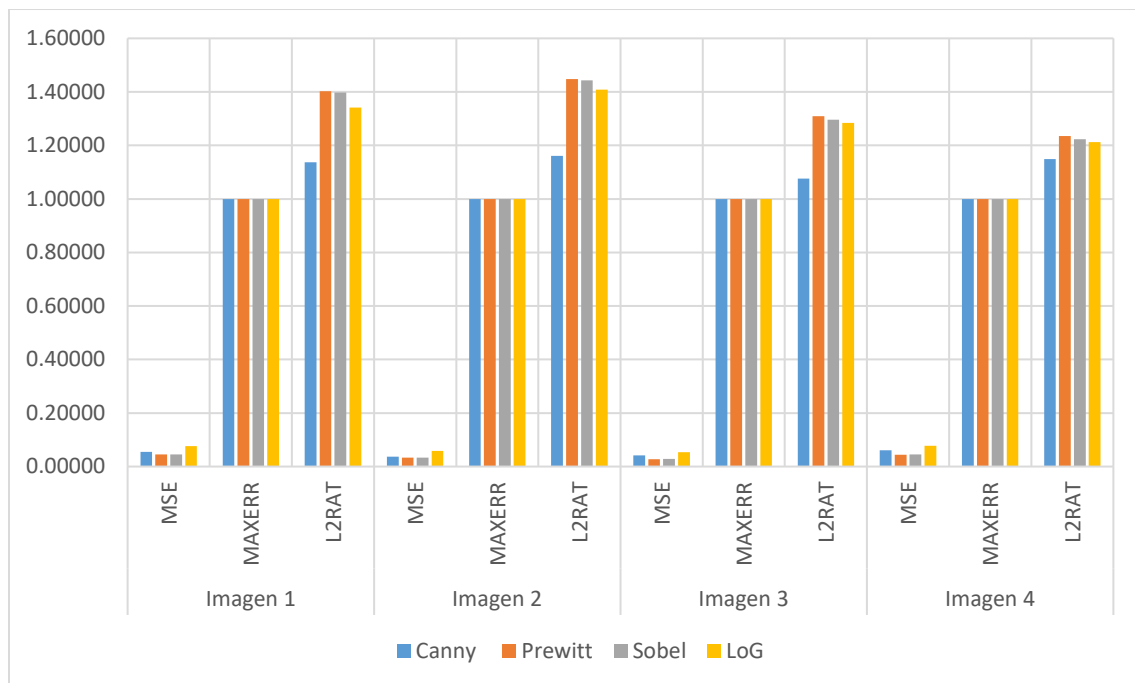


Figura 66: Grafica de los resultados de métricas utilizadas al comparar el Algoritmo Propuesto versus los métodos convencionales. **Fuente.** Elaboración propia.

Para este caso, el Error Máximo en todas las comparaciones realizadas fue nulo, ya que su valor, siempre tendió a uno (1), lo que quiere decir, que las imágenes resultantes del Algoritmo Propuesto, no presentaban errores en la definición y calidad de los bordes respecto a las evaluadas, es decir, no existen huecos o saltos en la definición del contorno del objeto encontrado (el trazo es continuo).

En cuanto al Error Cuadrático Medio, los datos arrojados muestran que el error con menor valor se presenta en el momento de comprar la imagen resultante frente a los métodos de Prewitt y Sobel, sin embargo, los valores arrojados para las demás comparaciones no sobrepasa el error en 0,08 unidades, lo que nos indica que para las imágenes de prueba el Algoritmo Propuesto tiene la capacidad de detectar bordes entre un 92.30% y un 97,21%, resultado bastante efectivo en términos cualitativos.

Los valores representados por medio del indicador L2RAT representan la aproximación entre los píxeles en la misma posición de las imágenes comparadas. En otras palabras, el Algoritmo Propuesto en los diferentes casos se asemeja a la imagen resultante de Canny, mientras que se presentan mayores diferencias al compararlo con Prewitt y Sobel. Aun así, en general los valores no se alejan demasiado de uno (1), es decir que casi se puede establecer una relación uno a uno de los resultados obtenidos en más de la mitad de los puntos que se presentan en la imagen a la hora de formar un borde. En términos porcentuales, la diferencia se encuentra ubicada entre más o menos 7,55% y un 44,76%, lo cual ratifica que la exactitud del Algoritmo Propuesto al momento de encontrar los bordes no por encima del 60% en la mayoría de los casos.

Capítulo 6

Aportes originales y discusión

En este apartado se expresan los factores que hacen que esta investigación presente novedades en el campo del procesamiento de imágenes. Cabe resaltar que en cada una de ellas existe la posibilidad de elaborar un desarrollo más amplio.

6.1. Aportes originales

Para el desarrollo del proyecto, fue necesaria una investigación exhaustiva sobre los diferentes algoritmos, métodos y técnicas existentes hasta la fecha, con el fin de determinar aquellos que han realizado mayores aportes en este ámbito científico.

En concordancia con lo anterior, este proyecto en esencia contiene dos (2) componentes que pueden resultar novedosos: a) El comportamiento adaptativo con el que cuenta el autómata celular y b) la combinación de metodologías para llevar a cabo la labor de detectar/eliminar ruido impulsivo junto con la identificación de contornos, en un proceso que no afecta la complejidad computacional. En primer lugar, aunque el tema de la implementación de autómatas celulares en el procesamiento de imágenes es común, hasta el momento no existen investigaciones que incluyan el comportamiento adaptativo dentro del autómata, lo cual trajo consigo resultados bastante favorables en comparación con los existentes, además fue de gran logro, encontrar que, al establecer esta característica en el algoritmo, el resultado en imágenes con alto nivel de ruido fue

destacable (para más detalles ver sección 4.6). Finalmente, la combinación de las técnicas (eliminación de ruido y detección de contornos), así como, la implementación de un método para la detección de bordes adaptativo, trajo consigo resultados y bordes en imágenes en escala de grises, que hasta el momento no se han visto.

Los aportes originales de este trabajo se presentan en dos (2) puntos:

Propuesta de un algoritmo adaptativo: se presenta este aporte de manera reiterada como aporte original, dado que hasta la fecha existe un alto número de aportes realizados por otros autores donde se abarca el funcionamiento de los autómatas celulares, pero no se explota todo su potencial. Los autómatas celulares dentro de sus capacidades básicas, pueden ampliar su vecindario. Sin embargo, trabajos que presentan características semejantes solamente contemplan los vecindarios estáticos, bien sea desde la perspectiva original (entiéndase por original, lo referente a lo planteado inicialmente por Neumann o Moore, es decir $n=1$) o la extendida. Se destaca que, desde el punto de vista de la biología, los seres vivos dependiendo de las circunstancias evolucionan de tal forma se adaptan a su entorno. Y en el caso de esta investigación, se le otorga esta capacidad al autómata de realizar esta función, que está enfocada en una técnica que se fundamenta en el mundo natural.

Implementación de un modelo adaptativo que tiene la capacidad de eliminar ruido y detectar bordes: los resultados que se alcanzaron en esta investigación, se dieron gracias al ensayo y error de diferentes metodologías, las cuales se encuentran fuertemente fundamentadas. No obstante, la combinación de dos (2) inclinaciones como lo son la identificación de contornos y la depuración de ruido, no ha sido comúnmente tratada como un complemento directo entre ellas. Por ende, la unión de estas técnicas de procesamiento de imágenes más el aporte de un comportamiento adaptativo, resaltan la labor establecida en este proyecto.

De igual manera, el proyecto atravesó por un proceso evolutivo basado en el método científico, para alcanzar los resultados obtenidos hasta la finalización de este trabajo. Aunque, queda abierta la posibilidad de ahondar en diferentes vertientes que contempla este trabajo.

6.2. Discusión

Para llevar a cabo este proyecto investigativo, se establecieron algunas acotaciones con el fin de enfocar y definir el camino del mismo. Inicialmente se exploró la posibilidad de aplicar los algoritmos sobre imágenes en escala RGB y en escala blanco y negro, no obstante, la documentación existente y las investigaciones llevadas a cabo por otros investigadores, dieron una pauta consistente para usar las imágenes en escala de grises como base experimental. Asimismo, se contemplaron los diferentes tipos de ruido, pero por razones similares se optó por el ruido impulsivo de tipo sal y pimienta como base experimental.

Por otro lado, en la sección de análisis de resultados y comparación se tenía la intención de comparar con una cantidad mayor de algoritmos, pero comúnmente los autores no presentan explícitamente sus algoritmos, por lo que solamente se tuvieron presentes aquellos donde el pseudocódigo, diagrama de flujo o modelos matemáticos permitían replicar las condiciones para obtener los resultados presentes. Propiamente desde la perspectiva de los resultados correspondientes a la detección de contornos, existe un enfoque subjetivo; esto se debe a que hasta el momento no existe algún acercamiento que pueda considerarse “perfecto”, lo cual limita que exista una base de comparación robusta que permita comparar objetivamente los resultados tanto propios como los de los presentados por otros autores. Sin embargo, por medio de otros indicadores comparativos se llevó a cabo la labor.

Adicionalmente, este proyecto cuenta con factores novedosos (el comportamiento adaptativo de un autómata celular y la combinación de dos (2) técnicas de procesamiento de imágenes), que no se han explorado ampliamente por otros investigadores, aun así, los resultados presentados en las secciones anteriores garantizan que las tareas realizadas a lo largo de la investigación han presentado resultados satisfactorios tanto a niveles cuantitativos como cualitativos.

