



EDUCACIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

Tecnológico Nacional de México

Centro Nacional de Investigación
y Desarrollo Tecnológico

Tesis de Doctorado

Cálculo Fraccionario Discreto Aplicado en
Procesamiento de Imágenes y Señales

presentada por

MC. Luis Felipe Avalos Ruiz

como requisito para la obtención del grado de

**Doctor en Ciencias en Ingeniería
Electrónica**

Director de tesis

Dr. José Francisco Gómez Aguilar

Codirector de tesis

Dr. Carlos Jesús Zúñiga Aguilar

Cuernavaca, Morelos, México. Diciembre de 2023.

 <small>Centro Nacional de Investigación y Desarrollo Tecnológico</small>	ACEPTACIÓN DE IMPRESIÓN DEL DOCUMENTO DE TESIS DOCTORAL	Código: CENIDET-AC-006-D20
		Revisión: 0
	Referencia a la Norma ISO 9001:2008 7.1, 7.2.1, 7.5.1, 7.6, 8.1, 8.2.4	Página 1 de 1

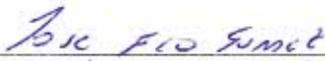
Cuernavaca, Mor., a 26 de enero de 2024

Dr. Carlos Manuel Astorga Zaragoza
Subdirector Académico
Presente

At'n: Dr. Víctor Manuel Alvarado Martínez
Presidente del Claustro Doctoral del Departamento De Ing. Electrónica

Los abajo firmantes, miembros del Comité Tutorial del estudiante Luis Felipe Ávalos Ruiz manifiestan que después de haber revisado el documento de tesis titulado "Cálculo Fraccionario Discreto Aplicado en Procesamiento de Imágenes y Señales", realizado bajo la dirección del Dr. José Francisco Gómez Aguilar y la codirección del Dr. Carlos Jesús Zúñiga Aguilar, el trabajo se ACEPTA para proceder a su impresión.

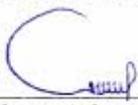
ATENTAMENTE



 Dr. José Francisco Gómez Aguilar
 CENIDET



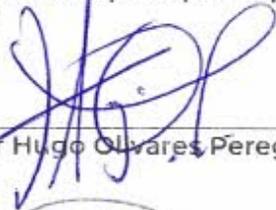
 Dr. Carlos Jesús Zúñiga Aguilar
 Panzura Data Service



 Dra. Ma. Guadalupe López López
 CENIDET



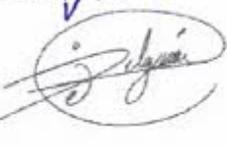
 Dr. Víctor Manuel Alvarado Martínez
 CENIDET



 Dr. Víctor Hugo Olivares Peregrino
 CENIDET



 Dr. Janniel García Morales
 CENIDET



 Dr. Jorge Enrique Lavín Delgado
 UPEGuerrero

cc.p: MTI María Elena Gómez Torres/Jefa del Departamento de Servicios Escolares.
 cc.p: Dr. Janniel García Morales/Jefe del Departamento de Ingeniería Electrónica
 cc.p: Expediente



Cuernavaca, Mor.,
No. De Oficio:
Asunto:

30/enero/2024
SAC/018/2024
Autorización de
impresión de tesis

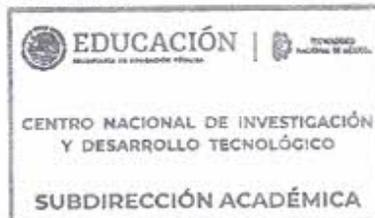
LUIS FELIPE ÁVALOS RUIZ
CANDIDATO AL GRADO DE DOCTOR EN CIENCIAS
EN INGENIERÍA ELECTRÓNICA
P R E S E N T E

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado **"Cálculo Fraccionario Discreto Aplicado en Procesamiento de Imágenes y Señales"**, ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

ATENTAMENTE

Excelencia en Educación Tecnológica®
"Conocimiento y tecnología al servicio de México"



CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO

C. c. p. Departamento de Ingeniería Electrónica
Departamento de Servicios Escolares

CMAZ/lmz



Dedicatoria

A mis padres, mis hermanos
y mi hermana

Agradecimientos

Siempre estaré agradecido de recibir el apoyo de mi maravillosa familia. Gracias a mis padres por su amor incondicional y a impulsarme a tomar nuevos desafíos. A mi hermana Claudia y mis hermanos Carlos y Daniel, por siempre estar dispuestos a escucharme, alentarme y velar por mi salud.

A los amigos con los que compartí alegrías y desvelos durante todo este tiempo, Iván, Fabiola, Cesar, Héctor, Jesús, Sabido, Diana, Chava, Gonzalo, Alexis, Santiago. Gracias por sus consejos, aprecio y amistad a lo largo de estos años.

A Andrea, por acompañarme en los desvelos y no permitirme quebrarme bajo el estrés. Gracias por ser mi confidente y por ayudarme a superar los momentos difíciles.

Al Dr. Carlos Jesús Zúñiga Aguilar, quien ha sido una figura a seguir para mi desde que cursaba mi licenciatura. Le estoy profundamente agradecido por todo su tiempo, paciencia, amistad y confianza. Es sin duda una de las personas mas nobles y dedicadas que he conocido en mi vida. Gracias por todos los valiosos consejos que me ha brindado, que me han permitido crecer y aprender.

Al Dr. José Francisco Gómez Aguilar por su tiempo, apoyo y su paciencia. Por abrirme las puertas a su grupo de trabajo, donde he tenido la oportunidad de aprender y crecer académica y profesionalmente. Su compromiso siempre ha sido admirable y su experiencia invaluable para mi desarrollo académico.

A mis revisores: Dra. María Guadalupe López López, Dr. Víctor Manuel Alvarado Martínez, Dr. Víctor Hugo Olivares Peregrino y al Dr. Jarniel García Morales por sus acertados comentarios y sugerencias que sin duda, enriquecieron este trabajo.

Al Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) por permitirme crecer en esta institución durante todas las etapas de mi educación superior.

Al Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT) por el apoyo económico otorgado mediante la beca doctoral y al tecnológico Nacional de México (TecNM) por facilitar las instalaciones y equipo que permitieron este trabajo.

Resumen

Las metodologías de procesamiento de señales basadas en técnicas de cálculo fraccionario de orden discreto presentan propiedades atractivas. Por ejemplo, en la mejora de características de imágenes, estos algoritmos permiten mejorar ciertas características, como los bordes, al tiempo que mantienen detalles de baja frecuencia relacionados con la textura que suelen ser descartados por algoritmos de orden entero equivalentes. Otra característica importante de estos algoritmos radica en su robustez frente a diferentes tipos de ruido.

Este trabajo se divide en dos partes principales. En primer lugar, se propone un esquema de sincronización unilateral basado en controladores para sistemas caóticos discretos de orden fraccionario y variable. Para demostrar la relativamente baja complejidad algorítmica del método, se realizó su implementación en placas de desarrollo Arduino UNO. Se muestra una posible aplicación en la generación de secuencias pseudoaleatorias para algoritmos de criptografía. Los resultados obtenidos por el algoritmo de encriptación propuesto muestran un aumento en la seguridad contra ciertos tipos de ataques sin comprometer el tiempo de computación.

La segunda parte del documento consiste en la propuesta de dos metodologías basadas en el cálculo fraccional de orden discreto en problemas de clasificación. Primero, se propone una metodología de suavizado de señales basada en un observador de orden fraccionario para la etapa de preprocesamiento, que se aplicó a fonocardiogramas no segmentados de la base de datos *PhysioNet/CinC Challenge 2016*. Las características de clasificación se obtuvieron en el dominio tiempo-frecuencia empleando la Transformada Fraccionaria de Fourier (FRFT). Los modelos de clasificación resultantes se utilizaron para la detección de enfermedades relacionadas con el corazón. Finalmente, se presenta un esquema de clasificación para la base de datos SAT-6, que consta de imágenes satelitales de diversos tipos de terreno etiquetadas en seis clases diferentes. Las características de clasificación se extrajeron basándose en el gradiente de orden fraccionario. Se entrenaron tres CNNs en un esquema de clasificación jerárquica. Los resultados obtenidos muestran una mayor precisión, incluso en comparación con CNNs preentrenadas.

Abstract

Signal processing methodologies based on discrete order fractional calculus techniques present attractive properties. For instance in image feature enhancement, these algorithms allow for the improvement of certain characteristics such as borders while maintaining low frequency details related to texture which are usually discarded by equivalent integer order algorithms. Another important feature of these algorithms lie in their robustness for different types of noise.

This work is divided in two main parts. First a controller based unilateral synchronization scheme for fractional, variable order discrete chaotic systems is proposed. To demonstrate the relatively low algorithmic complexity of the method its implementation in Arduino UNO development boards was realized. A possible application in the generation of pseudorandom sequences for cryptography algorithms is shown. The obtained results by the proposed encryption algorithm exhibit increased security against certain types of attacks without compromising computing time.

The second part of the document consists on the proposal of two methodologies based on discrete fractional order calculus in classification problems. First, a signal smoothing methodology is proposed based on a fractional order observer for the preprocessing stage. which was applied to unsegmented phonocardiograms from the *PhysioNet/CinC Challenge 2016* database. The classification features were obtained in the time-frequency domain employing the Fractional Order Fourier Transform (FRFT). The resulting classification models were used for the detection of heart related diseases. Finally, a classification scheme is presented for the SAT-6 database, which consists of satellite images of various types of terrain labeled into six different classes. The classification features were extracted based on the fractional order gradient. Three CNNs were trained in a hierarchical classification scheme. The obtained results demonstrate higher accuracy, even when compared to pre-trained CNNs.

Contenido

Resumen	VII
Abstract	VIII
Contenido	IX
Lista de Figuras	XI
Lista de Tablas	XIII
1. Introducción	1
1.1. Estado del Arte	2
1.1.1. Antecedentes Institucionales	7
1.2. Planteamiento del Problema	8
1.3. Justificación	8
1.4. Hipótesis	9
1.5. Objetivos	9
1.6. Metas	10
1.7. Organización del Documento	10
2. Cálculo Fraccionario Discreto	11
2.1. Cálculo en Escalas de Tiempo	11
3. Procesamiento de Señales	17
3.1. Transformada de Fourier Fraccionaria	17
3.2. Clasificación	19
3.2.1. Modelos de Clasificación	19
4. Control de Sistemas Caóticos Discretos de Orden Fraccionario y su Aplicación en Sistemas Criptográficos	23
4.1. Controlador Propuesto	24
4.1.1. Caso 1. Mapa de Tinkerbelle	26
4.1.2. Caso 2. Mapa del Ave Mítica	27
4.1.3. Caso 3. Mapa Ave del Paraíso	31

4.1.4. Implementación	33
4.2. Aplicación en Criptografía	35
4.2.1. Generador de Secuencias Pseudoaleatorias	35
4.2.2. Codificación Basada en ADN	37
4.2.3. Análisis de Seguridad	39
4.3. Conclusiones Parciales	43
5. Clasificación de Señales Basado en Cálculo Discreto de Orden Fraccionario	56
5.1. Caso 1. Fonocardiogramas	57
5.1.1. Base de Datos	58
5.1.2. Preprocesamiento	59
5.1.3. Extracción de Características	62
5.1.4. Clasificación	63
5.2. Caso 2. Imágenes Satelitales	68
5.2.1. Base de Datos	69
5.2.2. Preprocesamiento y Extracción de Características	70
5.2.3. Clasificación	73
5.3. Conclusiones Parciales	76
6. Conclusiones	77
6.1. Trabajos Futuros	78
Bibliografía	79
Anexos	88

Lista de Figuras

1.1.	Diagrama del estado del arte de análisis de sistemas caóticos de orden fraccionario y su aplicación en criptografía.	4
1.2.	Diagrama del estado del arte de aplicaciones de cálculo discreto de orden fraccionario en clasificación basada en señales.	6
2.1.	Función $a^{(v)}(k) \forall k \in [0, 20]$ para distintos valores de $v \in [0, 1]$. Si bien la función mostrada es discreta, líneas continuas son utilizadas para claridad.	14
2.2.	Comparación de trayectorias del mapa logístico para diferentes valores de v	15
3.1.	Rotación del dominio tiempo-frecuencia al aplicar la transformada de Fourier de orden fraccionario.	18
3.2.	Estructura básica de un árbol de decisión.	21
3.3.	Esquema simplificado de la estructura de la CNN.	22
4.1.	Esquema general de encriptación con generación de secuencias pseudoaleatorias basada en sistemas caóticos.	24
4.3.	Retrato fase del mapa caótico de orden fraccionario Tinkerbell con ordenes de diferencia $v(k)$ definidos por funciones escogidas arbitrariamente.	27
4.4.	Sincronización de dos mapas Tinkerbell de orden fraccionario mediante la metodología propuesta.	28
4.5.	Retrato fase del mapa caótico ave mítica de orden fraccionario para diversos valores de $v(k)$, escogidos de manera arbitraria.	29
4.6.	Resultados del esquema de control propuesto para la sincronización de dos mapas ave mítica de orden fraccionario.	30
4.7.	Retrato fase del mapa caótico de orden fraccionario ave del paraíso con ordenes de diferencia $v(k)$ definidos por funciones escogidas arbitrariamente.	32
4.8.	Esquema de sincronización propuesto para dos mapas caóticos ave del paraíso de orden fraccionario.	33
4.9.	Implementación del esquema de sincronización propuesto en tarjetas Arduino UNO.	34
4.10.	Sincronización de dos mapas ave del paraíso de orden fraccionario mediante la metodología propuesta. Se emplean líneas continuas por claridad.	35
4.11.	Diagrama de bifurcación para el mapa caótico fraccionario de Tinkerbell con respecto a cambios en el orden $v(k)$	36

4.12. Esquema de la metodología propuesta para encriptación de imágenes RGB.	40
4.13. Resultados obtenidos aplicando la metodología propuesta en la imagen “Airplane”.	44
4.14. Resultados obtenidos aplicando la metodología de encriptación propuesta en la imagen “Baboon”.	45
4.15. Resultados obtenidos aplicando la metodología propuesta en la imagen “House”.	46
4.16. Resultados obtenidos aplicando la metodología propuesta en la imagen “Peppers”.	47
4.17. Resultados obtenidos aplicando la metodología propuesta en la imagen “Sailboat On Lake”.	48
4.18. Resultados obtenidos aplicando la metodología propuesta en la imagen “Splash”.	49
4.19. Resultados obtenidos aplicando la metodología de encriptación propuesta en la imagen “Tree”.	50
4.20. Resultados obtenidos aplicando la metodología propuesta en la imagen “White”.	51
4.21. Resultados ante ruido en el medio de transmisión para la imagen “Airplane”.	52
4.22. Resultados ante ruido en el medio de transmisión para la imagen “Baboon”.	53
4.23. Resultados ante ruido en el medio de transmisión para la imagen “House”.	54
4.24. Resultados ante ruido en el medio de transmisión para la imagen “Peppers”.	55
5.1. Fases de la clasificación basada en características.	56
5.2. Fonocardiograma sintético obtenido mediante funciones de Gabor.	58
5.3. Señal de fonocardiograma suavizada con metodología propuesta.	61
5.4. Esquema general del algoritmo PSO.	65
5.5. Esquema general de optimización utilizado para la selección de ordenes v y α para las etapas de prerprocesamiento y extracción de características respectivamente.	65
5.6. Comparación de medias por clase entre metodología de orden entero y propuesta.	66
5.7. Matriz de confusión de mejor modelo encontrado y ordenes optimizados con algoritmo PSO.	67
5.8. Esquema general de la metodología propuesta para la clasificación de fonocardiogramas.	68
5.9. Imágenes muestra de cada clase de la base de datos SAT-6.	70
5.10. Comparación de gradientes en imágenes ante diversos ordenes de v con acercamientos a objetos cercanos y lejanos en la imagen.	72
5.11. Ejemplos de características empleadas para cada clase.	73
5.12. Matriz de confusión de la clasificación de imágenes provenientes de la base de datos SAT-6 con el modelo entrenado	75
5.13. Esquema general de la metodología propuesta para la clasificación de imágenes satelitales basada en el gradiente discreto de orden fraccionario.	75

Lista de Tablas

3.1. Comparación de algunas características relevantes de diferentes metodologías de discretización para la transformada fraccionaria de Fourier.	18
4.1. Valores-P obtenidos para diversos análisis incluidos en el banco de pruebas de NIST. . .	37
4.2. Configuraciones posibles de codificación basadas en ADN.	38
4.3. Tabla de verdad para el operador XOR empleando la segunda regla de codificación. . . .	38
4.4. Índices de desempeño ante ataques estadísticos para la metodología de encriptación propuesta.	42
4.5. Comparación de la metodología propuesta con trabajos similares reportados en la literatura.	42
5.1. Comparación de índices de desempeño para modelos comunes de clasificación empleando metodología propuesta de preprocesamiento con extracción de características en el dominio de Fourier Fraccionario.	64
5.2. Comparación de resultados de metodología propuesta contra algunos trabajos relevantes del estado del arte.	67
5.3. Estructura de CNN utilizada para la clasificación de imágenes satelitales.	73
5.4. Comparación de resultados obtenidos con trabajos relevantes reportados en la literatura.	75

Nomenclatura

A	Adenina	PST	Transformada de Estiramiento de Fase
ADAM	Estimación de Momento Adaptativo	PWM	Modulación por Ancho de Pulso
ADN	Ácido Desoxirribonucleico	RGB	Rojo, Verde, Azul
ANN	Red Neuronal Artificial	RTV	Variación Total Relativa
C	Citosina	S1	Primer sonido del ciclo cardiaco
CENIDET	Centro Nacional de Investigación y Desarrollo Tecnológico	S2	Segundo sonido del ciclo cardiaco
CNN	Red Neuronal Convolutiva	SRAM	Memoria Estática de Acceso Aleatorio
DCT	Transformada Coseno Discreta	SVM	Máquina de Vectores de Soporte
DFRFT	Transformada Discreta Fraccionaria de Fourier	T	Timina
DNN	Red de Aprendizaje Profundo	α	Orden de transformación
EEG	Electroencefalograma	\bar{x}, \bar{y}	Estados de sistema líder
EEPROM	Memoria de Solo Lectura Programable y Borrable Eléctricamente	Δ^v	Diferencia de orden fraccionario
FOBDS	Diferencia/Suma de Orden Fraccionario	$\Gamma(\cdot)$	Función Gamma
FPGA	Matriz de Puerta Programable en Campo	λn_i	Ventana normalizada en rango $[0, 1]$
FRFT	Transformada Fraccionaria de Fourier	λ_i	i-ésima ventana de una señal
G	Guanina	\mathbb{R}	Números enteros
GL	Grünwald-Letnikov	\mathbb{R}	Números reales
KNN	K Vecinos Más Cercanos	\mathbb{T}	Escala de tiempo
LFSR	Registro de Desplazamiento con Retroalimentación Lineal	$\mathcal{F}\{\cdot\}$	Transformada de Fourier
NIST	Instituto Nacional de Estándares y Tecnología	$\mathcal{F}_\alpha\{\cdot\}$	Transformada de Fourier Fraccionaria
Núcleo IP	Núcleo de propiedad intelectual de semiconductores	$\mu(\cdot)$	Distancia entre los instantes actual y siguiente
OMS	Organización Mundial de la Salud	Ψ_i	Sistema caótico
		$\rho(\cdot)$	Valor anterior en una escala de tiempo
		$\rho(\cdot, \cdot)$	Coefficiente de correlación de Pearson
		$\sigma(\cdot)$	Valor siguiente en una escala de tiempo
		$\Upsilon(\cdot, \cdot)$	Matriz de características
		ζ_i	Funciones de diseño para controlador
		a, b, c, d, l, r	Parámetros
		$a^{(v)}(k)$	Función auxiliar para calcular coeficientes de la derivada Δ^v
		$ACC(\cdot)$	Exactitud
		$C_i(\cdot, \cdot)$	Imagen Cifrada
		$D(\cdot, \cdot)$	Distancia euclidiana
		e_i	Función de error

$Ex(\cdot)$	Exhaustividad	$M_i(\cdot, \cdot)$	Máscara
$f^\Delta(\cdot)$	Derivada de Hilger de $f(\cdot)$	$mod_i(\cdot)$	Operador módulo con base i
$F_1(\cdot)$	Valor-F	$NPCR$	Tasa de Cambio de Píxeles
f_i, g_i	Funciones auxiliares	$P(\cdot)$	Precisión
$G(\cdot)$	Función gaussiana	s_1, s_2	Parámetros de función seno
$GN(\cdot)$	Función gaussiana normalizada	S_i	Suma del error
$H(\cdot)$	Entropía	t	Tiempo
I_i	Ganancia integral	u_i	Ley de control
$J(\cdot)$	Función costo	$UACI$	Cambio de intensidad promedio ponderado
k	Instante en tiempo discreto	$V(\cdot)$	Suma del error
$K_\alpha(\cdot, \cdot)$	Kernel de transformación para orden α de la FRFT	$v(\cdot)$	Orden de diferencia
k_i	Ganancia proporcional	v_{in}^k	Velocidad de la partícula en el instante k
L	Ganancia	x, y	Estados de sistema seguidor
M	Matriz de confusión	x_{in}^k	Dimensión n de la partícula i

Capítulo 1

Introducción

Por señal nos referimos a una función que contiene información sobre un sistema o fenómeno. La clasificación de señales puede realizarse de acuerdo con la continuidad de la variable dependiente que las define, obtenemos así las categorías de señales continuas y discretas. El uso de computadoras para el procesamiento y análisis de señales demanda el uso de señales discretas, por lo que su estudio resulta importante.

El procesamiento de señales se refiere al conjunto de algoritmos y técnicas de análisis y transformación de señales, los cuales son con frecuencia utilizados para obtener la mejor información posible para la toma de decisiones. Por ejemplo, en un contexto médico, algoritmos de procesamiento de señales pueden ser empleados como herramientas de apoyo a diagnóstico de enfermedades. Debido a que las señales pueden provenir de una variedad amplia de fuentes y exhibir comportamientos complejos (como autosimilitud), el desarrollo de metodologías de procesamiento de señales es un tema que se encuentra en constante evolución.

El cálculo es la rama de las matemáticas que estudia sistemas que cambian. Así como la aritmética tiene sus bases en las operaciones de suma y resta, el cálculo tiene su base en los operadores diferenciales e integrales. El operador derivada representa el cambio instantáneo de una función. El operador integral se refiere a la suma acumulada de dichos incrementos. Estos operadores pueden ser empleados para describir sistemas en términos de sus cambios con relación a una variable independiente. Sin embargo, a medida que aumenta la complejidad de los sistemas a analizar, también aumenta la necesidad de técnicas de modelado más sofisticadas. Ciertas características de los sistemas, como las impurezas en materiales, factores ambientales o comportamientos de memoria, pueden plantear desafíos al intentar describir con precisión un fenómeno.

El cálculo fraccionario es el campo matemático que aborda problemas que involucran con órdenes diferenciales o integrales no enteros. Aunque este concepto es prácticamente tan antiguo como el cálculo de orden entero, en años recientes ha adquirido popularidad gracias a sus propiedades. En términos simples, el cálculo fraccionario permite describir el efecto de memoria en la dinámica de los sistemas, dicho fenómeno puede estar asociado con disipación. Diversos fenómenos que ocurren naturalmente se pueden representar de manera más precisa por medio de herramientas del cálculo fraccionario.

En el caso particular del procesamiento digital de señales, las técnicas basadas en cálculo fraccionario presentan propiedades atractivas, como la robustez contra el ruido y mejores características para la representación de información. Por ejemplo, en aplicaciones de mejora de bordes, se ha encontrado que las metodologías centradas en el cálculo fraccionario discreto permiten mantener ciertas características en imágenes relacionadas con componentes de baja frecuencia, las cuales suelen ser descartadas como ruido en metodologías equivalentes de orden entero. La principal desventaja de los algoritmos basados en el cálculo fraccionario suele radicar en su elevado costo computacional.

En este trabajo se discuten dos metodologías de procesamiento de señales. Primero se mostrará el uso de técnicas de cálculo fraccionario discreto para la generación de secuencias pseudoaleatorias para la encriptación de imágenes. En segundo lugar se presentarán los resultados al emplear técnicas de cálculo fraccionario discreto para la clasificación de señales e imágenes.

1.1. Estado del Arte

El procesamiento de señales se refiere al conjunto de algoritmos y metodologías empleados para el análisis y transformación de señales, usualmente empleados para extraer información útil que pueda ser aprovechada para la toma de decisiones. En la literatura se ha encontrado que metodologías de procesamiento de señales construidas en torno a técnicas de cálculo fraccionario discreto ofrecen alternativas atractivas debido a sus propiedades.

El análisis de sistemas caóticos de orden fraccionario permite describir dinámicas adicionales debido al efecto de memoria de la definición de derivada utilizada. Los sistemas caóticos son, por definición, sensibles a las variaciones de parámetros [1], y su representación de orden fraccionario no solo agrega un parámetro dado por el orden de diferenciación, sino que también permite la descripción de trayectorias presentes en el comportamiento del sistema que no se pueden obtener con equivalentes de orden entero [2–5]. Diversos trabajos en la literatura lidian también con la simulación de sistemas caóticos de orden fraccionario en tarjetas de implementación. En el trabajo de Chu et al. [3] se desarrolló un modelo macroeconómico artificial para representar las relaciones dinámicas entre ingresos, consumo e inversión. El modelo diseñado exhibe comportamientos caóticos y emplea la definición de diferencia de orden fraccionario de Caputo. Los diagramas de bifurcación en relación al orden fraccionario fueron estudiados con el fin de evidenciar su influencia en las regiones caóticas del sistema. Finalmente el sistema propuesto fue implementado en una placa Arduino Due sin embargo, debido a las limitaciones de *hardware* se optó por truncar la memoria de la definición fraccionaria para reducir la carga computacional. Por su parte el trabajo de Mohamed et al. [6] presenta la implementación de un núcleo IP para una derivada genérica de orden fraccionario basada en una aproximación de Grünwald-Letnikov en una placa FPGA Artix-7. La metodología propuesta fue probada con tres sistemas caóticos de orden fraccionario con órdenes conmensurados e incommensurados. Liu et al. [7] presentó un sistema caótico de orden fraccionario con atractores ocultos y atractores autoexcitados, analizando sus comportamientos dinámicos e implementando el sistema en un procesador de señal digital. Un problema recurrente al implementar métodos numéricos radica en la complejidad algorítmica de los mismos, lo cual lleva a simplificaciones o cambios en las metodologías con la finalidad de que los equipos utilizados sean capaces de ejecutarlos. Estas

modificaciones sin embargo pueden ocasionar cambios en el comportamiento dinámico de las señales obtenidas.

El control de sistemas caóticos de orden fraccionario presenta un reto debido a la complejidad y sensibilidad presentada por los sistemas. Diversos enfoques de control han sido utilizados tanto para la supresión de caos [8–11] como para la sincronización de estos sistemas [12–16]. El trabajo de Zuñiga-Aguilar et al. [8] desarrolló una estrategia de control robusta para la estabilización de diversos sistemas caóticos fraccionarios de orden variable en el tiempo basados en la función de estabilidad de Lyapunov. El trabajo de Cao en [12] presenta un método de sincronización para sistemas de orden fraccionario descritos por la definición de cálculo fraccionario de Caputo. La metodología propuesta emplea una ganancia proporcional en el término de acoplamiento de un esquema de sincronización de Pecora-Carroll. El esquema propuesto fue probado mediante simulación numérica en una familia de sistemas financieros. Las limitaciones para la selección tanto del orden fraccionario, así como para la ganancia proporcional fueron establecidos mediante el análisis de la transformada de Laplace de los sistemas escogidos. Kumar et al. [13] estudió el comportamiento caótico que surge de un sistema de satélites de orden fraccionario. Se diseñó un controlador lineal de retroalimentación de estados con el objetivo de suprimir el comportamiento caótico del sistema. Se propuso además un esquema de sincronización basado en un controlador adaptativo que permite cumplir la condición de sincronización idéntica para dos sistemas con diferentes parámetros. Akinlar et al. [14] empleó el modelo de orden fraccionario para la rueda hidráulica de Malkus, basado en la definición de derivada fraccionaria de Caputo, como caso de estudio. La estabilidad del sistema fue analizada y posteriormente se desarrolló un esquema de anti-sincronización basado en un controlador robusto. Además, se resolvió una formulación de controlador óptimo para la supresión de caos aplicando el método de estados adjuntos. Los resultados obtenidos fueron validados mediante simulación numérica. Rahman et al. [15] presentó un sistema caótico de orden fraccionario sin puntos de equilibrio descrito mediante la definición de derivada de Caputo. Se empleó un controlador adaptable para la sincronización de dos sistemas caóticos de estructura y orden fraccionario iguales pero con parámetros y condiciones iniciales diferentes. Se propuso también un esquema de cifrado el cual posteriormente fue implementado en una placa Arduino Due, obteniendo un error de $\mp 1.56\%$ en la simulación del sistema al compararlo con los valores de simulación obtenidos en MATLAB. Al-Khedhairi et al. [11] investigó la dinámica caótica del modelo de campo geomagnético de orden fraccionario. Se diseñó un controlador lineal con la intención de suprimir el comportamiento caótico del modelo estudiado. El controlador propuesto es capaz de estabilizar el sistema estudiado en cualquiera de sus cinco puntos de equilibrio y los resultados se mostraron mediante simulación numérica. Xiong et al. [16] analizó un sistema caótico de orden fraccionario con múltiples estados estables y se propuso una nueva técnica de control de modos deslizantes basado en redes neuronales para la sincronización de sistemas reduciendo el *chattering*.

Los sistemas caóticos de orden fraccionario así como las técnicas de control utilizadas para suprimir sus comportamientos o establecer esquemas de sincronización pueden ser empleadas para la generación de secuencias pseudoaleatorias, las cuales pueden ser utilizadas en aplicaciones criptográficas. En esteganografía se diseñan algoritmos con el propósito de ocultar información dentro de un medio para la transmisión segura de datos [17] o bien, para incorporar marcas de agua ocultas a las señales de interés [18]. El trabajo presentado por Pu en [19] implementa un algoritmo esteganográfico basado en

la derivada de Grünwald-Letnikov. Los datos ocultos corresponden a un código en un medio impreso que permite detectar productos falsificados. La metodología propuesta fue probada contra ataques estadísticos, de distorsión, rotación, recorte, escala y ruido, mostrando robustez ante ellos. En el campo de la criptografía una de las aplicaciones más populares es la encriptación, en donde se diseñan algoritmos con la finalidad de transformar información de tal manera que su naturaleza original resulte imposible de discernir. Idealmente, la información original podría ser recuperada únicamente si se posee la llave de encriptación, protegiendo así los datos contra individuos no deseados. La generalización de sistemas caóticos para ordenes de derivada no enteros permite aumentar el espacio de las claves utilizadas [20, 21], proporcionando así mayor seguridad contra ataques de fuerza bruta. La combinación de un algoritmo de generación de secuencias pseudoaleatorias con algoritmos de confusión y desplazamiento robustos es esencial en algoritmos criptográficos. Una etapa de confusión interesante surge de la analogía del codificado de ADN con operadores binarios [22–25], en la que se proponen reglas de codificación basadas en la propiedad complementaria de las bases de ADN.

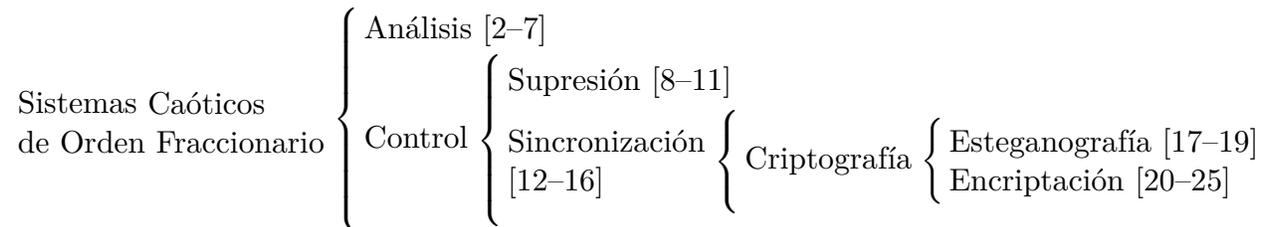


Figura 1.1: Diagrama del estado del arte de análisis de sistemas caóticos de orden fraccionario y su aplicación en criptografía.

Por otro lado, el problema de clasificación se refiere a el proceso en el cual una etiqueta de membresía de grupo es asignada a cierta información. Con esta finalidad es posible emplear algoritmos de aprendizaje automático, en los cuales se entrena a un modelo matemático con datos conocidos para que posteriormente sean capaces de asignar etiquetas a información nueva. El trabajo presentado por Raubitzek [26] realiza una revisión de avances recientes en metodologías de aprendizaje automático que emplean cálculo de orden fraccionario, omitiendo las redes neuronales como modelos de clasificación. El trabajo se divide en tres partes principales: preprocesamiento, aprendizaje automático y dinámicas fraccionarias. En la etapa de preprocesamiento, se destacan las aplicaciones en espectroscopía, imágenes médicas e ingeniería. En esta aplicación, la definición de Grünwald-Letnikov se ha utilizado predominantemente y a menudo ha mejorado el rendimiento de clasificadores basados en árboles de decisiones. El cálculo fraccionario también se utiliza con frecuencia para la generalización de algoritmos de optimización tanto para métodos basados en gradientes como para métodos sin gradiente. Los problemas comunes encontrados por los autores son la falta de especificidad al discutir las definiciones de derivadas fraccionario y el uso incorrecto de palabras clave; lo primero lleva a confusiones con trabajos que no emplean definiciones de cálculo fraccionario. A grandes rasgos, el problema de clasificación puede separarse en tres partes: preprocesamiento, extracción de características y el entrenamiento del modelo de aprendizaje. Es posible emplear técnicas de cálculo fraccionario en cada una de las etapas.

En el caso de los modelos de aprendizaje el trabajo de Antil [27] presenta una red neuronal profunda

fraccionaria mediante el uso de la definición de derivada de Caputo, lo que conlleva a introducir un efecto similar a la memoria en el entrenamiento de una red de aprendizaje profundo (DNN, por sus siglas en inglés). Esto resulta útil para abordar el problema del gradiente desvaneciente, en el cual, durante el paso de entrenamiento el gradiente obtenido mediante la derivada parcial del error con respecto al peso sináptico es extremadamente pequeño, tanto que el peso no cambia su valor. Dicho problema puede llevar a una paralización en el entrenamiento de la red neuronal. La DNN propuesta se prueba en problemas de clasificación de dos conjuntos de datos diferentes, y se obtienen resultados favorables en comparación con las metodologías de orden entero. Se observa que el costo computacional para el marco diseñado es mayor que el de algoritmos similares.

Diversos trabajos han empleado algoritmos basados en el cálculo discreto de orden fraccionario para mejorar diversas características en información. El trabajo de Wei Wang [28] propuso un algoritmo de mejora de calidad de imágenes basado en la generalización de la Transformada de Estiramiento de Fase (PST) para órdenes fraccionarios de transformación. La transformada PST fraccionaria es utilizada para extraer información relacionada con los bordes de la imagen mientras se preservan los componentes de baja frecuencia. Las características extraídas son posteriormente combinadas con la señal original mediante un modelo de variación total relativa (RTV). Para esta aplicación, el orden fraccionario de transformación proporciona robustez contra ciertos tipos de ruido presentes en la imagen. En su contribución en [29], Lavin-Delgado propone una discretización de la derivada Caputo-Fabrizio de orden fraccionario para el operador Sobel de primer orden. La máscara obtenida se utiliza luego para la detección de bordes en imágenes médicas de pacientes con problemas de mama, aneurismas cerebrales y retinopatía. El método propuesto mejora los resultados obtenidos en imágenes con bordes débiles en comparación con su contraparte de orden entero en todos los casos. Como muestran estos trabajos, el cálculo fraccionario proporciona herramientas para la mejora de imágenes en dominios de frecuencia y espaciales. El trabajo de Chandra [30] presentó una máscara basada en el orden de diferenciación fraccionario para la segmentación de imágenes. El algoritmo propuesto muestra ser capaz de manejar datos contaminados con ruido y al mismo tiempo ser sensible a pequeñas variaciones de intensidad en la imagen, lo que permite la detección temprana de tumores cerebrales con un costo computacional relativamente bajo. Las transformadas discretas de orden fraccionario permiten el análisis de señales en un dominio entre el tiempo y la frecuencia. Estas transformaciones resultan útiles para la extracción de características. En señales de fonocardiogramas el trabajo de Abduh [31] combinó el análisis de coeficientes Mel en el dominio tiempo-frecuencia fraccionario para el diagnóstico de enfermedades cardíacas. Las comparaciones con metodologías de orden entero muestran que el algoritmo propuesto proporciona robustez contra problemas de sobre ajuste. El trabajo presentado por Erdal Dinç [32] realiza el análisis no invasivo de productos farmacéuticos empleando transformadas wavelet de orden fraccionario discretas, que muestran una mayor tolerancia al ruido y las perturbaciones en comparación con sus contrapartes de orden entero.

En medicina, se han empleado transformadas integrales fraccionarias para el análisis y extracción de características de diversas señales. La transformada discreta fraccionaria de Fourier (DFRFT) se ha implementado en señales neurológicas y cardiovasculares [33]. Los trabajos de Zhang [34, 35] emplearon modelos de aprendizaje automático para la detección de patologías cerebrales a partir de características basadas en la DFRFT extraídas de resonancias magnéticas. Fei en [36] se ocupa del análisis de electro-

encefalogramas (EEG) para la predicción de convulsiones epilépticas; estas señales son particularmente interesantes ya que poseen dinámicas caóticas. El algoritmo del exponente de Lyapunov más grande se modificó para la selección de órdenes fraccionarios. Incluso sin parámetros óptimos, el sistema propuesto logró alcanzar rendimientos de clasificación de 90 %.

Un objetivo común en el preprocesamiento de señales es la supresión de ruido. Por ejemplo, en el trabajo de Qi Wang en [37] empleó el gradiente diferencial fraccionario para la detección de ruido y técnicas de integración fraccionario para suprimir los componentes no deseados en imágenes mediante una modificación en un filtro de orden entero. La metodología propuesta preserva las texturas de la imagen que podrían ser confundidas como ruido por algoritmos equivalentes de orden entero. Waheed en [38] generalizó el operador Laplaciano empleando la transformada discreta del coseno (DCT), para posteriormente utilizarla en la construcción de un filtro de tendencia de orden fraccionario para la supresión de ruido en una señal unidimensional. El operador propuesto también se utiliza para la generalización del detector de bordes Marr-Hildreth, la comparación con otros métodos de derivadas de segundo orden muestra una mayor tolerancia al ruido gaussiano debido a la propiedad de memoria encontrada en las definiciones de cálculo fraccionario.

