



Universidad
Tecnológica de Bolívar
CARTAGENA DE INDIAS

**IMPLEMENTACIÓN Y EVALUACIÓN DE MÉTODOS DE FILTRADO
DIGITAL PARA IMÁGENES ULTRASÓNICAS**

HANSEL CASTILLA CABALLERO

UNIVERSIDAD DE TECNOLÓGICA DE BOLÍVAR
FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
CARTAGENA – COLOMBIA
2013

**IMPLEMENTACIÓN Y EVALUACIÓN DE MÉTODOS
DE FILTRADO DIGITAL PARA IMÁGENES
ULTRASÓNICAS**

Esta página ha sido dejada intencionalmente en blanco

**IMPLEMENTACIÓN Y EVALUACIÓN DE MÉTODOS DE FILTRADO
DIGITAL PARA IMÁGENES ULTRASÓNICAS**

HANSEL ARMANDO CASTILLA CABALLERO

DESARROLLADOR

TRABAJO DE GRADO COMO REQUISITO PARCIAL
PARA OPTAR EL TÍTULO DE INGENIERO ELECTRÓNICO

SONIA H. CONTRERAS ORTIZ

INGENIERA ELECTRÓNICA.PhD

DIRECTORA

PROCESAMIENTO DIGITAL DE SEÑALES (DSP)

BIOINGENIERÍA

ÁREAS DE INVESTIGACIÓN

**UNIVERSIDAD DE TECNOLÓGICA DE BOLÍVAR
FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**

PROGRAMA DE INGENIERÍA ELECTRÓNICA

CARTAGENA – COLOMBIA

2013

Nota de aceptación

Presidente del jurado

Jurado

Jurado

Cartagena de Indias, Febrero 28 de 2013.

**A Jehová, por ser mi guía, acompañarme en todo momento,
resguardarme y darme ánimos para no rendirme.**

**A mi madre y hermano, junto con el resto de mi familia
porque siempre me dieron su apoyo incondicional.**

**A mi tutora y modelo a seguir,
por haberme acogido, por permitirme trabajar a su lado y por
enseñarme de sus conocimientos y experiencias en
bioingeniería y en la vida en general.**

AGRADECIMIENTOS

A Dios, Señor Todopoderoso, que siempre me acompaña y me guía por el camino del bien. Él que me apoya en todo momento para poder avanzar en cada una de las etapas de mi vida, proveyéndome de sabiduría para esquivar obstáculos y romper barreras.

A mi madre y hermano, María y Deyler, quienes siempre han estado junto a mí y que a pesar de las dificultades tenidas siempre hemos estado juntos. A ellos muchas gracias por su apoyo y motivación. Son la mejor familia que pude haber deseado.

A mi tutora, a quien considero mi amiga, la Doctora Sonia Contreras, quien me demostró que una persona puede estar llena de conocimientos, puede ser alguien muy reconocido y ser la más sencilla de todas también. Muchas gracias por haberme permitido el honor de trabajar a su lado. Nunca olvidaré esta distinguida experiencia. Gracias a usted encontré mi vocación en esta hermosa carrera.

A mis tíos, tías, primos y primas, porque todos ellos han contribuido a mi desarrollo personal y profesional. En especial a Concepción por haberme ayudado durante todo este tiempo.

A mis amigos y amigas, en especial a Kati Cuadrado, Carolina Restrepo y Katty Castro, quienes han estado conmigo en todo momento y estoy seguro que se regocijan conmigo por este logro alcanzado.

Hansel Castilla Caballero

Esta página ha sido dejada intencionalmente en blanco

CONTENIDO

1	<i>PRESENTACIÓN</i>	6
	RESUMEN	6
	INTRODUCCIÓN	7
	ANTECEDENTES Y ESTADO DEL ARTE.....	8
	PLANTEAMIENTO DEL PROBLEMA.....	12
	OBJETIVOS.....	14
	JUSTIFICACIÓN.....	15
	ALCANCES DE LA INVESTIGACIÓN	16
2	<i>MARCO DE REFERENCIA</i>	17
2.1	Procesamiento Digital de Imágenes	17
2.2	Ultrasonido	18
2.3	Filtros Digitales	23
2.4	Evaluación de la Calidad de una Imagen	31
3	<i>METODOLOGÍA</i>	33
3.1	Filtro Adaptativo:	35
3.2	Filtro por Difusión Anisotrópica:	37
3.3	Filtro por Wavelet:.....	39
4	<i>RESULTADOS Y DISCUSIÓN</i>	43
5	<i>CONCLUSIONES y RECOMENDACIONES</i>	48
6	<i>BIBLIOGRAFÍA</i>	50
	<i>ANEXOS</i>	57