Capítulo 7

Conclusiones y expectativas

7.1. Conclusiones

El modelo desarrollado para la detección de bordes y eliminación de ruido en términos generales, supero los resultados de los algoritmos con los cuales se comparó. Teniendo en consideración que se realizaron símiles incrementando sucesivamente en 10% el nivel de ruido, la brecha de eficiencia se comienza a enmarcar una considerable diferencia, luego de alcanzar el 50% de ruido. Asimismo, cuando el nivel de ruido llega al 90%, normalmente algunos algoritmos no logran restaurar la calidad de la imagen a un punto aceptable. Sin embargo, el Algoritmo Propuesto pudo realizar una restauración más eficiente tanto a nivel cuantitativo como cualitativo. De acuerdo con el índice de similitud estructural (SSIM), el Algoritmo Propuesto supero entre un 1% y un 68% los resultados presentados por otros autores cuando el nivel de ruido es del 50%. En el caso de las imágenes con ruido al 90% el Algoritmo Propuesto supero los resultados entre un 9% y 66%.

En el apartado de la detección de contornos, se presenta una particularidad, ya que todavía no se ha desarrollado un algoritmo con resultados que puedan considerarse como perfectos y que sirvan como base fundamental y sólida, para establecer una comparación completamente objetiva. Sin embargo, existen algoritmos (Canny, Sobel, Prewitt o Laplaciano de Gauss) los cuales son comúnmente usados como base comparativa de los

resultados. No obstante, se debe recalcar que las condiciones de cada uno de estos algoritmos son diferentes, pero, aun así, por medio de comparaciones con “casos ideales” (donde las formas que se usan son elementales), se pueden considerar como unas comparaciones objetivas de resultados. Por un lado, se obtuvo el MSE, el cual en ninguna comparación superó el valor de 0,08. Lo que indica que el Algoritmo Propuesto tiene la capacidad de detectar bordes entre un 92,30% y un 97,21%, es decir, desde una perspectiva en términos cualitativos que es el resultado es bastante eficiente. En cuanto a los resultados de la variable L2RAT, estos oscilan entre un 7,55% y un 44,76%, lo que representa básicamente se puede establecer una relación 1 a 1 entre los pixeles de los algoritmos comparativos y el Algoritmo Propuesto. Y para el caso del error máximo (MAXERR) se mantuvo siempre en uno (1), esto significa que no existen huecos o saltos en la definición del contorno del objeto encontrado (el trazo es continuo).

Dentro de los alcances que se plantearon inicialmente en este proyecto, no contemplaba el hecho de unificar dos (2) algoritmos que abarcaran dos (2) técnicas del procesamiento de imágenes digitales, teniendo en cuenta esto, se dividió la forma como se comparan los resultados, además, el hecho de que muy pocos investigadores hayan abarcado las mismas temáticas dentro de sus proyectos, hace que esta técnica sea novedosa. Para la comparación de imágenes donde existía un nivel de ruido de 5% de ruido y se deseaban identificar los bordes (apartado 5.6.1), como se puede evidenciar, el Algoritmo Propuesto pudo realizar las tareas de depurar ruido y localizar los contornos de una manera eficiente y con la calidad de los bordes más detallada y robusta.

7.2. Expectativas

Teniendo en cuenta los aportes expuestos en este proyecto, los cuales se fundamentan primordialmente en autómatas celulares y su comportamiento en el procesamiento de imágenes, y lo novedoso del mismo, las posibilidades de investigación a partir de lo planteado, generaron una serie de vertientes que ampliaron el horizonte investigativo, así:

Niveles de intensidad y modelos de colores: el objeto de estudio que enmarcó este proyecto, se enfatizó en las imágenes en escala de grises. Por lo tanto, realizar ajustes que favorezcan la realización de labores de procesamiento de imágenes digitales en otras escalas de intensidad o modelos de color, tales como RGB, CMY, YIQ, HSI, entre otras.

Tipos de ruido: a pesar de que se realizaron algunos experimentos con ruido de tipo gaussiano, es necesario la modificación de parámetros y ajustes para lograr este objetivo. No obstante, es una posibilidad factible contemplar otros tipos de ruido como objeto central de investigación por medio de los autómatas celulares.

Optimización de resultados: teniendo en cuenta los resultados alcanzados, queda abierta la posibilidad de implementar técnicas que permitan mayores acercamientos. Dentro las posibilidades se encuentra el entrenamiento del autómata celular por medio de algoritmos genéticos, lo cual proporcione un comportamiento más intuitivo y asertivo. De igual manera otros métodos pueden ser implementados con el fin de optimizar los acercamientos realizados en esta investigación.

Modelo matemático: A raíz de la eficiencia del comportamiento adaptativo del autómata celular, es factible establecer un modelo matemático el cual brinde un soporte con mejor orientación al fundamento científico a este sistema bioinspirado.

Capítulo 8

Productos académicos

8.1. Realizados

En esta sección se encuentra la primera hoja de los productos que ya han sido publicados o se encuentran en proceso de publicación:

INTEGRATION OF AN ADAPTIVE CELLULAR AUTOMATON AND A CELLULAR NEURAL NETWORK FOR THE IMPULSIVE NOISE SUPPRESSION AND EDGE DETECTION IN DIGITAL IMAGES

Este paper fue presentado en el “2nd IEEE Colombian Conference on Applications in computational Intelligence - ColCACI 2019”, y en este momento, está en proceso de publicación. La versión completa se encuentra en la carpeta: *Productos académicos/1. Realizados/IEEE - ColCACI 2019*.

Integration of an adaptive cellular automaton and a cellular neural network for the impulsive noise suppression and edge detection in digital images

Danilo Gustavo Gil Sierra
Facultad de Ingeniería
Universidad Distrital Francisco José de
Caldas
Bogotá, Colombia
dggils@correo.udistrital.edu.co

Karen Vanessa Angulo Sogamoso
Facultad de Ingeniería
Universidad Distrital Francisco José de
Caldas
Bogotá, Colombia
kvangulos@correo.udistrital.edu.co

Helbert Eduardo Espitia Cuchango
Facultad de Ingeniería
Universidad Distrital Francisco José de
Caldas
Bogotá, Colombia
heespitiac@udistrital.edu.co

Abstract—This article proposes the combination of two bio-inspired computational models that are sequentially implemented to eliminate impulsive noise and edge detection in grayscale images. In general, this procedure consists of: (1) implementing a cellular automaton (CA) with an adaptive behavior that expands the Moore neighborhood when he considers that the information obtained from his first level neighbors is insufficient. Based on the above, the image affected by noise is processed, in order to eliminate the corrupted pixels and perform reprocessing that will lead to the improvement of the quality of the image, (2) the resulting image is defined as an input of the cellular neural network (CNN) together with the training images, so that by defining three templates (feedback (A), cloning (B) and threshold or bias (I)), you can perform contour detection of objects present within the image thrown by the initial method. The results for the noise elimination presents a restoration of the image that oscillates between 70.63% and 99.65%, indicating that the image does not lose its quality despite being exposed to high noise levels, similarly it occurs for the edge detection, which presents an approximate efficiency of 65% with respect to the algorithms established within the framework of comparison.

Keywords—image processing, cellular neural networks, adaptive algorithm, noise elimination, edge detection.

I. INTRODUCCIÓN

El procesamiento de imágenes se remonta a cientos de años cuando filósofos y científicos como Alhazen, Aristóteles, Roger Bacon y Leonardo Da Vinci, entre otros, centraron sus esfuerzos en el estudio de la formación de imágenes en el ojo humano (óptica) [1]. Siendo este el primer paso para que años más tarde, investigadores y expertos en el tema hicieran uso de estos conceptos como base fundamental, para así lograr desencadenar una corriente investigativa enfocada al tratamiento de fenómenos ópticos que fueron útiles posteriormente para la creación de imágenes [1]. Años más tarde, gracias a la teoría desarrollada por el matemático Jean-Baptiste-Joseph Fourier (1768-1830), se pudo verificar que el fundamento del procesamiento de imágenes tanto para medios ópticos como para medios digitales, está basado en el teorema de Fourier, ya que la representación de la variación de la irradiancia o brillantez de una imagen puede ser vista como una suma de distribuciones senoidales de varias frecuencias [2].

Actualmente, las imágenes desempeñan un papel importante en la percepción de la realidad, debido a que el movimiento se puede interpretar como una secuencia de imágenes (fotogramas) que se visualizan en un segundo y a partir de estos fotogramas es posible identificar características relevantes de un objeto en particular o de su entorno en

general. Por lo anterior, esta temática es de vital importancia en diversos ámbitos científicos tales como: análisis y diagnósticos médicos, exploración espacial, visión computacional, reconocimiento de imágenes, etc. [3], ya que con el uso de imágenes se han generado aportes significativos en estos campos y al mismo tiempo, se han desencadenado el desarrollo de técnicas que tienen como objeto mejorar el aspecto de la imagen y resaltar los detalles que se desean hacer más visibles de la misma [1], [2].

Dentro de las técnicas más frecuentemente utilizadas en el procesamiento de imágenes, se encuentran la eliminación de ruido y la detección de bordes, las cuales han sido empleadas en un sinnúmero de aplicaciones que abarcan desarrollos de metodologías que inician en simples y eficientes, y terminan en complejas y robustas [3].