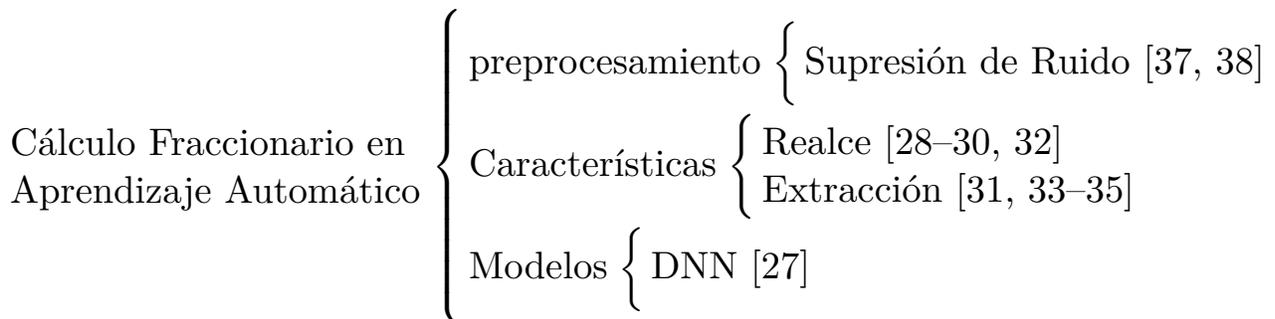


Figura 1.2: Diagrama del estado del arte de aplicaciones de cálculo discreto de orden fraccionario en clasificación basada en señales.

En el caso de preprocesamiento de señales los algoritmos utilizados varían considerablemente según la aplicación específica. En el contexto médico, las enfermedades relacionadas con el corazón se estima que causan 17,9 millones de muertes cada año según la Organización Mundial de la Salud (OMS) [39], lo que resalta la importancia del estudio y el análisis de las señales relacionadas con el corazón. Las señales de fonocardiogramas son particularmente atractivas ya que son no invasivas y de naturaleza mecánica. Desde un punto de vista de ingeniería, la clasificación de señales basada en modelos de aprendizaje automático es una herramienta útil para la creación de herramientas de asistencia al diagnóstico. Para el filtrado de fonocardiogramas se han explorado diferentes metodologías. Dependiendo de la aplicación, se puede utilizar la filtración de banda pasante para reducir eficazmente los componentes no deseados de la señal [40–42]. El trabajo presentado por Zheng en [43] utiliza una descomposición de valores singulares multinivel para el aislamiento del ruido, luego la señal se reconstruye basada en un algoritmo de muestreo comprimido. El marco resultante permite la reducción del ruido sin afectar la información original de la señal. Los trabajos de Kyzdarbekova [44] y Rouis [45] emplean la transformada wavelet para descomponer la señal. En el primer caso, se aplica un esquema de reducción adaptativa de ruido

utilizando la función Daubechies 4 para una descomposición de cuarto nivel. El segundo trabajo explora el uso de diferentes funciones madre de wavelet pertenecientes a la familia Symlet y diferentes niveles de descomposición. Las señales se procesan posteriormente en función de varios valores umbral para eliminar el ruido sintético. Los resultados obtenidos muestran que la función wavelet más efectiva necesaria para el filtrado depende en gran medida del tipo de ruido. Para ruido violeta y azul, Symlet 16 dio los mejores resultados; para ruido rojo, Symlet 3 y para ruido rosa, Symlet 9 presentaron la mejor reducción de ruido. Además de la supresión de ruido, los algoritmos de segmentación de fonocardiogramas también forman parte de su preprocesamiento. El trabajo de Akram [46] propone una metodología para la localización de los componentes S_1 y S_2 en los fonocardiogramas. El algoritmo propuesto utiliza un filtro homomórfico para el cálculo de una señal de envolvente. La diferenciación de los componentes se realiza obteniendo diferentes características estadísticas en el dominio del tiempo y con una descomposición wavelet de segundo nivel.

1.1.1. Antecedentes Institucionales

En esta sección, se destacan algunos trabajos realizados en el Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET) que son relevantes como antecedentes para esta tesis. En el estudio de los operadores de orden fraccionario y sus aplicaciones, el trabajo de Coronel en [47] presenta una metodología para la sincronización de sistemas caóticos de orden fraccionario. El sistema secundario propuesto se basa en un observador de estado. Se utilizan diferentes funciones suaves para definir el orden fraccionario de los sistemas. Se emplearon las definiciones de Liouville-Caputo y Atangana-Baleanu-Caputo para la generalización de los sistemas y se resolvieron numéricamente utilizando una modificación del método de Adams. El trabajo de Solís en [48] utiliza métodos numéricos basados en la interpolación polinómica de Lagrange para resolver ecuaciones diferenciales de orden fraccionario descritas por diferentes definiciones de cálculo fraccionario. Se realizan múltiples simulaciones numéricas en dos sistemas caóticos con orden de integración variable. Además, en el trabajo de Avalos [49], se utiliza una variación del método de Euler que permite el uso de métodos de cálculo de orden generalizado continuo en sistemas discretos. Se aplican diferentes operadores de cálculo fraccionario, conformable y fraccionario conformable con el propósito de encontrar comportamientos diferentes a los descritos en el caso clásico. Se destaca también que entre las ecuaciones de un sistema, tanto el orden de integración como la definición de cálculo utilizada pueden ser diferentes entre sí.

En el caso de la clasificación basada en señales, se destacan los siguientes trabajos: Pérez en [50] analiza señales electroencefalográficas medidas en el cuero cabelludo para la creación de una interfaz cerebro-computadora. Se abordan tres problemas de clasificación diferentes: estímulos visuales, movimientos musculares y perturbaciones oculares. Se utilizan tanto las transformadas de Stockwell como la transformada wavelet para un cambio de dominio antes de calcular diferentes características estadísticas que luego se utilizan para la clasificación de los estímulos. En la tesis de Ramos [51], se introduce una metodología de modelado para señales electrocardiográficas basada en la característica de Euler-Poincaré para campos aleatorios no gaussianos. La metodología propuesta se utiliza en combinación con un modelo de conjunto KNN para la clasificación de la apnea del sueño.

1.2. Planteamiento del Problema

En los últimos 60 años, el acelerado y continuo desarrollo de los computadores digitales y sus aplicaciones en el procesamiento de señales han presentado la necesidad del desarrollo de técnicas de análisis y procesamiento de señales discretas. Las señales pueden provenir de un número incontable de fuentes con comportamientos complejos, como fenómenos de autosimilitud. El interés por problemas de procesamiento crecientemente más complejos exige también el uso de técnicas de procesamiento de señales más sofisticadas.

El cálculo de orden fraccionario se refiere a la rama de las matemáticas que lidia con ordenes de integración y derivación no enteros. En su contraparte discreta, los operadores de integral y derivada son sustituidos por sumas y diferencias de orden fraccionario. El uso de operadores del cálculo discreto de orden fraccionario presenta retos importantes, como lo son su interpretación física y su elevado costo computacional. La complejidad presentada por los sistemas discretos de orden fraccionario puede ser aprovechada para fines criptográficos, por lo que su estudio e implementación resultan temas relevantes para su estudio.

El procesamiento de señales empleando operadores del cálculo fraccionario discreto presenta posibles aplicaciones prometedoras tanto para el análisis como para la transformación de datos. En el filtrado de señales, el cálculo fraccionario discreto permite rechazar efectivamente la información no deseada, manteniendo componentes de las señales que algoritmos de orden entero suelen descartar como ruido. En el contexto de clasificación, las técnicas basadas en cálculo fraccionario discreto permiten obtener características que representan de una manera más precisa las diferencias entre las etiquetas del problema a trabajar, lo cual permite obtener mejores índices de desempeño empleando modelos de clasificación computacionalmente más sencillos. En otras palabras, las características obtenidas con metodologías basadas en el cálculo discreto de orden generalizado pueden ser utilizadas para describir diversas particularidades de un problema de clasificación de tal manera que la etapa de clasificación es simplificada.

1.3. Justificación

La propiedad de no localidad que ofrece el cálculo fraccionario permite considerar la información histórica del fenómeno a estudiar, lo cual presenta ciertas ventajas en diversas aplicaciones. En el caso del modelado matemático, el cálculo de orden fraccionario permite describir comportamientos de memoria y autosimilitud, obteniendo así trayectorias dinámicas distintas a las obtenidas por los modelos de orden entero. La solución obtenida para una ecuación diferencial de orden entero depende directamente de su condición inicial, en el cálculo de orden fraccionario es posible encontrar soluciones distintas manteniendo las mismas condiciones iniciales debido a la propiedad de no localidad que presentan los operadores. El orden de derivada no entero permite ampliar la familia de soluciones posibles para los problemas descritos por operadores fraccionarios.

Las nuevas trayectorias obtenidas en modelos caóticos descritos con derivadas fraccionarias pueden ser aprovechadas para la generación de secuencias pseudoaleatorias que pueden posteriormente ser utiliza-

das para aplicaciones criptográficas o esteganográficas. Se ha encontrado que algoritmos criptológicos basados en técnicas de cálculo fraccionario discreto permiten aumentar el tamaño de la llave de encriptación, mejorando así la seguridad de la metodología.

En el contexto de procesamiento de señales, las metodologías de filtrado basadas en cálculo de orden fraccionario permiten conservar información que podría ser descartada como ruido por algoritmos equivalentes de orden entero. Técnicas de cálculo discreto de orden fraccionario permiten también la extracción de características que describen mas cercanamente señales encontradas en diversos fenómenos, lo cual resulta útil en aplicaciones de clasificación, en donde permiten reducir la complejidad de los modelos de clasificación a utilizar.

Existe un campo de oportunidad amplio en el desarrollo de expresiones matemáticas y algoritmos para el procesamiento de señales e imágenes basados en cálculo discreto de orden fraccionario. Por ejemplo, el uso de metodologías de filtrado, extracción de características, sistemas de encriptación y métodos numéricos.

1.4. Hipótesis

El diseño de técnicas de procesamiento de señales e imágenes basadas en cálculo discreto de orden general, permitirá una mejor representación y análisis de datos de sistemas con dinámicas complejas.

1.5. Objetivos

General

- Proponer algoritmos y técnicas de modelado basados en cálculo de orden generalizado discreto útiles para el análisis de señales e imágenes.

Específicos

- Definir técnicas de modelado de señales con base en el cálculo discreto de orden fraccionario con ordenes variables e inconmesurados.
- Utilizar métodos de optimización heurística para la selección del orden de derivación.
- Aplicar las técnicas desarrolladas en señales provenientes de sistemas con comportamientos complejos, como el comportamiento estocástico de un terremoto, mediciones de electrocardiogramas muestreados en pacientes con sepsis, señales de respiración obtenidos de pacientes con apneas, entre otros.
- Definir metodologías de procesamiento de imágenes orden de derivación y paso variables.
- Proponer una metodología para la selección del orden de derivación α y el paso de discretización basado en métodos de optimización heurísticos.
- Aplicar los métodos diseñados para el resalte y supresión de ruido en imágenes
- Comparar el desempeño de los enfoques clásicos y las técnicas desarrolladas.

1.6. Metas

- Desarrollo de algoritmos y técnicas de modelado útiles para el análisis de señales con efectos no lineales y efectos de memoria.
- Proponer algoritmos de procesamiento de imágenes tolerantes a ruido.
- Implementación de los algoritmos en software o hardware.

1.7. Organización del Documento

El presente documento se divide de la siguiente manera:

- **Capítulo 2:** Se presentan las bases del cálculo en escalas de tiempo y su generalización para órdenes de diferencia no enteros, así como la simulación de sistemas discretos de orden fraccionario empleando el operador de diferencia fraccionaria.
- **Capítulo 3:** Se presentan conceptos de procesamiento de señales discretas. En el caso del análisis de señales se muestra la transformada de Fourier y su generalización para órdenes de transformación no enteros. Para la toma de decisiones se presentan los puntos claves de la clasificación basada en características junto a algunos modelos populares.
- **Capítulo 4:** Se presenta un esquema de sincronización basado en controladores para sistemas caóticos de orden fraccionario. La simplicidad algorítmica se demuestra mediante la implementación del arreglo propuesto en una placa de desarrollo Arduino UNO. Se presenta una aplicación en la generación de números pseudoaleatorio para un algoritmo de imagen RGB.
- **Capítulo 5:** Se presenta una metodología de asistencia para el diagnóstico, que emplea operadores de orden no entero para el preprocesamiento y la extracción de características de los fonocardiogramas.
- **Capítulo 6:** Finalmente se presentan las conclusiones del presente trabajo.

Capítulo 2

Cálculo Fraccionario Discreto

Cálculo fraccionario es el campo matemático que aborda problemas que involucran ordenes de derivadas e integrales no enteros. El orden no entero en los operadores de cálculo fraccionario permite describir comportamientos interpretados como memoria encontrados en sistemas de diversos campos como: física [52, 53], química [54, 55], control [56, 57], finanzas [58, 59], entre otros. En los ejemplos presentados, los operadores basados en cálculo fraccionario permiten proponer modelos que describen datos experimentales de manera mas precisa.

Al discretizar las definiciones de integrales y derivadas de orden fraccionario se obtienen sumas y diferencias de orden no entero [60]. El uso de diferencias y sumas de orden fraccionario es más complicado en comparación con su contraparte de orden entero; sin embargo, también conlleva aplicaciones interesantes. Por ejemplo, en teoría de control, muchos problemas clásicos pueden abordarse utilizando estos operadores, tales como: análisis de sistemas dinámicos [61], controladores de orden fraccionario [62] y la estabilidad de sistemas de orden fraccionario [63].

2.1. Cálculo en Escalas de Tiempo

El cálculo en escalas de tiempo generalizadas, estudiado por el matemático alemán Stefan Hilger en 1988, permite unir los conceptos de ecuaciones diferenciales y ecuaciones en diferencias [64]. Una escala de tiempo es un subconjunto cerrado perteneciente a \mathbb{R} . Un punto cualquiera t en una escala de tiempo \mathbb{T} se representa en la Ecuación (2.1)

$$t : t \in \mathbb{T} \tag{2.1}$$

Si $\mathbb{T} = \mathbb{R}$ entonces la escala de tiempo es continua. Por otro lado, si $\mathbb{T} = \mathbb{Z}$ se tiene una escala de tiempo discreta. El siguiente punto en la escala de tiempo es obtenido por el operador $\sigma(\cdot)$ definido en la Ecuación (2.2)

$$\sigma(t) := \inf\{s \in \mathbb{T} : s > t\}, \tag{2.2}$$

mientras que el punto anterior es obtenido por el operador $\rho(\cdot)$ mostrado en la Ecuación (2.3)

$$\rho(t) := \sup\{s \in \mathbb{T} : s < t\}. \tag{2.3}$$

La distancia entre un punto t y $\sigma(t)$ se describe en la Ecuación (2.4)

$$\mu(t) = \sigma(t) - t. \quad (2.4)$$

En base a las ecuaciones (2.2) y (2.4), la derivada en escalas de tiempo generalizadas $f^\Delta(t)$ existe si y solo si $\forall \varepsilon > 0, \exists U \in t$ tal que la condición mostrada en la Ecuación (2.5) se cumpla $\forall s \in U$

$$|f(\sigma(t)) - f(s) - f^\Delta(t)(\sigma - s)| \leq \varepsilon |\sigma(t) - s|. \quad (2.5)$$

Definición 1. *Asumiendo $f : \mathbb{T} \rightarrow \mathbb{R}$ y $t \in \mathbb{T}^k$, la derivada de Hilger se define en la Ecuación (2.6)*

$$f^\Delta(t) := \lim_{s \rightarrow t} \frac{f(\sigma(s)) - f(t)}{\sigma(s) - t}, \quad \sigma(s) \neq t. \quad (2.6)$$

Donde \mathbb{T}^k es una escala de tiempo definida como: si \mathbb{T} tiene un máximo disperso a la izquierda M entonces $\mathbb{T}^k := \mathbb{T} / \{M\}$, de lo contrario $\mathbb{T}^k = \mathbb{T}$. Si $f^\Delta(t) \exists \forall t \in \mathbb{T}^k$ entonces se dice que f es delta diferenciable en \mathbb{T}^k . Si se considera entonces que la escala de tiempo es continua $\mathbb{T} = \mathbb{R}$ entonces se recupera el caso de derivada del cálculo clásico $f^\Delta = f'$. Si en cambio $\mathbb{T} = \mathbb{Z}$ se recupera el operador de diferencia finita adelantada $f^\Delta = \Delta f$.

Definición 2. *Sea $[a, b]$ un intervalo cerrado en \mathbb{T} . La función $F : [a, b] \rightarrow \mathbb{R}$ es la antiderivada de la función $f : [a, b] \rightarrow \mathbb{R}$ si F es continua en $[a, b]$, delta diferenciable en $[a, b]$ y $F^\Delta(t) = f(t) \forall t \in [a, b]$. Entonces la integral Δ de f en $[a, b]$ se define en la Ecuación (2.7)*

$$\int_a^b f(t) \Delta t := F(b) - F(a). \quad (2.7)$$

suponiendo entonces que $b > a$ y que $f(t)$ es continua en $[a, b]$ entonces

$$\int_a^b f(t) \Delta t = [\sigma(a) - a]f(a) + \int_{\sigma(a)}^b f(t) \Delta t. \quad (2.8)$$

Definición 3. *Suponiendo que $[a, b]$ es un intervalo de la escala de tiempo \mathbb{T} y que f es una función integrable en $[a, b]$. La integral fraccionaria de orden α de f se define en la Ecuación (2.9)*

$${}_{\mathbb{T}}I_a^v f(t) := \frac{1}{\Gamma(v)} \int_a^t (t - s)^{v-1} f(s) \Delta s. \quad (2.9)$$

En donde $\Gamma(\cdot)$ es la función gamma mostrada en la Ecuación (2.10).

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t}, \quad (2.10)$$

Esta función permite la generalización del operador factorial para números complejos para el dominio complejo, exceptuando los enteros negativos, como se muestra en la Ecuación (2.11)

$$\Gamma(n) = (n - 1)!, \quad \forall n \in \mathbb{C} \setminus \mathbb{Z}^-. \quad (2.11)$$

De manera análoga, se define el operador de suma fraccionaria de orden $\alpha > 0$ en la Ecuación (2.12)

$$\Delta^v f(t) = \frac{1}{\Gamma(v)} \sum_{s=a}^{t-v} (t - \sigma(s))^{v-1} f(s). \quad (2.12)$$

Si k es un punto en la escala de tiempo $\mathbb{T} = \mathbb{Z}$ es posible combinar los operadores de sumatoria y diferencia de orden fraccionario.

Definición 4. Considerando $f : k \mapsto \mathbb{R}$, $v \in \mathbb{R}$. $k \in \mathbb{N}$. La diferencia/suma de orden fraccionario en su forma Grünwald-Letnikov (GLFOBDS, por sus siglas en inglés) se define en la Ecuación (2.13)

$${}_{k_0}^{GL} \Delta_k^v f(k) = \sum_{i=0}^{k-k_0} a^{(v)}(i) f(k-i), \quad (2.13)$$

En donde $a^{(v)}(k)$ es una función auxiliar que calcula los coeficientes de la GLFOBDS como se muestra en la Ecuación (2.14)

$$a^{(v)}(k) = \begin{cases} 0 & k < 0, \\ 1 & k = 0, \\ (-1)^k \frac{v(v-1)\cdots(v-k+1)}{k!} & k = 1, 2, 3, \dots \end{cases} \quad (2.14)$$

es importante aclarar que (v) se refiere a un argumento de la función y no a un exponente, esta notación es utilizada para enfatizar su relación con el orden fraccionario. La función $a^{(v)}(k)$ calcula los coeficientes tanto para tanto el caso de diferencia como de suma de orden fraccionario. $\forall v \in \mathbb{R}^-$ se tiene el caso de sumatoria, mientras que $\forall v \in \mathbb{R}^+$ se tiene la definición de diferencia de orden fraccionario. La principal diferencia encontrada $\forall v \in \mathbb{R}^+/\mathbb{N}$ radica en el kernel de convolución presente en la definición, el cual suele ser asociado con propiedades de memoria presentes en sistemas reales. La Ecuación (2.14) puede ser calculada de manera recursiva como muestran las Ecuaciones (2.15) y (2.16)

$$a^{(v)}(k) = (-1)^k \frac{v(v-1)(v-2)\cdots(v-k+1)}{k!} \times (-1)^{k-1} \frac{v(v-1)(v-2)\cdots(v-k+2)}{(k-1)!} (-1)^{\frac{v-k+1}{k}}, \quad (2.15)$$

$$a^{(v)}(k) = a^{(v)}(k-1) (-1)^{\frac{v-k+1}{k}} = a^{(v)}(k-1) \left(1 - \frac{v+1}{k}\right). \quad (2.16)$$

Entonces la Ecuación (2.14) puede reescribirse como

$$a^{(v)}(k) = \begin{cases} 1 & \text{for } k = 0, \\ \left(1 - \frac{v+1}{k}\right) a^{(v)}(k-1) & \text{for } k = 1, 2, 3, \dots \end{cases} \quad (2.17)$$

La Figura 2.1 presenta los valores obtenidos por la Ecuación (2.17) para valores de $k \in [0, 20]$ y $v \in [0, 1]$

Para valores de $v > 0$ la función $a^{(v)}(k)$ se presenta la siguiente propiedad

$$\lim_{k \rightarrow \infty} a^{(v)}(k) = 0, \quad (2.18)$$

lo cual permite reescribir la Ecuación (2.17) como

$$a_L^{(v)}(k) = \begin{cases} 1 & \text{for } k = 0, \\ \left(1 - \frac{v+1}{k}\right) a^{(v)}(k-1) & \text{for } k = 1, 2, \dots, L, \\ 0 & \text{for } k > L, \end{cases} \quad (2.19)$$

donde $L \in \mathbb{N}$. La propiedad mostrada en la Ecuación (2.19) permite truncar la convolución presente en la GLFOBDS sin perder información de la dinámica del sistema descrito, esto resulta útil para la simulación numérica de sistemas dinámicos discretos de orden fraccionario.

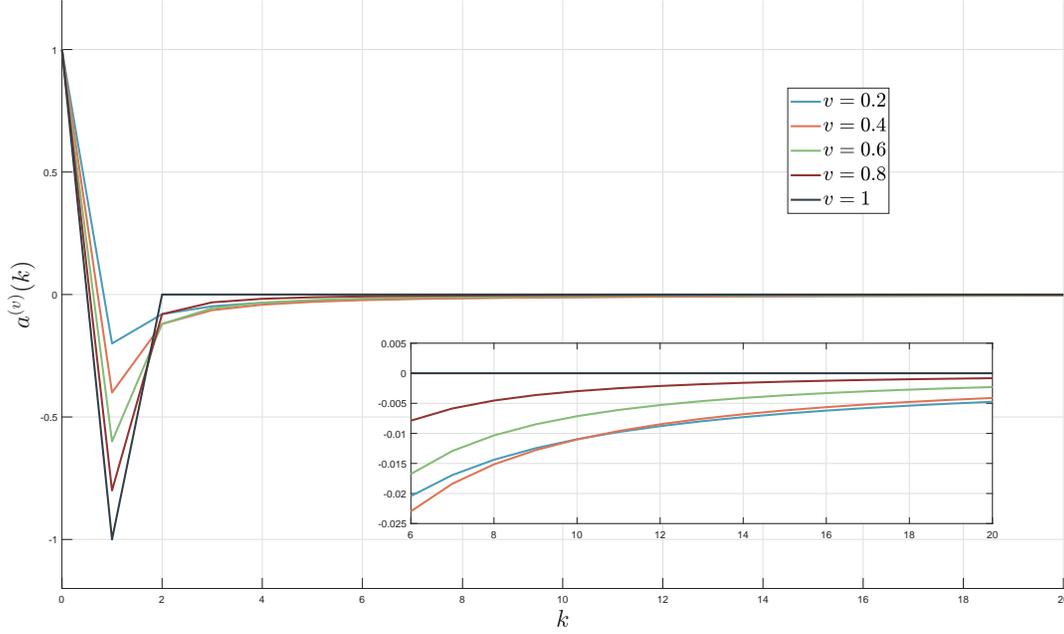


Figura 2.1: Función $a^{(v)}(k) \forall k \in [0, 20]$ para distintos valores de $v \in [0, 1]$. Si bien la función mostrada es discreta, líneas continuas son utilizadas para claridad.

Considerando un sistema de ecuaciones en diferencias como se muestra en la Ecuación (2.20)

$$X(k) = \varphi(X(k-1)), \quad (2.20)$$

donde $X(k) = [x_1(k), x_2(k), \dots, x_n(k)]^T$ y $\varphi(\cdot)$ denota una función auxiliar que describe el comportamiento del sistema. La Ecuación (2.21) reescribe la Ecuación (2.20) en su forma en diferencias

$$\Delta X(k) = \varphi(X(k-1)) - X(k-1), \quad (2.21)$$

generalizando el operador de diferencia para ordenes $v \notin \mathbb{N}$ con la definición de GLFOBDS mostrada en la Ecuación (2.13) se obtiene la Ecuación (2.22)

$$\Delta^{v(k)} X(k) = \varphi(X(k-1)) - X(k-1), \quad (2.22)$$

donde $v(k) : k \mapsto \mathbb{R}, k \in \mathbb{N}$. En este caso $v(k)$ considera ordenes de diferencia variables en el tiempo.

El cálculo discreto de orden fraccionario provee de herramientas para el modelado de dinámicas de sistemas complejos. El orden fraccionario en la definición de derivada permite describir el comportamiento del efecto de memoria e interacciones no locales presentes en sistemas encontrados en el mundo real, lo cual permite una representación más fiel en comparación con modelos de orden entero. La capacidad del cálculo fraccionario para capturar la memoria y las interacciones no locales en sistemas complejos ofrece una base teórica para comprender fenómenos más realistas.

Al modelar fenómenos reales es posible encontrar dinámicas complejas debido a las no linealidades inherentes del sistema. Específicamente los sistemas caóticos se caracterizan por ser gobernados por ecuaciones deterministas pero con una sensibilidad relativamente alta a sus condiciones iniciales y parámetros, lo cual los vuelve aparentemente impredecibles. Ejemplos de estos sistemas pueden ser encontrados en campos diversos como la física, biología, ecología, finanzas, etc. El mapa logístico, definido en la Ecuación (2.23), permite ejemplificar la aparición de dinámicas caóticas en sistemas no lineales con estructura simple. Estudiado por Robert May en 1976, este sistema describe de manera simplificada el comportamiento de una población con respecto a una escala de tiempo discreta considerando reproducción y hambruna.

$$x(k+1) = rx(k)(1-x(k)), \quad (2.23)$$

donde $x(k) \in [0, 1]$ representa población normalizada en el instante k y $r \in [0, 4]$ es la razón de crecimiento demográfico. De manera general, el mapa logístico presenta dinámicas caóticas para valores de $r \in [3.57, 4]$.

Generalizando la Ecuación (2.23) para ordenes de diferencia no entero se obtiene la Ecuación (2.24)

$$\Delta^{v(k)}x(k+1) = rx_k(1-x(k)) + x(k), \quad (2.24)$$

La figura 2.2 muestra la comparación de trayectorias del mapa logístico para diversos órdenes de diferencia arbitrariamente cercanos a 1, en todos los casos las condiciones iniciales se encuentran definidas como $x(0) = 0.5$ y $r = 3.6$. Como se menciona anteriormente, un cambio relativamente pequeño en los parámetros del sistema (en este caso el orden de diferencia) produce trayectorias distintas en el sistema después de cierto tiempo finito.

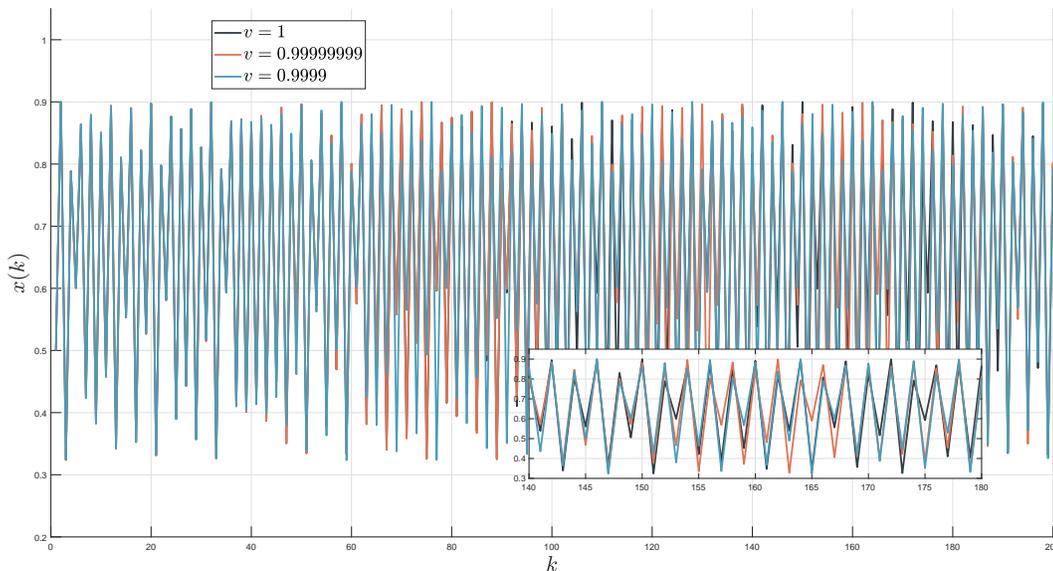


Figura 2.2: Comparación de trayectorias del mapa logístico para diferentes valores de v .

El orden de diferencia fraccionario permite describir comportamientos de memoria no locales presentes en diversos sistemas, sin embargo su interpretación propone un desafío que depende directamente del caso específico a trabajar. En el contexto de la física, el orden fraccionario de la derivada de una función puede estar asociado con comportamientos disipativos, lo que puede ser aprovechado para el modelado de diversos fenómenos como impurezas en los materiales que componen el sistema o dinámicas de disipación intrínsecas del sistema. Específicamente en sistemas caóticos, el efecto de memoria afecta de manera significativa la dinámica obtenida, lo cual tiene aplicaciones en generadores de secuencias pseudoaleatorias.

Diversas aplicaciones de los operadores de orden fraccionario en el procesamiento de señales han sido explorados. Por ejemplo, en el procesamiento de imágenes, filtros de orden fraccionario han probado ser útiles para la supresión de ruido preservando componentes de baja frecuencia usualmente asociadas con texturas. También en imágenes, algoritmos de detección de bordes basados en mascarar diferenciales de orden fraccionario han sido capaces de resaltar las características deseadas, presentando una robustez incrementada a algunos tipos de ruido.

Capítulo 3

Procesamiento de Señales

Una señal puede ser definida como una función que contiene información del comportamiento o características de un sistema o de los atributos de un fenómeno [65]. Una señal puede ser representada matemáticamente como una función de una variable independiente y su clasificación puede depender de la continuidad de dicha variable. Una señal como se muestra en la Ecuación (3.1) puede ser definida como una función real de una variable independiente real discreta $k \in \mathbb{T}_{k_0, k, h}$, la cual puede representar instantes de tiempo o distancia discreta por ejemplo.

$$f : \mathbb{T}_{k_0, k, h} \rightarrow \mathbb{R}, \quad (3.1)$$

donde $\mathbb{T}_{k_0, k, h}$ se refiere al dominio de la función y \mathbb{R} es el codominio. En ingeniería, el procesamiento de señales se refiere al campo de estudio centrado en el análisis y modificación de las mismas para extraer información útil o mejorar características específicas. Las técnicas de procesamiento de señales suelen tener como objetivo aplicaciones como la reducción de ruido, compresión, extracción o realce de características, reconocimiento de patrones, entre otros.

3.1. Transformada de Fourier Fraccionaria

Las transformadas integrales son herramientas empleadas en el área de procesamiento de señales que proporcionan representaciones alternativas de cierta información [66]. Estas transformadas permiten el cambio de dominio de la señal original a otro diferente en el cual su manipulación o análisis pueden resultar mas sencillos. Un ejemplo popular de transformada integral es la transformada de Fourier, definida en la Ecuación (3.2). La transformada de Fourier puede ser interpretada como la descomposición de una señal en sus componentes de frecuencia. En otras palabras, la transformada de Fourier descompone la señal en una sumatoria de senos y cosenos.

$$\mathcal{F}\{f(x)\} = F(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-i\omega x} f(x) dx. \quad (3.2)$$

En la Ecuación (3.3) se muestra la transformada de Fourier de orden fraccionario (FRFT), la cual extiende la definición de dicha transformada para ordenes no enteros. Esta transformada puede ser interpretada como una descomposición de la señal en componentes de frecuencia variable.

$$\mathcal{F}_\alpha\{f(x)\} = \sqrt{1 - i \cot(\alpha) e^{i\pi \cot(\alpha) u^2}} \int_{-\infty}^{\infty} e^{-i2\pi \left(\csc(\alpha) u x - \frac{\cot(\alpha)}{2} x^2 \right)} f(x) dx. \quad (3.3)$$

3.1. TRANSFORMADA DE FOURIER FRACCIONARIA

Si se consideran los dominios del tiempo y la frecuencia como perpendiculares, el orden de transformación fraccionario en la FRFT puede ser representado como un ángulo de rotación entre ambos dominios. Por ejemplo, para ordenes de $\alpha = \pi/2$ se obtiene la transformada de Fourier de orden entero, mientras que para $\alpha = \pi$ se obtiene un reflejo de la señal original con respecto al dominio del tiempo. Para órdenes $\alpha \neq n\pi/2; n \in \mathbb{Z}$, la señal se transforma a un dominio entre el tiempo y la frecuencia, comúnmente denotado como u . Esta rotación se ejemplifica en la Figura 3.1.

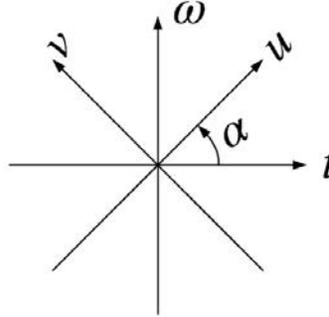


Figura 3.1: Rotación del dominio tiempo-frecuencia al aplicar la transformada de Fourier de orden fraccionario.

Existen diferentes métodos de discretización utilizados para utilizar el operador FRFT en el dominio discreto [67]. La Tabla 3.1 presenta una comparación de algunas propiedades de métodos de discretización para la FRFT encontrados en la literatura. La reversibilidad se refiere a la capacidad de recuperar la señal original de manera exacta a partir de su representación en el dominio de Fourier Fraccionario. La propiedad de aditividad indica que la FRFT de la suma de dos señales es igual a la suma de sus transformaciones individuales. La aditividad de ángulo se refiere a la capacidad de realizar una sola transformación enés de transformaciones consecutivas sumando los ángulos de cada una de ellas de manera individual. La propiedad de simetría al igual que en la transformada de Fourier de orden entero indica que la segunda mitad de la señal se compone de un reflejo de la primera mitad de la señal.

Tabla 3.1: Comparación de algunas características relevantes de diferentes metodologías de discretización para la transformada fraccionaria de Fourier.

Propiedades	Muestreo	Muestreo en forma cerrada	Combinación lineal	Descomposición por eigenvectores
Reversibilidad	No	Sí	Sí	Sí
Aditividad	No	No	Sí	Sí
Aditividad de Ángulo	No	Sí	No	Sí
Simetría	Sí	Sí	No	Sí
Similaridad FRFT	Sí	Sí	No	Sí
Complejidad Algorítmica	$O(N \log_2)$	$O(N \log_2)$	$O(N \log_2)$	$O(N^2)$

La complejidad algorítmica de cada método de discretización se muestra con la notación de Landau, también conocida como "Big O". La complejidad algorítmica en tiempo describe la relación entre el

número de iteraciones que el algoritmo tomará en obtener la salida deseada con el tamaño de la entrada. Como se muestra en la tabla, el algoritmo con mayor complejidad en tiempo corresponde a la descomposición por eigenvalores, donde el número total de iteraciones aumenta de manera cuadrática con el tamaño de la entrada.

3.2. Clasificación

El procesamiento de señales usualmente es aplicado con la finalidad de tomar decisiones basado en la mejor información disponible. Por ejemplo, la clasificación de datos se refiere al proceso en el que se asigna de manera automática una etiqueta de membresía de grupo a instancias de datos individuales. Esto puede realizarse, desde el punto de vista del aprendizaje automático, mediante el entrenamiento de un modelo de clasificación que toma datos como entrada y produce como salida la etiqueta correspondiente. Los modelos de aprendizaje automático se pueden dividir de manera general en dos categorías principales: algoritmos supervisados y no supervisados, la principal diferencia radica en que estos últimos no requieren datos etiquetados en sus fases de entrenamiento y prueba.

La clasificación supervisada basada en señales puede ser dividida en tres etapas principales: preprocesamiento, extracción de características y clasificación. En la etapa de preprocesamiento, las señales se modifican de acuerdo con los requisitos o necesidades del problema de clasificación particular. Algunos ejemplos de objetivos comunes en el preprocesamiento son: reducción de ruido, compresión de datos, cambios de dominio y la mejora o supresión de ciertas características presentes en la señal.

En la segunda etapa del proceso de clasificación se utilizan diversas técnicas para la extracción de características. Una característica se define como una propiedad medible que describe un fenómeno. Las características pueden presentarse de una amplia variedad de formas que dependen de cada problema, ya que dependen directamente de la naturaleza de las señales y clases deseadas. Por ejemplo, en problemas de reconocimiento de caracteres, las características pueden incluir variables relacionadas con los valores de intensidad de las imágenes de entrada. Por otro lado, en problemas relacionados con reconocimiento de voz pueden incluir cuantificaciones de relaciones de ruido, longitud y amplitud del sonido o valores frecuenciales de la señal pueden ser ejemplos de características. Al concluir esta etapa, las características extraídas junto con las etiquetas correspondientes se utilizan para entrenar un modelo de aprendizaje automático, el cual puede ser posteriormente utilizado para asignar etiquetas a nuevas instancias de datos.

3.2.1. Modelos de Clasificación

En la tercera etapa del proceso de clasificación automática, las características obtenidas en la etapa anterior son utilizadas para entrenar un modelo de clasificación que posteriormente será capaz de asignar etiquetas de clase a nuevas instancias de datos. Existen diversos modelos reportados en la literatura, cada uno con sus propias ventajas y desventajas. La selección de un modelo apropiado para el problema suele ser una tarea heurística, resulta una práctica común el entrenar varios modelos para la misma tarea y compararlos entre ellos para encontrar el mejor. En esta sección se presentan algunos modelos de clasificación relevantes para este trabajo.

Máquina de Vectores de Soporte

En términos generales, la máquina de vectores de soporte (SVM, por sus siglas en inglés) se encarga de seleccionar un hiperplano que maximiza el margen entre las clases deseadas. Los datos del conjunto de entrenamiento que se encuentran más cerca de la superficie de decisión se conocen como vectores de soporte [68]. Si se considera el caso simplificado, en el cual se tienen dos clases con una separación lineal dentro del espacio de las características y una entrada bidimensional. Los datos de entrenamiento para un SVM están compuestos por un conjunto de n elementos, que consisten en un par que contiene un vector de características x_i y su correspondiente etiqueta de clase y_i , como se muestra en la Ecuación (3.4).

$$X = \{x_i, y_i\}_{i=1}^n; \quad x_i \in \mathbb{R}^2, \quad y_i \in \{-1, 1\}. \quad (3.4)$$

El hiperplano de separación óptima para la entrada en el caso lineal se da mediante la Ecuación (3.5).

$$w^T x_i + b = 0. \quad (3.5)$$

Las condiciones de clasificación pueden expresarse como la Ecuación (3.6)

$$y_i(\langle w \cdot x_i \rangle + b) \geq 1, \quad \forall i. \quad (3.6)$$

Con estas condiciones, el margen geométrico se representa de la siguiente manera

$$\begin{aligned} \gamma_i &= \frac{1}{2} \left(\left\langle \frac{w}{\|w\|} \cdot x^+ \right\rangle - \left\langle \frac{w}{\|w\|} \cdot x^- \right\rangle \right), \\ \gamma_i &= \frac{1}{\|w\|}. \end{aligned} \quad (3.7)$$

donde w se refiere al hiperplano de separación óptimo para el problema y b es el sesgo de la función. La distancia entre el plano elegido y los puntos de datos más cercanos a él se llama margen y se denota como los hiperplanos H_1 y H_2 . Optimizar el margen se interpreta como minimizar la norma del vector w .