GRÁFICAS Y TABLAS	58
CÓDIGO DE LA INTERFAZ IMPLEMENTADA	95

ÍNDICE DE TABLAS

<i>Tabla 1. SNR y CNR para Quiste Mamario. Ejemplo#1. Filtro Adaptativo.....</i>	<i>44</i>
<i>Tabla 2. SNR y CNR para Quiste Mamario. Ejemplo#1. Filtro por Difusión Anisotrópica.....</i>	<i>45</i>
<i>Tabla 3. SNR y CNR para Quiste Mamario. Ejemplo#1. Filtro por Wavelet.....</i>	<i>45</i>
<i>Tabla 4. SNR y CNR para Quiste Mamario. Ejemplo#2. Filtro Adaptativo.....</i>	<i>58</i>
<i>Tabla 5. SNR y CNR para Quiste Mamario. Ejemplo#2. Filtro por Difusión Anisotrópica.....</i>	<i>59</i>
<i>Tabla 6. SNR y CNR para Quiste Mamario. Ejemplo#2. Filtro por Wavelet.....</i>	<i>59</i>
<i>Tabla 7. SNR y CNR para Quiste Mamario. Ejemplo#3. Filtro Adaptativo.....</i>	<i>61</i>
<i>Tabla 8. SNR y CNR para Quiste Mamario. Ejemplo#3. Filtro por Difusión Anisotrópica.....</i>	<i>62</i>
<i>Tabla 9. SNR y CNR para Quiste Mamario. Ejemplo#3. Filtro por Wavelet.....</i>	<i>62</i>
<i>Tabla 10. SNR y CNR para Quiste Mamario. Ejemplo#4. Filtro Adaptativo.....</i>	<i>64</i>
<i>Tabla 11. SNR y CNR para Quiste Mamario. Ejemplo#4. Filtro por Difusión Anisotrópica.....</i>	<i>65</i>
<i>Tabla 12. SNR y CNR para Quiste Mamario. Ejemplo#4. Filtro por Wavelet.....</i>	<i>65</i>
<i>Tabla 13. SNR y CNR para Riñón. Ejemplo#5. Filtro Adaptativo.....</i>	<i>67</i>
<i>Tabla 14. SNR y CNR para Riñón. Ejemplo#5. Filtro por Difusión Anisotrópica.....</i>	<i>68</i>
<i>Tabla 15. SNR y CNR para Riñón. Ejemplo#5. Filtro por Wavelet.....</i>	<i>68</i>
<i>Tabla 16. SNR y CNR para Riñón. Ejemplo#6. Filtro Adaptativo.....</i>	<i>70</i>
<i>Tabla 17. SNR y CNR para Riñón. Ejemplo#6. Filtro por Difusión Anisotrópica.....</i>	<i>71</i>
<i>Tabla 18. SNR y CNR para Riñón. Ejemplo#6. Filtro por Wavelet.....</i>	<i>71</i>
<i>Tabla 19. SNR y CNR para Riñón. Ejemplo#7. Filtro Adaptativo.....</i>	<i>73</i>
<i>Tabla 20. SNR y CNR para Riñón. Ejemplo#7. Filtro por Difusión Anisotrópica.....</i>	<i>74</i>
<i>Tabla 21. SNR y CNR para Riñón. Ejemplo#7. Filtro por Wavelet.....</i>	<i>74</i>
<i>Tabla 22. SNR y CNR para Quiste. Ejemplo#8. Filtro Adaptativo.....</i>	<i>76</i>
<i>Tabla 23. SNR y CNR para Quiste. Ejemplo#8. Filtro por Difusión Anisotrópica.....</i>	<i>77</i>
<i>Tabla 24. SNR y CNR para Quiste. Ejemplo#8. Filtro por Wavelet.....</i>	<i>77</i>
<i>Tabla 25. SNR y CNR para Quiste. Ejemplo#9. Filtro Adaptativo.....</i>	<i>79</i>
<i>Tabla 26. SNR y CNR para Quiste. Ejemplo#9. Filtro por Difusión Anisotrópica.....</i>	<i>80</i>
<i>Tabla 27. SNR y CNR para Quiste. Ejemplo#9. Filtro por Wavelet.....</i>	<i>80</i>
<i>Tabla 28. SNR y CNR para Quistes. Ejemplo#10. Filtro Adaptativo.....</i>	<i>82</i>
<i>Tabla 29. SNR y CNR para Quistes. Ejemplo#10. Filtro por Difusión Anisotrópica.....</i>	<i>83</i>
<i>Tabla 30. SNR y CNR para Quistes. Ejemplo#10. Filtro por Wavelet.....</i>	<i>83</i>