El objetivo principal de las técnicas de eliminación de ruido es reducir los píxeles que toman un nivel de intensidad diferente al de sus vecinos, bien sea por el proceso de captura, digitalización y/o transmisión de la imagen [4]. Para abarcar esta problemática existen distintos tipos de algoritmos que se alinean con dicho objeto. Los filtros de ruido más comunes son los filtros lineales que se dividen en: filtro de media y filtro gaussiano, los cuales realizan una operación de convolución entre la imagen a ser filtrada y una máscara, sin ofrecer resultados satisfactorios, debido a que la imagen resultante se presenta difuminada y en muchas ocasiones, los bordes se pierden [4], asimismo, es importante destacar que para imágenes que presentan bajas densidades de ruido estos métodos son bastante eficientes [5].

Por otro lado, se encuentran los filtros no lineales, como el filtro de mediana, el cual en términos generales, recorre cada píxel de la imagen y lo reemplaza por la mediana de los píxeles vecinos, reduciendo así los defectos de la imagen [4], [6]; no obstante, este proceso aumenta significativamente la complejidad computacional, porque para calcular la mediana, se deben ordenar los diferentes valores que aparecen dentro de la máscara y luego determinar cuál es el valor central [5]. Además, cabe destacar que este método presenta inconvenientes con imágenes que tienen altas densidades de ruido [8], puesto que se opera sobre todos los píxeles de la imagen.

Referente a la detección de bordes, el objetivo de esta técnica contempla la identificación de los cambios de intensidad en una imagen (variación de la luminancia), es decir que, a través de una serie de operaciones matemáticas se busca detectar las discontinuidades entre los píxeles [11].

Dentro de los métodos que se enfocan en resolver esta temática, se destacan los trabajos realizados por Lawrence Roberts en 1963 y John Canny en 1986, que hicieron uso de

INTEGRATION OF AN ADAPTIVE CELLULAR AUTOMATON AND A CELLULAR NEURAL NETWORK FOR THE IMPULSIVE NOISE SUPPRESSION AND EDGE DETECTION IN DIGITAL IMAGES – **EXTENDED VERSION**

Este paper fue presentado en el “2nd IEEE Colombian Conference on Applications in computational Intelligence - ColCACI 2019” y fue aceptado como versión extendida para la revista “**Communications in Computer and Information Sciences (CCIS) of Springer**”, en este momento, está en proceso de publicación. La versión completa se encuentra en la carpeta: *Productos académicos/1. Realizados/Communications in Computer and Information Sciences (CCIS) of Springer*.

Integration of an Adaptive Cellular Automaton and a Cellular Neural Network for the Impulsive Noise Suppression and Edge Detection in Digital Images

Karen Angulo^[0000-0002-3624-0554], Danilo Gil^[0000-0001-9106-3370] and Helbert Espitia²

¹ Universidad Distrital Francisco José de Caldas, Systems Engineering, Bogotá, Colombia
{kvangulos,dggils}@correo.udistrital.edu.co

² Universidad Distrital Francisco José de Caldas, Faculty of Engineering, Bogotá, Colombia
heespitiac@udistrital.edu.co

Abstract. This article proposes the combination of two bio-inspired computational models that are sequentially implemented to eliminate impulsive noise and edge detection in grayscale images. In general, this procedure consists of: (1) implementing a cellular automaton (CA) with an adaptive behavior that expands the Moore neighborhood when it considers that the information obtained from its first level neighbors is insufficient. Based on the above, the image affected by noise is processed, in order to eliminate the corrupted pixels and perform reprocessing that will lead to the improvement of the quality of the image, (2) the resulting image is defined as an input of the cellular neural network (CNN) together with the training images, so that by defining three templates (feedback (*A*), cloning (*B*) and threshold or bias (*I*)), contour detection of objects within the image thrown by the initial method can be performed. The results for the noise elimination present a restoration of the image that oscillates between 70.63% and 99.65%, indicating that the image does not lose its quality despite being exposed to high noise levels, similarly it occurs for the edge detection, which presents an approximate efficiency of 65% with respect to the algorithms established within the framework of comparison.

Keywords: Image Processing, Cellular Neural Networks, Adaptive Algorithm, Noise Elimination, Edge Detection.

1 Introduction

Image processing goes back hundreds of years when philosophers and scientists like Alhazen, Aristotle, Roger Bacon and Leonardo Da Vinci, among others, focused their efforts on the study of imaging in the human eye (optics) [1]. This being the first step for years later, researchers and experts in the field made use of these concepts as a fundamental basis to trigger a research trend focused on the treatment of optical phenomena that were subsequently useful for the creation of images [1]. Years later, thanks to the theory developed by the mathematician Jean-Baptiste-Joseph Fourier (1768-1830), it was verified that the basis of image processing for both optical media and digital media is based on the Fourier theorem, since the representation of the vari-

8.2. En proceso de aceptación

METHOD FOR EDGES DETECTION IN DIGITAL IMAGES THROUGH THE USE OF CELLULAR AUTOMATA

Este paper será presentado en el “CSEI'2019 - I Congreso de Ciencia de la Computación, Electrónica e Ingeniería Industrial” y para ser publicado en la revista “**RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao**”, el congreso que se realizará del 28 al 31 de octubre de 2019. La versión completa se encuentra en las carpetas: *Productos académicos/2. Aceptados en proceso de publicación/Revista Ibérica de Sistemas e Tecnologias de Informação – RISTI*.

Method for edges detection in digital images through the use of cellular automata

Abstract: In this article an algorithm based on cellular automata for noise elimination and edge detection in grayscale images is proposed. However, the focus of this project will be on the process of identifying contours, since this represents a higher challenge at the research level. The cellular automaton has an adaptive behavior, which allows it to expand when it considers that the information coming from its initial neighbors is insufficient to determine if the pixel in evaluation is a "border pixel" or not. The above along with a transition ruleset are important features and useful to accentuate relevant details within the image. Initially, a review of the different works carried out by other authors will be given. Afterwards, a brief explanation of the fundamentals that are part of the scope of this investigation will be shown. Finally, the proposed algorithm and the results obtained will be presented.

Keywords: cellular automaton; digital images; image processing; edges.

1. Introduction

In the field of image processing, edge detection is a relevant and widely used application because it is considered a low-level processing operation (Bhardwaj & Mittal, 2012). In other words, a contour is located in the first operation of the algorithm on the image, when examining each pixel and determining whether it has the properties of an edge. Thus, the main emphasis of this tool is to focus and identify sharp discontinuities present in the image (Diwakar, Kumar & Gupta, 2013). These discontinuities are defined by some authors as abrupt changes in the intensity of the pixels and are the main characteristic to find the limits of the objects that are immersed in an image. From this concept arises the need to denote an edge as which limits and segments the different regions or objects within the image under evaluation (Diwakar, Kumar & Gupta, 2013). The edge detection is usually the first step of many of computer vision algorithms (edge-based facial recognition, edge-based obstacle detection, edge-based objective recognition, and image compression, among others).

8.3. En revisión

PROPOSAL OF AN ALGORITHM BASED ON CELLULAR AUTOMATA TO ELIMINATE NOISE IN DIGITAL IMAGES

Un autómata celular es un sistema de procesamiento que toma decisiones basadas en la información de sus vecinos. La implementación de un autómata celular puede tener una relación directa con la forma de representación de imágenes digitales, por lo que es posible realizar aplicaciones de procesamiento de imágenes utilizando este concepto. Debido a esto, este documento presenta la propuesta de un algoritmo basado en autómatas celulares para la eliminación del ruido impulsivo en imágenes digitales, que utiliza el mecanismo de adaptación para ajustarse a las condiciones del entorno (en este caso, la imagen) en este momento. que la información adquirida por el autómata celular es insuficiente para tomar una decisión sobre el píxel en evaluación. Además, gracias a la consideración de trabajos relacionados, se puede establecer un método de evaluación para observar el desempeño del algoritmo propuesto en comparación con dos algoritmos presentados por otros autores. Las pruebas se llevan a cabo con cuatro imágenes que tienen características diferentes. Al evaluar las imágenes expuestas a diferentes niveles de ruido, los resultados obtenidos muestran que el algoritmo propuesto se presenta mejor de acuerdo con el Índice de similitud estructural (SSIM), ya que a niveles de ruido entre 10% y 90% las mejoras en el rango de reducción de ruido entre 15% y 68 %. La versión completa de este borrador encuentra en la carpeta, *Productos académicos*, en la subcarpeta 3. *En Revisión*.

PROPOSAL OF AN ALGORITHM BASED ON CELLULAR AUTOMATA TO ELIMINATE NOISE IN DIGITAL IMAGES

Karen V. Angulo-Sogamoso¹, Danilo G. Gil-Sierra¹, and Helbert E. Espitia-Cuchango²

¹Ingeniería de sistemas, Facultad de Ingeniería, Universidad Distrital Francisco José de Caldas, Bogotá, Colombia

²Facultad de Ingeniería, Universidad Distrital Francisco José de Caldas, Bogotá, Colombia

Email: kvangulos@correo.udistrital.edu.co, dggils@correo.udistrital.edu.co, heespitiac@udistrital.edu.co

ABSTRACT

A cellular automaton is a processing system that makes decisions based on the information of its neighbors. Implementing a cellular automaton can have a direct relationship with the form of digital images representation, whereby it is possible to perform image processing applications using this concept. Due to this, this document presents the proposal of an algorithm based on cellular automata for the elimination of impulsive noise in digital images, which makes use of the adaptation mechanism to adjust to the environment conditions (in this case, the image) at the moment that the information acquired by the cellular automaton is insufficient to make a decision about the pixel under evaluation. In addition, thanks to the consideration of related works, an evaluation method can be established to observe the performance of the proposed algorithm against two algorithms presented by other authors. Tests are carried out with four images that have different characteristics. In evaluating the images exposed to different noise levels, the results obtained show that the proposed algorithm presents better according to the Structural Similarity Index (SSIM), since at noise levels between 10% and 90% improvements in noise reduction range between 15% and 68%.