Árbol de Decisiones

Los clasificadores de árboles de decisión, son un algoritmo de aprendizaje supervisado en el cual cualquier camino que comience desde la raíz está descrito por una secuencia de separación de datos hasta que se obtiene una salida booleana [69]. En un árbol de decisión, cada nodo representa una característica, cada enlace representa una decisión y cada hoja muestra un resultado que se repite hasta que no se pueda extraer más información. La Figura 3.2 muestra la estructura básica de un árbol de clasificación.

Las principales ventajas de los árboles de decisión radican en la simplicidad de su estructura en comparación con otros métodos, lo que proporciona una velocidad de predicción y entrenamiento relativamente rápida, así como un uso reducido de memoria. Estos algoritmos también pueden procesar datos numéricos y categóricos y requieren poca preparación de datos, ya que las características irrelevantes se utilizan menos al construir el árbol.

Sin embargo, los árboles de clasificación tienen algunas desventajas. Son sensibles a los cambios en los datos de entrenamiento, lo que resulta en un cambio considerable en la estructura final del árbol.

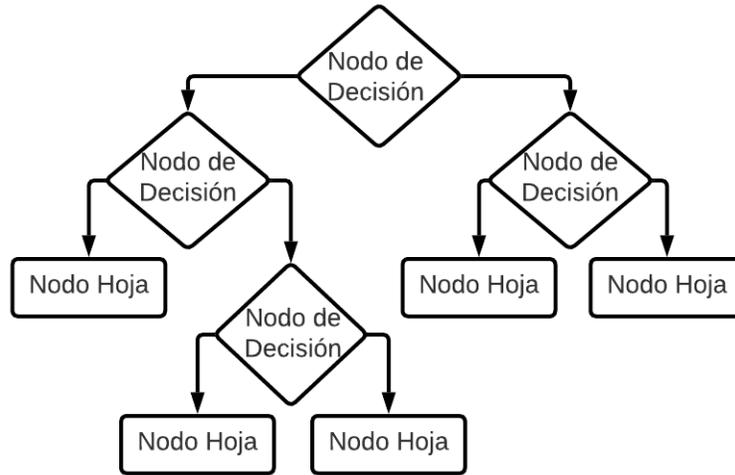


Figura 3.2: Estructura básica de un árbol de decisión.

También son propensos al sobreajuste, lo que se refiere a la sobrecomplicación del modelo de tal manera que no generaliza completamente el problema y solo funciona en los datos de entrenamiento. El sobreajuste se puede evitar mediante el uso de algoritmos de poda que eliminan secciones del árbol que son redundantes, reduciendo así la complejidad del clasificador final.

K Vecinos Más Cercanos

La idea principal del algoritmo del K -ésimo vecino más cercano (KNN, por sus siglas en inglés) es estimar la clase de una instancia no vista de datos utilizando la clasificación de las K instancias más cercanas a ella [70]. En su forma más básica, el algoritmo KNN primero encuentra los K puntos de datos de entrenamiento más cercanos a la instancia no vista y luego toma la clase más común. La precisión del algoritmo KNN se ve afectada negativamente por la presencia de ruido, características irrelevantes o si la distribución de clases está sesgada. Un valor grande de K reduce el impacto de la varianza en los datos, sin embargo, también aumenta el riesgo de ignorar patrones pequeños.

Considere un conjunto de datos de entrenamiento organizado en características y clases como se muestra en la Ecuación (3.8).

$$X = \{x_i, y_i\}_{i=1}^n; \quad x_i \in \mathbb{R}^d, \quad y_i \in \{-1, 1\}. \quad (3.8)$$

La distancia euclidiana en un espacio de dimensión d entre dos puntos dados en coordenadas cartesianas se calcula de la siguiente manera

$$D(X_1, X_2) = \sqrt{\sum_{j=1}^d (x_{1,j} - x_{2,j})^2}. \quad (3.9)$$

Un algoritmo KNN entrenado clasificaría una instancia no vista $X_q = \{x\}$ como la moda de la clase y de los k elementos más cercanos del conjunto de entrenamiento X .

Redes Neuronales Convolucionales

Un tipo de modelo de aprendizaje automático que se puede utilizar para la clasificación son las redes neuronales artificiales (ANNs). Una red neuronal está compuesta por nodos llamados neuronas artificiales que se organizan en capas. Durante la fase de entrenamiento, el algoritmo de entrenamiento calcula pesos sinápticos y cruces por cero que satisfacen ciertos criterios deseados. La arquitectura exacta de la red neuronal varía según la aplicación [71].

Las redes neuronales convolucionales (CNNs) son un tipo especializado de ANN que realiza operaciones de convolución en los datos de entrada y se utilizan con frecuencia en aplicaciones relacionadas con imágenes. Apilando múltiples capas convolucionales, la CNN puede aprender representaciones jerárquicas de los datos de entrada. En una imagen, por ejemplo, se puede interpretar como que las capas iniciales aprenden patrones simples, como bordes y texturas, mientras que las capas más profundas aprenden patrones más complejos, como formas.

Además de las capas convolucionales, las CNN incorporan capas de agrupación, que reducen las dimensiones espaciales de las capas anteriores, reduciendo el tamaño de las características extraídas. Las capas finales de la CNN suelen estar compuestas por capas totalmente conectadas, y en tareas de clasificación, a menudo se sigue con una función de activación softmax que produce una distribución de probabilidad sobre las posibles clases. La Figura 3.3 muestra la arquitectura básica de una CNN. Las capas convolucionales y de agrupación se pueden interpretar como la red aprendiendo sus propias características de los datos de entrada, luego la capa completamente conectada se entrena para etiquetar estas características.

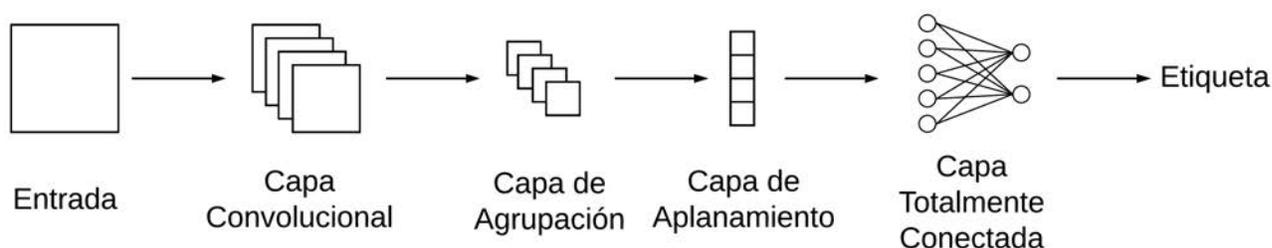


Figura 3.3: Esquema simplificado de la estructura de la CNN.

Una de las principales ventajas de utilizar CNN como modelos de clasificación es su capacidad para aprender y extraer sus propias características a partir de datos en bruto, lo cual es útil cuando se presentan problemas en los que los patrones en las características comúnmente utilizadas no se presentan de manera natural. Por otro lado, durante el paso de entrenamiento, las CNN necesitan grandes cantidades de datos etiquetados y a menudo también requieren conjuntos de datos equilibrados. Este proceso también es intensivo en recursos computacionales. Las CNN se consideran modelos de caja negra, lo que significa que los pesos obtenidos para la red a menudo carecen de interpretabilidad. Estos modelos también son propensos al sobreajuste, lo que se refiere a la situación en la que el modelo se entrena para resolver el problema deseado específicamente para los datos de entrenamiento, perdiendo generalidad y fallando al rendir cuando se presentan nuevos datos.

Capítulo 4

Control de Sistemas Caóticos Discretos de Orden Fraccionario y su Aplicación en Sistemas Criptográficos

Los sistemas caóticos son sistemas deterministas no lineales que presentan una sensibilidad relativamente elevada a sus parámetros y condiciones iniciales [72]. Estos sistemas, para ciertos conjuntos de parámetros, presentan comportamientos aparentemente impredecibles a pesar de ser deterministas y se presentan en una amplia variedad de campos como economía, ingeniería, biología, física, meteorología, entre otras. Para valores iniciales arbitrariamente cercanos los sistemas caóticos presentan dinámicas completamente distintas. En otras palabras, un cambio relativamente pequeño en las condiciones iniciales o parámetros del sistema genera trayectorias cada vez más distintas al avanzar el tiempo. Debido a esto, la predicción de la trayectoria exacta de un sistema caótico a largo plazo resulta prácticamente imposible si no se conocen los valores exactos de sus parámetros.

La imprevisibilidad inherente de los sistemas caóticos plantea desafíos en su análisis, control y aplicación práctica. Sin embargo, para ciertas condiciones, algunos sistemas caóticos acoplados presentan el fenómeno de sincronización, el cual consiste en el establecimiento de correlaciones temporales entre las variables dinámicas de los sistemas acoplados. Este fenómeno resulta contrastante con la naturaleza no lineal y sensible de los sistemas caóticos. El problema de sincronización idéntica entre dos sistemas caóticos de orden fraccionario en un esquema de acoplamiento unidireccional puede ser interpretado desde el punto de vista de teoría de control como la necesidad de diseñar un controlador que manipule al sistema seguidor de tal manera que este imite la dinámica generada por un sistema líder. En otras palabras, el sistema líder en el esquema provee una referencia cambiante en el tiempo la cual debe ser alcanzada por el sistema seguidor. Las dinámicas obtenidas en este tipo de esquemas de sincronización pueden ser posteriormente aprovechadas para la generación de secuencias pseudoaleatorias que pueden ser posteriormente utilizadas por algoritmos de encriptación.

La Figura 4.1 presenta el esquema general para la encriptación imágenes empleando generadores de secuencias pseudoaleatorias basados en sistemas caóticos, en donde los parámetros del sistema pueden ser utilizados como llaves. La trayectoria obtenida por el sistema líder es utilizada para la creación

de la secuencia pseudoaleatoria, la cual es entonces utilizada para codificar la información deseada. Posteriormente, es posible recuperar la misma secuencia pseudoaleatoria del sistema líder utilizando un esquema de sincronización idéntica; en este caso a partir de uno de los estados del sistema líder. Aplicando el proceso inverso a la imagen cifrada se obtiene la imagen original.

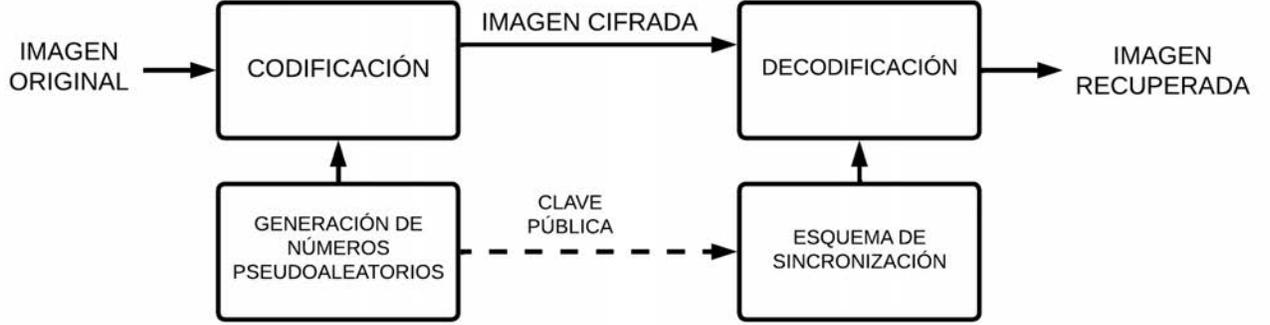


Figura 4.1: Esquema general de encriptación con generación de secuencias pseudoaleatorias basada en sistemas caóticos.

4.1. Controlador Propuesto

Definición 5. Considerando un sistema caótico $\Phi_i(k)$, la condición de sincronización idéntica se define en la Ecuación (4.1)

$$\|\Phi_2(k) - \Phi_1(k)\| \rightarrow 0; \quad k \rightarrow \infty. \quad (4.1)$$

Esto significa que, independientemente de sus condiciones iniciales o trayectorias, la condición de sincronización idéntica se cumple si ambos sistemas exhiben los mismos valores en un tiempo finito k [73]. Los esquemas de sincronización pueden categorizarse según cómo estén acoplados los sistemas. Los esquemas de sincronización unidireccional están compuestos por un sistema seguidor que recibe información de un sistema líder de tal manera que se cumple la condición de sincronización. Desde el punto de vista de la teoría de control, los esquemas de sincronización unidireccional pueden interpretarse como un problema de control en el cual el sistema líder produce una señal de referencia que varía con el tiempo, mientras que el sistema seguidor a menudo necesita una ley de control que garantice la condición de sincronización. El controlador propuesto para la sincronización idéntica de dos sistemas caóticos fraccionarios discretos con orden variable en un esquema unidireccional se presenta a continuación. Considere la forma general de un mapa caótico definido en el dominio temporal $k \in \mathbb{N}$ con orden de diferencia variable, como se muestra en la Ecuación (4.2), que se utilizará como sistema líder.

$$\begin{aligned} \Delta^{v(k)} \bar{x}(k) &= \sum_{i=1}^n f_i(\bar{x}(k), \bar{y}(k)), \\ \Delta^{v(k)} \bar{y}(k) &= \sum_{j=1}^m g_j(\bar{x}(k), \bar{y}(k)), \end{aligned} \quad (4.2)$$

donde f_i y g_j son funciones auxiliares que describen el comportamiento del sistema caótico. El sistema seguidor se define entonces en la Ecuación (4.3).

$$\begin{aligned} \Delta^{v(k)} x(k) &= \sum_{i=1}^n f_i(x(k), y(k)) + u_x, \\ \Delta^{v(k)} y(k) &= \sum_{j=1}^m g_j(x(k), y(k)) + u_y, \end{aligned} \quad (4.3)$$

donde u_x y u_y representan la ley de control deseada que garantiza la convergencia de las trayectorias de ambos sistemas. A partir de las Ecuaciones (4.2) y (4.3), la función de error para cada estado se define en la Ecuación (4.4).

$$\begin{aligned} e_x(k) &= \sum_{i=1}^n f_i(\bar{x}(k), y(k)) - f_i(x(k), y(k)), \\ e_y(k) &= \sum_{j=1}^m g_j(\bar{x}(k), y(k)) - f_j(x(k), y(k)). \end{aligned} \quad (4.4)$$

La dinámica del error se puede representar en la Ecuación (4.5).

$$\begin{aligned} \Delta^{v(k)} e_x(k) &= \sum_{i=1}^n f_i(e_{xi}(k)) + u_x(k), \\ \Delta^{v(k)} e_y(k) &= \sum_{j=1}^m g_j(e_{yj}(k)) + u_y(k). \end{aligned} \quad (4.5)$$

A partir de las Ecuaciones (4.3) y (4.5), la ley de control se propone de la siguiente manera:

$$\begin{aligned} u_x &= -\sum_{i=1}^n f_i(e_{xi}(k)) - k_x e_x(k) - I_x S_x(k), \\ u_y &= -\sum_{j=1}^m g_j(e_{yj}(k)) - k_y e_y(k) - I_y S_y(k), \end{aligned} \quad (4.6)$$

donde I_x , I_y , k_x y k_y son constantes positivas. $S_x(k)$ y $S_y(k)$ representan la suma del error, lo que significa que I_x e I_y pueden interpretarse como ganancias integrales que permiten al controlador compensar cambios en la señal de referencia, mientras que k_x y k_y son ganancias proporcionales.

Teorema 1. *La ley de control presentada en la Ecuación (4.6) cumple con la condición de sincronización mostrada en la Ecuación (4.1) para el sistema líder (ver Ecuación (4.2)) y el seguidor (ver Ecuación (4.3)).*

Prueba. La función de Lyapunov mostrada en la Ecuación (4.7) demuestra la estabilidad asintótica del error entre ambos sistemas. Si $e(k) \rightarrow 0$, $k \rightarrow \infty$, entonces el controlador diseñado satisface la condición de sincronización idéntica.

$$V(k) = \frac{1}{2} \left(e_x^2(k) + e_y^2(k) \right). \quad (4.7)$$

La derivada fraccionaria discreta de la función de Lyapunov propuesta se presenta en la Ecuación (4.8).

$$\Delta^{v(k)} V(k) = e_x(k) \Delta^{v(k)} e_x(k) + e_y(k) \Delta^{v(k)} e_y(k). \quad (4.8)$$

A partir de las Ecuaciones (4.5) y (4.6), la dinámica de las funciones de error puede describirse como se muestra en la Ecuación (4.9).

$$\begin{aligned} \Delta^{(v)} e_x(k) &= \sum_{i=1}^n f_i(e_{xi}(k)) - \sum_{i=1}^n f_i(e_{xi}(k)) - k_x e_x(k) - I_x S_x(k), \\ \Delta^{(v)} e_y(k) &= \sum_{j=1}^m g_j(e_{yj}(k)) - \sum_{j=1}^m g_j(e_{yj}(k)) - k_y e_y(k) - I_y S_y(k). \end{aligned} \quad (4.9)$$

La dinámica de la función de error puede reducirse como se muestra en la Ecuación (4.10).

$$\begin{aligned} \Delta^{(v)} e_x(k) &= -k_x e_x(k) - I_x S_x(k), \\ \Delta^{(v)} e_y(k) &= -k_y e_y(k) - I_y S_y(k), \end{aligned} \quad (4.10)$$

Al utilizar la Ecuación (4.10) y reescribir la Ecuación (4.8), obtenemos la Ecuación (4.11), que se presenta como sigue:

$$\Delta^{v(k)} V(k) = e_x(k) (-k_x e_x(k) - I_x S_x(k)) + e_y(k) (-k_y e_y(k) - I_y S_y(k)). \quad (4.11)$$

A partir de la Ecuación (4.11), se derivan las siguientes condiciones:

$$\begin{aligned} -k_x e_x^2(k) - I_x S_x e_x &< 0, \\ -k_y e_y^2(k) - I_y S_y e_y &< 0. \end{aligned} \quad (4.12)$$

Para cumplir con las condiciones de estabilidad de Lyapunov, se especifican las siguientes restricciones: $k_i > 0$ e $I_i > 0$. \square

4.1.1. Caso 1. Mapa de Tinkerbell

El mapa Tinkerbell es un sistema discreto caótico de dos estados que muestra comportamiento caótico. Se dice que recibe su nombre debido a la semejanza de su gráfico de fase con la trayectoria de vuelo de un hada [74]. El mapa Tinkerbell fraccionario con orden variable se muestra en la Ecuación (4.13).

$$\begin{aligned} \Delta^{v(k)} \bar{x}(k) &= \bar{x}^2(k-1) - \bar{y}^2(k-1) + a\bar{x}(k-1) + b\bar{y}(k-1) + \bar{x}(k-1), \\ \Delta^{v(k)} \bar{y}(k) &= 2\bar{x}(k-1)\bar{y}(k-1) + c\bar{x}(k-1) + d\bar{y}(k-1) + \bar{y}(k-1), \end{aligned} \quad (4.13)$$

donde a, b, c, d son parámetros del sistema. La Figura 4.3 muestra el retrato de fase de la Ecuación (4.13) para diferentes valores de $v(k)$ descritos por funciones arbitrariamente elegidas. En todos los casos, las condiciones iniciales se establecen en $[-0.7, -0.6]$ y los parámetros en $[a, b, c, d] = [0.9, -0.6013, 2, 0.5]$, $T_{sim} = 10000$. En el caso de la Figura 4.2d, se eligió un orden variable y se describió en la Ecuación (4.14) con parámetros $[s_1, \omega, s_2] = [0.0002, 0.001, 0.9998]$.

$$S(k) = s_1 \sin(2\pi\omega k) + s_2. \quad (4.14)$$

Para el esquema de sincronización propuesto, el sistema seguidor Tinkerbell se define entonces como se muestra en la Ecuación (4.15).

$$\begin{aligned} \Delta^{v(k)} x(k) &= x^2(k-1) - y^2(k-1) + ax(k-1) + by(k-1) + x(k-1) + u_x(k-1), \\ \Delta^{v(k)} y(k) &= 2x(k-1)y(k-1) + cx(k-1) + dy(k-1) + y(k-1) + u_y(k-1). \end{aligned} \quad (4.15)$$

Las funciones auxiliares mostradas en la Ecuación (4.2) se definen mediante el sistema Tinkerbell fraccionario de la siguiente manera:

$$\begin{aligned} f_1(x(k), y(k)) &= x^2(k), & g_1(x(k), y(k)) &= 2x(k)y(k), \\ f_2(x(k), y(k)) &= -y^2(k), & g_2(x(k), y(k)) &= cx(k), \\ f_3(x(k), y(k)) &= ax(k), & g_3(x(k), y(k)) &= dy(k), \\ f_4(x(k), y(k)) &= by(k), & & \end{aligned} \quad (4.16)$$

las funciones auxiliares de error definidas en la Ecuación (4.4) se proponen en la Ecuación (4.17).

$$\begin{aligned} e_1(k) &= x(k) - \bar{x}(k), \\ e_2(k) &= y(k) - \bar{y}(k), \\ e_3(k) &= x^2(k) - \bar{x}^2(k), \\ e_4(k) &= y^2(k) - \bar{y}^2(k), \\ e_5(k) &= x(k)y(k) - \bar{x}(k)\bar{y}(k). \end{aligned} \quad (4.17)$$

Finalmente, la ley de control propuesta para el objetivo de sincronización se muestra en la Ecuación (4.18).

$$\begin{aligned} u_x(k) &= -e_3(k) + e_4(k) - ae_1(k) - be_2(k) - e_1(k) - k_x e_1(k) - I_x S_x, \\ u_y(k) &= -2e_5(k) - ce_1(k) - de_2(k) - e_2(k) - k_y e_2(k) - I_y S_y. \end{aligned} \quad (4.18)$$

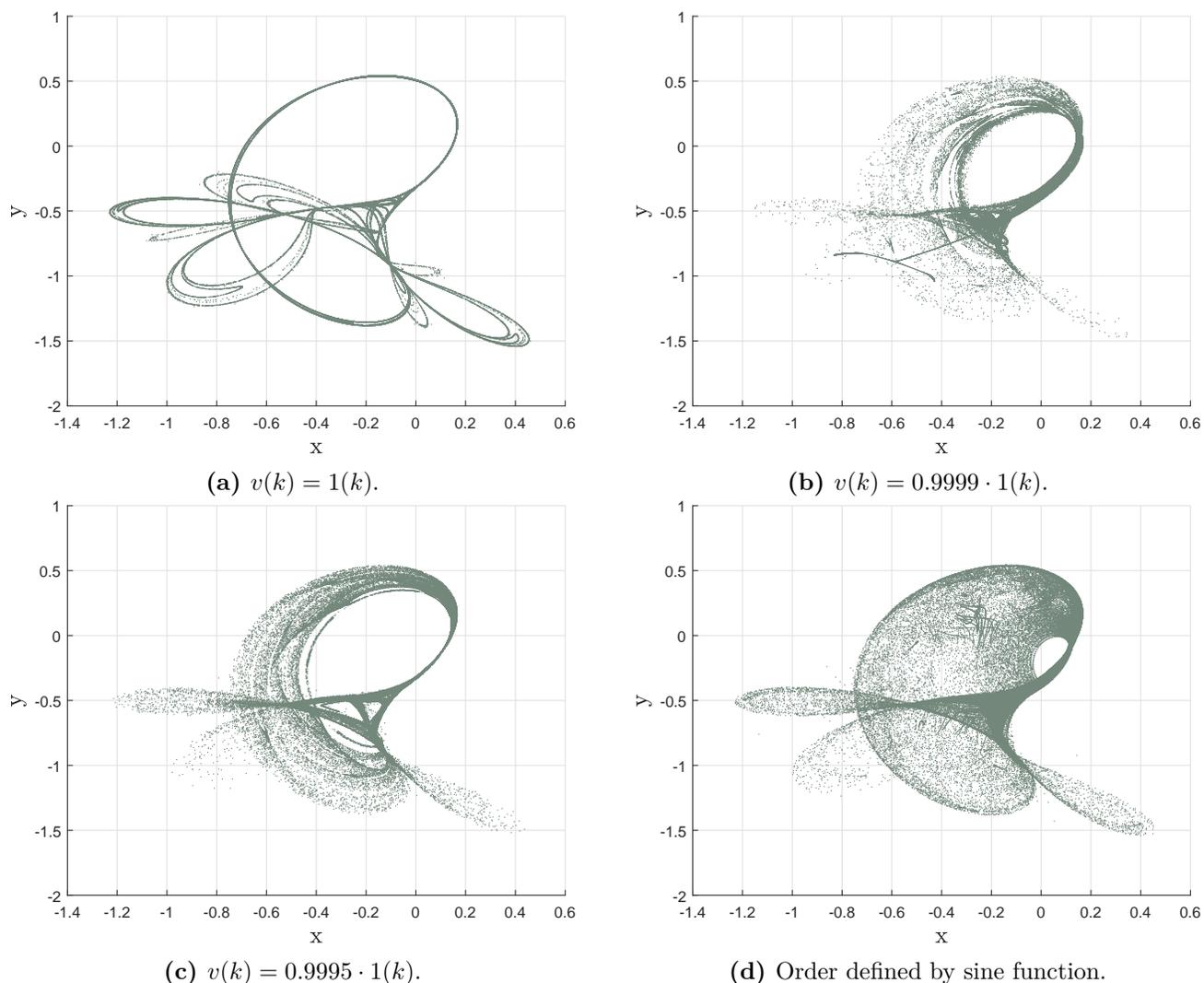


Figura 4.3: Retrato fase del mapa caótico de orden fraccionario Tinkerbell con ordenes de diferencia $v(k)$ definidos por funciones escogidas arbitrariamente.

La Figura 4.4 muestra los resultados del esquema de sincronización propuesto basado en el controlador. En este caso, los parámetros para ambos sistemas se establecieron iguales: $[a, b, c, d] = [0.9, -0.6013, 2, 0.5]$, con condiciones iniciales $[x_0, y_0] = [-0.7, -0.6]$. La diferencia entre los sistemas radica en el orden fraccionario. Para el sistema líder, $v(k) = 0.99 \cdot 1(k)$, mientras que para el sistema seguidor, $v(k) = 0.98 \cdot 1(k)$. Los parámetros del controlador se seleccionaron empíricamente como $[k_x, k_y, I_x, I_y] = [0.05, 0.05, 0.2, 0.5]$ con tiempo de activación $k = 1000$.

4.1.2. Caso 2. Mapa del Ave Mítica

Ambos sistemas caóticos presentados en las siguientes subsecciones fueron presentados en el coloquio "Transformations Ponctuelles et Leurs Applications"(Mapeo de Puntos y sus Aplicaciones), celebrado

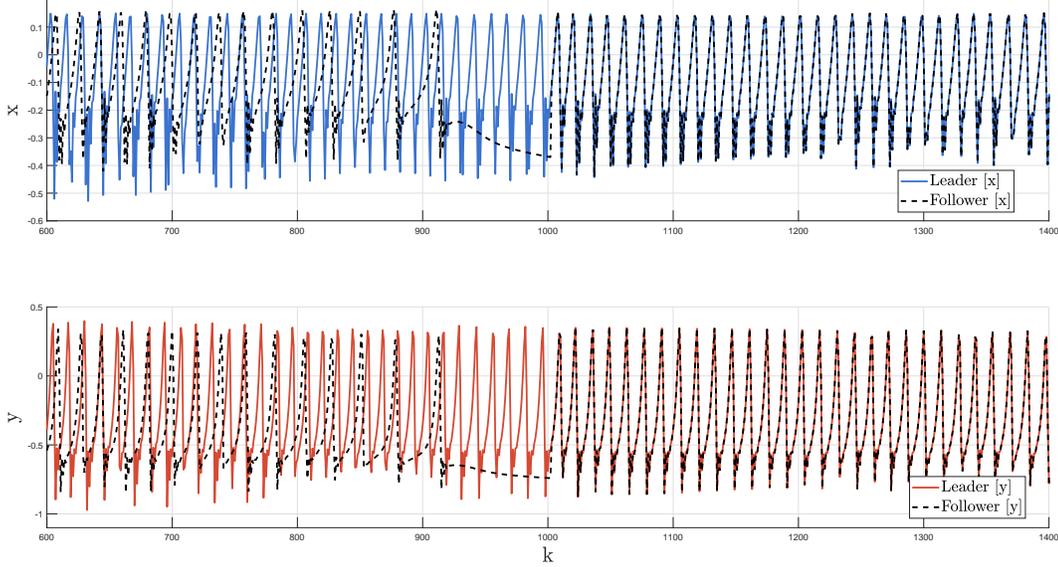


Figura 4.4: Sincronización de dos mapas Tinkerbell de orden fraccionario mediante la metodología propuesta.

en París en 1976 [75]. El mapa del ave mítica se describe de la siguiente manera:

$$\begin{aligned}\Delta^{v(k)}\bar{x}(k) &= \bar{x}(k-1) + \bar{y}(k-1) + a(1 - 0.05\bar{y}^2(k-1))\bar{y}(k-1) + \eta\bar{x}(k-1) + \frac{2(1-\eta)\bar{x}^2(k-1)}{1+\bar{x}^2(k-1)}, \\ \Delta^{v(k)}\bar{y}(k) &= \bar{y}(k-1) - \bar{x}(k-1) + \eta\bar{x}(k) + \frac{2(1-\eta)\bar{x}^2(k)}{1+\bar{x}^2(k)},\end{aligned}\quad (4.19)$$

donde tanto a como η corresponden a constantes del sistema. La Figura 4.5 muestra el mapa del ave mítica de orden fraccionario con condiciones iniciales $[0.1, 0]$ y parámetros $[a, \eta] = [0.009, \cos(4\pi/5) + 0.008]$ en un tiempo total de simulación $T_{sim} = 10000$. El orden $v(k)$ se definió arbitrariamente utilizando diferentes funciones. La Figura 4.5c muestra la dinámica del sistema para un orden definido por una función gaussiana, como se describe en la Ecuación (4.20), pero normalizado según se muestra en la Ecuación (4.21), con parámetros $[n_1, l, r, n_2] = [0.3, 5000, 0.0007, 0.7]$.

$$G(k) = \frac{1}{r\sqrt{2\pi}} \exp\left(-\frac{(k-l)^2}{2r^2}\right). \quad (4.20)$$

$$GN(k) = n_1 \frac{G(k)}{\max(G(k))} + n_2. \quad (4.21)$$

En el caso de la Figura 4.5d, se seleccionó el orden como una función senoidal, como se muestra en la Ecuación (4.14), con parámetros $[s_1, \omega, s_2] = [0.15, 0.001, 0.85]$.

Siguiendo el esquema de sincronización propuesto, tomando la Ecuación (4.19) como el sistema líder, el sistema seguidor se define como se muestra en la Ecuación (4.22).

$$\begin{aligned}\Delta^{v(k)}x(k) &= x(k-1) + y(k-1) + a(1 - 0.05y^2(k-1))y(k-1) + \eta x(k-1) + \frac{2(1-\eta)x^2(k-1)}{1+x^2(k-1)} + u_x(k-1), \\ \Delta^{v(k)}y(k) &= y(k-1) - x(k-1) + \eta x(k) + \frac{2(1-\eta)x^2(k)}{1+x^2(k)} + u_y(k-1).\end{aligned}\quad (4.22)$$

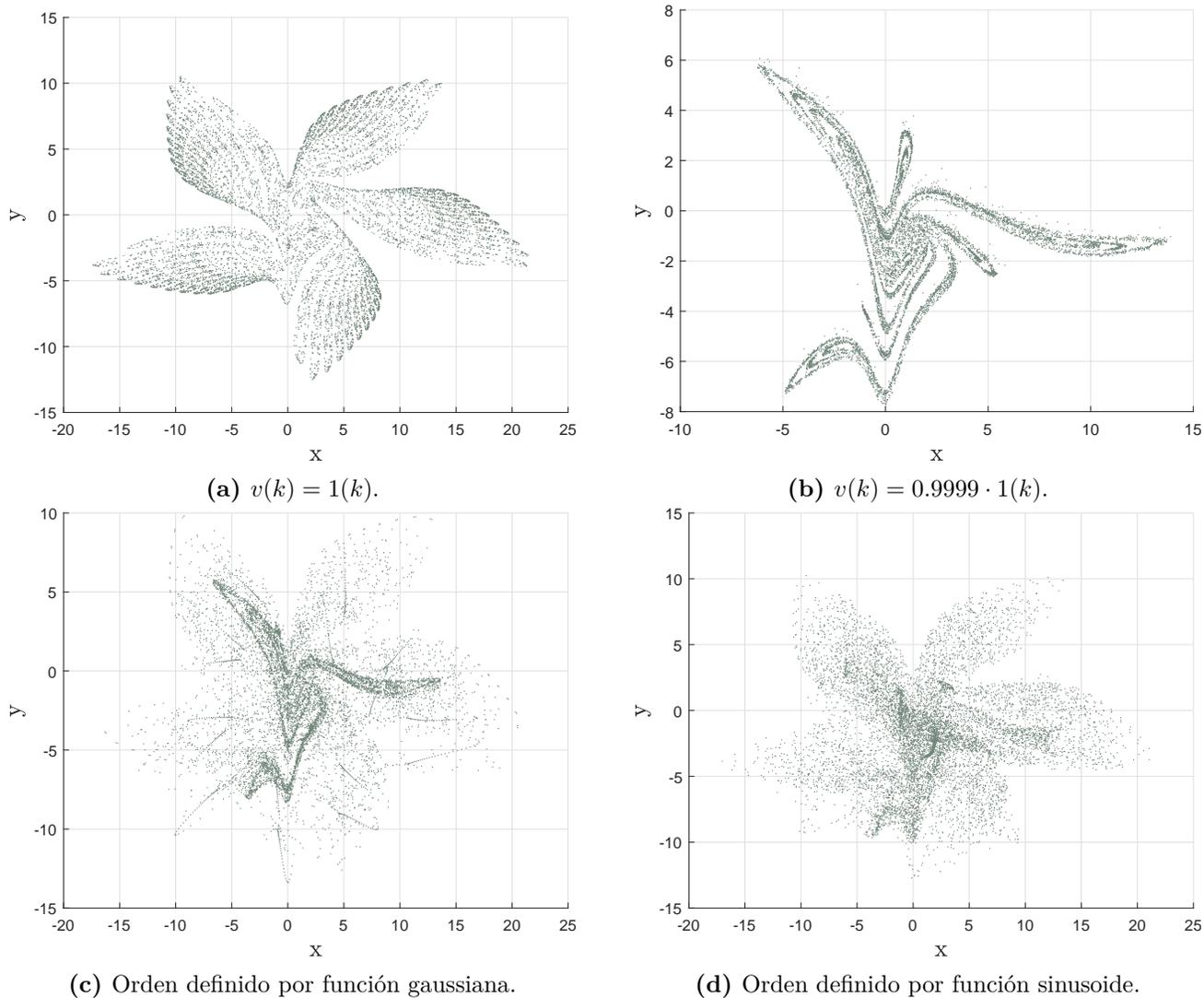


Figura 4.5: Retrato fase del mapa caótico ave mítica de orden fraccionario para diversos valores de $v(k)$, escogidos de manera arbitraria.

Las funciones auxiliares que describen el comportamiento del sistema del ave mítica se proponen en la Ecuación (4.23).

$$\begin{aligned}
 f_1(x(k), y(k)) &= y(k), & g_1(x(k), y(k)) &= -x(k), \\
 f_2(x(k), y(k)) &= ay(k), & g_2(x(k), y(k)) &= \eta x(k+1), \\
 f_3(x(k), y(k)) &= -0.05ay^3(k), & g_3(x(k), y(k)) &= \frac{2(1-\eta)x^2(k+1)}{1+x^2(k+1)}. \\
 f_4(x(k), y(k)) &= \eta x(k), & & \\
 f_5(x(k), y(k)) &= \frac{2(1-\eta)x^2(k)}{1+x^2(k)}, & &
 \end{aligned} \tag{4.23}$$

4.1. CONTROLADOR PROPUESTO

Las funciones de error auxiliares se proponen de la siguiente manera:

$$\begin{aligned}
 e_1(k) &= x(k) - \bar{x}(k), \\
 e_2(k) &= y(k) - \bar{y}(k), \\
 e_3(k) &= x^2(k) - \bar{x}^2(k), \\
 e_4(k) &= y^2(k) - \bar{y}^2(k), \\
 e_5(k) &= y^3(k) - \bar{y}^3(k), \\
 e_6(k) &= \frac{x^2(k)}{1+x^2(k)} - \frac{\bar{x}^2(k)}{1+\bar{x}^2(k)}.
 \end{aligned} \tag{4.24}$$

Finalmente, la ley de control propuesta se muestra en la Ecuación (4.18)

$$\begin{aligned}
 u_x(k) &= e_1(k) - (1+a)e_2(k) + 0.05ae_5(k) - \eta e_1(k) - 2(1-\eta)e_6(k) - k_x e_1(k) - I_x S_x(k), \\
 u_y(k) &= e_2(k) + e_1(k) - \eta e_1(k+1) - 2(1-\eta)e_6(k+1) - k_y e_2(k) - I_y S_y(k).
 \end{aligned} \tag{4.25}$$

La Figura 4.6 presenta el esquema de sincronización propuesto para dos mapas caóticos fraccionarios del ave mítica. Una vez más, los parámetros se establecieron iguales entre los sistemas con $[a, \eta] = [0.009, \cos(4\pi/5) + 0.008]$ y condiciones iniciales $[x_0, y_0] = [0.15, 0.1]$. La ley de control se inicia en $k = 1000$. El orden de diferencia para el sistema líder se eligió arbitrariamente como una onda senoidal, como se muestra en la Ecuación (4.14), con parámetros $[s_1, \omega, s_2] = [0.1, 0.001, 0.9]$, mientras que el orden del sistema seguidor se definió como constante con un valor $v(k) = 0.95$.