Keywords: cellular automaton, digital images, processing, noise.

1. INTRODUCTION

For several years, digital images have played an important role into the research environment, thanks to the fact that, like any type of data, it can reveal valuable information for any field [1]. With the arrival of information systems and continuous technological advances, it became necessary to capture an image in a computational way and, at the same time, create systematized techniques or applications that will help extract information from it. From this need, the processing of digital images arises. In the field of image processing, there is great variety of techniques that fulfill different purposes and, in turn, influence different forms on an image, all aiming the improvement of the especial conditions that this presents. This is the case of techniques focused on the elimination of noise, that seek a corrupt image (image that is affected by some external factors which generate noise on it, making it impossible to read it properly [1]) exposes its best shape, by varying or changing the value of noisy pixels by one that is more in line with its surroundings.

When discussing noise elimination, it is important to mention that within the field of noise typifications of image processing, there is great variety, however, among the most common are: Gaussian noise and impulsive noise, since work and research are frequently presented focused on solving this problem and, at the same time in many real life cases, often the images are affected by impulsive noise, which can be caused by different factors such as errors or interferences in the transmission sections (that is, by transmitting the information bits), defective memory locations in the hardware, errors associated with the equipment used to capture images or malfunction of the pixels in the camera sensors [2]. However, within the classification of impulsive noise there is a special and well known case

called "salt and pepper", which is observed in a contaminated image through pixels that can only take the maximum or minimum values; that is, they usually appear as black and white dots on the image [1], [2].

Taking into account that the presence of noise can significantly affect the quality of an image, various techniques have been proposed for the restoration of images that have been affected by impulsive noise, which have made solid contributions to this field, through the creation of new methods from the variation of conventional techniques or the formulation of new types of filters, which have been continuously improved in order to obtain better results in less time. Among the most used methods to suppress this type of noise is the non-linear filter called "median filter", because its strength lies in efficiently solving the effects on images with low noise densities [2]. However, when applied in images with high noise densities, a slight deformation of the image details (lines, corners, edges) is generated [3], since the operation is carried out indiscriminately over the whole image and does not discriminate if the pixel is noisy or not. Based on the above, and observing the problems that still occur in the basic filtering of salt and pepper noise, several researchers have developed new techniques and approaches to obtain improvements in this topic.

Particularly, Garnett, Huegerich, Chui and He in [4], proposed a filter which through statistical values obtained from the quantification of the differences between pixel intensities with respect to their more similar neighbors. When applied to an image, it can be seen that this filter is extended in such a way that it is possible to eliminate both Gaussian and Impulsive noise, the results were outstanding at a quantitative and qualitative level. However, the filter has some faults when the noise level is too high, because if most of its neighbors are noisy pixels, the calculated statistical value is not optimal.

In [5] Echeverri, Rudas, Toscano and

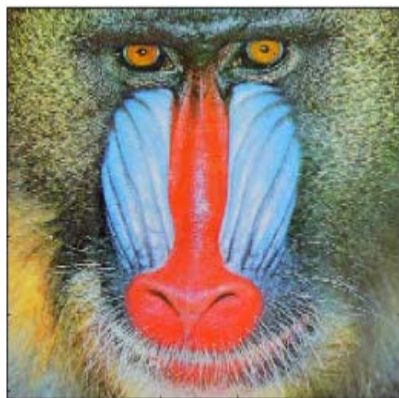
Apéndice A

Anexo 1

A.1. Resultados del método propuesto para la eliminación de ruido impulsivo en imágenes a escala de grises

A continuación, se adjuntan las imágenes producto de la simulación del algoritmo basado en autómatas celulares para la eliminación de ruido. En las Figuras A.1., A.2., A.3., y A.4., se observan los resultados de la simulación en dos (2) imágenes comúnmente utilizadas (Babuino y Cameraman). Acorde con los datos de obtenidos en la sección 4.6., el mayor porcentaje de restauración de una imagen, se encuentra cuando esta tiene un ruido inferior al 50%. Sin embargo, los resultados en imágenes con ruidos superiores muestran una clara identificación de la imagen expuesta.

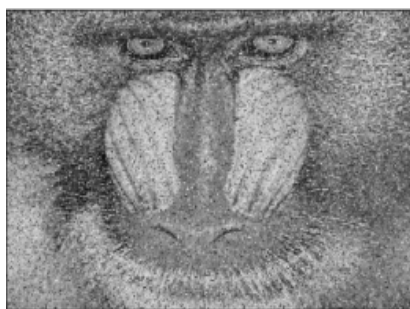
Además, es importante mencionar que las siguientes ilustraciones muestran el resultado en imágenes afectadas con 50% y 90% de ruido, siendo estos los puntos centrales de la investigación.



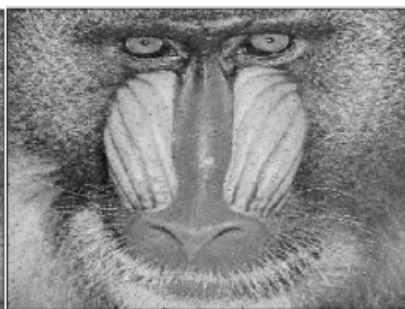
(a) Gráfica 1.



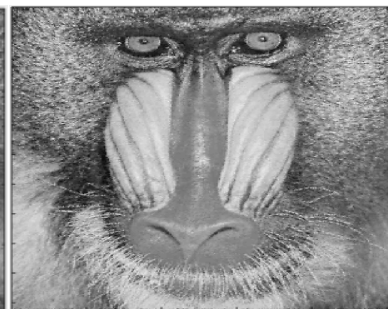
(b) Gráfica 2.



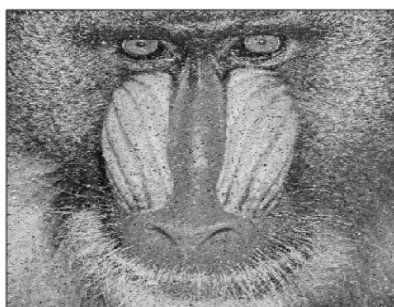
(c) Gráfica 3.
Algoritmo 1



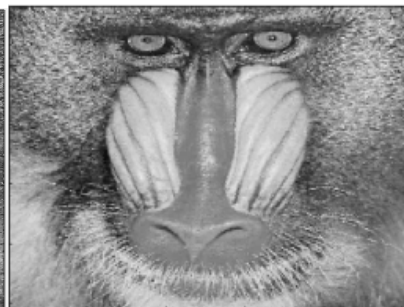
(d) Gráfica 4.
Algoritmo 2



(e) Gráfica 5.
Algoritmo 3

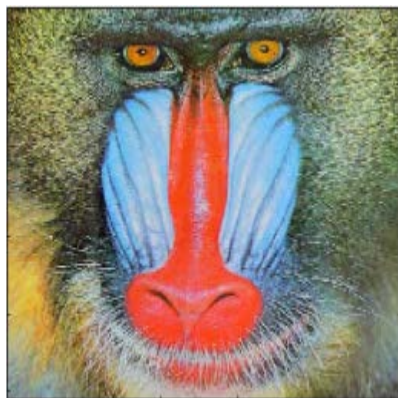


(f) Gráfica 6.
Algoritmo 4

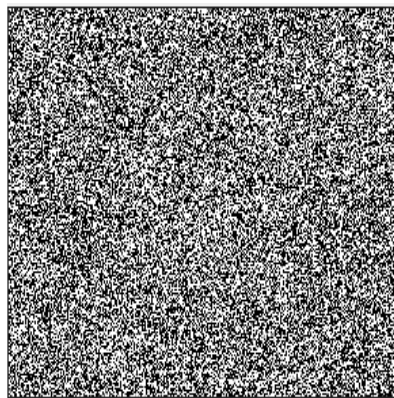


(g) Gráfica 7.
Algoritmo Propuesto

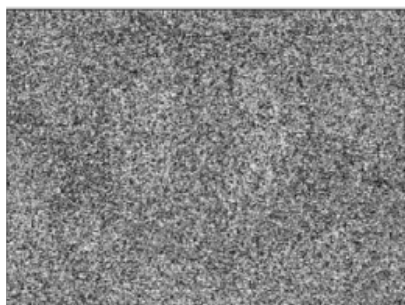
Figura 67 – A.1: Resultados del algoritmo aplicado en la imagen del babuino (512 x 512 píxeles) con un ruido de 50%. (a) Imagen original. (b) Imagen en escala de grises con el ruido 50%. (c) Imagen resultante del Algoritmo 2. (d) Imagen resultante del Algoritmo 1. (e) Imagen resultante del Algoritmo Propuesto. **Fuente.** Elaboración propia.



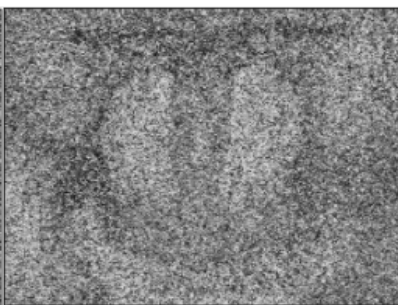
(a) Gráfica 1.



(b) Gráfica 2.