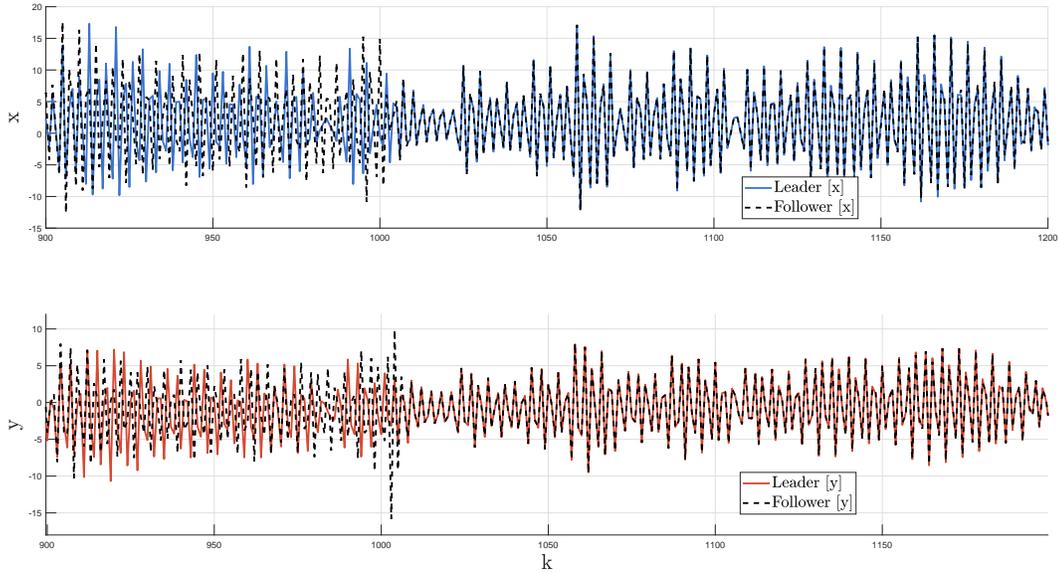


Figura 4.6: Resultados del esquema de control propuesto para la sincronización de dos mapas ave mítica de orden fraccionario.

4.1.3. Caso 3. Mapa Ave del Paraíso

El mapa del ave del paraíso, descrito en [75], se representa mediante la Ecuación (4.26).

$$\begin{aligned}\Delta^{v(k)}\bar{x}(k) &= \bar{x}(k-1) + \bar{y}(k-1) + a(1 - \sigma\bar{y}^2(k-1))\bar{y}(k-1) + \eta\bar{x}(k-1) + (1 - \eta)\bar{x}^2(k-1)e^{\frac{1-\bar{x}^2(k-1)}{4}}, \\ \Delta^{v(k)}\bar{y}(k) &= \bar{y}(k-1) - \bar{x}(k-1) + \eta\bar{x}(k) + \frac{2(1-\eta)\bar{x}^2(k)}{1+\bar{x}^2(k)},\end{aligned}\tag{4.26}$$

donde a , η y σ se refieren a los parámetros del sistema. La Figura 4.7 muestra diferentes simulaciones del mapa del ave del paraíso de orden fraccionario con valores de parámetros $[a, \eta, \sigma] = [0.0083, -0.38, 0.1]$. El sistema se simuló durante $T_{sim} = 10000$ con condiciones iniciales $[0.15, 0]$. En todos los casos, el orden fraccionario $v(k)$ se eligió arbitrariamente. Las Figuras 4.7a y 4.7b presentan órdenes de diferencia constantes. Por otro lado, la Figura 4.7c se definió como se muestra en la Ecuación (4.20) con parámetros $[n_1, l, r, n_2] = [-0.1, 5000, 0.005, 1]$. Finalmente, la Figura 4.7d presenta el mapa caótico del ave del paraíso con un orden variable definido por una onda senoidal, como se muestra en la Ecuación (4.14) con parámetros $[s_1, \omega, s_2] = [0.02, 0.001, 0.98]$.

Siguiendo la metodología propuesta, el sistema seguidor se define como se muestra en la Ecuación (4.27).

$$\begin{aligned}\Delta^{v(k)}x(k) &= x(k-1) + y(k-1) + a(1 - \sigma y^2(k-1))y(k-1) + \eta x(k-1) + \dots \\ &\quad (1 - \eta)x^2(k-1)e^{\frac{1-x^2(k-1)}{4}} + u_x(k-1), \\ \Delta^{v(k)}y(k) &= y(k-1) - x(k-1) + \eta x(k) + \frac{2(1-\eta)x^2(k)}{1+x^2(k)} + u_y(k-1).\end{aligned}\tag{4.27}$$

Las funciones auxiliares se muestran en la Ecuación (4.28).

$$\begin{aligned}f_1(x(k), y(k)) &= y(k), & g_1(x(k), y(k)) &= -x(k), \\ f_2(x(k), y(k)) &= ay(k), & g_2(x(k), y(k)) &= \eta x(k+1), \\ f_3(x(k), y(k)) &= -\sigma ay^3(k), & g_3(x(k), y(k)) &= \frac{2(1-\eta)x^2(k+1)}{1+x^2(k+1)}, \\ f_4(x(k), y(k)) &= \eta x(k), \\ f_5(x(k), y(k)) &= (1 - \eta)x^2(k)\exp\left(\frac{1-x^2(k)}{4}\right),\end{aligned}\tag{4.28}$$

La Ecuación (4.29) presenta las funciones de error auxiliares para el sistema del ave del paraíso.

$$\begin{aligned}e_1(k) &= x(k) - \bar{x}(k), \\ e_2(k) &= y(k) - \bar{y}(k), \\ e_3(k) &= x^2(k) - \bar{x}^2(k), \\ e_4(k) &= y^2(k) - \bar{y}^2(k), \\ e_5(k) &= y^3(k) - \bar{y}^3(k), \\ e_6(k) &= x^2(k)\exp\left(\frac{1-x^2(k)}{4}\right) - \bar{x}^2(k)\exp\left(\frac{1-\bar{x}^2(k)}{4}\right), \\ e_7(k) &= \frac{x^2(k)}{1+x^2(k)} - \frac{\bar{x}^2(k)}{1+\bar{x}^2(k)}.\end{aligned}\tag{4.29}$$

A partir de las Ecuaciones (4.6) y (4.29), la Ecuación (4.30) presenta la ley de control propuesta para el mapa fraccionario del ave del paraíso.

$$\begin{aligned}u_x(k) &= -e_1(k) - (a+1)e_2(k) + \sigma e_5(k) - \eta e_1(k) - (1-\eta)e_6(k) - k_x e_1(k) - I_x S_x(k), \\ u_y(k) &= -e_2(k) + e_1(k) - \eta e_1(k+1) - (1-\eta)e_7(k+1) - k_y e_2(k) - I_y S_y(k).\end{aligned}\tag{4.30}$$

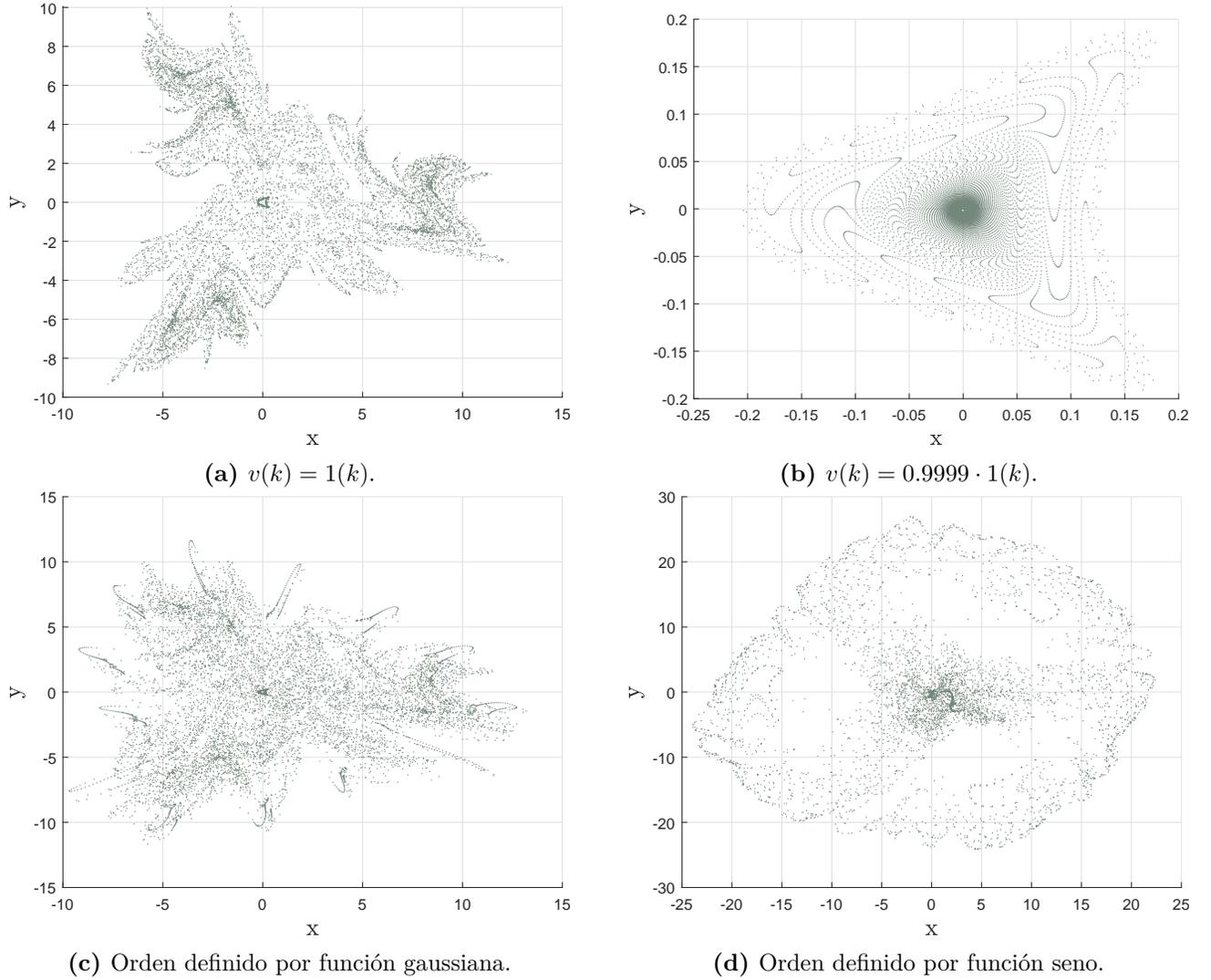


Figura 4.7: Retrato fase del mapa caótico de orden fraccionario ave del paraíso con ordenes de diferencia $v(k)$ definidos por funciones escogidas arbitrariamente.

La Figura 4.8 muestra la simulación del esquema de control propuesto con los parámetros de ambos sistemas configurados en los mismos valores: $[a, \eta, \sigma] = [0.0083, -0.38, 0.1]$, con condiciones iniciales $[x_0, y_0] = [0.15, 0.1]$. El tiempo total de simulación fue $T_{sim} = 2000$ y la acción del controlador comenzó en el instante $k = 1000$. El orden de diferencia del sistema líder se seleccionó arbitrariamente como un valor constante $v(k) = 0.99$, mientras que al sistema seguidor se le asignó un orden fraccionario definido por una función gaussiana (como se muestra en la Ecuación (4.20)) con parámetros $[n_1, l, r, n_2] = [-0.1, 1000, 0.005, 1]$. Los parámetros del controlador se eligieron de manera heurística con los siguientes valores: $[k_x, k_y, I_x, I_y] = [0.005, 0.05, 0.7, 0.4]$.

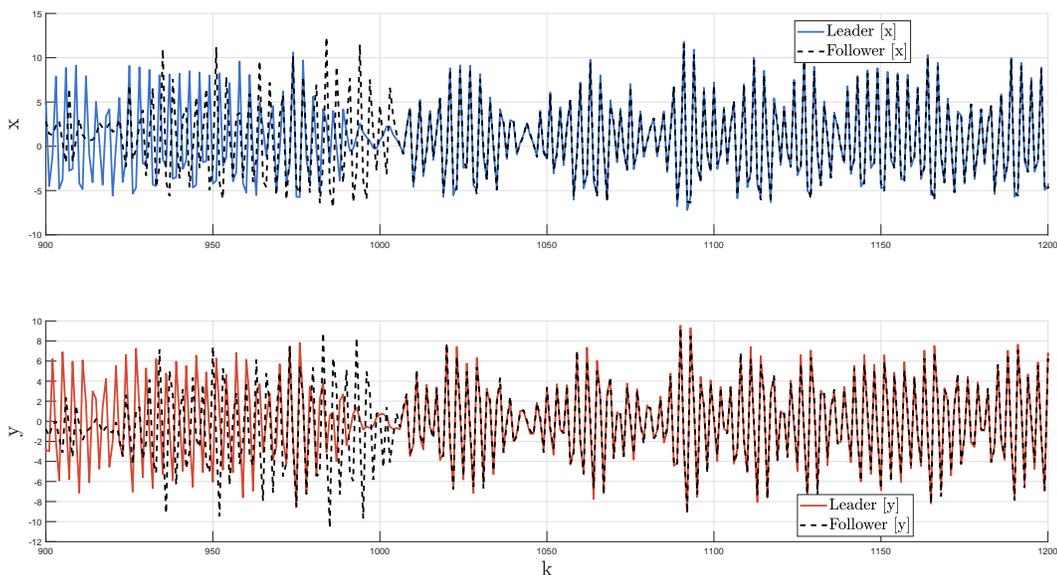


Figura 4.8: Esquema de sincronización propuesto para dos mapas caóticos ave del paraíso de orden fraccionario.

4.1.4. Implementación

Las placas de control Arduino son una elección popular para probar esquemas de implementación en diferentes campos de la ingeniería [76, 77]. Para este trabajo, se eligió una placa Arduino Uno para la implementación del esquema de sincronización propuesto. Basada en el microcontrolador ATmega328P, esta placa cuenta con las siguientes especificaciones técnicas relevantes:

- 6 pines PWM.
- 6 pines de entrada analógica.
- 32KB de memoria flash.
- 2KB de SRAM.
- 1KB de EEPROM.
- Velocidad de reloj de 16MHz.

El esquema de implementación se puede describir con los siguientes pasos:

1. Se configuran los parámetros del sistema tanto para el sistema líder como para el seguidor, junto con el orden de diferencia para cada sistema y la longitud del búfer de memoria. Arduino 1 simula el sistema líder, mientras que Arduino 2 simula el seguidor.
2. Se inicia la comunicación serial para ambas placas Arduino Uno.
3. Ambos sistemas actualizan sus valores individuales para la función auxiliar $a^v(k)$ mostrada en la Ecuación (2.19).
4. Arduino 1 actualiza los valores del sistema líder mostrados en la Ecuación (4.2) y transmite el valor calculado a través de la comunicación serial a Arduino 2.

4.1. CONTROLADOR PROPUESTO

5. Arduino 2 recibe el valor de Arduino 1 y actualiza el valor del sistema seguidor presentado en la Ecuación (4.3).
6. Arduino 2 envía tanto los valores actualizados del sistema líder como del sistema seguidor a través de la comunicación serial a la computadora.
7. Se actualizan los búferes de memoria en ambas placas.

Las limitaciones de hardware en la placa Arduino UNO, específicamente la precisión con la que puede representar y adquirir datos, no permiten la representación correcta de los sistemas caóticos elegidos. La transferencia de datos entre las placas y la computadora se realizó mediante comunicación por puerto serie.

La Figura 4.9 muestra la configuración utilizada. Los pines PWM de las placas Arduino UNO se utilizaron para emitir los valores de simulación tanto del sistema líder como del seguidor con fines representativos. La señal obtenida se filtra a través de un filtro paso bajo y se muestra en el osciloscopio. La computadora grafica los valores obtenidos de ambas placas a través de la comunicación serial a una velocidad de baudios de 9600.

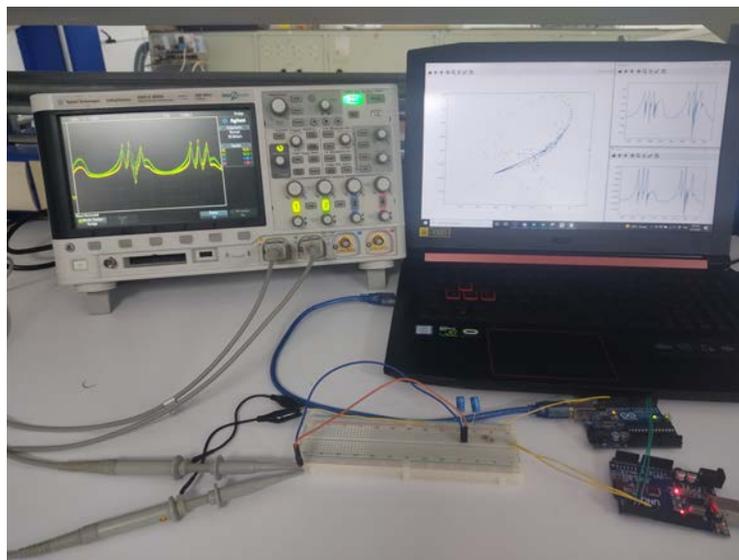


Figura 4.9: Implementación del esquema de sincronización propuesto en tarjetas Arduino UNO.

La Figura 4.10 muestra el segundo estado tanto para el sistema líder como para el sistema seguidor del sistema Tinkerbell fraccionario con el esquema de control propuesto activado en el instante de simulación $k = 100$. Las señales se obtuvieron directamente de la comunicación por puerto serie de Arduino UNO. En este caso, los valores de los parámetros para los sistemas eran iguales. La diferencia entre los sistemas radica en el orden de diferencia: para el sistema líder se seleccionó un orden variable $v(k) = 0.01 \sin(\pi k/50) + 0.99$, mientras que para el sistema seguidor se seleccionó un orden fraccionario como $v(k) = 0.999 \cdot 1(k)$. La ley de control propuesta es capaz de compensar los cambios dinámicos generados por el orden fraccionario variable en el tiempo.

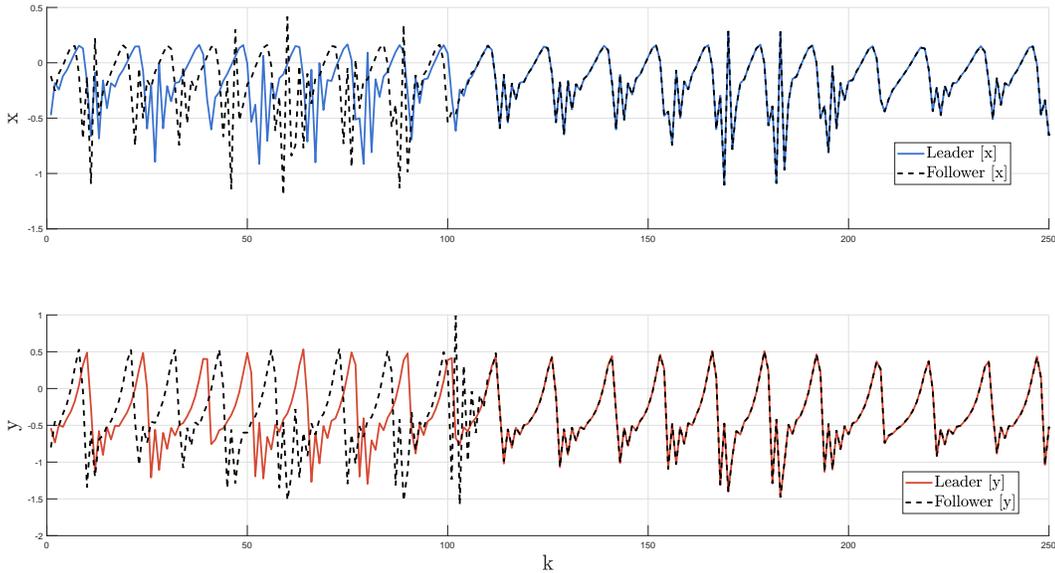


Figura 4.10: Sincronización de dos mapas ave del paraíso de orden fraccionario mediante la metodología propuesta. Se emplean líneas continuas por claridad.

4.2. Aplicación en Criptografía

El objetivo principal de un algoritmo de cifrado es transformar cierta información de interés de tal manera que solo pueda ser recuperada por un usuario autorizado. Los sistemas criptográficos basados en el caos suelen constar de dos partes principales: la generación de secuencias pseudoaleatorias, que posteriormente se utilizan en una etapa de codificación [78]. Esta sección presenta una aplicación criptográfica para la metodología de sincronización propuesta para la generación de secuencias pseudoaleatorias para el cifrado de imágenes RGB.

4.2.1. Generador de Secuencias Pseudoaleatorias

Los sistemas caóticos a menudo exhiben atractores caóticos para conjuntos específicos de parámetros y condiciones iniciales. Para sistemas caóticos de orden fraccionario, el orden fraccionario $v(k)$ también afecta su trayectoria [79–81]. La Figura 4.11 muestra el diagrama de bifurcación para el estado $x(k)$ del mapa Tinkerbell fraccionario con $0.99 \leq v(k) \leq 1$ como variable de bifurcación. Las simulaciones se realizaron de acuerdo con las Ecuaciones (2.19), (2.22) y (4.13), durante un tiempo total de $N = 10000$ muestras, con longitud de memoria $L = 50$. El sistema presenta comportamiento caótico para órdenes $v(k) > 0.9966$.

Al simular el mapa fraccionario de Tinkerbell, se obtienen tres máscaras diferentes a partir del segundo estado del sistema, como se muestra en la Ecuación (4.31), para el cifrado de una imagen I de

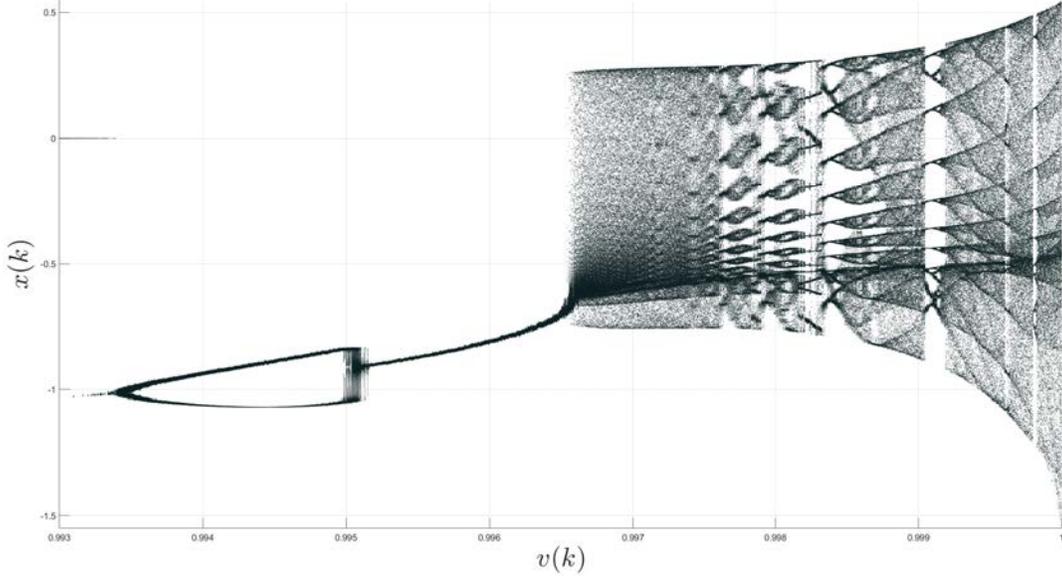


Figura 4.11: Diagrama de bifurcación para el mapa caótico fraccionario de Tinkerbell con respecto a cambios en el orden $v(k)$.

dimensiones $N \times M \times 3$.

$$\begin{aligned} M_1(i, j) &= \text{mod}_{256}(\text{floor}(1000y(p))), \\ M_2(i, j) &= M_1(j, i), \\ M_3(i, j) &= M_1(M - j, i), \end{aligned} \tag{4.31}$$

donde $p = Ni + j + \gamma$ y $\gamma = \sum_{k=0}^p y(k) \geq 768$.

Para evaluar si la secuencia obtenida es apropiada para la encriptación de datos se empleó el banco de pruebas del Instituto Nacional de Estándares y Tecnología (NIST por sus siglas en inglés), el cual evalúa una multitud de características que permiten definir si las secuencias obtenidas son lo suficientemente aleatorias para aplicaciones criptográficas [82]. Las pruebas pueden ser divididas en cuatro categorías: pruebas criptográficas, de patrones secuenciales, de comparación de plantillas y otras. Todos los análisis del banco de pruebas NIST toman como entrada un vector de valores binarios, por lo que la secuencia de datos obtenidos del vector $y(k)$ son convertidos de UINT8 a binario.

A continuación se describen de manera general algunas de las pruebas del banco de NIST. El análisis de complejidad lineal evalúa la complejidad de la secuencia de bits en base a la longitud máxima de un registro lineal de desplazamiento (LFSR) capaz de generar dicha secuencia. Un LFSR corto que permita caracterizar la secuencia implica que esta no es lo suficientemente aleatoria. Por su parte, las pruebas de patrones secuenciales permiten analizar si existe un patrón recurrente en la secuencia. Las pruebas de frecuencia determinan la proporción de ceros y unos que aparecen en la secuencia y lo comparan con el valor de probabilidad esperado (0.5). Las pruebas de rachas analizan la presencia de ocurrencias consecutivas del mismo símbolo en la secuencia. Las pruebas de comparación de plantillas se aplican

con la finalidad de detectar patrones específicos en la secuencia binaria, esto con la finalidad de detectar generadores de secuencias que producen mas instancias de un patrón específico.

Además de las pruebas antes mencionadas existen algunas otras que no encajan dentro de las categorías mencionadas. Por ejemplo, es posible examinar la secuencia mediante la transformada discreta de Fourier, en donde componentes frecuenciales sesgados aparecerían en una secuencia no aleatoria. Es posible también calcular de manera aproximada la entropía de la frecuencia. Es posible interpretar esta característica como la cuantificación de la impredecibilidad de la secuencia analizada. Las pruebas de excursiones aleatorias contienen conjuntos de pruebas que se encargan de analizar caminos aleatorios generados a partir de la suma acumulada de la secuencia binaria. La prueba de suma acumulada, como su nombre indica, analiza las desviaciones de la sumatoria de la secuencia. Por ultimo, la prueba de rango de la matriz binaria construye matrices utilizando subconjuntos de la secuencia original y verifica si existe dependencia lineal entre las filas o columnas de la matriz.

Tabla 4.1: Valores-P obtenidos para diversos análisis incluidos en el banco de pruebas de NIST.

Prueba	Valor-P	Prueba	Valor-P
Prueba de frecuencia	0.172611	Prueba de plantilla superpuesta	0.886588
Frecuencia en un bloque	0.454307	Prueba de complejidad lineal	0.124647
Prueba de rachas	0.091083	Entropía aproximada	0.535449
Rachas en bloque	0.121830	Sumas acumuladas	0.281498
Prueba de rango de matriz binaria	0.291891	Excursiones aleatorias	0.532271
Prueba de transformada de Fourier	0.880616	Variante de excursiones aleatorias	0.920344
Prueba de plantilla no superpuesta	0.887663		

La Tabla 4.1 muestra los valores-P resultantes de aplicar los análisis encontrados en el banco de pruebas de NIST a una secuencia pseudoaleatoria obtenida empleando la metodología propuesta. En cada uno de los casos la hipótesis nula propone que la secuencia probada es aleatoria, mientras que la hipótesis alternativa es que la secuencia no es aleatoria. Para todas las pruebas si el Valor-P obtenido es mayor que 0.01 se considera que se cumple la hipótesis nula y por lo tanto es posible aceptar la secuencia como aleatoria. Como se muestra en la tabla, la secuencia analizada cumple con el criterio aprobatorio en todos los casos, por lo tanto es lo suficientemente aleatoria para ser aplicada en un algoritmo de encriptación de datos.

4.2.2. Codificación Basada en ADN

Las secuencias de ADN están compuestas por cuatro bases: adenina (A), citosina (C), guanina (G) y timina (T). Se forman relaciones complementarias entre A y C, al igual que entre C y G. Basándose en esto, pueden representarse como los siguientes pares binarios: 00, 01, 10 y 11. Existen ocho configuraciones posibles para la correspondencia entre las bases y los pares binarios [83], que se muestran en la Tabla 4.2.

Siguiendo estas reglas, es posible definir operadores lógicos tradicionales. La Tabla 4.3 muestra la operación lógica XOR empleando la segunda regla de codificación mostrada en la Tabla 4.2.

Tabla 4.2: Configuraciones posibles de codificación basadas en ADN.

	1	2	3	4	5	6	7	8
00	A	A	C	C	G	G	T	T
01	C	G	A	T	A	T	C	G
10	G	C	T	A	T	A	G	C
11	T	T	G	G	C	C	A	A

Tabla 4.3: Tabla de verdad para el operador XOR empleando la segunda regla de codificación.

\oplus	A	G	C	T
A	A	G	C	T
G	G	A	T	C
C	C	T	A	G
T	T	C	G	A

Cada valor de intensidad de una imagen RGB $I(i, j, k) \in \mathbb{Z}, [0, 255]$ se transforma luego con el cifrado elegido de la Tabla 4.2 y las máscaras obtenidas en la Ecuación (4.31). La imagen obtenida después de una ronda de XOR se presenta en la Ecuación (4.32).

$$\begin{aligned}
 C_1(i, j, 1) &= I(i, j, 1) \oplus M_1(i, j), \\
 C_1(i, j, 2) &= I(i, j, 2) \oplus M_2(i, j), \\
 C_1(i, j, 3) &= I(i, j, 3) \oplus M_3(i, j).
 \end{aligned} \tag{4.32}$$

La imagen obtenida es recorrida espacialmente en dirección horizontal como se muestra en la Ecuación (4.33)

$$C_2(i, j, k) = C_1(i, \text{mod}_N(j + M_1(1, j, k)), k). \tag{4.33}$$

Finalmente, una segunda ronda de codificación XOR entrega la imagen encriptada, como se muestra en la Ecuación (4.34)

$$\begin{aligned}
 C_3(i, j, 1) &= C_2(i, j, 1) \oplus M_3(i, j), \\
 C_3(i, j, 2) &= C_2(i, j, 2) \oplus M_1(i, j), \\
 C_3(i, j, 3) &= C_2(i, j, 3) \oplus M_2(i, j).
 \end{aligned} \tag{4.34}$$

La Figura 4.12 muestra el esquema general de la metodología propuesta, donde la generación de la secuencia pseudoaleatoria para calcular los valores de las máscaras empleadas se realiza a partir de uno de los estados del sistema líder, en este caso y . Se generan tres máscaras basado en permutaciones del estado compartido, las cuales son transformadas mediante una de las reglas de codificación basadas en ADN mostradas en la Tabla 4.2. Los canales de color de la imagen son entonces encriptados en base a las máscaras obtenidas para eliminar tendencias en los niveles de intensidad de píxeles, se intercaló la aplicación de las mascarar con un desplazamiento espacial en dirección horizontal y vertical con la finalidad de eliminar correlaciones de ubicación en los píxeles de la imagen encriptada. Después de aplicar este proceso se obtienen como salida el estado del sistema líder x y la imagen encriptada.

Para el proceso de desencriptación se recupera la secuencia pseudoaleatoria a partir de la sincronización del sistema seguidor. Esto se logra mediante el diseño de la ley de control propuesta mostrada en la

Ecuación (4.6) y utilizando el estado compartido x , que en este esquema puede ser interpretado como la llave de encriptación utilizada para la metodología. Debido al uso de operadores XOR para aplicar las máscaras de encriptación, es posible recuperar la información original mediante la aplicación de las operaciones correspondientes en el orden inverso al utilizado para el proceso de encriptación. Por último las señales obtenidas son decodificadas aplicando la regla de codificación de ADN correspondiente.

4.2.3. Análisis de Seguridad

Tamaño de la Llave

Las condiciones iniciales, el orden y los coeficientes del sistema pueden utilizarse como claves secretas. Sin embargo, al considerar solo el orden de diferencia, que puede tomar valores entre $1 > v(k) > 0.9966$, y asumiendo un formato de punto flotante de doble precisión, el espacio total de claves para una imagen de $N \times N$ se representa como 2^{40N} .

Ataques Estadísticos

La resistencia a algunos ataques estadísticos comúnmente utilizados puede probarse mediante el análisis del histograma y la correlación espacial de la imagen cifrada. Las Figuras 4.13 - 4.20 muestran los resultados para el esquema de cifrado de imágenes RGB propuesto aplicado a diferentes imágenes obtenidas de la base de datos de imágenes USC-SIPI [84], que está compuesta por imágenes en color de 24 bits por píxel. Específicamente, las imágenes seleccionadas se extrajeron del subconjunto "Miscellaneous". La Subfigura (a) en cada instancia representa la imagen original. La Subfigura (b) representa la imagen cifrada y la Subfigura (c) muestra la imagen recuperada. Las Subfiguras (d-f) muestran los histogramas de cada canal para la imagen original, el equivalente para la imagen cifrada se puede ver en las Subfiguras (g-i). En este caso, es deseable que los valores de intensidad de la imagen cifrada presenten una distribución similar al ruido blanco. Idealmente, la imagen cifrada no debe transmitir información fácilmente accesible sobre el sesgo en la distribución de la imagen original. Las Subfiguras (j-l) presentan la correlación de píxeles adyacentes en la imagen original, mientras que las Subfiguras (m-o) muestran la misma información para la imagen cifrada. La correlación espacial se calcula en tres direcciones: horizontal, vertical y diagonal. Estas figuras se obtuvieron seleccionando aleatoriamente 5000 pares de píxeles adyacentes. Los puntos en las diagonales principales indican una fuerte correlación espacial en la dirección analizada, que la imagen original muestra, esta característica no está presente en la imagen cifrada.

Sensibilidad de la Llave

Considerando dos imágenes cifradas C_1 y C_2 , que representan dos imágenes cifradas para la misma imagen de entrada y con los mismos parámetros para el generador de secuencias pseudoaleatorias, cambiando la clave de cifrado en un bit. Los siguientes índices de rendimiento se pueden utilizar para evaluar la sensibilidad de la salida cifrada a la clave de cifrado [85]. El Índice de Cambio de Intensidad Promedio Unificado (UACI) cuantifica la intensidad promedio de los cambios entre dos imágenes y se describe mediante la Ecuación (4.35).

$$UACI = \frac{100}{M \times N} \left(\frac{|C_1(i, j) - C_2(i, j)|}{2^n - 1} \right). \quad (4.35)$$

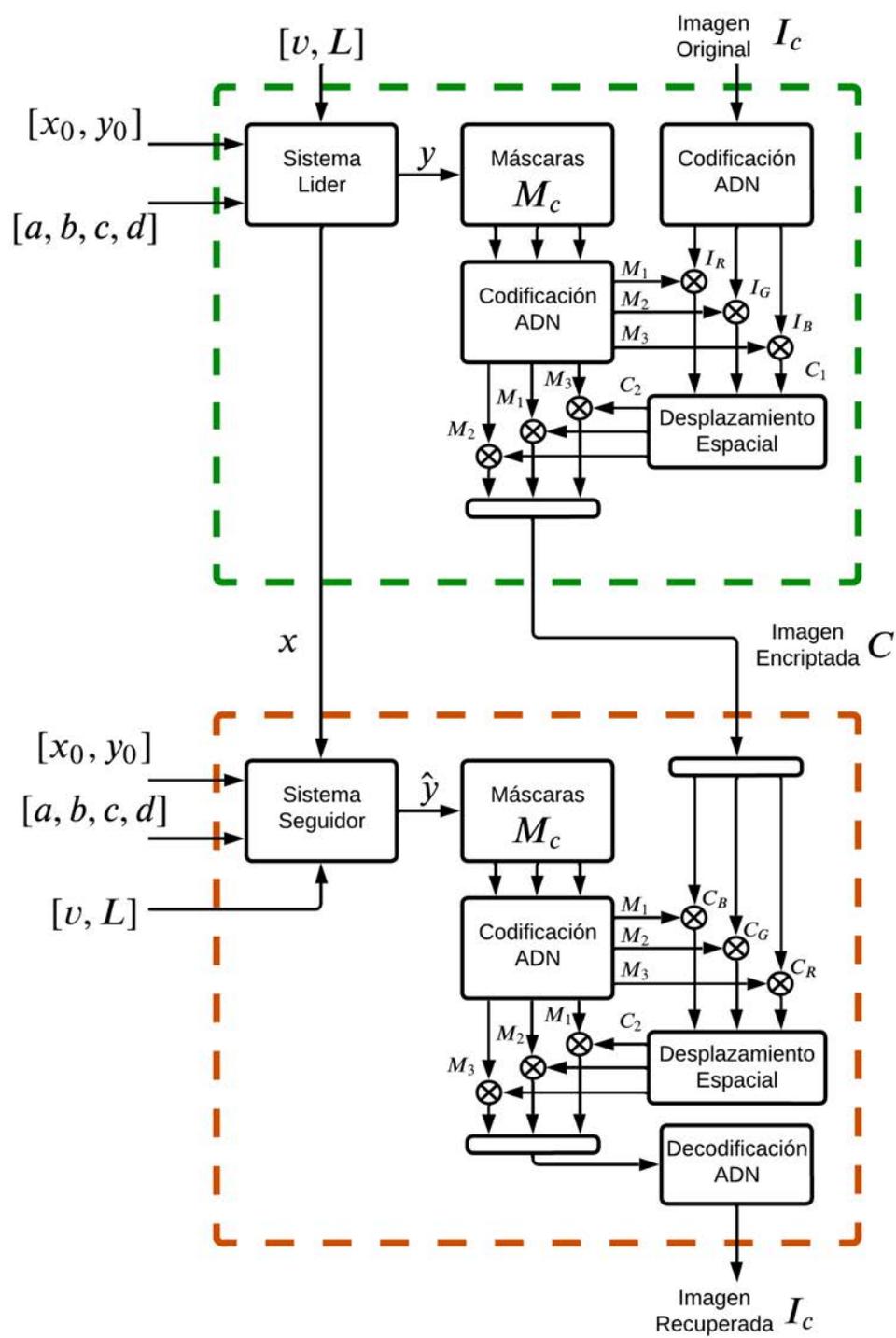


Figura 4.12: Esquema de la metodología propuesta para encriptación de imágenes RGB.

La Tasa de Cambio de Píxeles (NPCR) mostrada en las Ecuaciones (4.36) y (4.37) representa en porcentaje el número de píxeles con las mismas posiciones pero con valores de intensidad diferentes.

$$NPCR = \frac{100}{M \times N} \sum_{i,j} G(i, j), \quad (4.36)$$

$$G(i, j) = \begin{cases} 1, & C_1(i, j) = C_2(i, j), \\ 0, & C_1(i, j) \neq C_2(i, j). \end{cases} \quad (4.37)$$

La entropía, como se muestra en la Ecuación (4.38), está relacionada con la cantidad promedio de información o incertidumbre en un proceso [86]. En este caso, debido a los posibles valores de intensidad, su valor máximo y deseable es $H(s) = 8$.

$$H(s) = \sum_{i=1}^N P(s_i) \log_2 \frac{1}{P(s_i)}. \quad (4.38)$$

El Coeficiente de Correlación de Pearson (mostrado en la Ecuación (4.39)) se utiliza para medir la correlación lineal entre las imágenes originales y cifradas [87]. Idealmente, $\rho(I, C) \rightarrow 0$, lo que significa que la imagen original tiene una influencia débil en los valores de intensidad de la imagen cifrada.

$$\rho(I, C) = \frac{\sum_{j=1}^M \sum_{i=1}^N (I(i, j) - \bar{I}) (C(i, j) - \bar{C})}{\sqrt{\sum_{j=1}^M \sum_{i=1}^N (I(i, j) - \bar{I})^2 \sum_{j=1}^M \sum_{i=1}^N (C(i, j) - \bar{C})^2}}, \quad (4.39)$$

donde $(\bar{\cdot})$ indica el valor promedio de intensidad de la imagen.