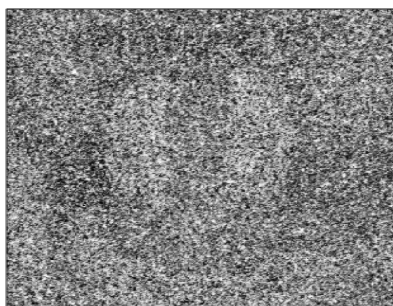
(c) Gráfica 3.
Algoritmo 1



(d) Gráfica 4.
Algoritmo 2



(e) Gráfica 5.
Algoritmo 3



(f) Gráfica 6.
Algoritmo 4



(g) Gráfica 7.
Algoritmo Propuesto

Figura 68 – A.2: Resultados del algoritmo aplicado en la imagen del babuino (512 x 512 píxeles) con un ruido de 90%. (a) Imagen original. (b) Imagen en escala de grises con el ruido 90%. (c) Imagen resultante del Algoritmo 2. (d) Imagen resultante del Algoritmo 1. (e) Imagen resultante del Algoritmo Propuesto. **Fuente.** Elaboración propia.



(a) Gráfica 1.



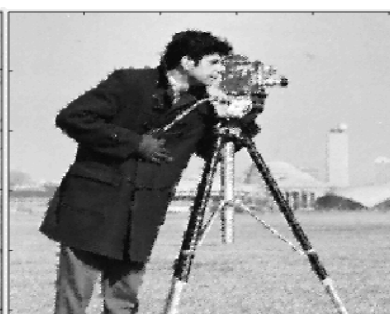
(b) Gráfica 2.



(c) Gráfica 3.
Algoritmo 1



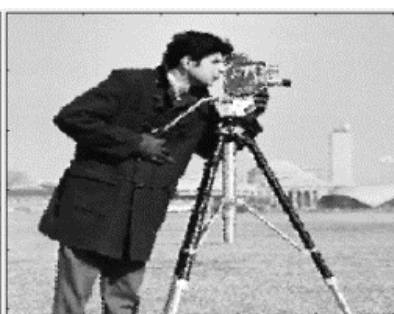
(d) Gráfica 4.
Algoritmo 2



(e) Gráfica 5.
Algoritmo 3



(f) Gráfica 6.
Algoritmo 4

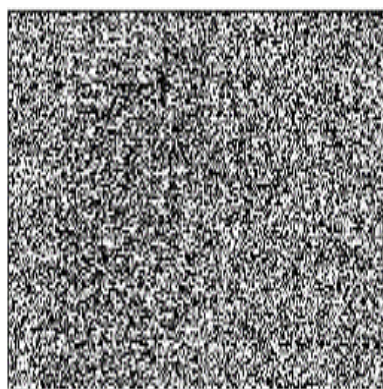


(g) Gráfica 7.
Algoritmo Propuesto

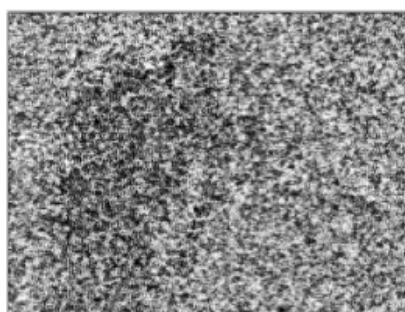
Figura 69 – A.3: Resultados del algoritmo aplicado en la imagen del Cameraman (256 x 256 pixeles) con un ruido de 50%. (a) Imagen original. (b) Imagen en escala de grises con el ruido 50%. (c) Imagen resultante del Algoritmo 2. (d) Imagen resultante del Algoritmo 1. (e) Imagen resultante del Algoritmo Propuesto. **Fuente.** Elaboración propia.



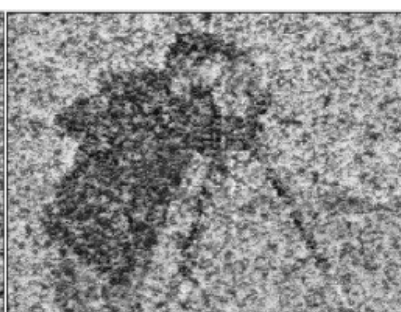
(a) Gráfica 1.



(b) Gráfica 2.



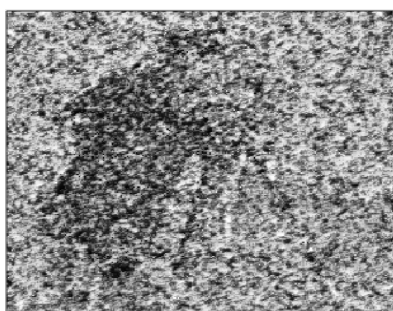
(c) Gráfica 3.
Algoritmo 1



(d) Gráfica 4.
Algoritmo 2



(e) Gráfica 5.
Algoritmo 3



(f) Gráfica 6.
Algoritmo 4



(g) Gráfica 7.
Algoritmo Propuesto

Figura 70 – A.4: Resultados del algoritmo aplicado en la imagen del Cameraman (256 x 256 píxeles) con un ruido de 90%. (a) Imagen original. (b) Imagen en escala de grises con el ruido 90%. (c) Imagen resultante del Algoritmo 2. (d) Imagen resultante del Algoritmo 1. (e) Imagen resultante del Algoritmo Propuesto. **Fuente.** Elaboración propia.

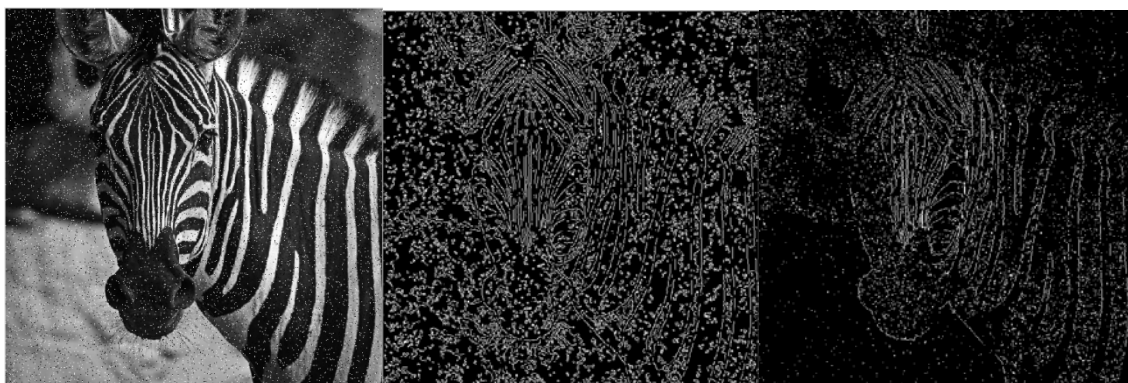
Apéndice B

Anexo 2

B.1. Resultados del método propuesto para la eliminación de ruido y la detección de bordes en imágenes a escala de grises

A continuación, se adjuntan las imágenes producto de la simulación del algoritmo basado en autómatas celulares, el cual permite la eliminación de ruido impulsivo y la detección de bordes. En las Figuras B.1., B.2., B.3., y B.4., se observan los resultados de la simulación en dos (2) imágenes comumente usadas en este proceso (Cebra y Lenna). Conforme con los datos de obtenidos en la sección 5.6., la calidad de la restauración y detección de bordes esta entre un 92.30% y un 97,21%, dado que el Error Cuadrático Medio para estas imágenes no supera el 0.08. De forma adicional, en la Figura B.5, se hace una comparación de algunas imágenes médicas confortando los resultados de los algoritmos Canny, Sobel y Prewitt frente a los arrojados por el Algoritmo Propuesto.

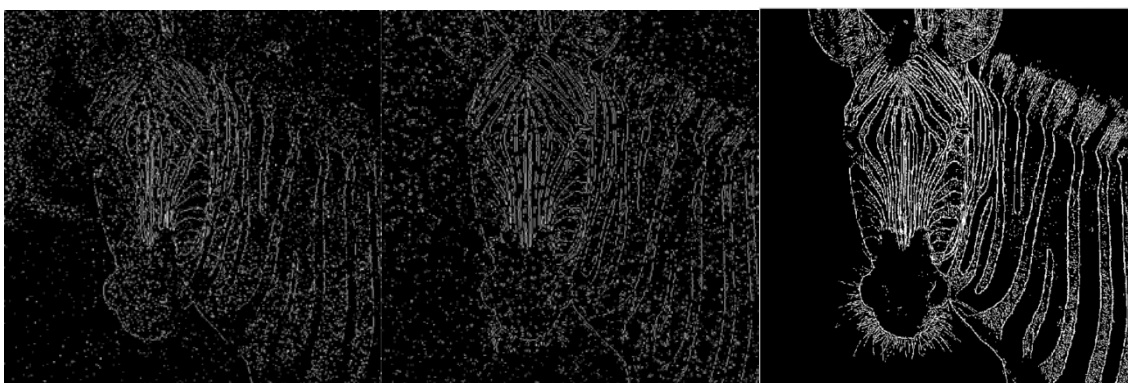
Cabe la pena resaltar, e que las siguientes ilustraciones muestran el resultado en imágenes afectadas con 5% de ruido, ya que acorde con el objetivo de este método, no sólo se suprime el ruido, sino detecta de forma efectiva los bordes presentes en una imagen.



(a) Gráfica 1.

(b) Gráfica 2.

(c) Gráfica 3.

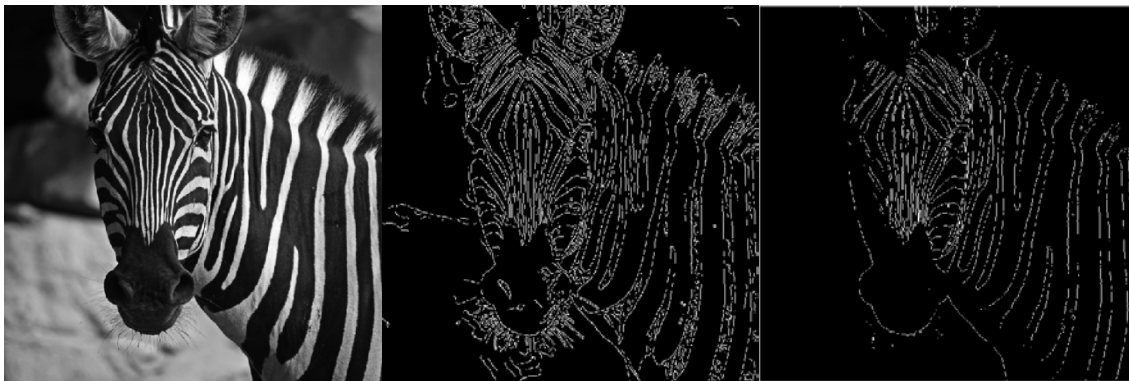


(d) Gráfica 4.

(e) Gráfica 5.

(f) Gráfica 6.

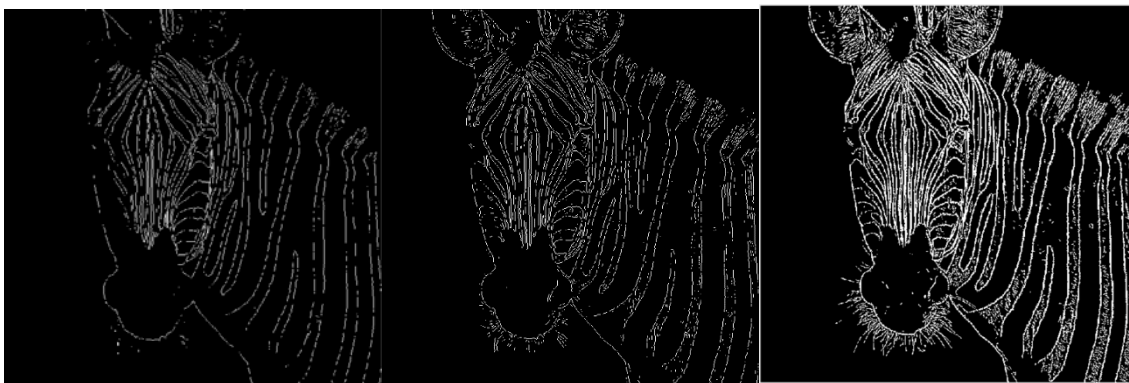
Figura 71 – B.1: Detección de bordes con ruido al 5%, cebra (a) Imagen base. (b) Imagen resultante algoritmo de Canny. (c) Imagen resultante algoritmo de Prewitt. (d) Imagen resultante algoritmo de Sobel. (e) Imagen resultante algoritmo Laplaciano de Gauss. (f) Imagen resultante algoritmo de propuesto. **Fuente.** Elaboración propia.



(a) Gráfica 1.

(b) Gráfica 2.

(c) Gráfica 3.

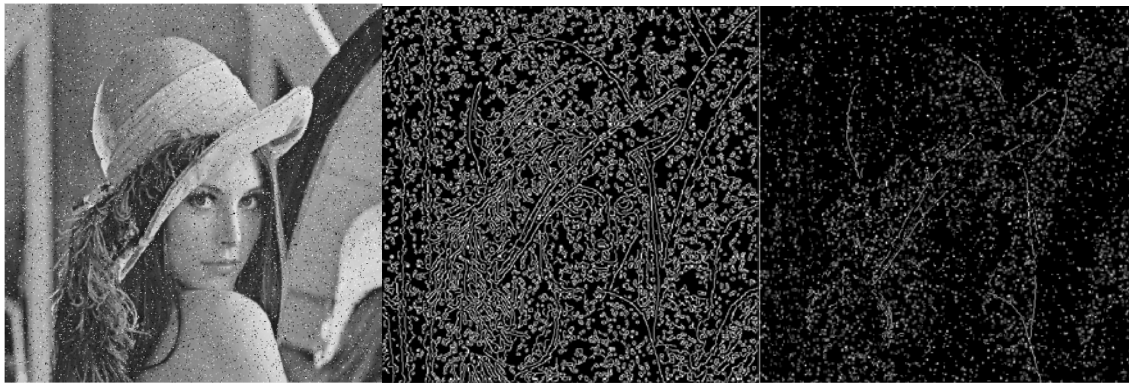


(d) Gráfica 4.

(e) Gráfica 5.

(f) Gráfica 6.

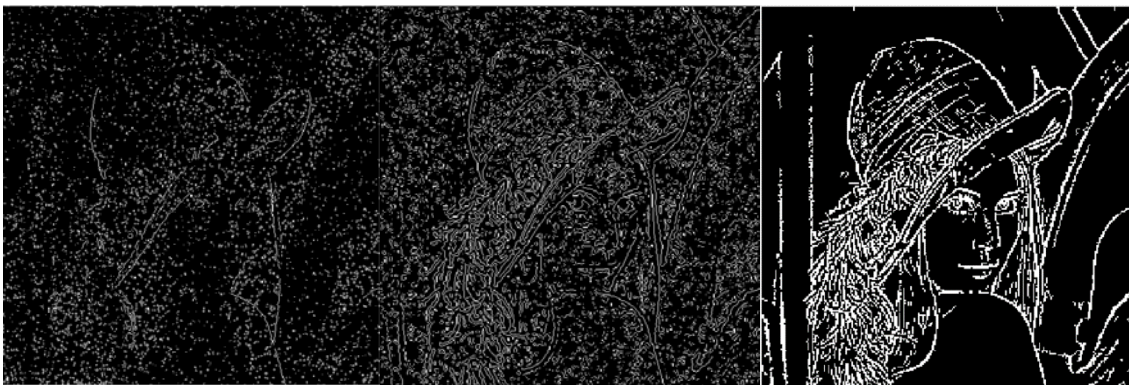
Figura 72 – B.2.: Detección de bordes sin ruido, cebra (a) Imagen base. (b) Imagen resultante algoritmo de Canny. (c) Imagen resultante algoritmo de Prewitt. (d) Imagen resultante algoritmo de Sobel. (e) Imagen resultante algoritmo Laplaciano de Gauss. (f) Imagen resultante algoritmo de propuesto. **Fuente.** Elaboración propia.



(a) Gráfica 1.

(b) Gráfica 2.

(c) Gráfica 3.



(d) Gráfica 4.

(e) Gráfica 5.

(f) Gráfica 6.

Figura 73 – B.3: Detección de bordes con ruido al 5%, Lenna (a) Imagen base. (b) Imagen resultante algoritmo de Canny. (c) Imagen resultante algoritmo de Prewitt. (d) Imagen resultante algoritmo de Sobel. (e) Imagen resultante algoritmo Laplaciano de Gauss. (f) Imagen resultante algoritmo de propuesto. **Fuente.** Elaboración propia.



(a) Gráfica 1.

(b) Gráfica 2.

(c) Gráfica 3.



(d) Gráfica 4.

(e) Gráfica 5.

(f) Gráfica 6.

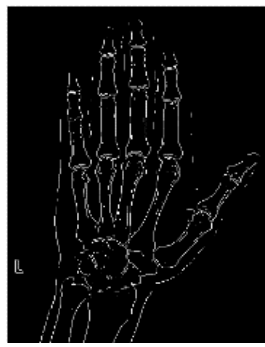
Figura 74 – B.4: Detección de bordes sin ruido, Lenna (a) Imagen base. (b) Imagen resultante algoritmo de Canny. (c) Imagen resultante algoritmo de Prewitt. (d) Imagen resultante algoritmo de Sobel. (e) Imagen resultante algoritmo Laplaciano de Gauss. (f) Imagen resultante algoritmo de propuesto. **Fuente.** Elaboración propia.



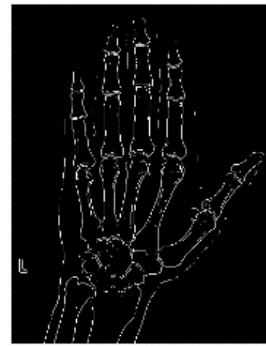
Gráfica (A-a)



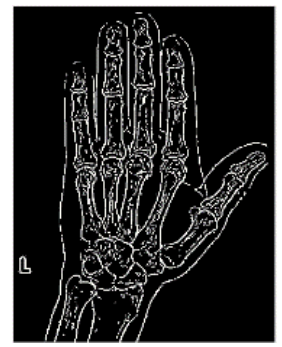
Gráfica (A-b)



Gráfica (A-c)



Gráfica (A-d)



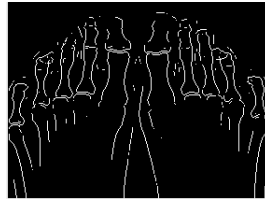
Gráfica (A-e)



Gráfica (B-a)



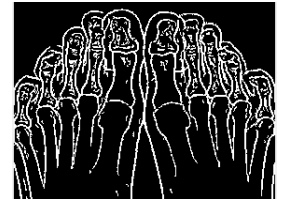
Gráfica (B-b)



Gráfica (B-c)



Gráfica (B-d)



Gráfica (B-e)



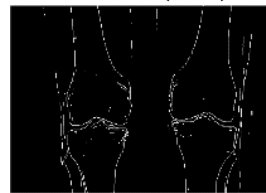
Gráfica (C-a)



Gráfica (C-b)



Gráfica (C-c)



Gráfica (C-d)