La Tabla 4.4 presenta los índices previamente descritos al evaluar cada canal de color por separado de las imágenes de prueba seleccionadas. Los resultados obtenidos muestran que el esquema propuesto es resistente a las técnicas de criptoanálisis estadístico.

Pérdidas en Imagen Cifrada

Las Figuras 4.21 - 4.24 presenta los resultados obtenidos de la metodología propuesta considerando pérdidas de información entre el sistema de encriptación y el sistema de desencriptación. Para cada caso la Subfigura (a) presenta la imagen original, mientras que la Subfigura (b) muestra la imagen encriptada con la metodología propuesta. En la Subfigura (c) se tiene la imagen recuperada al desencriptar la imagen encriptada con una llave diferente en un solo bit a la llave utilizada en la etapa de encriptación. Las Subfiguras (d) y (f) consideran que le imagen cifrada se encuentra contaminada con ruido. Las Subfiguras (e) y (g) presentan las imágenes recuperadas a partir de las imágenes encriptadas con ruido. Las Figuras 4.21 y 4.22 consideran ruido de tipo Gaussiano de media 0 con varianza 0.1 para la Subfigura (d) y con varianza 0.5 para la Subfigura (f). Las Figuras 4.23 y 4.24 consideran ruido de tipo sal y pimienta con densidad de 0.1 para la Subfigura (d) y 0.5 para la Subfigura (g).

Comparativa

La Tabla 4.5 muestra una comparación de la metodología de cifrado propuesta con algunos esquemas similares encontrados en otros trabajos. El esquema propuesto está diseñado para cifrar imágenes RGB,

Tabla 4.4: Índices de desempeño ante ataques estadísticos para la metodología de encriptación propuesta.

Image	Channel	UACI	NPCR	Entropy	ρ
Airplane	Red	99.616241	33.584433	7.991030	-0.000450
	Green	99.625397	33.667570	7.990969	0.000011
	Blue	99.610901	33.740432	7.990352	-0.000598
Baboon	Red	99.602509	33.041723	7.991374	0.004225
	Green	99.599838	33.071741	7.990765	-0.001085
	Blue	99.603653	33.215900	7.991585	0.004105
House	Red	99.616623	33.267110	7.991347	0.005256
	Green	99.615860	33.338497	7.991425	0.003098
	Blue	99.607086	33.409630	7.991572	0.006336
Peppers	Red	99.607849	33.072016	7.991094	0.006253
	Green	99.623871	33.667241	7.991559	0.001916
	Blue	99.609375	33.516193	7.991404	-0.002171
Sailboat on Lake	Red	99.618149	32.864957	7.990370	0.003620
	Green	99.620819	33.696675	7.991221	0.000940
	Blue	99.603271	33.766327	7.990970	0.006050
Splash	Red	99.612808	33.680767	7.991025	-0.000067
	Green	99.601746	33.839895	7.990510	0.005145
	Blue	99.620056	33.602163	7.991293	0.001372
Tree	Red	99.598694	33.273004	7.989015	0.005808
	Green	99.612427	33.708065	7.987948	-0.000247
	Blue	99.604797	33.393938	7.989694	0.001535
White	Red	99.612427	34.241937	7.987071	-
	Green	99.600220	34.260371	7.986789	-
	Blue	99.600220	34.227480	7.987160	-

lo que, debido a los tamaños de entrada, conlleva un mayor costo computacional en comparación con algoritmos para imágenes en escala de grises. Sin embargo, al emplear la propiedad de la derivada $\Delta^{v(k)}$ mostrada en la Ecuación (2.18), las máscaras mostradas en la Ecuación (4.31) se calculan en complejidad lineal. La principal ventaja presentada por el algoritmo propuesto radica en el orden variable, que permite aumentar sustancialmente el tamaño de la clave de cifrado manteniendo la sensibilidad.

Tabla 4.5: Comparación de la metodología propuesta con trabajos similares reportados en la literatura.

	Proposed	[88]	[89]	[90]	[91]	[92]	[93]
Imagen a Color	✓	×	✓	×	×	✓	✓
Codificación ADN	✓	✓	×	✓	×	✓	×
Orden Variable	✓	×	×	×	✓	×	×

4.3. Conclusiones Parciales

En este trabajo se presenta un esquema de sincronización basado en control en una arquitectura líder-seguidor para mapas caóticos de orden fraccionario. El controlador propuesto se probó en tres mapas caóticos diferentes. Los resultados de simulación obtenidos muestran que la ley de control propuesta permite la sincronización idéntica de los sistemas presentados. Para demostrar la relativa baja complejidad algorítmica del esquema propuesto, se implementó en dos placas Arduino Uno, una para el sistema líder y otra para el sistema seguidor. La comunicación entre las placas se realizó a través del puerto serial y las variables relacionadas con el comportamiento del sistema no superaron el 50% de la memoria dinámica de la placa. Además, se presentó una posible aplicación del esquema propuesto en un sistema de cifrado de imágenes RGB, que aprovecha el comportamiento caótico del sistema para la generación de secuencias pseudoaleatorias. La metodología propuesta utiliza un esquema de codificación basado en la propiedad complementaria de las bases de la molécula de ADN. En este caso, el tamaño de la clave puede aumentarse considerablemente al considerar un orden de diferencia variable en el operador $\Delta^{v(k)}$, mejorando así la robustez de la metodología contra diversos tipos de ataques de fuerza bruta, manteniendo al mismo tiempo la resistencia contra otros tipos de técnicas de criptoanálisis.

4.3. CONCLUSIONES PARCIALES

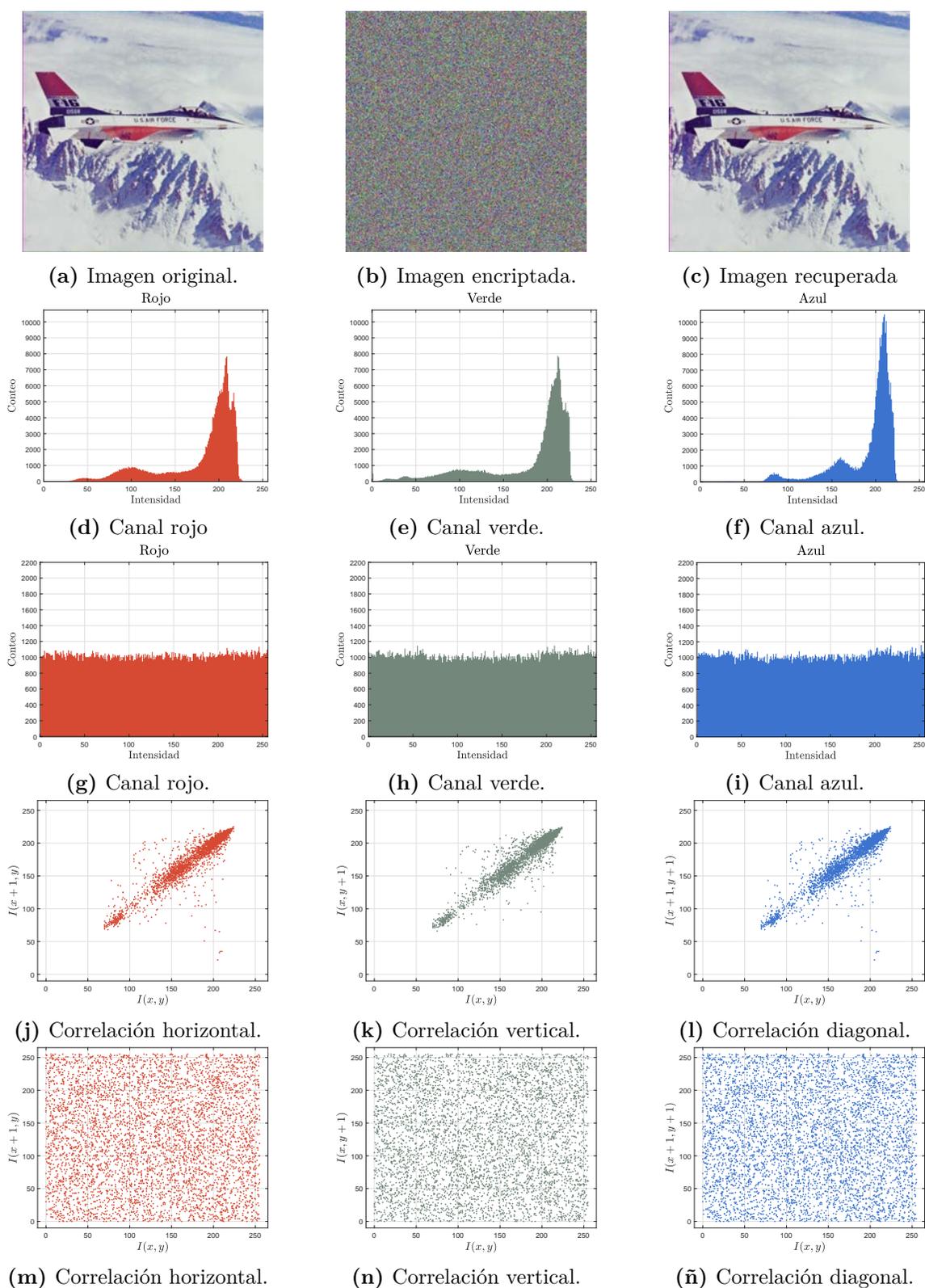


Figura 4.13: Resultados obtenidos aplicando la metodología propuesta en la imagen “Airplane”.

4.3. CONCLUSIONES PARCIALES

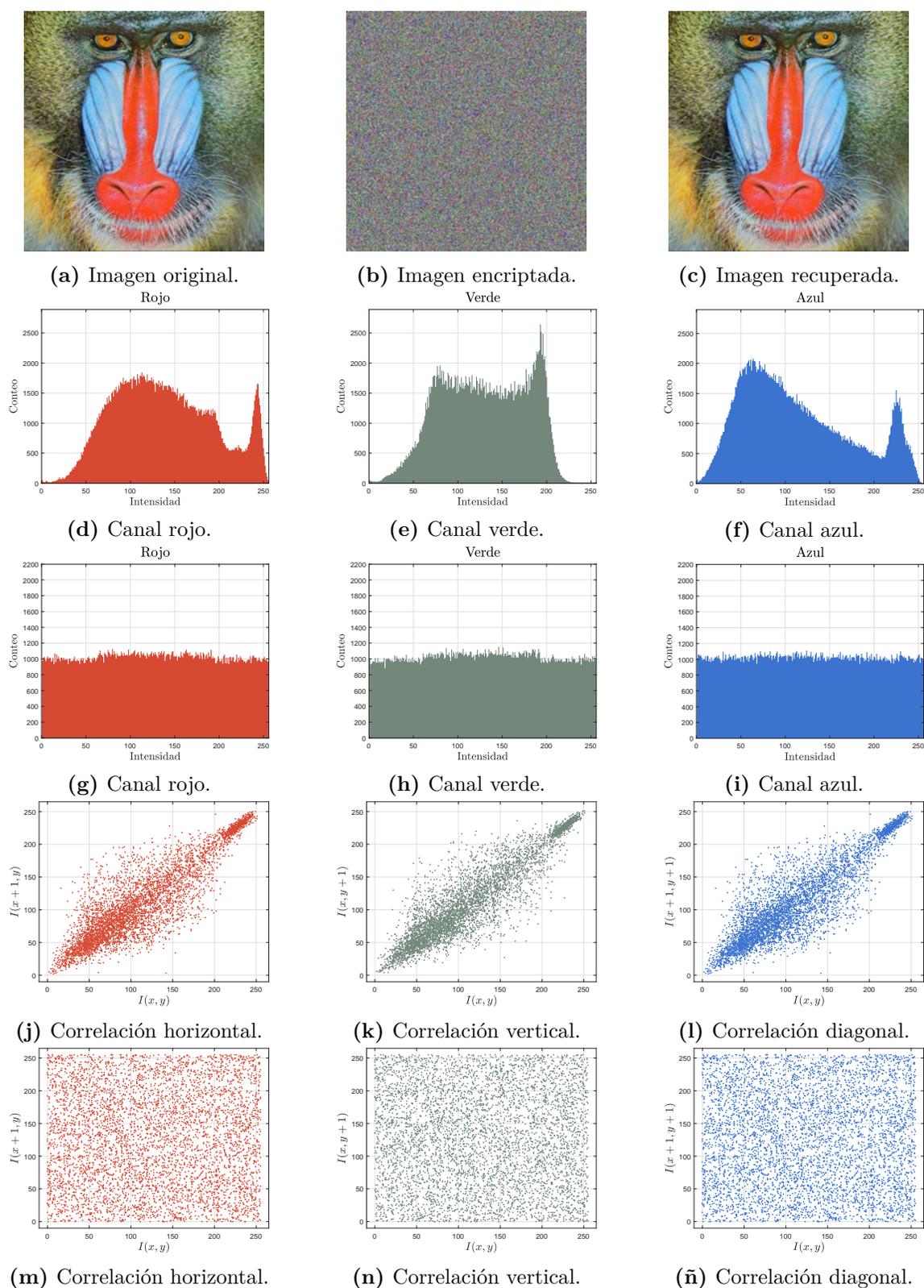


Figura 4.14: Resultados obtenidos aplicando la metodología de encriptación propuesta en la imagen “Baboon”.

4.3. CONCLUSIONES PARCIALES

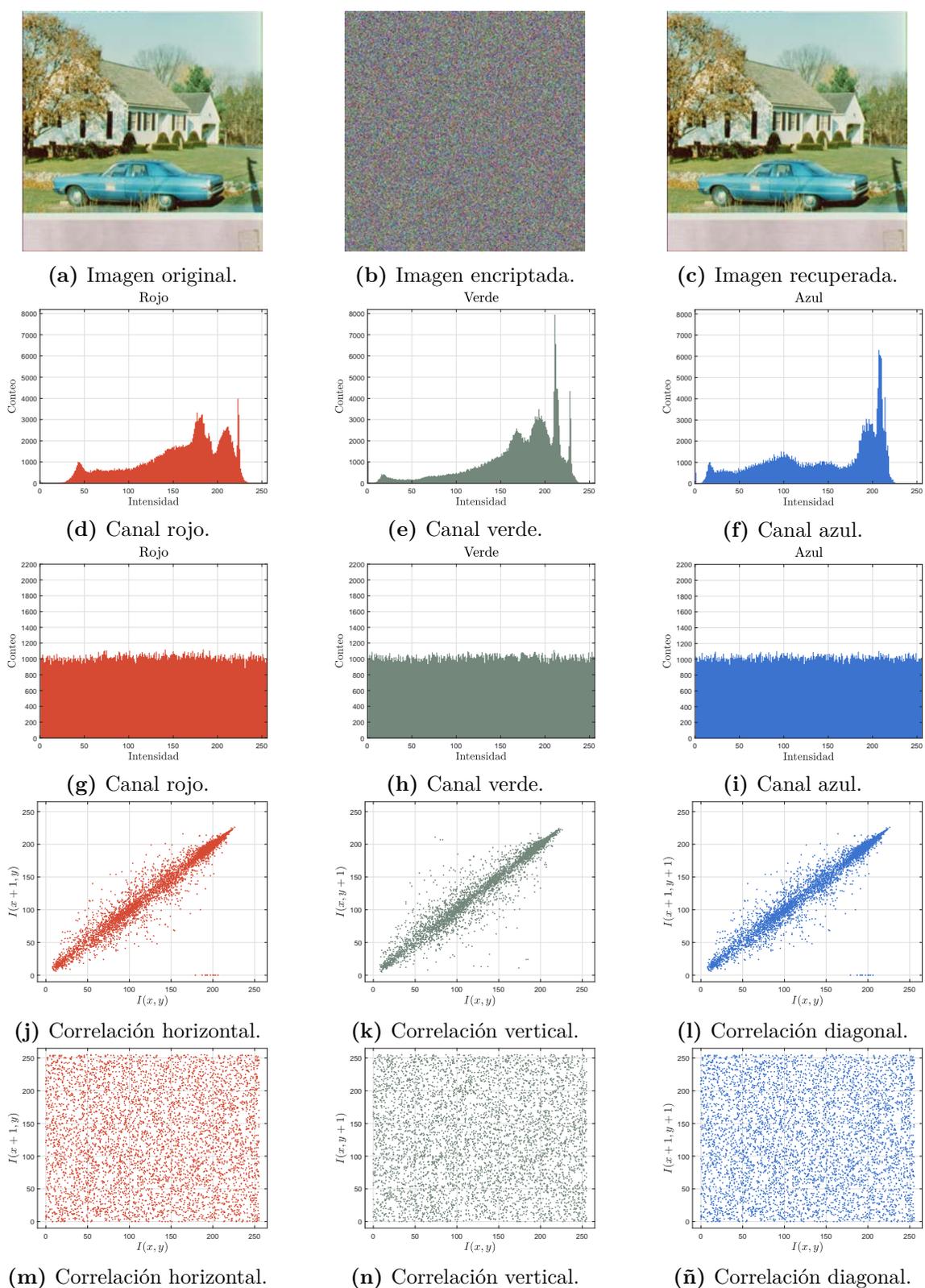


Figura 4.15: Resultados obtenidos aplicando la metodología propuesta en la imagen “House”.

4.3. CONCLUSIONES PARCIALES

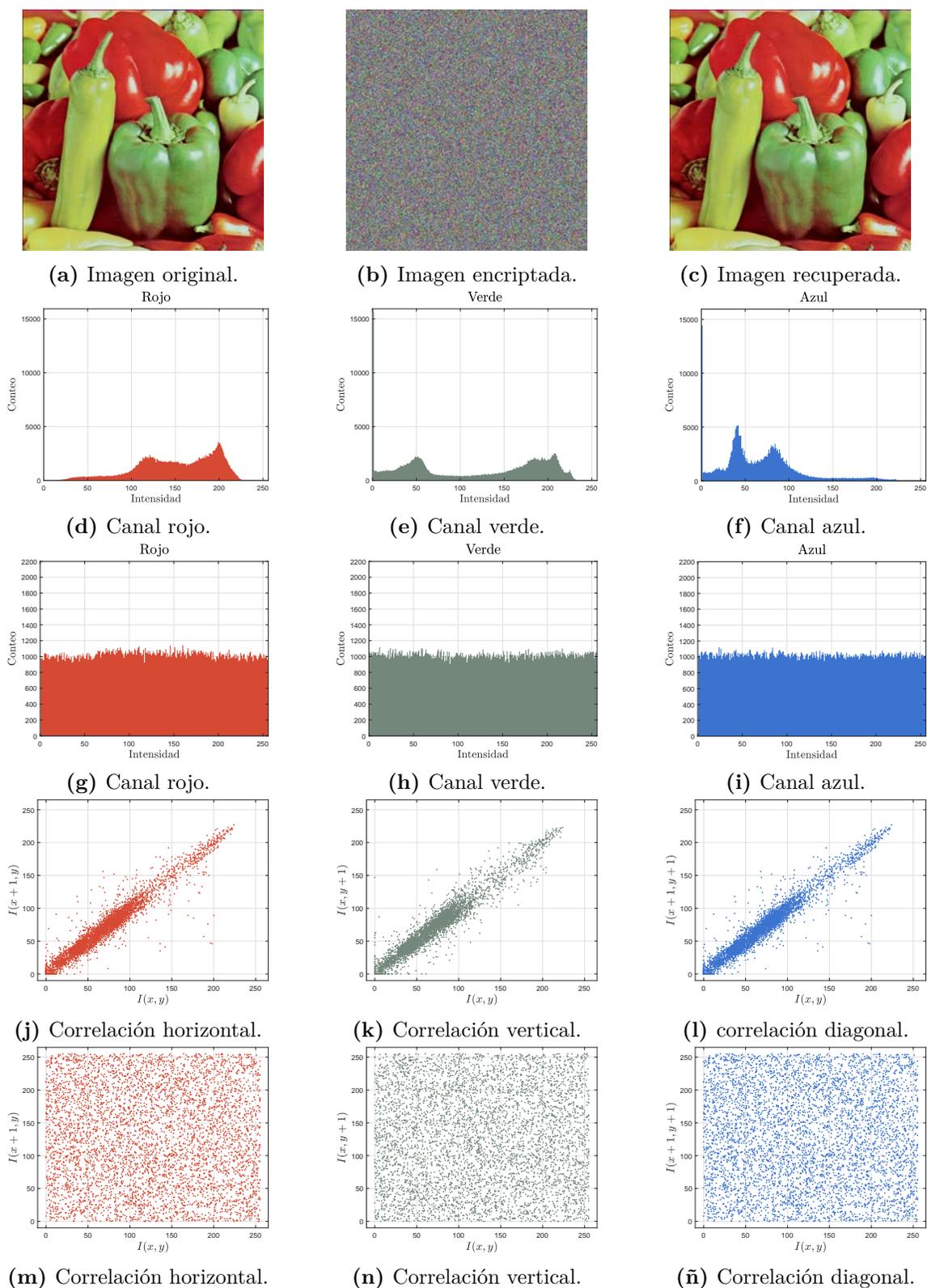


Figura 4.16: Resultados obtenidos aplicando la metodología propuesta en la imagen “Peppers”.

4.3. CONCLUSIONES PARCIALES

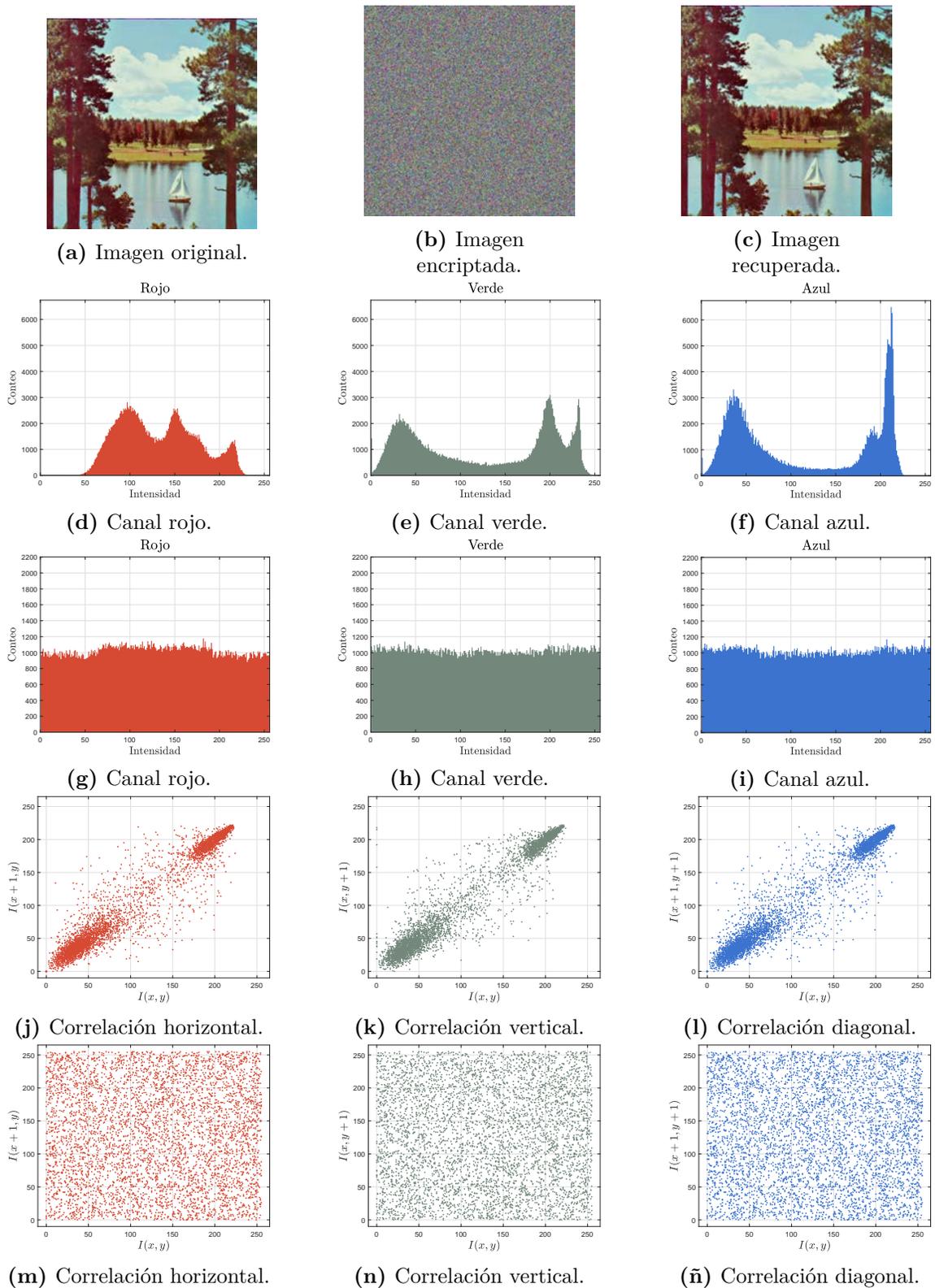


Figura 4.17: Resultados obtenidos aplicando la metodología propuesta en la imagen “Sailboat On Lake”.

4.3. CONCLUSIONES PARCIALES

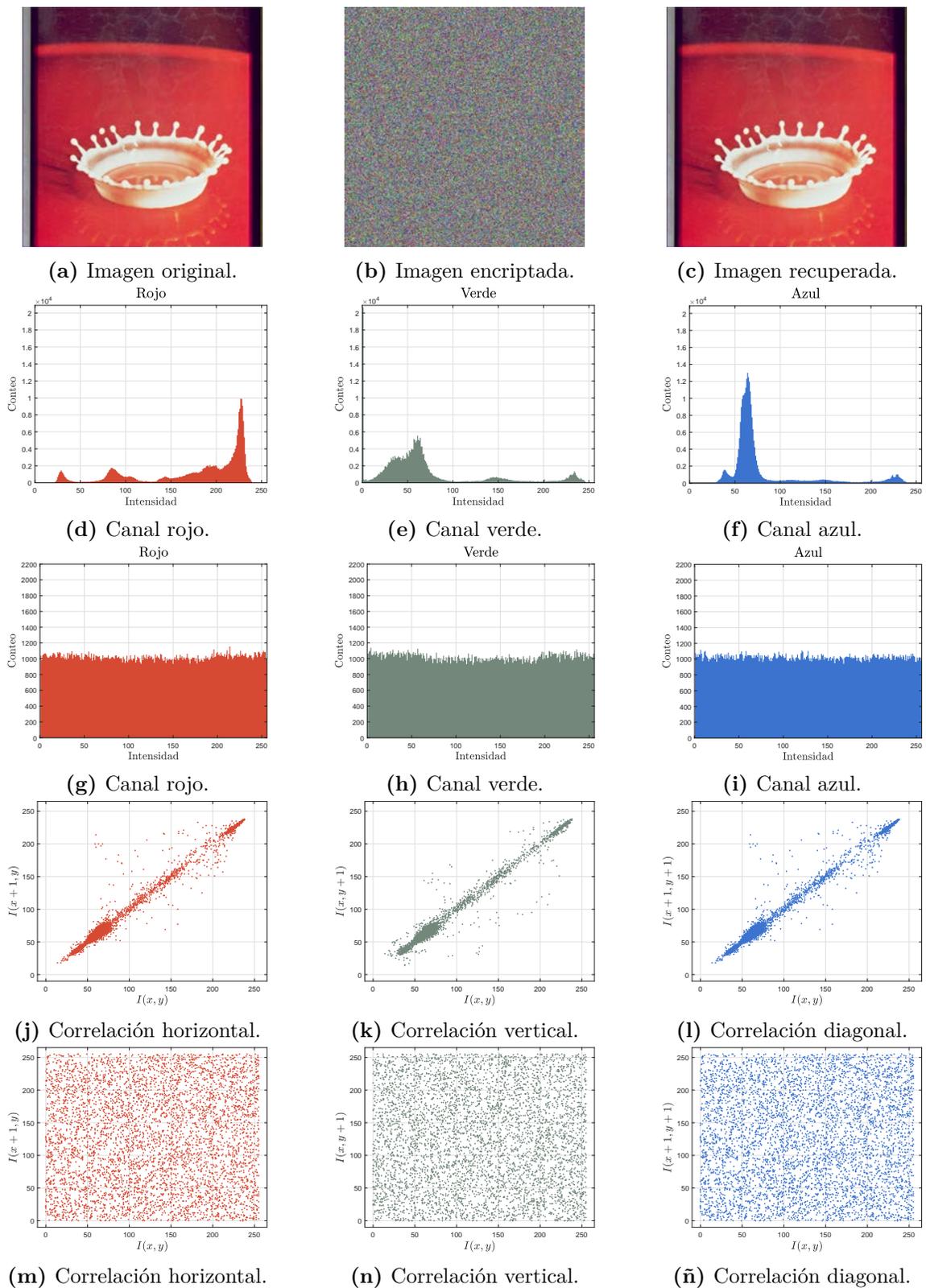


Figura 4.18: Resultados obtenidos aplicando la metodología propuesta en la imagen “Splash”.

4.3. CONCLUSIONES PARCIALES

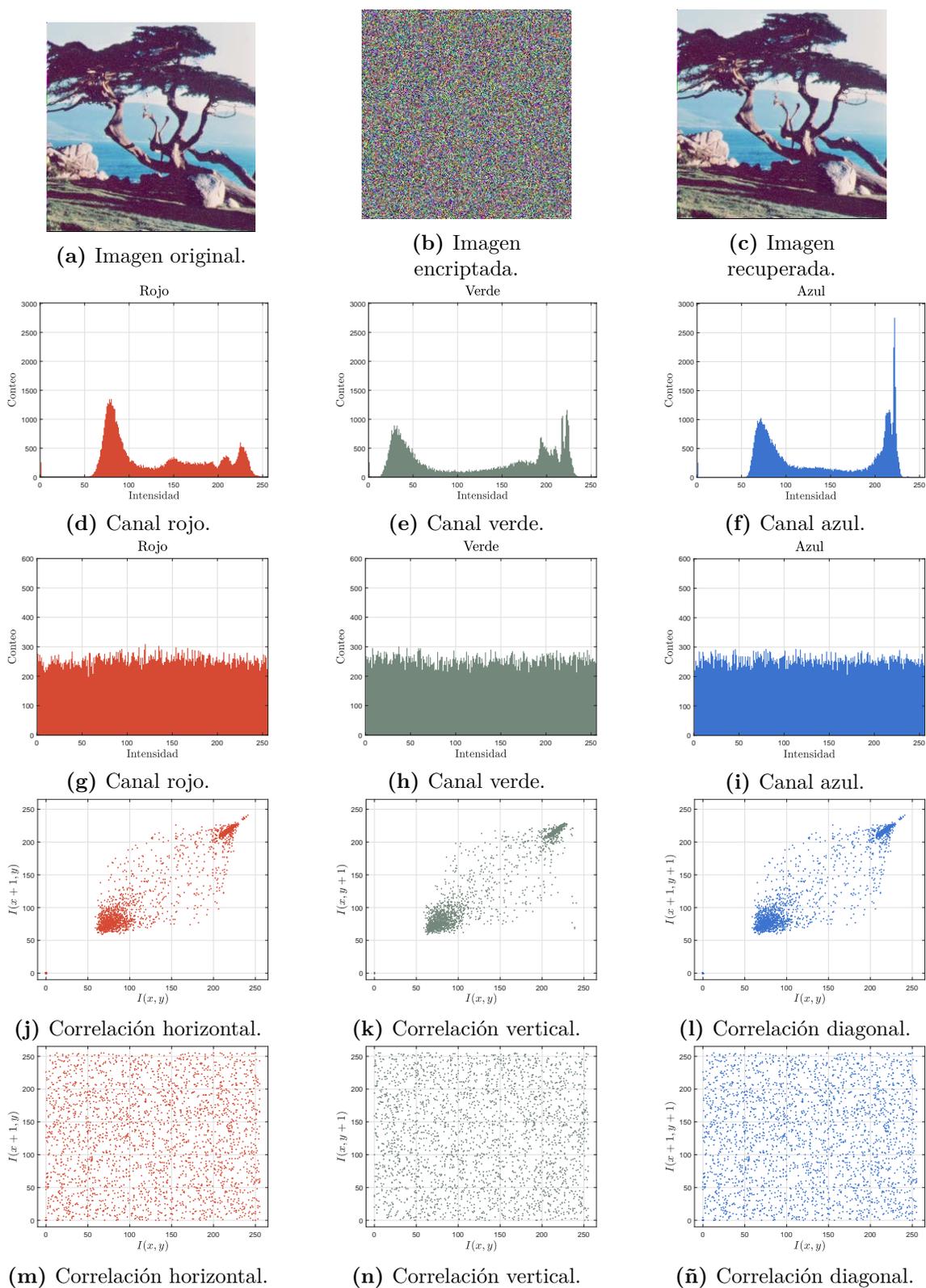


Figura 4.19: Resultados obtenidos aplicando la metodología de encriptación propuesta en la imagen “Tree”.

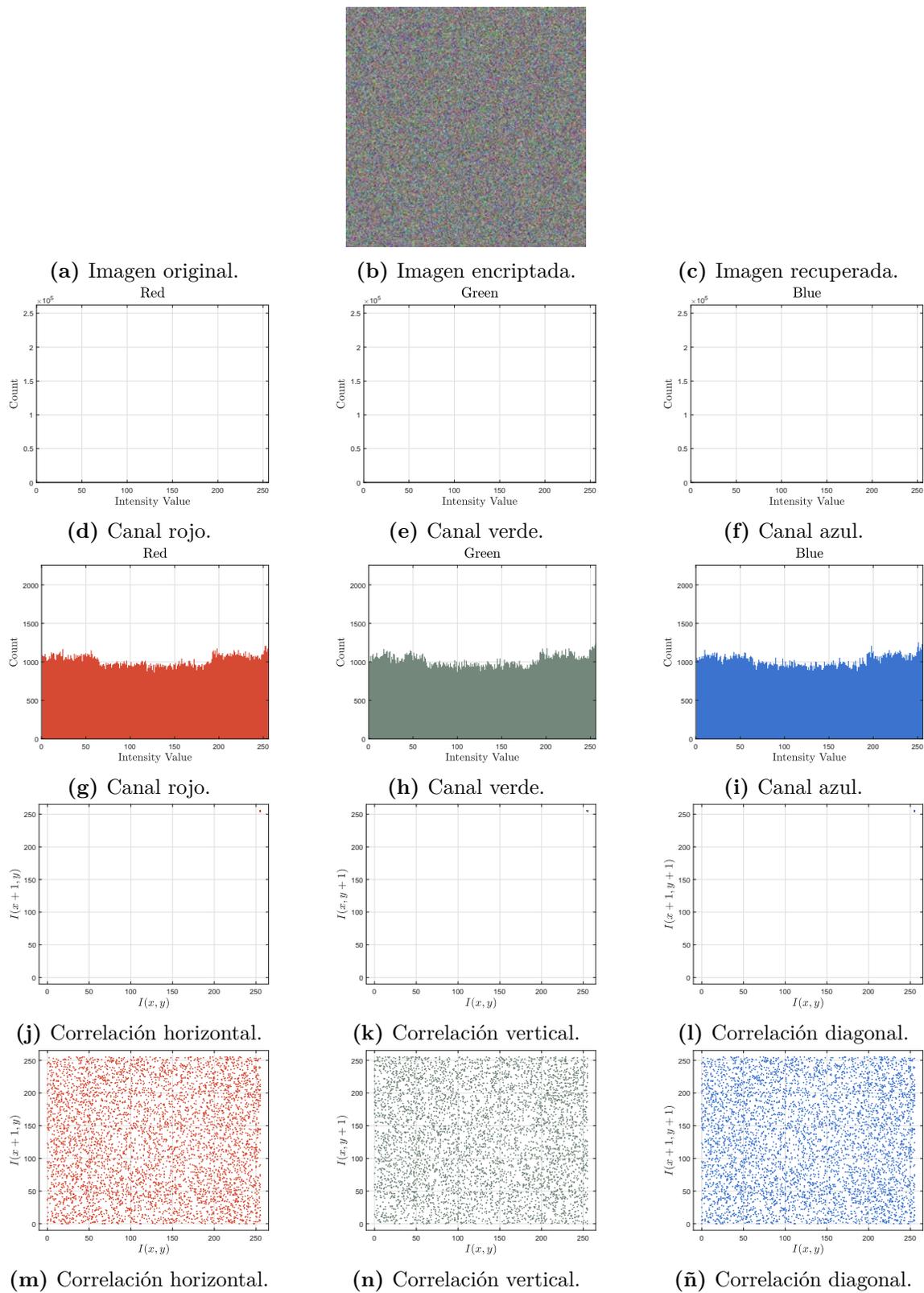
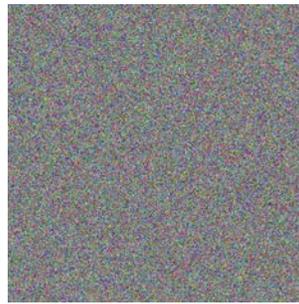


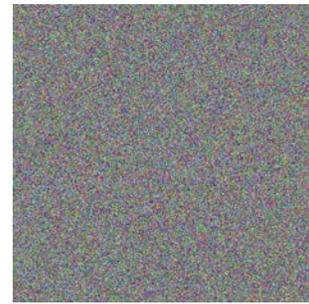
Figura 4.20: Resultados obtenidos aplicando la metodología propuesta en la imagen “White”.



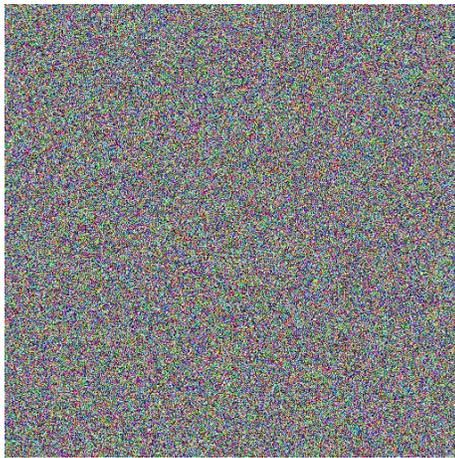
(a) Imagen original.



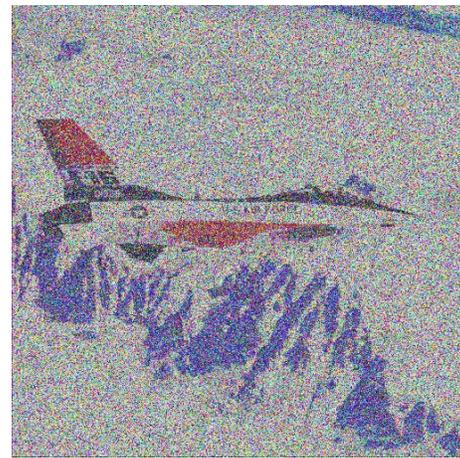
(b) Imagen encriptada.



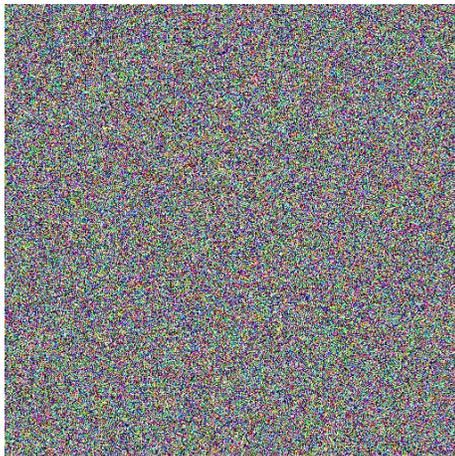
(c) Imagen recuperada.



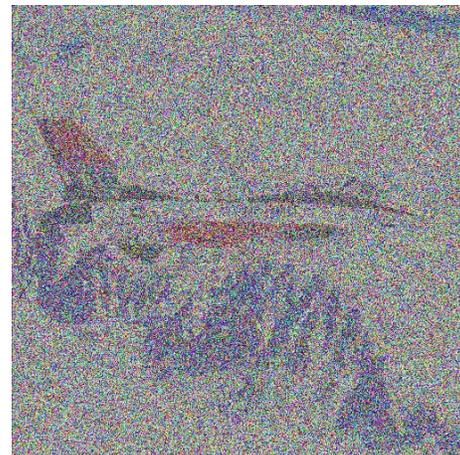
(d) Imagen cifrada con ruido Gaussiano. $Varianza = 0.1$.



(e) Imagen recuperada.

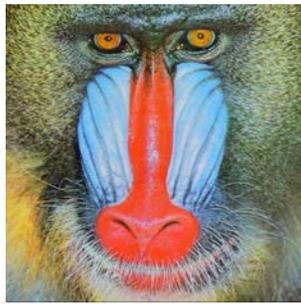


(f) Imagen cifrada con ruido Gaussiano. $Varianza = 0.5$.

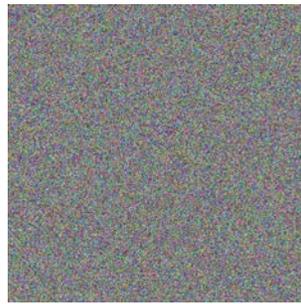


(g) Imagen recuperada.

Figura 4.21: Resultados ante ruido en el medio de transmisión para la imagen "Airplane".



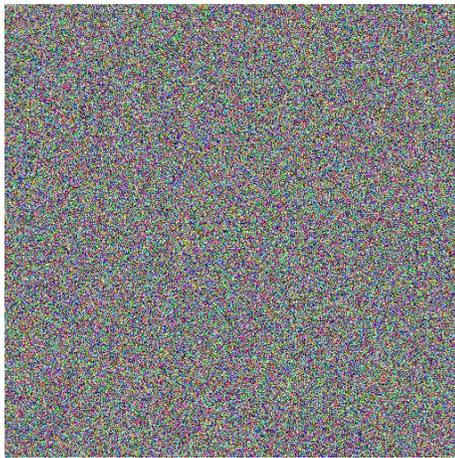
(a) Imagen original.



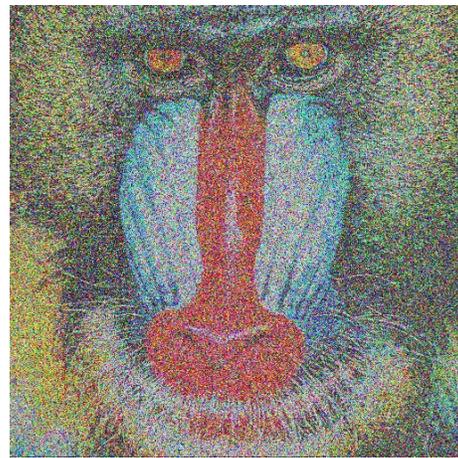
(b) Imagen encriptada.



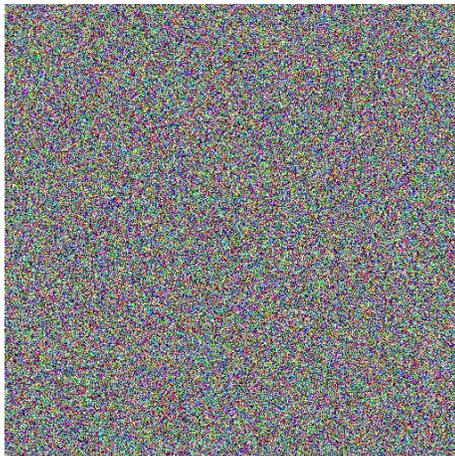
(c) Imagen recuperada.



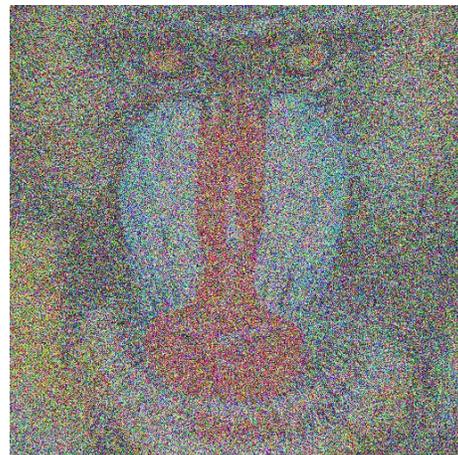
(d) Imagen cifrada con ruido Gaussiano. $Varianza = 0.1$.



(e) Imagen recuperada.

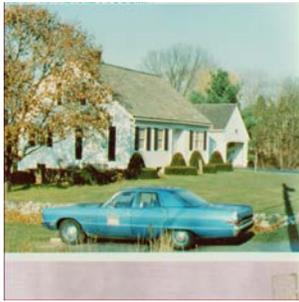


(f) Imagen cifrada con ruido gaussiano. $Varianza = 0.5$.

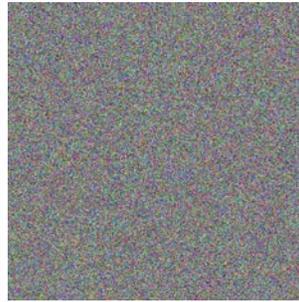


(g) Imagen recuperada.

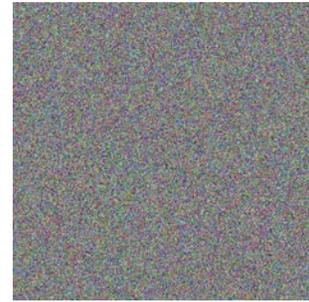
Figura 4.22: Resultados ante ruido en el medio de transmisión para la imagen “Baboon”.



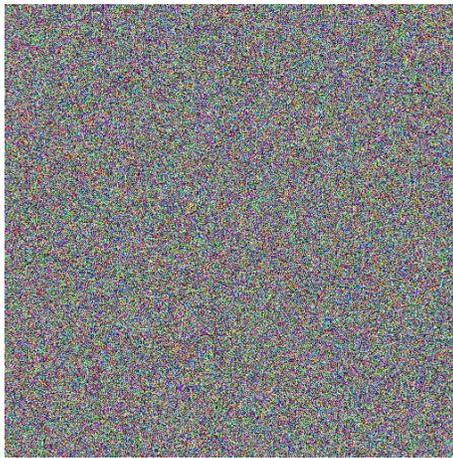
(a) Imagen original.



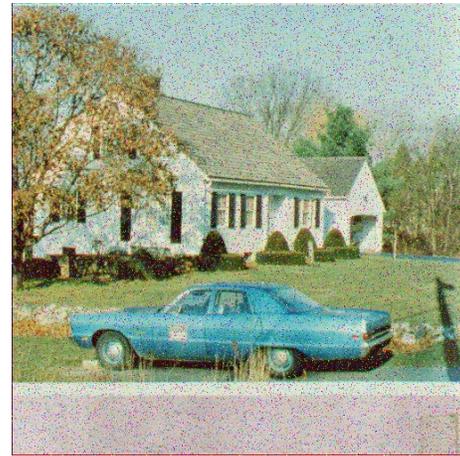
(b) Imagen encriptada.



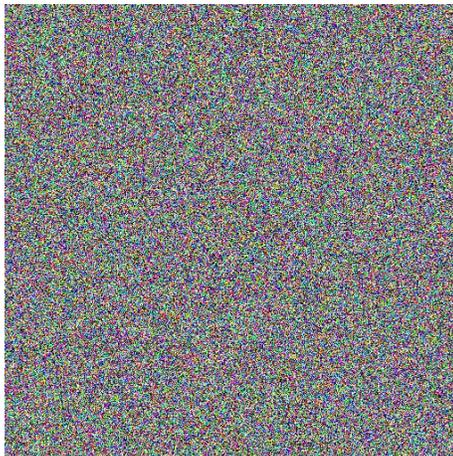
(c) Imagen recuperada.



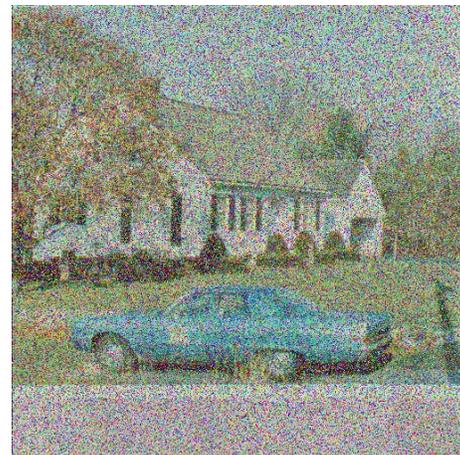
(d) Imagen cifrada con ruido sal y pimienta. $densidad = 0.1$.



(e) Imagen recuperada.



(f) Imagen cifrada con ruido sal y pimienta. $densidad = 0.5$.

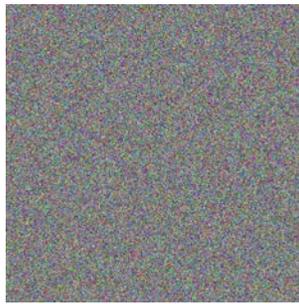


(g) Imagen recuperada.

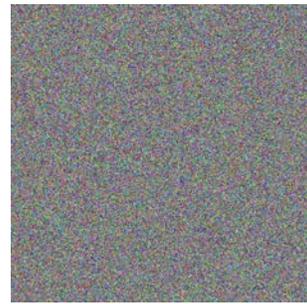
Figura 4.23: Resultados ante ruido en el medio de transmisión para la imagen “House”.



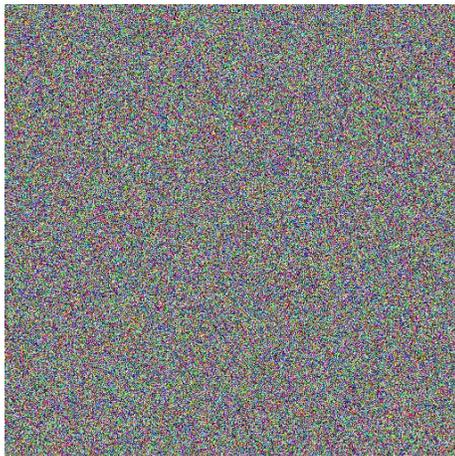
(a) Imagen original.



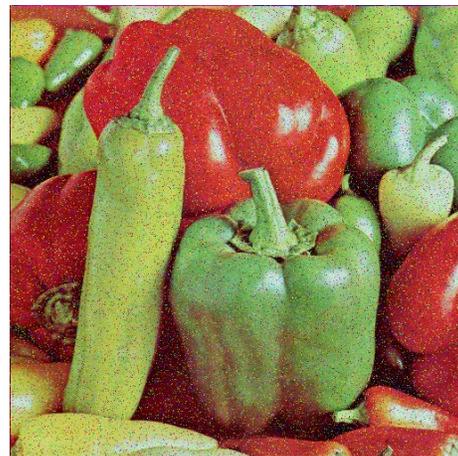
(b) Imagen encriptada.



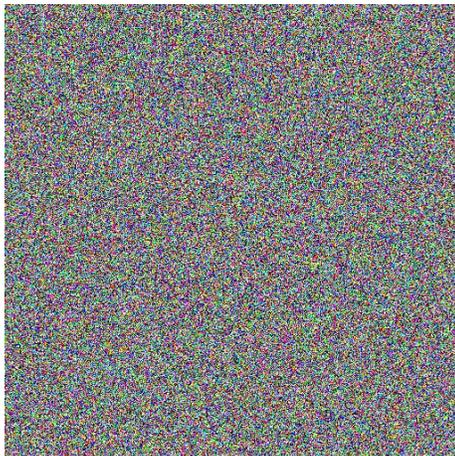
(c) Imagen recuperada.



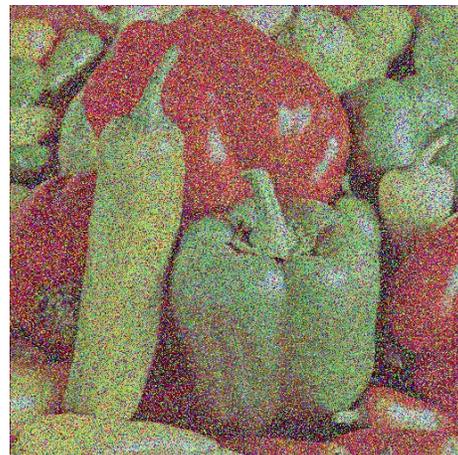
(d) Imagen cifrada con ruido sal y pimienta. $densidad = 0.1$.



(e) Imagen recuperada.



(f) Imagen cifrada con ruido sal y pimienta. $densidad = 0.1$.



(g) Imagen recuperada.

Figura 4.24: Resultados ante ruido en el medio de transmisión para la imagen “Peppers”.

Capítulo 5

Clasificación de Señales Basado en Cálculo Discreto de Orden Fraccionario

Como se menciona en el Capítulo 3, la clasificación de datos se refiere al proceso en el que se asigna de manera automática una etiqueta de membresía de grupo a instancias de datos individuales. En el aprendizaje automático esto se logra mediante el entrenamiento de un modelo de clasificación. De manera general, la clasificación automática puede dividirse en tres partes principales como se muestra en la Figura 5.1.

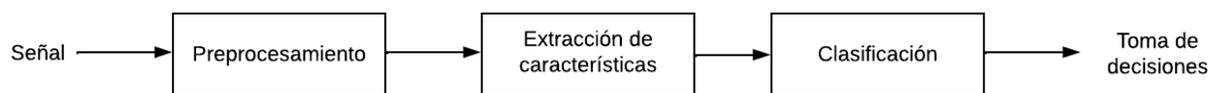


Figura 5.1: Fases de la clasificación basada en características.

El cálculo discreto de orden fraccionario es una herramienta útil en todas las fases de la metodología de clasificación. En el preprocesamiento, filtros de orden fraccionario permiten suprimir ruido sin descartar componentes que poseen información útil de la señal, en el caso específico de imágenes estas componentes suelen estar asociadas con texturas propias de la imagen. En la extracción de características, herramientas como la FRFT permiten un análisis de los datos en un dominio intermedio entre los dominios tiempo y frecuencia, permitiendo considerar características que poseen información de ambos dominios. En modelos de clasificación se han estudiado por ejemplo métodos de gradiente descendente de orden fraccionario para el entrenamiento de CNNs. Este trabajo se limitará a aplicaciones del cálculo discreto de orden generalizado en las etapas de preprocesamiento y extracción de características.

Los pasos exactos en cada etapa de la clasificación automática varían considerablemente dependiendo del tipo de señales a trabajar. En esta tesis se escogieron como caso de estudio los fonocardiogramas (señales acústicas provenientes del corazón) e imágenes satélites para proponer metodologías de preprocesamiento y extracción de características basadas en el cálculo discreto de orden fraccionario.

5.1. Caso 1. Fonocardiogramas

En México y en el mundo las enfermedades cardiovasculares se ubican en el primer lugar como causa de muerte. La Organización Mundial de la Salud (OMS) estima que para el año 2030 el número de decesos debido a infartos al miocardio y accidentes cerebro vasculares ascenderá a 23.6 millones [94]. México se encuentra dentro de los países con una tendencia creciente en enfermedades cardiovasculares, por lo tanto el estudio de métodos de diagnóstico en un tema de interés.

Existen diversas enfermedades cardíacas, siendo las más prevalentes el infarto de miocardio o ataque al corazón, caracterizado por el estrechamiento de los vasos sanguíneos que suministran sangre al corazón [95]. Otras afecciones comunes incluyen la angina de pecho, manifestada como dolor en el área del pecho debido a la disminución del flujo sanguíneo al corazón; las arritmias, que implican irregularidades en el ritmo cardíaco; y la insuficiencia cardíaca, que se manifiesta cuando el corazón no puede bombear la cantidad adecuada de sangre para cubrir las necesidades del organismo. La arterioesclerosis también es frecuente, donde se acumulan placas de grasa, colesterol, calcio y otras sustancias en las arterias, limitando el flujo de sangre rica en oxígeno. Esta condición puede desencadenar problemas graves, como la enfermedad de las arterias coronarias, encargadas de transportar sangre al corazón, y su obstrucción puede resultar en angina de pecho o un infarto cardíaco.

La examinación física del paciente es uno de los primeros pasos en el diagnóstico de cualquier enfermedad. Debido a su naturaleza, las señales acústicas provenientes del corazón contienen información mecánica del mismo, lo cual puede ser utilizado para el diagnóstico de diversas enfermedades como arritmias o insuficiencia cardíaca. La Figura 5.2 muestra el comportamiento esperado de un fonocardiograma obtenido de manera sintética mediante funciones de Gabor [96].

Un ciclo cardíaco completo se compone de las siguientes segmentos [97]:

- **S1:** El primer sonido del ciclo corresponde al cierre de la válvula mitral seguido de la apertura de la válvula aórtica. Este periodo es conocido como contracción isovolumétrica, en donde la presión dentro del ventrículo aumenta sin cambio en su volumen.
- **Sístole:** Durante el periodo de silencio entre los sonidos S1 y S2 sucede la eyección rápida de sangre del ventrículo.
- **S2:** El segundo sonido del ciclo cardíaco corresponde al cierre de la válvula aórtica, seguido de la apertura de la válvula mitral. A este periodo se le conoce como relajación isovolumétrica.
- **Diástole:** Durante el tiempo entre S2 y S1 el ventrículo se llena nuevamente de sangre para repetir el ciclo cardíaco.

Uno de los problemas principales encontrados en el análisis de fonocardiogramas consiste en encontrar los tiempos de inicio y fin de cada uno de los segmentos que comprenden el ciclo cardíaco. Una técnica simple consiste en monitorear la energía de la señal y emplear umbrales para la detección de audio, posteriormente se analiza la distancia entre ambos sonidos para determinar el momento actual del ciclo cardíaco. Sin embargo, diversos padecimientos cardíacos dan origen a alteraciones significativas del

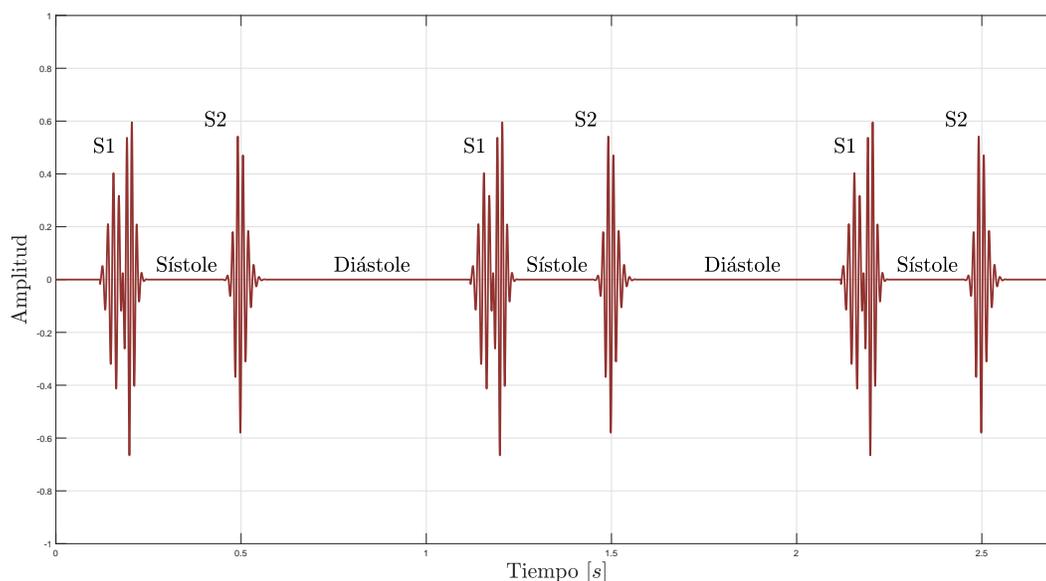


Figura 5.2: Fonocardiograma sintético obtenido mediante funciones de Gabor.

comportamiento esperado de la señal.

Comúnmente, los padecimientos cardiacos se presentan en la forma de murmullos o clics. Los murmullos cardiacos son sonidos adicionales generados por el flujo turbulento de sangre a través de las válvulas del corazón, las cámaras o los vasos sanguíneos y pueden tener diversas causas como el estrechamiento de las válvulas cardiacas o fuga en las mismas. Por su parte los clics son sonidos relacionados con la apertura o el cierre súbito de las válvulas cardiacas. Este tipo de sonidos puede indicar la presencia de válvulas cardiacas rígidas o calcificadas.

En esta sección se propone una metodología de clasificación automática como herramienta auxiliar para la detección de padecimientos cardiovasculares a partir de señales acústicas. La técnica propuesta emplea herramientas basadas en el cálculo discreto de orden generalizado para las etapas de preprocesamiento y extracción de características con la finalidad de mejorar la eficacia en la detección de enfermedades cardiacas.

5.1.1. Base de Datos

La base de datos empleada corresponde al reto *Classification of Heart Sound Recordings: The Physio-Net/Computing in Cardiology Challenge 2016* [98], la cual se compone de un total de nueve subconjuntos de datos recolectados de manera independiente por siete equipos de investigación distintos, en tres continentes diferentes a lo largo de mas de una década. Como consecuencia los equipos para extraer y almacenar los sonidos, los lugares donde se obtuvieron, los tipos de paciente y la calidad de los datos obtenidos varía sustancialmente dentro de la misma base de datos.

La información obtenida proviene tanto de pacientes adultos como niños. Existen diversas áreas en el cuerpo humano de los cuales es posible registrar un fonocardiograma, de manera general se consideraron el área aórtica, el área pulmonar, el área tricúspide y el área mitral. Dentro de la base de datos existen señales tomadas en ambientes clínicos así como señales provenientes de ambientes no controlados tales como consultas médicas a domicilio. La población enferma disponible presenta en su mayoría defectos en las válvulas del corazón, algunos de los padecimientos encontrados mas comunes incluyen: prolapso mitral, insuficiencia mitral y la estenosis aórtica.

Los datos públicos para el entrenamiento del reto consisten en cinco subconjuntos de datos con un total de 3126 señales de audio con duraciones distintas, con valores desde los 5 hasta 120 segundos. Todas las señales se encuentran remuestreadas a una frecuencia de $2,000Hz$. Se proporcionan dos etiquetas binarias para el entrenamiento de modelos de clasificación: la primera indica si la señal proviene de un paciente sano o con un padecimiento cardiovascular, la segunda etiqueta considera la cantidad de ruido en la señal y determina si el fonocardiograma es útil para algoritmos de clasificación. Es importante resaltar que la base de datos no se encuentra balanceada es decir, existe mayor cantidad de señales con padecimientos cardiacos que de pacientes sanos.

Es relevante destacar que la diversidad de contextos en los que se recopilieron las señales contribuye a la complejidad y la heterogeneidad de la base de datos. La variabilidad en los entornos clínicos y no controlados, así las variables en los equipos de captura de señales y en los pacientes, agrega un elemento desafiante a la tarea de clasificación. Estas consideraciones adicionales resaltan la importancia de implementar estrategias que permitan su procesamiento y análisis.

5.1.2. Preprocesamiento

Debido a la gran cantidad de variables en el proceso de recopilación de las señales se realizaron diversas consideraciones. Primero se obtuvo un subconjunto de datos descartando señales en base a la etiqueta binaria que describe la calidad de la señal. Las señales restantes fueron truncadas a la misma duración total con la finalidad de evitar sesgos en los datos de entrenamiento al tener mas datos de un paciente específico. Posteriormente se obtuvieron ventanas de la señal como se muestra en la Ecuación (5.1) donde λ_i representa la ventana i de la señal $x(k)$ para $i \in \mathbb{N}$

$$\lambda_i = x(\psi), \quad \forall \psi \in [ai, ai + b], \quad (5.1)$$

donde $a = 1000$ y $b = 4000$, es decir se obtuvieron ventanas de la señal con duración de 2 segundos y solapamiento de 0.5 segundos. Esta longitud de ventana fue escogida en base al tiempo promedio de duración de el ciclo cardiaco en reposo.

Con la finalidad de eliminar sesgos ocasionados por la variabilidad de los equipos de medición se restó la media de cada ventana a sí misma y se normalizaron los valores de amplitud en el rango $[0, 1]$ como se muestra en la Ecuación (5.2) donde el operador $(\bar{\cdot})$ denota la media aritmética

$$\lambda n_i = \frac{\xi - \min(\xi)}{\max(\xi)}, \quad \xi = \lambda_i - \bar{\lambda}_i. \quad (5.2)$$

Después de aplicar estas consideraciones se obtuvieron 4560 señales, 2466 provenientes de pacientes sanos y 2094 de pacientes que presentaban algún padecimiento cardiaco. Todas los segmentos se encuentran normalizados en amplitud en un rango de $[0, 1]$ y acotadas en tiempo a 2 segundos de duración,

con un solapamiento de ventana de 0.5 segundos.

Debido a la naturaleza de las señales acústicas así como de los ambientes en los cuales fueron recopiladas las señales, estas se encuentran contaminadas con ruido de diversas fuentes; desde el movimiento del estetoscopio, sonidos de los aparatos respiratorio y digestivo del paciente, e incluso voces. Se propone para el suavizado de la señal un algoritmo basado en una red neuronal de orden fraccionario [99].

Considerando que la señal a procesar se encuentra descrita por una Ecuación dinámica no lineal discreta de orden generalizado como se muestra en la Ecuación (5.3)

$$\begin{aligned}\Delta^{v(k)}x(k) &= f(x(k)), \\ y(k) &= x(k).\end{aligned}\tag{5.3}$$

el sistema secundario propuesto se describe en la Ecuación (5.4)

$$\begin{aligned}\Delta^{v(k)}\hat{x}(k) &= A\hat{x}(k) + \zeta_1(k)\varphi(\hat{x}(k)) + \zeta_2(k) + Le(k), \\ \hat{y}(k) &= \hat{x}(k).\end{aligned}\tag{5.4}$$

En donde ζ_i son funciones auxiliares y L la ganancia, que serán utilizados para el diseño del observador.

La Ecuación (5.5) define el error entre los sistemas presentados anteriormente

$$e(k) = y(k) - \hat{y}(k).\tag{5.5}$$

Derivando la Ecuación (5.5) se obtiene la dinámica del error presentada en la Ecuación (5.6)

$$\Delta^{v(k)}e(k) = \Delta^{v(k)}y(k) - \Delta^{v(k)}\hat{y}(k),\tag{5.6}$$

sustituyendo las salidas de los sistemas (5.3) y (5.4) en la dinámica del error se obtiene la Ecuación (5.7)

$$\Delta^{v(k)}e(k) = f(x) - A\hat{x}(k) - \zeta_1(k)\varphi(\hat{x}(k)) - \zeta_2(k) - Le(k).\tag{5.7}$$

Para garantizar la convergencia del sistema descrito en la Ecuación (5.4) a la trayectoria descrita por la señal presentada en la Ecuación (5.3) se emplea el análisis de una función de Lyapunov. Como se menciona en la Sección 4 existen tres condiciones para verificar la estabilidad de Lyapunov en un sistema:

$$\begin{aligned}V(x) &= 0 \quad \Leftrightarrow \quad x = 0, \\ V(x) &> 0, \quad \forall \quad x \in \mathbb{R}, \\ \dot{V}(x) &< 0, \quad \forall \quad x \in \mathbb{R}.\end{aligned}\tag{5.8}$$

Para la selección de las funciones auxiliares de la Ecuación (5.4) se analizó la siguiente función candidata de Lyapunov

$$V(e, \zeta_1) = \frac{1}{2}e^2 + \frac{1}{2}\zeta_1^2.\tag{5.9}$$

Asumiendo $\zeta_1(k) \neq e(k)$ entonces $V(0, 0) = 0$ y $V(e, \zeta_1) \in \mathbb{R}^+$ es decir, la Ecuación (5.9) cumple con las primeras dos condiciones de estabilidad de Lyapunov. Con la finalidad de evaluar la condición restante se deriva la Ecuación (5.9) como se muestra en la Ecuación (5.10)

$$\Delta V(e, \zeta_1) = e\Delta e + \zeta_1\Delta\zeta_1,\tag{5.10}$$

sustituyendo la Ecuación (5.7) y agrupando sus términos se obtiene la Ecuación (5.11)

$$\Delta V(e, \zeta_1) = e(\lambda - \zeta_2) + e\zeta_1(\Delta\zeta_1 - \varphi(\hat{x})) - Le^2, \quad (5.11)$$

donde $\lambda(k) = f(k) - A\hat{x}(k)$. Se remarca que término $-Le^2$ es definido negativo es decir, $-Le^2 \in \mathbb{R}^-$; $\forall L \in \mathbb{R}^+$. Para el primer término de la función se asume que existe una constante $\sigma > \lambda$ que puede ser seleccionada a partir de conocimiento previo de la señal a procesar. Se selecciona la función auxiliar ζ_2 como se muestra en la Ecuación (5.12)

$$\zeta_2 = \text{sign}(e)\sigma. \quad (5.12)$$

Para el término faltante es posible obtener la condición $e\zeta_1(\Delta\zeta_1 - \varphi(\hat{x})) = 0$ a partir de la cual es posible seleccionar la función auxiliar ζ_1 resolviendo la Ecuación (5.13)

$$\Delta\zeta_1 = \varphi(\hat{x}). \quad (5.13)$$

Con base en estas consideraciones para la selección de los parámetros de diseño se garantiza la estabilidad de Lyapunov para el sistema descrito en la Ecuación (5.4). La Figura 5.3 presenta un fonocardiograma suavizado con la metodología propuesta empleando un orden $\nu(k) = 1(k) \cdot 0.92$ escogido de manera arbitraria. Como se ha mencionado anteriormente, el efecto de memoria en las definiciones de cálculo fraccionario se ve reflejado en el comportamiento de los sistemas descritos como pérdidas en la dinámica del sistema, lo que puede estar asociado a fenómenos de subdifusión. En este caso la dinámica reducida, consecuencia del orden fraccionario permite suavizar la señal de interés.

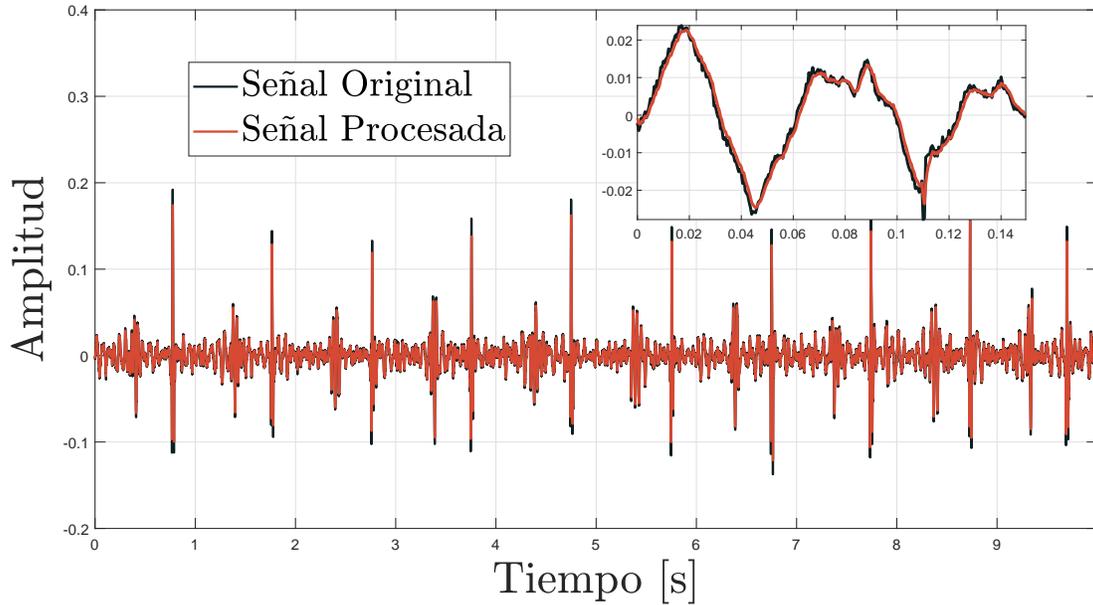


Figura 5.3: Señal de fonocardiograma suavizada con metodología propuesta.

La etapa de preprocesamiento concluye al aplicar la metodología de suavizado de señales propuesta a las ventanas obtenidas de los fonocardiogramas analizados. Estos sonidos pueden originarse debido al funcionamiento correcto del corazón: o a diversos padecimientos cardiaco

5.1.3. Extracción de Características

Si bien los sonidos S1 y S2 son los más reconocibles en el ciclo cardíaco existen una multitud de sonidos provenientes de la naturaleza mecánica del corazón y pueden o no indicar la presencia de una patología. Por ejemplo el sonido S3 es ocasionado por la interacción entre el flujo sanguíneo y la pared ventricular durante el llenado ventricular. Por su parte el sonido S4 se encuentra relacionado con la resistencia presentada durante la contracción atrial. Las componentes de S1, S2, S3 y S4 se superponen entre ellas en el dominio de la frecuencia, por lo que para el problema de clasificación es necesario considerar características distintas que permitan discernir entre ellos de manera adecuada.

Se propone en este trabajo el uso de características basadas en la transformada fraccionaria de Fourier. Como se menciona en el Capítulo 3, la FRFT permite transformar la señal a una representación alternativa en un dominio entre el tiempo y la frecuencia. Al aplicar la FRFT a las señales cardíacas se obtiene una representación que resalta las características tiempo-frecuencia en función del orden de transformación.

Para la primera etapa correspondiente a la extracción de características se calculó la FRFT para cada una de las ventanas obtenidas en la etapa de preprocesamiento. El cálculo numérico de la FRFT se realizó mediante el cálculo de la matriz de transformación basado en la descomposición de eigenvectores, la cual se basa en la discretización de la Ecuación (5.14)

$$K_\alpha(x, u) = \sum_{n=0}^{\infty} H_n(x)H_n(u), \quad (5.14)$$

donde α denota el ángulo de transformación asociado con la FRFT y $H_n(\cdot)$ representa la función de Hermite normalizada de orden n presentada en la Ecuación (5.15)

$$H_n(x) = \frac{1}{2^n n! \sigma \sqrt{\pi}} h_n\left(\frac{x}{\sigma}\right) e^{-(x^2/2\sigma^2)}, \quad (5.15)$$

donde $h_n(\cdot)$ se refiere al polinomio de Hermite de orden n . Con base a esto la transformada fraccionaria de Fourier puede ser calculada como se muestra en la función (5.16)

$$\mathcal{F}_\alpha \{f(x)\} (u) = \int_{-\infty}^{\infty} f(x)K_\alpha(x, u)dx = \sum_{n=0}^{\infty} H_n(u)e^{-in\alpha} \left(\int_{-\infty}^{\infty} f(x)H_n(x)dx \right). \quad (5.16)$$

La discretización de la FRFT mediante la Ecuación (5.16) puede ser interpretada como la sumatoria ponderada de funciones de Hermite. Como se muestra en la Tabla 3.1, esta metodología de discretización posee una complejidad algorítmica, sin embargo presenta diversas ventajas como su reversibilidad, aditividad y similitud. Además, una vez calculada la matriz de transformación para un orden y longitud, esta puede ser almacenada para aplicaciones futuras. Esto quiere decir que una vez seleccionado el orden de transformación deseado y calculada la matriz de transformación, la complejidad algorítmica de aplicar la FRFT se reduce a la multiplicación de la matriz de transformación por el vector a transformar. La Ecuación (5.17) expresa entonces la transformación mediante la FRFT de las ventanas obtenidas, donde $F_{i,c}$ se refiere a la muestra c de la transformada fraccionaria de Fourier de la ventana i .

$$F_{i,c} = FRFT_\alpha(\lambda_i), \quad (5.17)$$

La Ecuación (5.18) presenta la matriz de características obtenidas, expresada como $\Upsilon(i, c)$. En donde se restó la media de cada característica a las señales obtenidas.

$$\Upsilon(i, c) = F_{i,c} - \frac{1}{N} \sum_{i=0}^N F_{i,c}. \quad (5.18)$$

Se destaca que normalmente en metodologías de clasificación de fonocardiogramas, la segmentación de los mismos en la etapa de preprocesamiento o extracción de características suele ser un paso importante para garantizar resultados aceptables. En la metodología propuesta no se emplea la segmentación de fonocardiogramas en función de sus segmentos, si no en ventanas temporales que pueden comprender cualquier periodo del ciclo cardiaco.

5.1.4. Clasificación

La selección del algoritmo de clasificación para un problema particular suele ser una tarea empírica en mucho casos. En este caso se consideraron modelos con estructura de árbol de decisiones, K-vecinos más cercanos y maquinas de vectores de soporte. La evaluación del desempeño de los modelos obtenidos fue efectuada por medio de los índices de desempeños presentados en las Ecuaciones (5.19)-(5.22).

En una clasificación binaria los resultados obtenidos por el modelo de clasificación pueden ser ordenados en una matriz M de dimensiones 2×2 conocida como matriz de confusión, en donde las filas representan la clase real de la señal a clasificar y las columnas la predicción realizada por el modelo de clasificación. Las predicciones correctas se ubican en la diagonal principal de la matriz, mientras que $M(1, 2)$ representa falsos positivos en la predicción y $M(2, 1)$ constituye los falsos negativos. En base a lo anterior la exactitud se define en la Ecuación (5.19), donde N se refiere en este caso al número total de ventanas obtenidas.

$$Acc(M) = \frac{tr(M)}{N}. \quad (5.19)$$

El valor de predicción positivo (5.20), también conocido como precisión, relaciona los falsos positivos de predicción con los verdaderos positivos.

$$P(M) = \frac{M(2, 2)}{M(2, 2) + M(1, 2)}. \quad (5.20)$$

Por su parte la exhaustividad relaciona los positivos con los falsos negativos.

$$Ex(M) = \frac{M(2, 2)}{M(2, 2) + M(2, 1)}. \quad (5.21)$$

El valor-F relaciona la precisión y la exhaustividad como se muestra en la Ecuación (5.22). Este índice permite una evaluación de modelo mas precisa ante bases de datos con distribución de clase sesgada.

$$F_1(M) = 2 \cdot \frac{P(M) \cdot Ex(M)}{P(M) + Ex(M)}. \quad (5.22)$$

Para evitar sobre entrenamiento con respecto a subconjunto de datos de entrenamiento se empleó un esquema de validación cruzada de cinco secciones, en donde el conjunto de datos de entrenamiento se

divide en cinco partes iguales, manteniendo la proporción de distribución de los datos en cada subconjunto. Posteriormente el modelo es entrenado con cuatro de los subconjuntos, empleando el restante para la validación de los resultados. Este proceso es iterado hasta que todas las secciones se utilicen para la etapa de validación.