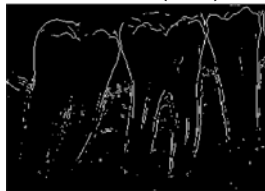
Gráfica (C-e)



Gráfica (D-a)



Gráfica (D-b)



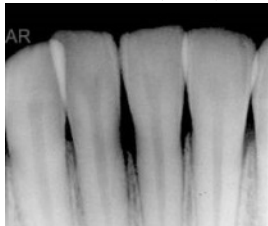
Gráfica (D-c)



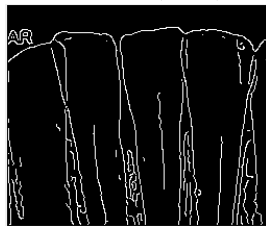
Gráfica (D-d)



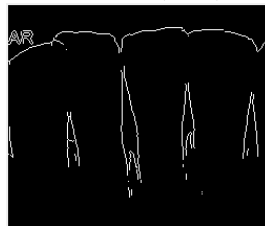
Gráfica (D-e)



Gráfica (E-a)



Gráfica (E-b)



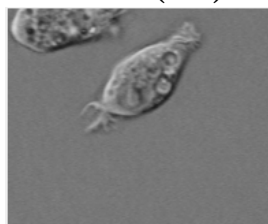
Gráfica (E-c)



Gráfica (E-d)



Gráfica (E-e)



Gráfica (F-a)



Gráfica (F-b)



Gráfica (F-c)



Gráfica (F-d)



Gráfica (F-e)

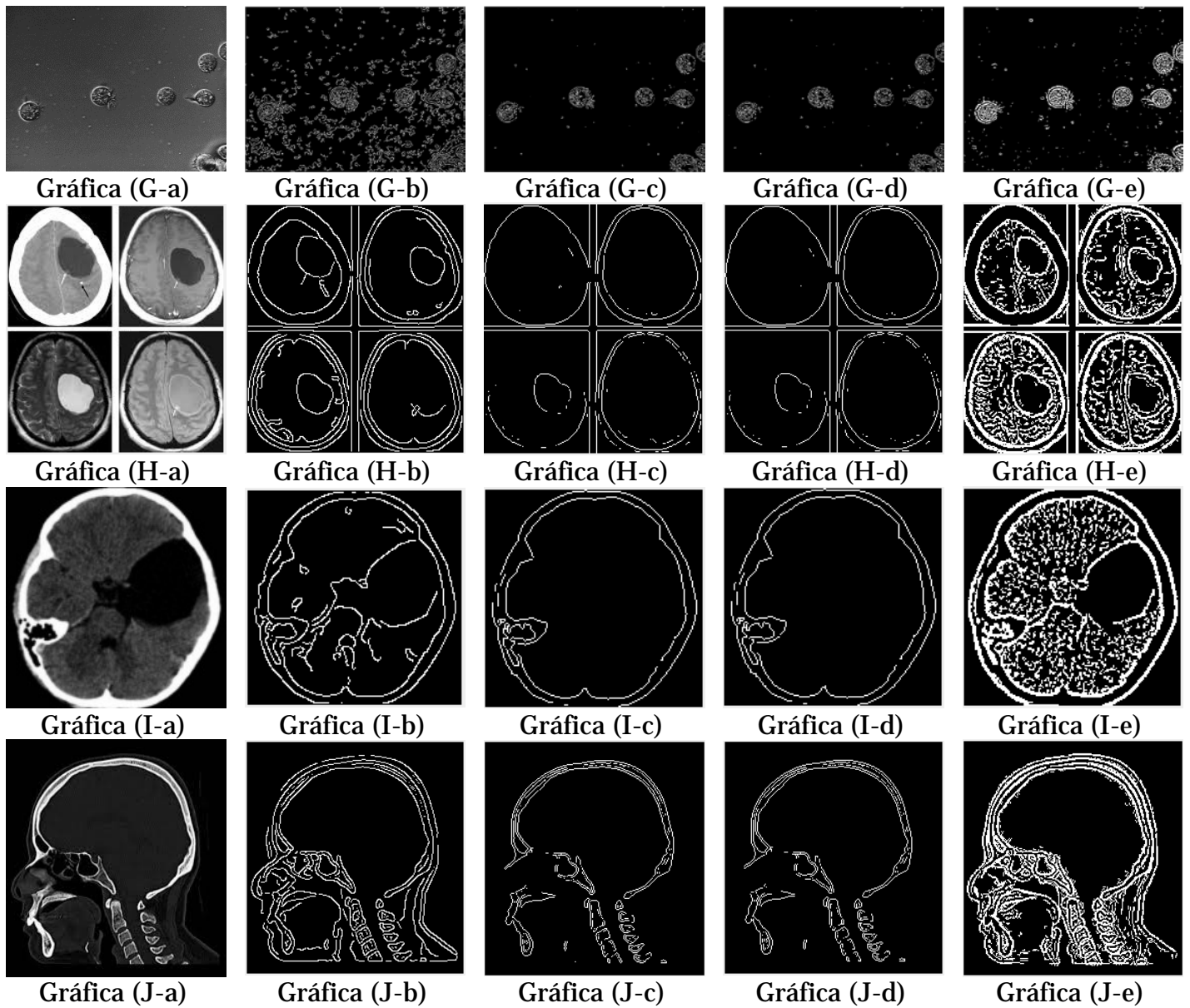


Fig. 75 – B.4: Rayos X. (A) Mano izquierda con la presencia de un tumor canceroso. (B) Pie derecho e izquierdo. (C) Rodillas. (D) Molares inferiores. (E) Incisivos inferiores. (F) Celda. (G) Microorganismos. (H) Tomografías del cerebro con presencia de tumor. (I) Tomografía de la presencia de tumor en la base del cráneo. (J) IRM de la cabeza desde la perspectiva lateral. Las columnas de la (a) a la (e) muestran los diferentes algoritmos: Canny, Sobel, Prewitt y Propuesta de Algoritmo. **Fuente.** Elaboración propia.

Bibliografía

- [Batouche et al., 2006] Batouche M., Meshoul S., Abbassene A. (2006) “On Solving Edge Detection by Emergence”. In: *Advances in Applied Artificial Intelligence. IEA/AIE 2006. Lecture Notes in Computer Science*, vol 4031. Springer, Berlin, Heidelberg. doi: 10.1007/11779568_86.
- [Benavides et al., 2009] Benavides, J., Calienes, W., Silva, C. (2009). Diseño de una Arquitectura para la Implementación de un Filtro Adaptativo RLS sobre un FPGA. In: *XV Workshop Iberchip*, pp. 25 -27, Buenos Aires, Argentina.
- [BenHamida and Schoenauer, 2000] BenHamida, S., Schoenauer, M. (2000). An Adaptive Algorithm for Constrained Optimization Problems. In: *Parallel Problem Solving from Nature PPSN VI. PPSN 2000*. Lecture Notes in Computer Science, vol 1917. Springer, Berlin, Heidelberg.
- [Betancourt et al, 2003] A. Betancourt, A. Mujica, H. Tapias, (2003). Procesamiento difuso de imágenes: filtro difuso para eliminar el ruido impulsivo. *Revista Ciencia, Investigación, Academia y Desarrollo. Ingeniería*, vol. 8, no. 2, pp. 40-46.
- [Bhardwaj and Mittal, 2012] Bhardwaj, S., Mittal, A. (2012). A Survey on Various Edge Detector Techniques. *Procedia Technology*, vol. 4, pp. 220-226. doi: 10.1016/j.protcy.2012.05.033.
- [Cano and Díaz, 2013] Cano, S.A. Díaz, N.E. (2013). *Implementación de un algoritmo para eliminación de ruido impulsivo en imágenes y análisis comparativo de tiempos de respuesta bajo arquitectura GPU y CPU*. Facultad de Ingeniería, Ingeniería de Sistemas, Universidad de Medellín.
- [Castillo et al., 2013] Castillo, R., Hernández, J. M., Inzunza, E., Torres, J. P. (2013). Procesamiento Digital de Imágenes Empleando Filtros Espaciales. In: *Décima Segunda*

Conferencia Iberoamericana en Sistemas, Cibernética e Informática: CISCI 2013, Orlando, Florida, EE.UU.

- [Chacón et al., 2015] Chacón M. M., Sandoval R. R, Vega P. J. (2015). *Percepción visual aplicada a la robótica*. Editorial: Alfaomega Grupo Editor, Mexico.
- [Chaikalis and Riera-Palou, 2009] Chaikalis, C., Riera-Palou, F. (2009). Efficient Receiver Implementation for Mobile Applications. *Handbook of Research on Heterogeneous Next Generation Networking: Innovations and Platforms*. Editorial: Advisory Board.
- [Chan et al., 2005]. Chan, R.H. Ho, C.W. Nikolova, M. (2005). Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization. *IEEE Transactions on Image Processing*, vol. 14, no. 10, pp. 1479-1485. doi: 10.1109/TIP.2005.852196.
- [Costa, 2009]. Costa, G. A. (2009). *Una Introducción a la Simetría*. Madrid, España: UNED - Universidad Nacional de Educación a Distancia.
- [Dalhoum et al., 2012] Dalhoum, A., Mahafzah, B.A., Awwad, A., Al-Dhamari, I., Ortega, A. Alfonseca, M. (2012). Digital image scrambling method based on two-dimensional cellular automata: a test of the lambda value. *IEEE Multimedia*, vol. 19, no. 4, pp. 28-36. doi: 10.1109/MMUL.2011.54.
- [Delorme, 1998] M. Delorme, (1998). An introduction to Cellular Automata, Reserach Report, *École Normale Supérieure de Lyon*, vol 98-37.
- [Dhillon, 2012] Dhillon, P. K. (2012). A Novel framework to Image Edge Detection using Cellular Automata. In: *Special Issue of International Journal of Computer Applications (0975 –8887). 3rd International IT Summit Confluence 2012 -The Next Generation Information Technology Summit*, pp. 1 – 5.
- [Diwakar et al., 2013] Diwakar, M., Kumar, P., Gupta, K. (2013, octubre). “Cellular automata based edge-detection for brain tumor”. In: *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Mysore, India. doi: 10.1109/ICACCI.2013.6637146.
- [Echeverri et al., 2011] Echeverri, J.A. Rudas, J.E. Toscano, R. Ballesteros, R. (2011). Eliminación de ruido impulsivo en imágenes a color, utilizando interpolación con funciones de base radial. *Revista Ingeniería*, vol. 16, no. 1, pp. 27-35. doi: 10.14483/23448393.3690.