La Tabla 5.1 muestra los resultados obtenidos de los modelos considerados en un entrenamiento preliminar, para esto los ordenes α y ν fueron seleccionados de manera arbitraria como $\nu = 0.92$ y $\alpha = 1$. En base a los resultados obtenidos se descartaron los modelos de estructura KNN y SVM. Los mejores resultados en todas las métricas se presentaron el modelo de árbol de decisiones.

Tabla 5.1: Comparación de índices de desempeño para modelos comunes de clasificación empleando metodología propuesta de preprocesamiento con extracción de características en el dominio de Fourier Fraccionario.

Clasificador	Acc	P	Ex	F_1
Árbol de Decisiones	93.8377 %	94.5807 %	91.9771 %	93.2607 %
KNN	74.1885 %	92.4930 %	47.6600 %	62.9057 %
SVM	83.9254 %	86.9235 %	76.5043 %	81.3818 %

La metodología propuesta para el preprocesamiento y extracción de características posee dos órdenes generalizados: ν corresponde al orden de derivada fraccionaria de el algoritmo de suavizado de señales presentado en la Ecuación (5.4) utilizado en la etapa de preprocesamiento, mientras que α se refiere al orden de transformación de la FRFT empleada para obtener las características de las señales. Los valores de estos ordenes afectan drásticamente el desempeño del clasificador a entrenar, por lo que su selección óptima resulta de principal interés para este trabajo.

La optimización matemática se refiere a la selección de los mejores elementos de un conjunto de acuerdo a algún criterio preestablecido. En su caso mas simple la tarea consiste en maximizar o minimizar una función. Los algoritmos metaheurísticos de optimización son aquellos diseñados para encontrar o seleccionar una solución lo suficientemente buena para un problema de optimización. Estos algoritmos son útiles para lidiar con problemas en los cuales el posible numero de combinaciones de soluciones es muy grande para ser evaluado de manera individual o si la información del problema es limitada. A diferencia de los algoritmos iterativos, los algoritmos metaheurísticos no garantizan encontrar la mejor solución global en algunos problemas, sin embargo su costo computacional puede ser menor.

La optimización por enjambre de partículas (PSO) se refiere a un algoritmo de optimización metaheurístico basado en una analogía del movimiento de un enjambre [100]. Este algoritmo funciona teniendo una población de soluciones candidatas, llamadas partículas, las cuales se mueven dentro del espacio de búsqueda. El movimiento de las partículas esta guiado por la mejor solución de la iteración anterior así como por la mejor solución de la partícula.

Cada partícula es un punto en un espacio de dimensión N y se representa como $X_I = (x_{i1}, x_{i2}, \dots, x_{iN})$, las posiciones se representan con el vector $P_I = (p_{i1}, p_{i2}, \dots, p_{iN})$ con esta información, las partículas

5.1. CASO 1. FONOCARDIOGRÁMAS

actualizan sus posiciones como se muestra en las ecuaciones (5.23) y (5.24)

$$v_{in}^{k+1} = wv_{in}^k + c_1r(p_{in}^k - x_{in}^k) + c_2r(p_{gn}^k - x_{in}^k), \quad (5.23)$$

donde c_1 y c_2 son constantes que usualmente toman el valor 1.494, mientras que r representa un numero aleatorio entre 0 y 1 y k representa la iteración actual. El término wv_{in}^k se interpreta como la inercia de cada partícula. $c_1r(p_{in}^k - x_{in}^k)$ se refiere a la "memoria" de cada partícula, ya que depende del mejor resultado histórico de la misma. De manera similar $c_2r(p_{gn}^k - x_{in}^k)$ describe la "memoria" del enjambre. La actualización de la posición de cada partícula se describe en la Ecuación (5.24)

$$x_{in}^{k+1} = x_{in}^k + v_{in}^{k+1}, \quad (5.24)$$

en donde c_1 y c_2 son constantes que toman el valor 1.494, mientras que r representa un numero aleatorio entre 0 y 1 y k representa la iteración actual.



Figura 5.4: Esquema general del algoritmo PSO.

Empleando un modelo de clasificación de tipo árbol de decisiones y empleando las metodologías de pre-procesamiento y extracción de características propuestas se utilizó el algoritmo PSO para la selección de los valores de v y α . El esquema general de la composición de cada partícula del problema se presenta en la Figure 5.5. Cada partícula se compone del proceso completo de entrenamiento del modelo de clasificación empleando las metodologías propuestas y entrega como resultado el valor-F obtenido por el modelo.

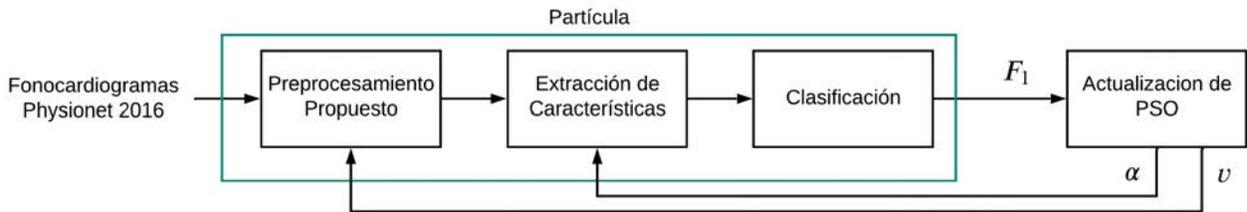


Figura 5.5: Esquema general de optimización utilizado para la selección de ordenes v y α para las etapas de preprocesamiento y extracción de características respectivamente.

La función costo a minimizar se muestra en la Ecuación (5.25), donde M representa la matriz de confusión obtenida por el modelo de clasificación de la partícula.

$$J(M) = 1 - F1(M). \quad (5.25)$$

5.1. CASO 1. FONOCARDIOGRÁMAS

Aplicando el algoritmo PSO para la selección del orden de diferencia fraccionario y el orden de transformación correspondiente a la FRFT con las partículas descritas se seleccionaron los valores $[\alpha, v] = [0.9573524, 0.944362]$

La Figura 5.6 muestra la comparación entre las medias de clase de la metodología propuesta para la extracción de características contra una metodología similar de orden entero, reportada en [102]. En la etapa de preprocesamiento de la metodología de orden entero se optó por utilizar un filtro pasa banda con frecuencias de corte $f_{c1} = 25Hz$ y $f_{c2} = 400Hz$. Las características mostradas en ambos casos se encuentran normalizadas en el rango $[0, 1]$. Visualmente, las características obtenidas con la metodología propuesta son similares a las obtenidas con la metodología de orden entero, sin embargo existe una mayor separación de las clases a partir de la muestra 80. En el caso entero, la información de las frecuencias superiores a los $4000Hz$ se encuentra truncada debido al filtro pasa banda utilizado en la etapa de preprocesamiento. En el caso de la metodología propuesta, el algoritmo de suavizado de señales empleado permite mantener información útil de las señales que se encuentra en la banda de frecuencia descartada por la contraparte de orden entero; Esto resulta aparente a partir de la muestra 1400, donde la media de las clases se distancia.

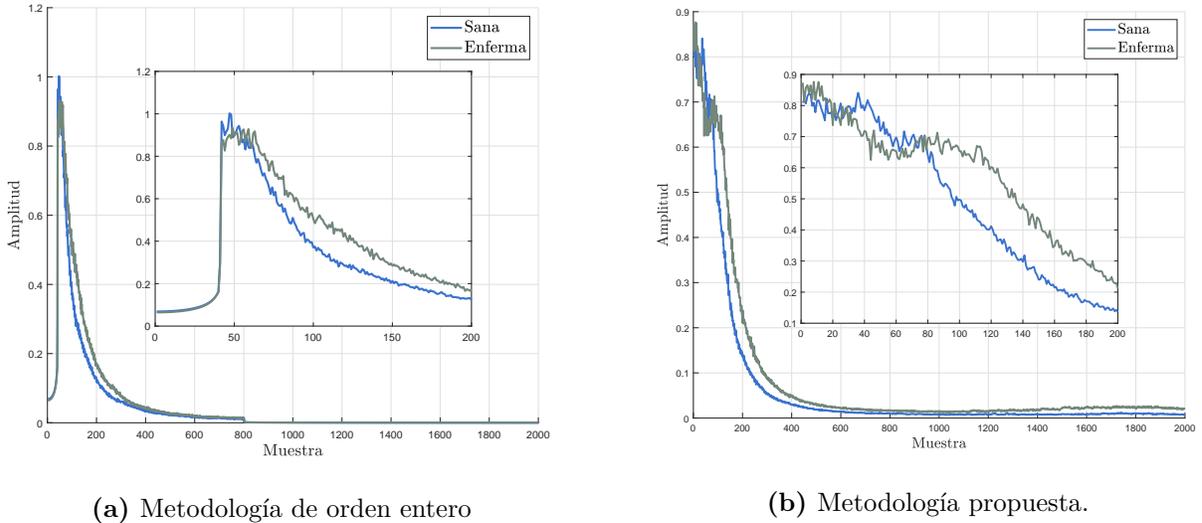


Figura 5.6: Comparación de medias por clase entre metodología de orden entero y propuesta.

La Ecuación (5.26) presenta la sumatoria del absoluto de la diferencia entre los promedios de clase obtenidos en ambas metodologías, donde $\bar{\Upsilon}_s$ se refiere al vector de promedio de las características para la clase sana y $\bar{\Upsilon}_e$ denota lo mismo para la clase de pacientes enfermos.

$$B(\Upsilon) = |\bar{\Upsilon}_s - \bar{\Upsilon}_e|. \quad (5.26)$$

Para las características de la metodología de orden entero mostrada en la Figura 5.6a se obtuvo un valor de $B = 40.988432$ mientras que para las características obtenidas empleando la metodología propuesta se obtuvo un valor de $B = 90.040138$. A grandes rasgos esta distancia entre clases permite al modelo de clasificación discernir entre ambas clases con mayor facilidad.

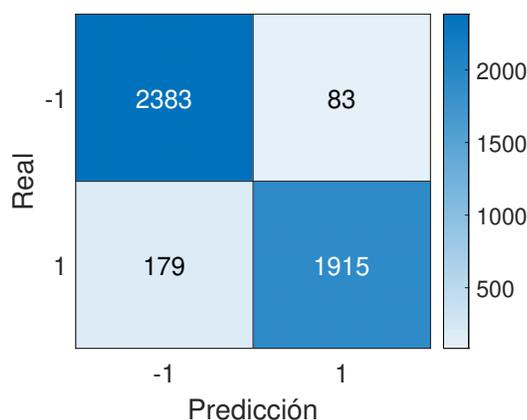


Figura 5.7: Matriz de confusión de mejor modelo encontrado y ordenes optimizados con algoritmo PSO.

La matriz de confusión del modelo obtenido se muestra en la Figura 5.7. La etiqueta 1 corresponde a señales provenientes de pacientes enfermos, mientras que la etiqueta -1 corresponde a las señales de pacientes sanos. El sistema de clasificación presenta un mayor error ante casos de falsos positivos, lo cual se traduce en un mejor índice de precisión pero menor índice de exhaustividad.

La Tabla 5.2 muestra una comparativa de índices de desempeño de diversas metodologías de clasificación de señales y los índices del mejor modelo obtenido por la metodología propuesta. Si bien el trabajo presentado en [104] presenta índices de desempeño mas elevados, la metodología utilizada para la clasificación se basa en el uso de redes neuronales convolucionales que son comparativamente más complejas y conllevan un costo computacional mayor que la metodología propuesta.

Tabla 5.2: Comparación de resultados de metodología propuesta contra algunos trabajos relevantes del estado del arte.

Referencia	Modelo de Clasificación	Segmentación	P	Ex	F_1	Acc
[103]	SVM	✓	90.07 %	88.45 %	89.25 %	89.265 %
[104]	CNN	✓	93.20 %	95.12 %	94.15 %	97 %
[101]	FDA-ANN	✓	76.96 %	88.31 %	82.24 %	82.63 %
[102]	Árbol de Decisiones	✓	77 %	80 %	78.47 %	79 %
[105]	SVM	×	88 %	87 %	87.49 %	88 %
Propuesto	Árbol de Decisiones	×	95.84 %	91.45 %	93.59 %	94.25 %

El esquema de la metodología propuesta se presenta en la Figura 5.8. En la etapa de preprocesamiento se aplica un ventaneado a las señales. Las ventanas obtenidas son posteriormente normalizadas y procesadas mediante la metodología de suavizado se señal propuesta. Posteriormente se obtienen las características a partir de la DFRFT y se obtienen los perfiles promedio por clase por separado. Las características obtenidas se emplean finalmente para el entrenamiento de un modelo de clasificación, en este caso un árbol de decisiones

Las metodologías propuestas para el preprocesamiento y la extracción de características basado en ope-

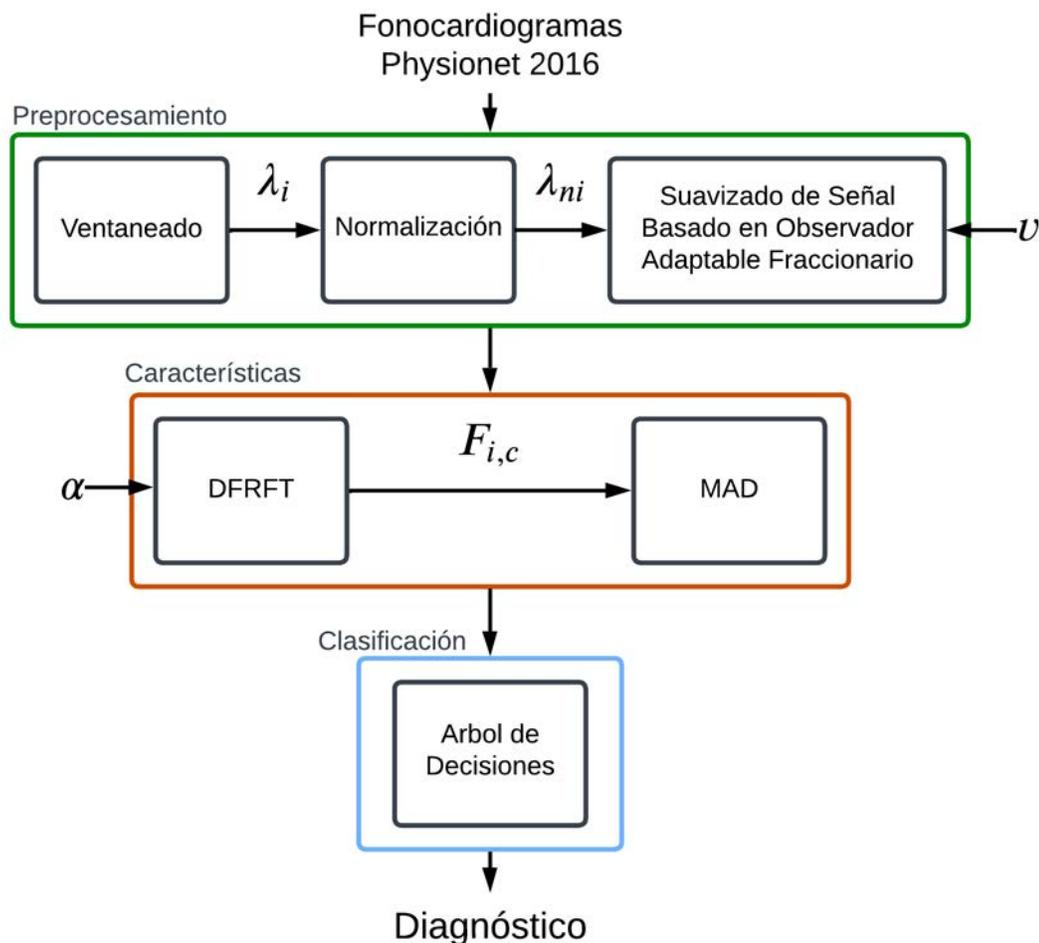


Figura 5.8: Esquema general de la metodología propuesta para la clasificación de fonocardiogramas.

radores de orden generalizado permiten obtener representaciones de la señal útiles para su clasificación. Se resalta que el método propuesto permite la clasificación de fonocardiogramas sin la necesidad de agregar una etapa de segmentación a la metodología, en este caso el uso de la representación en el dominio tiempo-frecuencia fraccionario permite al modelo de clasificación discernir entre las señales.

5.2. Caso 2. Imágenes Satelitales

La primera fotografía aérea registrada fue tomada en 1858 desde un globo aerostático a 262 pies de altura a las afueras de París [106]. Desde entonces, la captura de imágenes satelitales ha evolucionado de manera extraordinaria, impulsada por avances tecnológicos que han permitido la órbita de satélites equipados con sensores de alta resolución. A finales de 2022, 6718 satélites orbitaban la Tierra [107]. Los satélites proporcionan una visión detallada de la superficie terrestre, permitiendo la monitorización a

escala global y la obtención de datos cruciales para diversas disciplinas, como la climatología, la agricultura, la gestión de desastres y la planificación urbana [108]. La proliferación de tecnologías satelitales ha proporcionado un flujo constante de datos de alta resolución, generando oportunidades sin precedentes para extraer información valiosa.

La clasificación y análisis de patrones en las imágenes satelitales se han vuelto esenciales para comprender fenómenos terrestres complejos. Desde la detección de cambios en el uso del suelo hasta el monitoreo de la salud de los cultivos, la información extraída de estas imágenes contribuye significativamente a la toma de decisiones en una variedad de sectores [109]. El uso de imágenes satelitales también ha sido fundamental en la gestión de desastres naturales. Desde la predicción de eventos climáticos extremos hasta la evaluación de daños post-desastre, estas imágenes proporcionan una visión global en tiempo real, facilitando una respuesta rápida y efectiva.

El análisis efectivo de imágenes satélites presenta desafíos debido a diversos factores como variabilidad espectral causada por condiciones atmosféricas y temporales, la variación en la resolución espacial de las imágenes, la presencia de ruido y artefactos, la dificultad en la transferencia de modelos entre regiones geográficas y los cambios temporales en la cobertura del suelo. En esta sección se explora una metodología para la extracción de características en imágenes satelitales basada en el gradiente de orden fraccionario.

Los métodos de procesamiento de imágenes basados en gradiente emplean derivadas espaciales para realzar características relacionadas con el cambio de intensidad en la imagen. Estos métodos pueden ser utilizados para estudiar cambios de intensidad en los píxeles, cambios abruptos o de alta frecuencia suelen relacionarse con bordes de objetos en la imagen, mientras que cambios de baja frecuencia pueden ser interpretadas como texturas. Algoritmos basados en gradiente pueden ser utilizados para la detección de bordes, así como para el realce de características.

Los métodos basados en gradiente pueden ser categorizados de manera general en función del orden utilizado para el operador de derivada involucrado. En el caso clásico, operadores de primer orden suelen presentar dificultades para realzar bordes finos; si bien los operadores de segundo orden suelen ser capaces de resaltar estos detalles, su sensibilidad a ciertos tipos de ruido suele ser mayor. El cálculo de orden generalizado, al considerar ordenes de diferencia no enteros, permite incorporar comportamientos no locales en los algoritmos basados en gradiente, permitiendo realzar las características deseadas, manteniendo robustez contra ciertos tipos de ruido.

5.2.1. Base de Datos

La base de datos SAT-6 [110] recopila imágenes de distintas bases de datos de imágenes satelitales para obtener un total de 405000 secciones de dimensiones 28×28 píxeles. Las imágenes originales muestran fotografías satelitales de paisajes estadounidenses, presentando secciones de distintas áreas urbanas y rurales, así como de zonas densamente boscosas, terreno montañoso, cuerpos de agua de distintos tamaños, zonas agrícolas, entre otras. Las imágenes que componen la base de datos fueron etiquetadas manualmente, evitando que el subconjunto de prueba y el de entrenamiento tuvieran mosaicos adyacentes, esto se realizó con la finalidad de evitar sesgos durante la etapa de entrenamiento de clasificadores.

Posteriormente los subconjuntos fueron desordenados empleando un generador de secuencias pseudo-aleatorias para eliminar cualquier potencial estructura en el orden de los datos.

La base de datos se encuentra etiquetada con seis clases diferentes: tierra, arboles, pradera, carreteras, edificios y cuerpos de agua, cada imagen corresponde a una ventana deslizante no superpuesta. Las imágenes fueron adquiridas con una resolución de 1 metro por píxel y constan de 4 bandas: rojo, verde, azul e infrarrojo. La Figura 5.9 muestra algunas imágenes de cada clase de la base de datos.

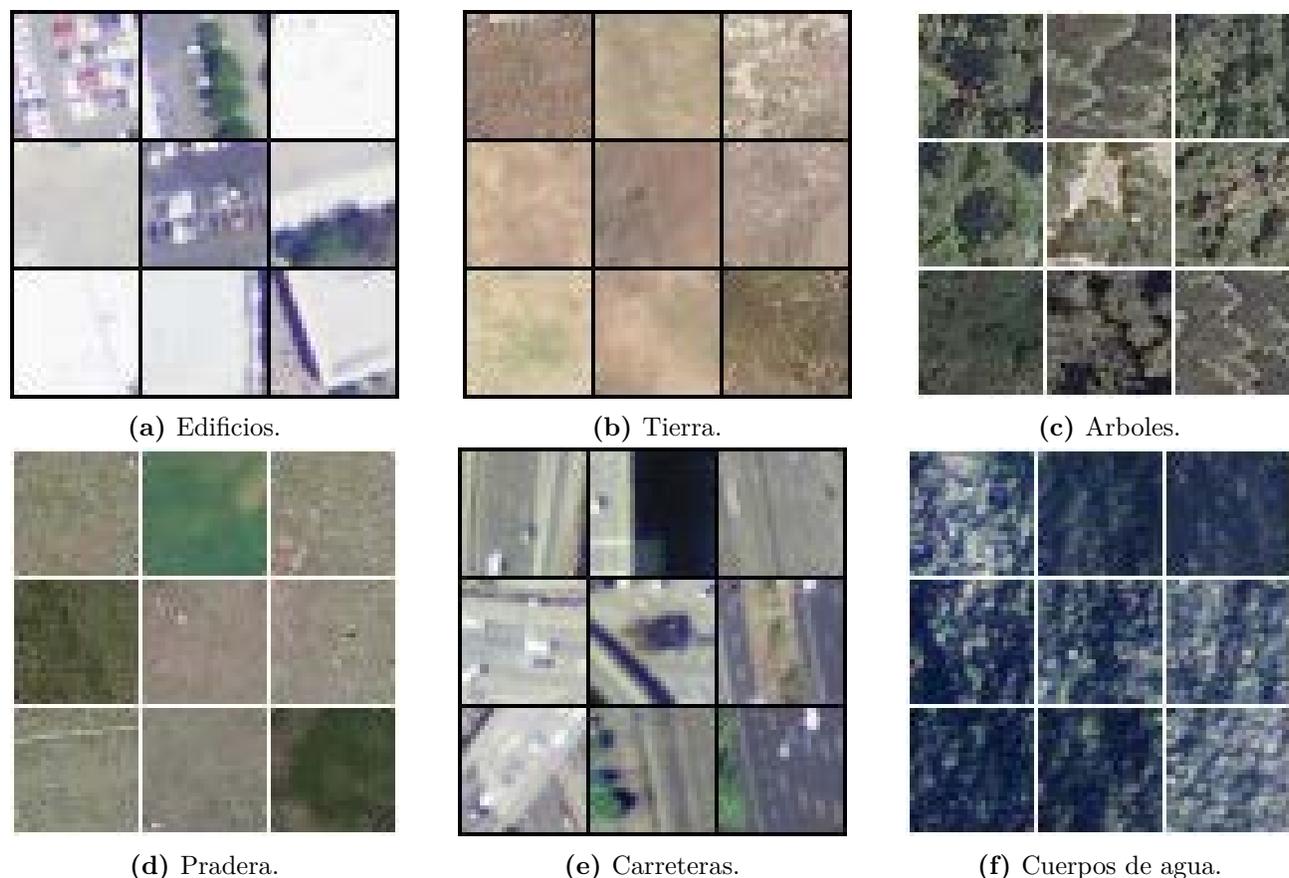


Figura 5.9: Imágenes muestra de cada clase de la base de datos SAT-6.

5.2.2. Preprocesamiento y Extracción de Características

En el preprocesamiento de la base de datos SAT-6, se llevaron a cabo dos pasos para transformar la información contenida en las imágenes. En primer lugar, se optó por descartar el canal de color infrarrojo, simplificando así la representación de cada píxel a tres canales (rojo, verde y azul). Esta decisión se tomó con el objetivo de reducir la complejidad computacional y focalizarse en los aspectos visuales más relevantes para las clasificaciones de cobertura de tierra. En segundo lugar, se balancearon los datos para lidiar con disparidades en la distribución de las clases. Este equilibrio de número de elementos en las clases se realizó para evitar sesgos en la etapa de entrenamiento, donde se consideraron un total de

12,000 imágenes.

Una imagen en escala de grises puede ser representada como una matriz bidimensional, donde cada elemento se encuentra asociado con el nivel de intensidad de ese píxel; el cual normalmente se encuentra acotado en el rango $[0,255]$. Una imagen RGB puede ser vista como una matriz de tres variables independientes $I(x, y, c) \in \mathbb{N}^{N \times M \times 3}$ donde $x, y \in \mathbb{N}$ indican las coordenadas espaciales de la imagen, por su parte c corresponde al canal de color. Una imagen RGB puede interpretarse como tres imágenes en escala de grises de las mismas dimensiones.

Para la etapa de extracción de características se propone el uso del gradiente de orden fraccionario, definido en la Ecuación (5.27) para imágenes en escala de grises

$$\nabla^v I(x, y, c) = \frac{\partial^v I(x, y)}{\partial x^v} + \frac{\partial^v I_c(x, y)}{\partial y^v}. \quad (5.27)$$

Para este trabajo se considera la generalización de los operadores de derivadas parciales empleando la definición de derivada fraccionaria discreta Δ^v mostrada en la Ecuación (2.22). La Ecuación (5.28) presenta el computo de la derivada en las direcciones horizontal y vertical.

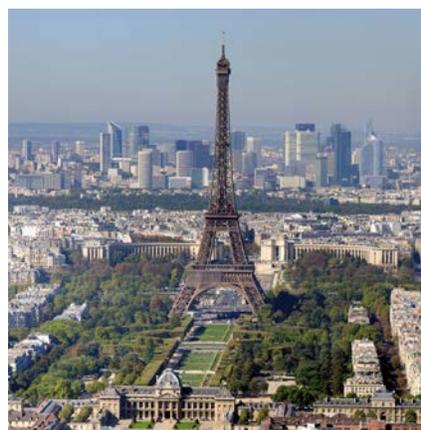
$$\begin{aligned} \frac{\partial^v}{\partial x^v} I(x, y, c) &= \sum_{i=0}^N a^{(v)}(i) f(N - i), \\ \frac{\partial^v}{\partial y^v} I(x, y, c) &= \sum_{i=0}^M a^{(v)}(i) f(M - i). \end{aligned} \quad (5.28)$$

Para el caso clásico ($v = 1$) el gradiente de una imagen representa los cambios de intensidad de píxeles adyacentes en cada canal de color. Cambios relativamente grandes pueden ser considerados bordes, mientras que cambios relativamente pequeños pueden referirse a texturas dentro de las mismas regiones de la imagen. En el caso de el gradiente de orden fraccionario para valores de $v > 0$ el efecto de el filtro puede ser utilizado para resaltar componentes de alta frecuencia, manteniendo las componentes de baja frecuencia. La Figura 5.10 compara el gradiente de orden entero de una imagen contra el gradiente de orden fraccionario con $v = 0.8$. En el caso, la imagen fue convertida previamente a escala de grises. El gradiente de orden entero este preprocesamiento permite realzar los bordes fuertes de la imagen, sin embargo al ser un filtro diferencial también puede incrementar el ruido presente en la imagen. En el caso del gradiente de orden fraccionario, es posible mitigar la sensibilidad al ruido.

El gradiente fraccionario presentado en la Ecuación (5.27) es utilizado como característica para la clasificación de las imágenes de la base de datos SAT-6. La Figura 5.11 muestra una imagen correspondiente a cada una de las clases de la base de datos junto con su gradiente de orden $v = 0.5$. El gradiente fue calculado de manera individual para cada canal de color, posteriormente los niveles de intensidad por canal fueron normalizados en el rango $[0, 255]$. Finalmente los gradientes normalizados de cada canal de color se concatenaron para formar una sola imagen RGB.

Con la finalidad de mantener las componentes de color de cada imagen se empleó como característica también la imagen original. La Ecuación (5.29) muestra las características empleadas (denotadas como γ). Para emplear estas características se normalizó la amplitud de los valores de intensidad de píxel en el rango $[0, 1]$

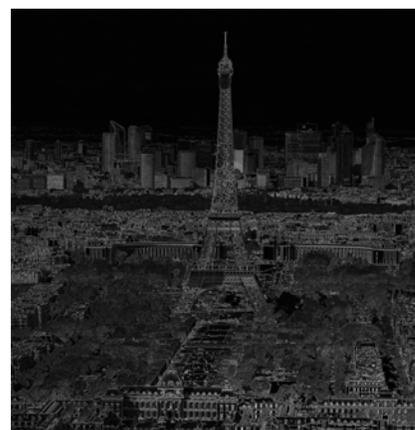
$$\gamma = [I(x, y, c) \quad \nabla I(x, y, c)] / 255. \quad (5.29)$$



(a) Imagen original.



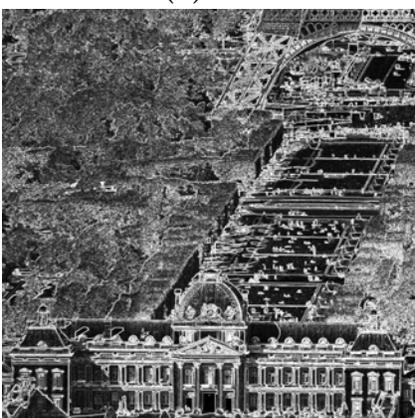
(b) $v = 1$.



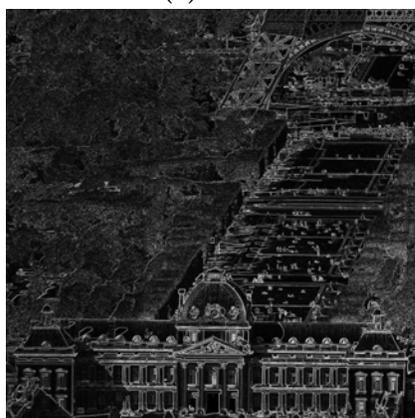
(c) $v = 0.8$.



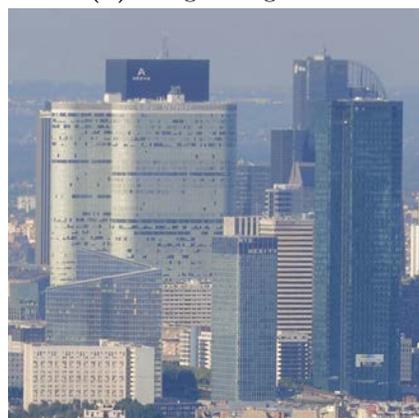
(d) Imagen original.



(e) $v = 1$.



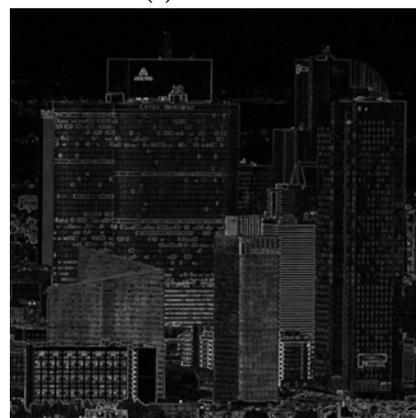
(f) $v = 0.8$.



(g) Imagen original.



(h) $v = 1$.



(i) $v = 0.8$.

Figura 5.10: Comparación de gradientes en imágenes ante diversos ordenes de v con acercamientos a objetos cercanos y lejanos en la imagen.

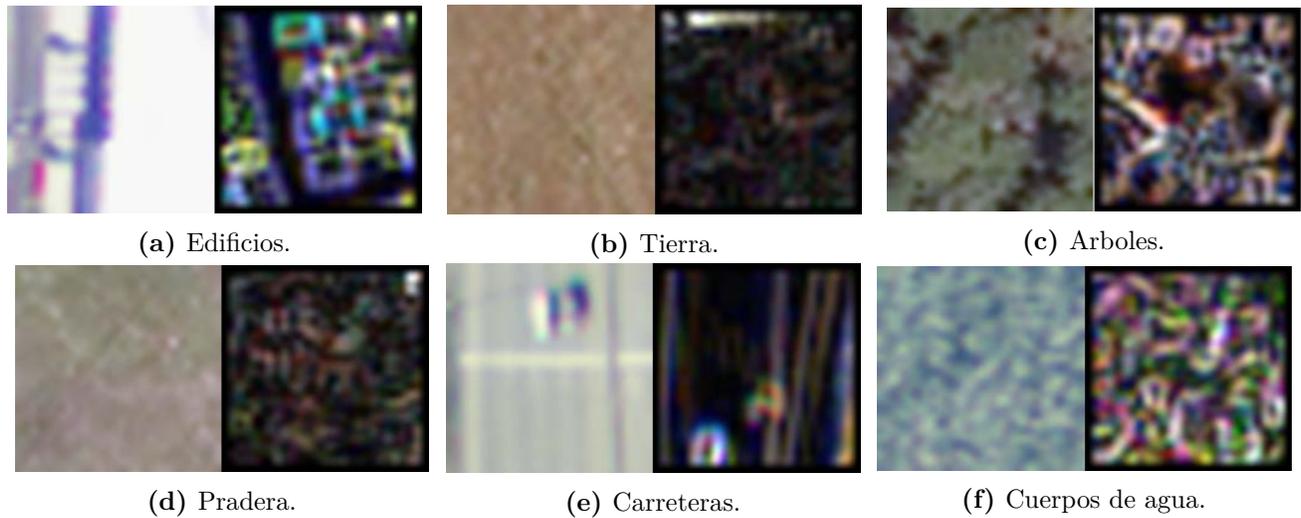


Figura 5.11: Ejemplos de características empleadas para cada clase.

5.2.3. Clasificación

Para la etapa de clasificación se optó por el uso de redes neuronales convolucionales en un esquema de clasificación jerárquica. Las estructuras de las CNNs utilizadas se describen brevemente a continuación: Para la capa de convolución se tomo un tamaño de filtro de 4×4 , con un número total de filtros de 64, cada filtro aprende a detectar ciertos patrones o características específicas de las imágenes de entrada. Para esta capa se empleo una función de activación ReLu. Posteriormente se tiene una capa de agrupación máxima (Max Pooling), la cual reduce las dimensiones de las características extraídas por la primera capa. Dicha operación es lograda, para este caso, dividiendo la salida de la capa anterior en regiones de 2×2 para después tomar el valor máximo de cada región. En algunas aplicaciones esto permite que la red reconozca ciertos patrones a pesar de que no aparezcan en las mismas coordenadas de la imagen. La salida de esta capa es posteriormente convertida a un arreglo unidimensional, que posteriormente se alimenta a una capa de 128 neuronas totalmente conectadas. Por ultimo la capa de salida se compone tantas neuronas como clases de salida se tienen en el modelo. En las dos ultimas capas se emplean funciones de activación de tipo sigmoide con la finalidad de producir salidas acotadas en el rango $[0, 1]$, lo cual facilita su interpretación como probabilidades de pertenencia a cada clase. Esta estructura se resume en la Tabla 5.3.

Tabla 5.3: Estructura de CNN utilizada para la clasificación de imágenes satelitales.

Capa	Tipo	Configuración	Función de Activación
1	Convolutacional	64 filtros. kernel 4×4 .	Relu
2	Max Pooling	ventana 2×2	-
3	Aplanado	-	-
4	Completamente Conectada	128 neuronas.	Sigmoide
5	Completamente Conectada	Neuronas igual a numero de clases	Softmax

Para el entrenamiento de las CNNs se empleo el método de optimización ADAM (Estimación de Mo-

mento Adaptativo), el cual es un método de optimización popular para aplicaciones de aprendizaje automático. Este método combina otros dos algoritmos de optimización, el gradiente descendente estocástico y el RMSprop. En términos generales, el algoritmo permite actualizar las tasas de aprendizaje de cada parámetro de manera individual basado en sus momentos de primer y segundo orden. La aleatoriedad correspondiente al gradiente estocástico previene que el algoritmo se detenga en posibles mínimos locales. El Algoritmo 1 presenta el pseudocódigo del método de optimización ADAM.

Pseudocódigo 1 Entrenamiento con método de ADAM.

Considerando: $\alpha \in \mathbb{R}^+$, $\epsilon \in \mathbb{R}$. $\beta_1\beta_2 \in (0,1)$, $w_0 \in \mathbb{R}^n$ y la función $f(w) \in C^2(\mathbb{R}^n, \mathbb{R})$

- 1: $m_0 = 0, v_0 = 0, t = 0$
- 2: **while w no converge do**
- 3: $m_{t+1} = \beta_1 m_t + (1 - \beta_1) \nabla_w f(w_t)$
- 4: $v_{t+1} = \beta_2 v_t + (1 - \beta_2) \nabla_w f(w_t) \otimes \nabla_w f(w_t)$
- 5: $w_{t+1} = w_t - \alpha \frac{\sqrt{1 - \beta_2^{t+1}}}{(1 - \beta_1^{t+1})} m_{t+1} \oslash \sqrt{v_{t+1} \oplus \epsilon}$
- 6: $t = t + 1$
- 7: **end while**

Se agregó además un factor de decaimiento de la tasa de aprendizaje de $D_f = 1/100$. El entrenamiento en todos los casos se realizó a lo largo de 25 épocas. Para la etapa de extracción de características se consideró un orden de diferencia de $v = 0.5$. Debido a que los datos de entrenamiento fueron previamente balanceados se empleó la exactitud como métrica de desempeño del modelo a entrenar.

Se entrenaron tres CNN en un esquema de clasificación jerárquica en donde el primer modelo clasifica de manera binaria dos subconjuntos de clases, como salida del modelo la etiqueta 0 indica que la imagen analizada corresponde a la clase de edificios, arboles o carreteras. Mientras que la etiqueta 1 corresponde a las clases de tierra, pradera y cuerpos de agua. Las dos CNNs restantes son utilizadas en cada subconjunto de clases para asignar una de tres etiquetas posibles a la imagen.

La Figura 5.12 muestra la matriz de confusión obtenida durante la etapa de entrenamiento del modelo de clasificación previamente descrito. En donde los valores de predicción acertados se sitúan sobre la diagonal principal de la matriz. Las incidencias de error mas altas suceden con la predicción de pradera en imágenes provenientes de tierra (clases 1 y 3) y con predicción de carreteras en imágenes de edificios (clases 0 y 4). Durante la etapa de entrenamiento se obtuvo un valor de $Acc = 0.987833$.

La Tabla 5.4 presenta la comparativa de resultados, donde se contrastan los resultados obtenidos con algunos modelos reportados en la literatura [111]. Todos los modelos evaluados en la tabla corresponden a diferentes configuraciones de CNNs.