- [Esakkirajan et al., 2011] Esakkirajan, S. Veerakumar, T. Subramanyam, A.N. Prem Chand, C.H. (2011). Removal of High Density Salt and Pepper Noise Through Modified Decision Based Unsymmetric Trimmed Median Filter. *IEEE Signal Processing Letters*, vol. 18, no. 5, pp. 287-290. doi: 10.1109/LSP.2011.2122333.
- [Esqueda and Palafox, 2005] J. J. Esqueda, L. E. Palafox. (2005). *Fundamentos de Procesamiento de Imágenes*. Mexicali, Baja California: Universidad Autónoma de Baja California. 2005.
- [Friedrich, 2002] Friedrich, G. (2002). *Introducción al procesamiento digital de señales. Técnicas Digitales III*. Universidad Tecnológica Nacional. Facultad Regional Bahía Blanca.
- [García and Viteri, 2009] García, C. C., Viteri, J. S. (2009). *Análisis e implementación de algoritmos para distorsionar imágenes con distintos tipos de ruido y aplicación de filtros en dos dimensiones para restaurarlas*. Escuela Superior Politécnica Del Litoral. Facultad De Ingeniería En Electricidad Y Computación.
- [Garnett et al., 2005] Garnett, R., Huegerich, T., Chui, C., He, W. (2005). A universal noise removal algorithm with an impulse detector. *IEEE Transactions on Image Processing*, vol. 14, no. 11, pp. 1747-1754. doi: 10.1109/TIP.2005.857261.
- [Gómez, 2013] N. Gómez, (2013). *Vida artificial: ciencia e ingeniería de sistemas complejos*. Editorial Universidad del Rosario, Bogotá, Colombia.
- [González and Woods, 2006] González, R. C., Woods, R. E. (2006). *Digital Image Processing (3rd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- [Kumar and Sahoo, 2010] Kumar, T., Sahoo, G. (2010). A novel method of edge detection using cellular automata. *International Journal of Computer Applications*, vol. 9, no. 4, pp. 38–44.
- [Lahoz-Beltra, 2010] Lahoz-Beltra R. (2010). *Bioinformática: Simulación, vida artificial e inteligencia artificial*. Editorial: Ediciones Díaz de Santos.
- [Malacara, 2002] D. Malacara, (2002). *Óptica tradicional y moderna*. Fondo de Cultura Económica. Tercera edición abreviada.
- [Márquez, 2013] Márquez, F. J. (2013). *Procesamiento y Análisis de Señales e Imágenes. Instituto de Ciencias Aplicadas y Tecnología*, UNAM. Posgrados de Ciencias de la Computación, Ingeniería Eléctrica y Física.

- [Montillet et al., 2016] Montillet, J. P., Yu, K., Bonenberg, L. K., Roberts G. W. (2016). Optimization Algorithms in Local and Global Positioning. *Handbook of Research on Modern Optimization Algorithms and Applications in Engineering and Economics*. Editorial: Advisory Board and List of Reviewers.
- [Nieto and Díaz] Nieto, A., Díaz, I. (2019). “Evolución del procesamiento digital de imágenes,” unpublished.
- [Neumann, 1952] Neumann, J.V. (1952). *Theory of automata*. Urbana University Press.
- [Öztürk and Akdemir, 2015] Öztürk, Ş., Akdemir, B. (2015, July). “Comparison of Edge Detection Algorithms for Texture Analysis on Glass Production,” *Procedia - Social and Behavioral Sciences*, vol. 195, pp. 2675-2682.
- [Pan et al., 2009] Pan, P.Z., Feng X.T., Zhou H. (2009, abril). Solid Cellular Automaton Method for the Solution of Physical Field. In: *World Congress on Computer Science and Information Engineering*, Los Angeles, CA, USA. doi: 10.1109/CSIE.2009.1018.
- [Pérez and Nakato, 2009] Pérez, H., Nakato, M. (2009). Signal Processing Techniques for Audio and Speech Applications. *Encyclopedia of Information Science and Technology*, Second Edition. Editorial Advisory Board.
- [Pitas, 1990] I. Pitas, A.N. Venetsanopoulos, (1990). *Nonlinear Digital Filters Principles and Applications*. Kluwer, Norwell, MA.
- [Platero, 2009] Platero D. C. (2009). *Procesamiento morfológico*. Dpto. Electrónica, Automática e Informática Industrial. Universidad Politécnica de Madrid.
- [Popovici and Popovici, 2002] Popovici, A., Popovici, D. (2002). Cellular Automata in Image Processing. In: *15th International Symposium on the Mathematical Theory of Networks and Systems*, pp. 6.
- [Roberts and Tesman, 2011] Roberts, F., Tesman, B. (2011). *Applied Combinatorics*, (2nd Edition). Chapman & Hall/CRC.
- [Rosin, 2006]. Rosin, P. (2006). Training cellular automata for image processing. *IEEE Transactions on Image Processing*, vol. 15, no. 7, pp. 2076-2087. doi: 10.1109/TIP.2006.877040.
- [Rosin and Sun, 2013]. Rosin, P., Sun, X. (2013) Edge Detection Using Cellular Automata.

Cellular Automata in Image Processing and Geometry. Emergence, Complexity and Computation, vol. 10, pp. 85-103. doi: 10.1007/978-3-319-06431-4_5.

[Rosin and Sun, 2011]. Rosin, P., Sun, X. (2011). Cellular Automata as a Tool for Image Processing. *Emerging Topics in Computer Vision and its Applications*, vol. 1, pp. 233-251. doi: 10.1142/9789814343008_0012.

[Rueda and Torres, 2013] Rueda, A. Torres, W. (2013). Autómatas celulares para la segmentación y clasificación de imágenes multispectrales. In: *V Jornadas Nacionales de Geomática y IX Jornadas de Educación en Percepción Remota en el Ámbito de Mercosur*.

[Sahin et al., 2014] Sahin, U., Uguz, S., Sahin F., (2014). Salt and pepper noise filtering with fuzzy-cellular automata. *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 59-69.

[Sahota et al., 1994] Sahota, P., Daemi, M., Elliman, D. (1994). Training genetically evolving cellular automata for image processing. In: *Proceedings of ICSIPNN '94. International Conference on Speech, Image Processing and Neural Networks*, Hong Kong, Hong Kong. doi: 10.1109/SIPNN.1994.344802.

[Selvapeter and Hordijk, 2013] Selvapeter, J., Hordijk, W. (2013) “Genetically evolved cellular automata for image edge detection”. In: *Proceedings of the International Conference on Signal, Image Processing and Pattern Recognition*.

[Slatnia et al., 2007] Slatnia S., Batouche M., y Melkemi K.E. (2007) Evolutionary Cellular Automata Based-Approach for Edge Detection. In: *Applications of Fuzzy Sets Theory. WILF 2007*, vol. 4578, pp. 404–411. doi: 10.1007/978-3-540-73400-0_51.

[Shukla, 2016] Shukla, A. P. (2016). Training Cellular Automata for Image Edge Detection. *Romanian Journal of Information Science and Technology*, vol. 19, no. 4, pp. 338–359.

[Tosina, 2008] Tosina M. M. (2008). *Diseño de un filtro morfológico para la reducción de fluorescencia en espectros RAMAN*. (Tesis de grado). Departamento de Teoría de la Señal y Comunicaciones. Universidad Politécnica de Catalunya.

[Tourtownis et al., 2018] Tourtownis, D., Mitianaudis, N., Sirakaulis, G. (2018), Salt-n-pepper noise filtering using Cellular Automata. *Journal of Cellular Automata*, vol. 13, no. 1-2, pp. 81-101.

[Vásquez, 2016] Vásquez, F. J. (2016). *Localización eficiente en detección de bordes en imágenes adaptando el algoritmo ABC*. (Tesis de grado) Facultad de Ingeniería.

Universidad de Concepción.

- [Wongthanavasus and Sadananda, 2003] Wongthanavasus, S., Sadananda, R. (2003). A CA-based edge operator and its performance evaluation. *Journal of Visual Communication and Image Representation*, vol. 14, no. 2, pp. 83-96. doi: 10.1016/S1047-3203(03)00022-1.
- [Xiao and Hui, 2010] Xiao, W., Hui, X. (2010, December). "An improved canny edge detection algorithm based on predisposal method for image corrupted by gaussian noise," in 2010 World Automation Congress, Kobe, Japan, pp. 113-116.
- [Yang et al., 2002] Yang, C., Ye, H., Wang, G. (2002). Cellular automata modeling in edge recognition. In: *Seventh International Symposium on Artificial Life and Robotics*, pp. 128-132.