Es relevante señalar que las CNNs empleadas como puntos de referencia en la comparación corresponden a modelos pre-entrenados, una práctica común que generalmente conlleva un mejor rendimiento en tareas de clasificación. La metodología propuesta logra niveles de exactitud superiores sin depender de un proceso de pre-entrenamiento de la red. Esto destaca la ventaja de emplear características basadas en operadores de orden fraccionario en la clasificación de imágenes satelitales. La Figura 5.13 presenta

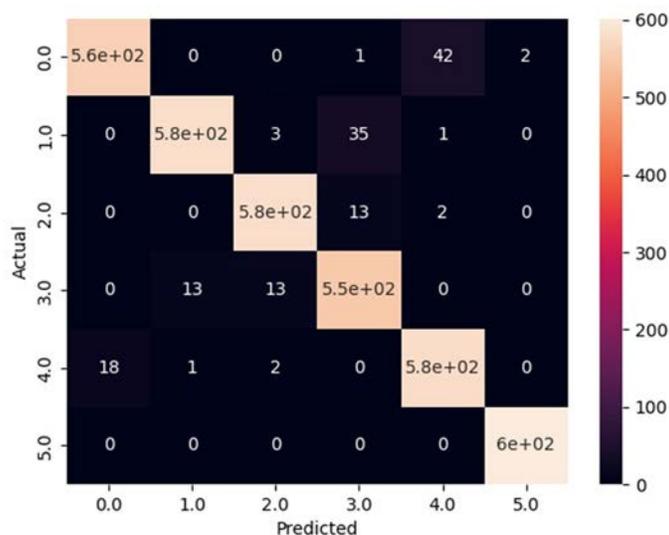


Figura 5.12: Matriz de confusión de la clasificación de imágenes provenientes de la base de datos SAT-6 con el modelo entrenado

Tabla 5.4: Comparación de resultados obtenidos con trabajos relevantes reportados en la literatura.

Modelo	Exactitud	Pre-entrenamiento	Tamaño [112]
AlexNet	82 %	si	271MB
VGG19	84 %	si	558.4MB
GoogleNet	89 %	si	47.1MB
Resnet50	94.1 %	si	94.3MB
Propuesto	95.9444 %	no	54.46MB

el esquema de la metodología propuesta basada en el gradiente de orden fraccionario para la extracción de características de imágenes satelitales

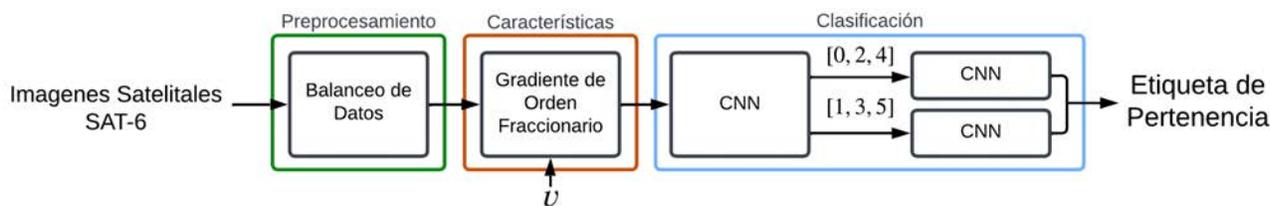


Figura 5.13: Esquema general de la metodología propuesta para la clasificación de imágenes satelitales basada en el gradiente discreto de orden fraccionario.

5.3. Conclusiones Parciales

En este capítulo se proponen metodologías de preprocesamiento y extracción de características basadas en el cálculo discreto de orden generalizado. En el caso de señales unidimensionales se propone un algoritmo de suavizado de señales basado en un tipo de observador adaptable, el cual permite atenuar el ruido de la señal, manteniendo componentes de interés. Para la extracción de características se empleó la representación en el dominio tiempo-frecuencia fraccionario otorgado por la FRFT. Para la elección del modelo de clasificación se consideraron estructuras de tipo SVM, KNN y árboles de decisiones, obteniendo mejores resultados con estos últimos. Se consideró como índice de desempeño para el entrenamiento de los modelos el valor-F (F_1). La selección de los órdenes tanto de diferencia como de transformación fueron seleccionados a partir del uso de un algoritmo de optimización heurístico (PSO).

La metodología propuesta fue aplicada para la clasificación de fonocardiogramas para la detección de patologías. Es relevante remarcar que algoritmos de clasificación con el mismo objetivo propuestos en la literatura suelen emplear una etapa de segmentación de la señal como parte del preprocesamiento que no resulta necesaria con la metodología propuesta. El modelo obtenido es comparativamente más simple a nivel computacional que otros modelos propuestos en la literatura, logrando resultados comparables.

Para la clasificación de imágenes se propone el uso del gradiente de orden fraccionario ∇^v empleando la definición de derivada Δ^v para la extracción de características en imágenes satelitales. El cálculo de orden generalizado, al considerar órdenes de diferencia no enteros, permite incorporar comportamientos no locales en los algoritmos basados en gradiente, lo que se traduce en un realce de las características deseadas de la imagen sin comprometer completamente información de texturas y manteniendo robustez contra ciertos tipos de ruido. Como clasificador se empleó un esquema de clasificación jerárquico basado en CNNs, obteniendo resultados comparables a CNNs pre-entrenadas reportadas en la literatura.

Capítulo 6

Conclusiones

El cálculo fraccionario ofrece ciertas ventajas para la construcción de metodologías y algoritmos de procesamiento tanto de imágenes como señales. Estas herramientas permiten por ejemplo, en el contexto de extracción de características, representar mas información de los datos analizados. Por otro lado, en el análisis de sistemas caóticos, es posible representar trayectorias distintas a las obtenidas con el sistema de orden entero. Esta propiedad sumada al hecho de que el orden de la derivada provee de grados de libertad adicional al sistema vuelve a estas herramientas atractivas para su uso en esquemas criptográficos. El presente trabajo discute tres metodologías de procesamiento de señales que emplean la definición de derivada discreta fraccionaria Δ^v .

El Capitulo 4 presenta un esquema de sincronización de acoplamiento unidireccional de dos mapas caóticos de orden de diferencia fraccionario basado en controlador. La implementación del esquema propuesto fue realizada en dos tarjetas Arduino UNO, en donde, aprovechando la propiedad de memoria truncada del operador Δ es posible la ejecución del sistema en tiempo real. Se propone una posible aplicación del esquema propuesto en la generación de secuencias pseudoaleatorias para la obtención de mascararas, las cuales fueron utilizadas para la encriptación de imágenes RGB. Para garantizar la suficiente aleatoriedad de las secuencias obtenidas se recurrió al banco de pruebas del Instituto Nacional de Estándares y Tecnología (NIST). En este caso, el tamaño de la llave puede ser incrementado considerablemente considerando un orden de diferencia variable en el operador $\Delta^{v(k)}$, lo cual incrementa la robustez de la metodología ante diversos tipos de ataques de fuerza bruta, manteniendo además resistencia contra otros tipos de ataques.

En el Capitulo 5 se presentan dos metodologías basadas en el cálculo discreto de orden generalizado en aplicaciones de clasificación automática. Primero se presenta una metodología de diagnóstico binario automático para padecimientos cardiacos basado en clasificación de fonocardiogramas obtenidos de la base de datos *PhysioNet / CinC Challenge 2016*. Para la etapa de preprocesamiento se propone un algoritmo de suavizado de señal basado en un observador adaptable de orden fraccionario, el cual permite reducir las variaciones rápidas de la señal, preservando información importante presente en ella. Para la etapa de extracción de características se considero la variación con respecto a la media de cada valor de la transformación al dominio fraccionario tiempo-frecuencia empleando la transformada fraccionaria de Fourier. Para la selección de modelo de clasificación se compararon diversos índices de desempeño. El

modelo elegido posteriormente fue utilizado para la optimización de los ordenes α y ν por medio de un algoritmo de optimización metaheurística PSO. Los resultados obtenidos son satisfactorios al compararse con metodologías encontradas en la literatura con un modelo de relativamente baja complejidad algorítmica.

Posteriormente se muestra una de clasificación de imágenes provenientes de la base de datos SAT-6 que se compone de imágenes satelitales de diversos tipos de terreno, etiquetadas en seis clases diferentes. Las características para la clasificación fueron obtenidas en base al gradiente de orden fraccionario con orden $\nu = 0.5$. Las imágenes procesadas se utilizaron para el entrenamiento de una red neuronal convolucional (CNN). Los resultados obtenidos muestran una mayor exactitud, incluso al ser comparados con CNNs pre-entrenadas.

Diversas características presentes en el cálculo discreto de orden generalizado resultan útiles para la mejora de algoritmos equivalentes de orden entero, así como para el planteamiento de nuevas técnicas de procesamiento de señales. En el primer caso presentado, la descripción de diferentes dinámicas en sistemas a través de la variación del orden de diferencia permite brindar mayor robustez en sistemas criptográficos. Por su parte en el procesamiento de señales la tolerancia al ruido de los operadores diferenciales de orden fraccionario permite una supresión de componentes no deseadas, sin perder información valiosa encontrada en los datos.

6.1. Trabajos Futuros

Se proponen diversos trabajos futuros:

- Diseño de controladores de orden fraccionario para sistemas de orden fraccionario inconmensurado.
- Metodología de selección de parámetros para el controlador propuesto en el Capítulo 4, ya sea de forma analítica o mediante métodos de optimización metaheurística.
- Aplicación del generador de secuencias pseudoaleatorias propuesto en otras aplicaciones criptográficas como la esteganografía.
- Aplicaciones de operadores del cálculo fraccionario en los modelos de aprendizaje para la clasificación automática.
- Analizar la interpretabilidad de los modelos de clasificación obtenidos.

Capítulo 6

Bibliografía

- [1] Chunhui Wen and Jinhai Yang. Complexity evolution of chaotic financial systems based on fractional calculus. Chaos, Solitons & Fractals, 128:242–251, 2019.
- [2] Yuexi Peng, Shaobo He, and Kehui Sun. Chaos in the discrete memristor-based system with fractional-order difference. Results in Physics, 24:104106, 2021.
- [3] Yu-Ming Chu, Stelios Bekiros, Ernesto Zambrano-Serrano, Onofre Orozco-López, Salim Lahmiri, Hadi Jahanshahi, and Ayman A Aly. Artificial macro-economics: A chaotic discrete-time fractional-order laboratory model. Chaos, Solitons & Fractals, 145:110776, 2021.
- [4] Shaobo He, Kehui Sun, and Yuexi Peng. Detecting chaos in fractional-order nonlinear systems using the smaller alignment index. Physics Letters A, 383(19):2267–2271, 2019.
- [5] Enli Chen, Wuce Xing, Meiqi Wang, Wenli Ma, and Yujian Chang. Study on chaos of nonlinear suspension system with fractional-order derivative under random excitation. Chaos, Solitons & Fractals, 152:111300, 2021.
- [6] Sara M Mohamed, Wafaa S Sayed, Lobna A Said, and Ahmed G Radwan. Reconfigurable fpga realization of fractional-order chaotic systems. IEEE Access, 9:89376–89389, 2021.
- [7] Tianming Liu, Huizhen Yan, Santo Banerjee, and Jun Mou. A fractional-order chaotic system with hidden attractor and self-excited attractor and its dsp implementation. Chaos, Solitons & Fractals, 145:110791, 2021.
- [8] Carlos Jesús Zuñiga-Aguilar, José Francisco Gómez-Aguilar, RF Escobar-Jiménez, and Héctor Manuel Romero-Ugalde. Robust control for fractional variable-order chaotic systems with non-singular kernel. The European physical journal plus, 133:1–13, 2018.
- [9] Zhimin Han, Shenggang Li, and Heng Liu. Composite learning sliding mode synchronization of chaotic fractional-order neural networks. Journal of Advanced Research, 25:87–96, 2020.
- [10] Manashita Borah, Antara Gayan, Jiv Siddhi Sharma, YangQuan Chen, Zhouchao Wei, and Viet-Thanh Pham. Is fractional-order chaos theory the new tool to model chaotic pandemics as covid-19? Nonlinear dynamics, 109(2):1187–1215, 2022.

- [11] A Al-Khedhairi, AE Matouk, and I Khan. Chaotic dynamics and chaos control for the fractional-order geomagnetic field model. Chaos, Solitons & Fractals, 128:390–401, 2019.
- [12] Yanli Cao. Chaotic synchronization based on fractional order calculus financial system. Chaos, Solitons & Fractals, 130:109410, 2020.
- [13] Sanjay Kumar, Ahmed E Matouk, Harindri Chaudhary, and Shashi Kant. Control and synchronization of fractional-order chaotic satellite systems using feedback and adaptive control techniques. International Journal of Adaptive Control and Signal Processing, 35(4):484–497, 2021.
- [14] Mehmet Ali Akinlar, Fairouz Tehier, and Mustafa Inc. Chaos control and solutions of fractional-order malkus waterwheel model. Chaos, Solitons & Fractals, 135:109746, 2020.
- [15] Zain-Aldeen SA Rahman, Basil H Jasim, Yasir IA Al-Yasir, Raed A Abd-Alhameed, and Bilal Naji Alhasnawi. A new no equilibrium fractional order chaotic system, dynamical investigation, synchronization, and its digital implementation. Inventions, 6(3):49, 2021.
- [16] Pei-Ying Xiong, Hadi Jahanshahi, Raúl Alcaraz, Yu-Ming Chu, JF Gómez-Aguilar, and Fawaz E Alsaadi. Spectral entropy analysis and synchronization of a multi-stable fractional-order chaotic system using a novel neural network-based chattering-free sliding mode technique. Chaos, Solitons & Fractals, 144:110576, 2021.
- [17] Thabet Abdeljawad, Santo Banerjee, and Guo-Cheng Wu. Discrete tempered fractional calculus for new chaotic systems with short memory and image encryption. Optik, 218:163698, 2020.
- [18] Zai-rong Wang, Babak Shiri, and Dumitru Baleanu. Discrete fractional watermark technique. Frontiers of Information Technology & Electronic Engineering, 21(6):880–883, 2020.
- [19] Yi-Fei Pu, Ni Zhang, and Huai Wang. Fractional-order spatial steganography and blind steganalysis for printed matter: Anti-counterfeiting for product external packing in internet-of-things. IEEE Internet of Things Journal, 6(4):6368–6383, 2018.
- [20] Sumit Kumar, Bhaskar Panna, and Rajib Kumar Jha. Medical image encryption using fractional discrete cosine transform with chaotic function. Medical & biological engineering & computing, 57:2517–2533, 2019.
- [21] Ankita Vaish and Manoj Kumar. Color image encryption using msvd, dwt and arnold transform in fractional fourier domain. Optik, 145:273–283, 2017.
- [22] Gururaj Maddodi, Abir Awad, Dounia Awad, Mirna Awad, and Brian Lee. A new image encryption algorithm based on heterogeneous chaotic neural network generator and dna encoding. Multimedia tools and applications, 77:24701–24725, 2018.
- [23] MA Ben Farah, R Guesmi, A Kachouri, and M Samet. A novel chaos based optical image encryption using fractional fourier transform and dna sequence operation. Optics & Laser Technology, 121:105777, 2020.

- [24] Liping Chen, Hao Yin, Tingwen Huang, Liguó Yuan, Song Zheng, and Lisheng Yin. Chaos in fractional-order discrete neural networks with application to image encryption. Neural Networks, 125:174–184, 2020.
- [25] Melih Yildirim. Dna encoding for rgb image encryption with memristor based neuron model and chaos phenomenon. Microelectronics Journal, 104:104878, 2020.
- [26] Sebastian Raubitzek, Kevin Mallinger, and Thomas Neubauer. Combining fractional derivatives and machine learning: A review. Entropy, 25(1):35, 2022.
- [27] Harbir Antil, Ratna Khatri, Rainald Löhner, and Deepanshu Verma. Fractional deep neural network via constrained optimization. Machine Learning: Science and Technology, 2(1):015003, 2020.
- [28] Wei Wang, Ying Jia, Qiming Wang, Pengfei Xu, et al. An image enhancement algorithm based on fractional-order phase stretch transform and relative total variation. Computational Intelligence and Neuroscience, 2021, 2021.
- [29] JE Lavín-Delgado, JE Solís-Pérez, José Francisco Gómez-Aguilar, and Ricardo Fabricio Escobar-Jiménez. A new fractional-order mask for image edge detection based on caputo–fabrizio fractional-order derivative without singular kernel. Circuits, Systems, and Signal Processing, 39:1419–1448, 2020.
- [30] Saroj Kumar Chandra and Manish Kumar Bajpai. Effective algorithm for benign brain tumor detection using fractional calculus. In TENCON 2018-2018 IEEE Region 10 Conference, pages 2408–2413. IEEE, 2018.
- [31] Zaid Abduh, Ebrahim Ameen Nehary, Manal Abdel Wahed, and Yasser M Kadah. Classification of heart sounds using fractional fourier transform based mel-frequency spectral coefficients and traditional classifiers. Biomedical Signal Processing and Control, 57:101788, 2020.
- [32] Erdal Dinç. Wavelet transforms and applications in drug analysis. FABAD Journal of Pharmaceutical Sciences, 38(3):159, 2013.
- [33] Tobón C. Gómez-Echavarría A., Ugarte J. P. The fractional fourier transform as a biomedical signal and image processing tool: A review. Biocybernetics and Biomedical Engineering, 40(3):1081–1093, 2020.
- [34] Yu-Dong Zhang, Shufang Chen, Shui-Hua Wang, Jian-Fei Yang, and Preetha Phillips. Magnetic resonance brain image classification based on weighted-type fractional fourier transform and nonparallel support vector machine. International Journal of Imaging Systems and Technology, 25(4):317–327, 2015.
- [35] Yudong Zhang, Yi Sun, Preetha Phillips, Ge Liu, Xingxing Zhou, and Shuihua Wang. A multilayer perceptron based smart pathological brain detection system by fractional fourier entropy. Journal of medical systems, 40:1–11, 2016.
- [36] Keling Fei, Wei Wang, Qiaoli Yang, and Shusen Tang. Chaos feature study in fractional fourier domain for preictal prediction of epileptic seizure. Neurocomputing, 249:290–298, 2017.

- [37] Qi Wang, Jing Ma, Siyuan Yu, and Liying Tan. Noise detection and image denoising based on fractional calculus. Chaos, Solitons & Fractals, 131:109463, 2020.
- [38] Waseem Waheed, Guang Deng, and Bo Liu. Discrete laplacian operator and its applications in signal processing. IEEE Access, 8:89692–89707, 2020.
- [39] Cardiovascular diseases (cvds). [https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-\(cvds\)](https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)). Consultado en: 2021-06-11.
- [40] Fatima Chakir, Abdelilah Jilbab, Chafik Nacir, Ahmed Hammouch, and Amir Hajjam El Hassani. Detection and identification algorithm of the s1 and s2 heart sounds. In 2016 International Conference on Electrical and Information Technologies (ICEIT), pages 418–420. IEEE, 2016.
- [41] Ravindra Manohar Potdar, Mekhram Meshram, and Ramesh Kumar. Design and analysis of a pcg acquisition and denoising circuit. Turkish Journal of Physiotherapy and Rehabilitation, 32(3):750–764, 2021.
- [42] Sinam Ajitkumar Singh, Takhellambam Gautam Meitei, and Swanirbhar Majumder. Short pcg classification based on deep learning. In Deep learning techniques for biomedical and health informatics, pages 141–164. Elsevier, 2020.
- [43] Yineng Zheng, Xingming Guo, Hong Jiang, and Benmei Zhou. An innovative multi-level singular value decomposition and compressed sensing based framework for noise removal from heart sounds. Biomedical Signal Processing and Control, 38:34–43, 2017.
- [44] Aydana S Kyzdarbekova, Dana M Dutbaeva, Kuralay B Kasymbekova, and Ulbosyn S Kyzdarbek. Adaptive noise reduction phonocardiograms based on wavelet transformation. In 2017 International Conference "Quality Management, Transport and Information Security, Information Technologies" (IT&QM&IS), pages 391–394. IEEE, 2017.
- [45] Mohamed Rouis, Salim Sbaa, and Nasser Edinne Benhassine. The effectiveness of the choice of criteria on the stationary and non-stationary noise removal in the phonocardiogram (pcg) signal using discrete wavelet transform. Biomedical Engineering/Biomedizinische Technik, 65(3):353–366, 2020.
- [46] Muhammad Usman Akram, Arslan Shaukat, Farhan Hussain, Sajid Gul Khawaja, Wasi Haider Butt, et al. Analysis of pcg signals using quality assessment and homomorphic filters for localization and classification of heart sounds. Computer methods and programs in biomedicine, 164:143–157, 2018.
- [47] A Coronel-Escamilla, JF Gómez-Aguilar, L Torres, RF Escobar-Jiménez, and M Valtierra-Rodríguez. Synchronization of chaotic systems involving fractional operators of liouville–caputo type with variable-order. Physica A: Statistical Mechanics and its Applications, 487:1–21, 2017.
- [48] JE Solís-Pérez, JF Gómez-Aguilar, and A Atangana. Novel numerical method for solving variable-order fractional differential equations with power, exponential and mittag-leffler laws. Chaos, Solitons & Fractals, 114:175–185, 2018.

- [49] LF Ávalos-Ruiz, JF Gomez-Aguilar, A Atangana, and Kolade M Owolabi. On the dynamics of fractional maps with power-law, exponential decay and mittag-leffler memory. Chaos, Solitons & Fractals, 127:364–388, 2019.
- [50] A. F. Pérez-Vidal. Identificación de señales en el cerebro humano asociadas a movimientos musculares y a la respuesta de estímulos externos. PhD thesis, CENIDET, 2019.
- [51] M. B. Ramos Martínez. Teoría de Identificación de Sistemas Aplicado al Diagnóstico y Tratamiento de Señales Bio-médicas. PhD thesis, CENIDET, 2021.
- [52] Ruifan Meng. Application of fractional calculus to modeling the non-linear behaviors of ferroelectric polymer composites: viscoelasticity and dielectricity. Membranes, 11(6):409, 2021.
- [53] Changqing Fang, Xiaoyin Shen, Kuai He, Chao Yin, Shasha Li, Xiaolong Chen, and Huiyu Sun. Application of fractional calculus methods to viscoelastic behaviours of solid propellants. Philosophical Transactions of the Royal Society A, 378(2172):20190291, 2020.
- [54] Khaled M Saad, HM Srivastava, and JF Gómez-Aguilar. A fractional quadratic autocatalysis associated with chemical clock reactions involving linear inhibition. Chaos, Solitons & Fractals, 132:109557, 2020.
- [55] Khaled M Saad. Comparative study on fractional isothermal chemical model. Alexandria Engineering Journal, 60(3):3265–3274, 2021.
- [56] Abhaya Pal Singh, Dipankar Deb, Himanshu Agrawal, Kishore Bingi, and Stepan Ozana. Modeling and control of robotic manipulators: A fractional calculus point of view. Arabian Journal for Science and Engineering, 46(10):9541–9552, 2021.
- [57] Pritesh Shah and Ravi Sekhar. Predictive modeling and control of clamp load loss in bolted joints based on fractional calculus. In Advances in Computing and Network Communications: Proceedings of CoCoNet 2020, Volume 1, pages 15–32. Springer, 2021.
- [58] Shu-Bo Chen, Hadi Jahanshahi, Oumate Alhadji Abba, JE Solís-Pérez, Stelios Bekiros, JF Gómez-Aguilar, Amin Yousefpour, and Yu-Ming Chu. The effect of market confidence on a financial system from the perspective of fractional calculus: Numerical investigation and circuit realization. Chaos, Solitons & Fractals, 140:110223, 2020.
- [59] Ting Chen and Derong Wang. Combined application of blockchain technology in fractional calculus model of supply chain financial system. Chaos, Solitons & Fractals, 131:109461, 2020.
- [60] Piotr Ostalczyk. Discrete fractional calculus: applications in control and image processing, volume 4. World scientific, 2015.
- [61] Lan-Lan Huang, Guo-Cheng Wu, Dumitru Baleanu, and Hong-Yong Wang. Discrete fractional calculus for interval-valued systems. Fuzzy Sets and Systems, 404:141–158, 2021.
- [62] Zhijie Li, Jie Ding, Min Wu, and Jinxing Lin. Discrete fractional order pid controller design for nonlinear systems. International Journal of Systems Science, 52(15):3206–3213, 2021.

- [63] Xingxing You, Qiankun Song, and Zhenjiang Zhao. Existence and finite-time stability of discrete fractional-order complex-valued neural networks with time delays. Neural Networks, 123:248–260, 2020.
- [64] Martin Bohner and Allan Peterson. Dynamic equations on time scales: An introduction with applications. Springer Science & Business Media, 2001.
- [65] Roland Priemer. Introductory signal processing, volume 6. World scientific, 1991.
- [66] Lokenath Debnath and Dambaru Bhatta. Integral transforms and their applications. Chapman and Hall/CRC, 2016.
- [67] Ashutosh Kumar Singh, Rajiv Saxena, et al. Dfrft: a classified review of recent methods with its application. Journal of Engineering, 2013, 2013.
- [68] Jair Cervantes, Farid Garcia-Lamont, Lisbeth Rodríguez-Mazahua, and Asdrubal Lopez. A comprehensive survey on support vector machine classification: Applications, challenges and trends. Neurocomputing, 408:189–215, 2020.
- [69] Bahzad Charbuty and Adnan Abdulazeez. Classification based on decision tree algorithm for machine learning. Journal of Applied Science and Technology Trends, 2(01):20–28, 2021.
- [70] Kashvi Taunk, Sanjukta De, Srishti Verma, and Aleena Swetapadma. A brief review of nearest neighbor algorithm for learning and classification. In 2019 international conference on intelligent computing and control systems (ICCS), pages 1255–1260. IEEE, 2019.
- [71] Keiron O’Shea and Ryan Nash. An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458, 2015.
- [72] Jesús M González-Miranda. Synchronization and control of chaos: an introduction for scientists and engineers. World Scientific, 2004.
- [73] Zakia Hammouch, Mehmet Yavuz, and Necati Özdemir. Numerical solutions and synchronization of a variable-order fractional chaotic system. Mathematical Modelling and Numerical Simulation with Applications, 1(1):11–23, 2021.
- [74] Adel Ouannas, Amina-Aicha Khennaoui, Samir Bendoukha, Thoai Phu Vo, Viet-Thanh Pham, and Van Van Huynh. The fractional form of the tinkerbell map is chaotic. Applied Sciences, 8(12):2640, 2018.
- [75] Ralph Abraham and Yoshisuke Ueda. The chaos avant-garde: Memories of the early days of chaos theory, volume 39. World scientific, 2000.
- [76] Waldemar Bauer and Aleksandra Kawala-Janik. Implementation of bi-fractional filtering on the arduino uno hardware platform. In Theory and Applications of Non-integer Order Systems: 8th Conference on Non-integer Order Calculus and Its Applications, Zakopane, Poland, pages 419–428. Springer, 2017.

- [77] Adel Ouannas, Amina Aicha Khennaoui, Taki-Eddine Oussaeif, Viet-Thanh Pham, Giuseppe Grassi, and Zohir Dibi. Hyperchaotic fractional grassi–miller map and its hardware implementation. *Integration*, 80:13–19, 2021.
- [78] Kanika Suneja, Shelza Dua, and Mohit Dua. A review of chaos based image encryption. In *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pages 693–698. IEEE, 2019.
- [79] Zai-Yin He, Abderrahmane Abbes, Hadi Jahanshahi, Naif D Alotaibi, and Ye Wang. Fractional-order discrete-time sir epidemic model with vaccination: Chaos and complexity. *Mathematics*, 10(2):165, 2022.
- [80] Amin Yousefpour, Hadi Jahanshahi, Jesus M Munoz-Pacheco, Stelios Bekiros, and Zhouchao Wei. A fractional-order hyper-chaotic economic system with transient chaos. *Chaos, Solitons & Fractals*, 130:109400, 2020.
- [81] Yuexi Peng, Shaobo He, and Kehui Sun. Chaos in the discrete memristor-based system with fractional-order difference. *Results in Physics*, 24:104106, 2021.
- [82] Lawrence E Bassham III, Andrew L Rukhin, Juan Soto, James R Nechvatal, Miles E Smid, Elaine B Barker, Stefan D Leigh, Mark Levenson, Mark Vangel, David L Banks, et al. *Sp 800-22 rev. 1a. a statistical test suite for random and pseudorandom number generators for cryptographic applications*. National Institute of Standards & Technology, 2010.
- [83] Xing-Quan Fu, Bo-Cheng Liu, Yi-Yuan Xie, Wei Li, and Yong Liu. Image encryption-then-transmission using dna encryption algorithm and the double chaos. *IEEE Photonics Journal*, 10(3):1–15, 2018.
- [84] Allan Weber. The usc-sipi image database. *Signal and Image Processing Institute of the University of Southern California*. URL: <http://sipi.usc.edu/services/database>, 1997.
- [85] Siju John and SN Kumar. 2d lorentz chaotic model coupled with logistic chaotic model for medical image encryption: Towards ensuring security for teleradiology. *Procedia Computer Science*, 218:918–926, 2023.
- [86] Alaa Kadhim Farhan, Nadia MG Al-Saidi, Abeer Tariq Maolood, Fahimeh Nazarimehr, and Iqtadar Hussain. Entropy analysis and image encryption application based on a new chaotic system crossing a cylinder. *Entropy*, 21(10):958, 2019.
- [87] Jiasen Liu, Chao Wang, Zheng Tu, Xu An Wang, Chuan Lin, and Zhihu Li. Secure knn classification scheme based on homomorphic encryption for cyberspace. *Security and Communication Networks*, 2021:1–12, 2021.
- [88] Yong Zhang. The image encryption algorithm based on chaos and dna computing. *Multimedia Tools and Applications*, 77(16):21589–21615, 2018.
- [89] Gururaj Maddodi, Abir Awad, Dounia Awad, Mirna Awad, and Brian Lee. A new image encryption algorithm based on heterogeneous chaotic neural network generator and dna encoding. *Multimedia tools and applications*, 77:24701–24725, 2018.

- [90] Guo-Cheng Wu, Zhen-Guo Deng, Dumitru Baleanu, and De-Qiang Zeng. New variable-order fractional chaotic systems for fast image encryption. Chaos: An Interdisciplinary Journal of Nonlinear Science, 29(8), 2019.
- [91] Wenqian Yu, Ye Liu, Lihua Gong, Miaomiao Tian, and Liangqiang Tu. Double-image encryption based on spatiotemporal chaos and dna operations. Multimedia Tools and Applications, 78:20037–20064, 2019.
- [92] Melih Yildirim. Dna encoding for rgb image encryption with memristor based neuron model and chaos phenomenon. Microelectronics Journal, 104:104878, 2020.
- [93] Mohamed Zakariya Talhaoui and Xingyuan Wang. A new fractional one dimensional chaotic map and its application in high-speed image encryption. Information Sciences, 550:13–26, 2021.
- [94] Centro Estatal de Vigilancia Epidemiológica y Control de Enfermedades. Panorama epidemiológico de la enfermedad cardiovascular en el estado de México. <https://cevece.edomex.gob.mx/sites/cevece.edomex.gob.mx/files/files/docs/documentostec/reportes/2019/PEECEM.pdf>. Consultado en: 2023-04-12.
- [95] Rosario Ríos-Prado, Álvaro Anzueto-Ríos, and Blanca Tovar-Corona. Metodología para discernir entre sonido cardiaco no patológico de regurgitación y estenosis aórtica, empleando dtw. Revista de la Facultad de Ciencias, 8(1):138–155, 2019.
- [96] Ali Almasi, Mohammad B Shamsollahi, and Lotfi Senhadji. A dynamical model for generating synthetic phonocardiogram signals. In 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pages 5686–5689. IEEE, 2011.
- [97] Soroor Behbahani. A hybrid algorithm for heart sounds segmentation based on phonocardiogram. Journal of Medical Engineering & Technology, 43(6):363–377, 2019.
- [98] Gari D Clifford, Chengyu Liu, Benjamin Moody, David Springer, Ikaro Silva, Qiao Li, and Roger G Mark. Classification of normal/abnormal heart sound recordings: The physionet/computing in cardiology challenge 2016. In 2016 Computing in cardiology conference (CinC), pages 609–612. IEEE, 2016.
- [99] CJ Zúñiga-Aguilar, JF Gómez-Aguilar, HM Romero-Ugalde, Hadi Jahanshahi, and Fawaz E Alsaadi. Fractal-fractional neuro-adaptive method for system identification. Engineering with Computers, pages 1–24, 2021.
- [100] Dongshu Wang, Dapei Tan, and Lei Liu. Particle swarm optimization algorithm: an overview. Soft computing, 22:387–408, 2018.
- [101] Mostafa Abdollahpur, Ali Ghaffari, Shadi Ghiasi, and M Javad Mollakazemi. Detection of pathological heart sounds. Physiological measurement, 38(8):1616, 2017.
- [102] Philip Langley and Alan Murray. Heart sound classification from unsegmented phonocardiograms. Physiological measurement, 38(8):1658, 2017.

- [103] Bradley M Whitaker, Pradyumna B Suresha, Chengyu Liu, Gari D Clifford, and David V Anderson. Combining sparse coding and time-domain features for heart sound classification. Physiological measurement, 38(8):1701, 2017.
- [104] Juan P Dominguez-Morales, Angel F Jimenez-Fernandez, Manuel J Dominguez-Morales, and Gabriel Jimenez-Moreno. Deep neural networks for the recognition and classification of heart murmurs using neuromorphic auditory sensors. IEEE transactions on biomedical circuits and systems, 12(1):24–34, 2017.
- [105] Elmar Messner, Matthias Zöhrer, and Franz Pernkopf. Heart sound segmentation—an event detection approach using deep recurrent neural networks. IEEE transactions on biomedical engineering, 65(9):1964–1974, 2018.
- [106] Olivia B Waxman. Aerial photography’s surprising role in history. Time Magazine, 31, 2018.
- [107] Teri Grimwood. Ucs satellite database. www.ucsusa.org/satellite_database. Consultado en: 2023-04-12.
- [108] Gerald K Moore. What is a picture worth? a history of remote sensing/quelle est la valeur d’une image? un tour d’horizon de télédétection. Hydrological Sciences Bulletin, 24(4):477–485, 1979.
- [109] Marshall Burke, Anne Driscoll, David B Lobell, and Stefano Ermon. Using satellite imagery to understand and promote sustainable development. Science, 371(6535):eabe8628, 2021.
- [110] Saikat Basu, Sangram Ganguly, Supratik Mukhopadhyay, Robert DiBiano, Manohar Karki, and Ramakrishna Nemani. Deepsat: a learning framework for satellite imagery. In Proceedings of the 23rd SIGSPATIAL international conference on advances in geographic information systems, pages 1–10, 2015.
- [111] Mohammed Abbas Kadhim and Mohammed Hamzah Abed. Convolutional neural network for satellite image classification. Intelligent Information and Database Systems: Recent Developments 11, pages 165–178, 2020.
- [112] Jie Hang, Dexiang Zhang, Peng Chen, Jun Zhang, and Bing Wang. Classification of plant leaf diseases based on improved convolutional neural network. Sensors, 19(19):4161, 2019.

Anexos



Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

journal homepage: www.elsevier.com/locate/chaos

A RGB image encryption technique using chaotic maps of fractional variable-order based on DNA encoding

L.F. Ávalos-Ruíz^a, C.J. Zúñiga-Aguilar^b, J.F. Gómez-Aguilar^{c,*}, H.M. Cortes-Campos^a, J.E. Lavín-Delgado^d

^a Tecnológico Nacional de México/CENIDET. Interior Internado Palmira S/N, Col. Palmira, C.P. 62490, Cuernavaca, Morelos, Mexico

^b Data Scientist at Panzura Data Services. Bv. Puerta de Hierro 5153, Puerta de Hierro. C.P. 45116 Zapopan, Jal., Mexico

^c CONAHCYT-Tecnológico Nacional de México/CENIDET. Interior Internado Palmira S/N, Col. Palmira, C.P. 62490, Cuernavaca, Morelos, Mexico

^d Universidad Politécnica del Estado de Guerrero (UPEG), Dirección de Ingeniería en Redes y Telecomunicaciones, Puente Campuzano, Carretera Federal Iguala-Taxco K.M. 105, Taxco de Alarcón, C.P. 40321, Guerrero, Mexico

ARTICLE INFO

Keywords:
Chaos theory
Arduino implementation
Control theory
Cryptography
Fractional variable-order calculus

ABSTRACT

This paper presents a controller-based synchronization scheme in a leader-follower architecture for chaotic maps of fractional variable-order. The proposed scheme demonstrates successful synchronization of three different chaotic maps through simulation results. The implementation on Arduino UNO boards is used to showcase the relatively low algorithm complexity of the proposed scheme. Additionally, the paper explores a potential application of the proposed scheme in an RGB image encryption system, leveraging the chaotic behavior of the system for pseudo-random sequence generation. The encoding scheme utilizes the complementary property of Deoxyribonucleic acid (DNA) molecule bases. The presented encryption scheme employs the variable-order of the fractional derivative as a key for improved safety. This enhancement improves the methodology resistance against brute-force attacks while maintaining resilience against other cryptanalysis techniques.

1. Introduction

Fractional calculus is the mathematical field that addresses problems involving non-integer differential or integral orders. The non-integer order found in fractional calculus definitions allows for the description of memory behaviors found in systems from vast interdisciplinary contexts, such as: physics [1,2], chemistry [3,4], control [5,6], finance [7,8], biology [9], epidemiology [10–14], among others. In these examples, fractional-order calculus allows for the creation of models that describe experimental data more accurately. Fractional-order chaotic systems allow for the representation of a wider variety of dynamics when compared to their integer order counterparts. Several studies have been reported focusing on exploring the properties and dynamics of fractional-order chaotic systems. For instance, in the work presented by Avalos et al. [15], a variation of Euler's method is used, which uses the continuous generalized order computation methods in discrete systems. Different operators from fractional, conformable and fractional-conformable calculus were applied with the purpose of finding behaviors different from those described by the classic case. It is remarked that among the equations of a system, both the order of integration and the definition of calculation used may be

different from each other. Fractional-order chaotic systems introduce fractional derivatives or integrals into the system equations, allowing for the modeling and analysis of complex behaviors beyond traditional integer-order systems. Several studies have focused on exploring the properties and dynamics of fractional-order chaotic systems. For instance, Peng et al. [16], designed a discrete memristor-based system with fractional-order difference, observing changes in the lobe area of the hysteresis loop with the fractional-order. Solís-Pérez et al. [17] proposed a numerical method based around the Lagrange polynomial for the simulation of conformable dynamic systems defined by the conformable Liouville-Caputo derivative, simulation results are tested with the Wang-Sun Attractor, furthermore an anti-synchronization scheme based on a conformable slide mode controller was presented. Chu et al. [18], developed an artificial macroeconomic model that represents the dynamic relations between income, consumption and investment. The designed model exhibits chaotic behavior and employs the fractional-order Caputo difference definition. The bifurcation diagrams regarding the fractional-order were studied, evidencing its influence on the chaotic regions of the system. Finally, the proposed system was implemented on an Arduino Due board, where due to hardware

* Corresponding author.

E-mail address: jose.ga@cenidet.tecnm.mx (J.F. Gómez-Aguilar).

<https://doi.org/10.1016/j.chaos.2023.114306>

Received 31 July 2023; Received in revised form 15 October 2023; Accepted 22 November 2023

0960-0779/© 2023 Elsevier Ltd. All rights reserved.