ÍNDICE DE FIGURAS

Figura 1. Imagen ultrasónica simulada de un feto humano.	20
Figura 2. Variación de brillo en imagen ultrasónica simulada de riñón.....	22
Figura 3. Variación de Contraste en Imagen ultrasónica de prototipo de lesión.....	23
Figura 4. Selección de la Ventana de Speckle.	24
Figura 5. Secuencialidad del Filtrado de un pixel.	25
Figura 6. Diagrama de bloques de Descomposición con Wavelet.....	30
Figura 7. Diagrama de bloques de la Transformada Inversa de Wavelet.	30
Figura 8. Diagrama de Flujo de Algoritmo para Filtro Adaptativo.....	36
Figura 9. Diagrama de Flujo de Algoritmo para Filtro por Difusión Anisotrópica.....	38
Figura 10. Diagrama de Flujo de Algoritmo para Filtro por transformada Wavelet.....	40
Figura 11. Interfaz gráfica para Filtrado de Imágenes Ultrasónicas.....	41
Figura 12. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #1.	44
Figura 13. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #2.	58
Figura 14. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #3.	61
Figura 15. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #4.	64
Figura 16. Imágenes ultrasónicas sin filtrar y filtradas de Riñón. Ejemplo #5.....	67
Figura 17. Imágenes ultrasónicas sin filtrar y filtradas de segmento de Riñón. Ejemplo #6.....	70
Figura 18. Imágenes ultrasónicas sin filtrar y filtradas de segmento de Riñón. Ejemplo #7.....	73
Figura 19. Imágenes ultrasónicas sin filtrar y filtradas de Quiste. Ejemplo #8.	76
Figura 20. Imágenes ultrasónicas sin filtrar y filtradas de Quiste. Ejemplo #9.	79
Figura 21. Imágenes ultrasónicas sin filtrar y filtradas de Quistes. Ejemplo #10.....	82
Figura 22. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #11.	85
Figura 23. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #12.	86
Figura 24. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #13.	87
Figura 25. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #14.	88
Figura 26. Imágenes ultrasónicas sin filtrar y filtradas. Ejemplo #15.	89
Figura 27. Imágenes ultrasónicas sin filtrar y filtradas de pelvis simulado. Ejemplo #16.....	90
Figura 28. Imágenes ultrasónicas sin filtrar y filtradas de quiste. Ejemplo #17.....	91
Figura 29. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #18.	92
Figura 30. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #19.	93
Figura 31. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #20.	94

ÍNDICE DE GRÁFICAS

Gráfica 1. Comparación de SNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#1.	46
Gráfica 2. Comparación de CNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#1.	46
Gráfica 3. Comparación de SNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#2.	60
Gráfica 4. Comparación de CNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#2.	60
Gráfica 5. Comparación de SNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#3.	63
Gráfica 6. Comparación de CNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#3.	63
Gráfica 7. Comparación de SNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#4.	66
Gráfica 8. Comparación de CNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#4.	66
Gráfica 9. Comparación de SNR para Imágenes de Riñón con y sin filtrado Ejemplo#5.	69
Gráfica 10. Comparación de CNR para Imágenes Riñón con y sin filtrado Ejemplo#5.	69
Gráfica 11. Comparación de SNR para Imágenes de Riñón con y sin filtrado Ejemplo#6.	72
Gráfica 12. Comparación de CNR para Imágenes de Riñón con y sin filtrado Ejemplo#6.	72
Gráfica 13. Comparación de SNR para Imágenes de Riñón con y sin filtrado Ejemplo#7.	75
Gráfica 14. Comparación de CNR para Imágenes de Riñón con y sin filtrado Ejemplo#7.	75
Gráfica 15. Comparación de SNR para Quiste con y sin filtrado Ejemplo#8.	78
Gráfica 16. Comparación de CNR para Quiste con y sin filtrado Ejemplo#8.	78
Gráfica 17. Comparación de SNR para Quiste con y sin filtrado Ejemplo#9.	81
Gráfica 18. Comparación de CNR para Quiste con y sin filtrado Ejemplo#9.	81
Gráfica 19. Comparación de SNR para Quiste con y sin filtrado Ejemplo#10.	84
Gráfica 20. Comparación de CNR para Quiste con y sin filtrado Ejemplo#10.	84

1 PRESENTACIÓN

RESUMEN

El presente proyecto de investigación, consistió en construir y evaluar los tres métodos de filtrado de imágenes ultrasónicas más sobresalientes, de acuerdo a la bibliografía consultada, para reducir el ruido granular generado en los procesos de adquisición de las señales de ultrasonido. Los filtros escogidos son Filtro Anisotrópico, Filtro por Difusión Anisotrópica y Filtro por Wavelet.

El diseño de los filtros se hizo considerando información y datos consultados en la literatura, con modificaciones realizadas para adaptarlo a las condiciones requeridas. La implementación se llevó a cabo siguiendo los parámetros establecidos en el diseño, y se evaluó empleando imágenes ultrasónicas reales e implementadas por el simulador Field II.

Para realizar los filtros, se procedió a efectuar los diseños de los mismos en una interfaz gráfica hecha en el software MatLab.v2010a en la cual se permite escoger de las opciones tenidas el tipo de filtro que se desea implementar, así como también la imagen a la que se quiere disminuir ruido granular.

Los resultados mostraron que los Filtros para imágenes ultrasónicas implementados reducen significativamente el ruido granular.

INTRODUCCIÓN

Existen varios métodos de adquisición de imágenes para visualizar los órganos internos de las personas, los más importantes son radiografías, resonancias magnéticas y ecografías. Los dos primeros mencionados utilizan señales de frecuencias muy altas en comparación con las radiofrecuencias empleadas por los métodos ultrasónicos. Los primeros mencionados utilizan señales con frecuencias que se encuentran en la denominada *Banda Ionizante*, lo que traduce en que pueden causar alteraciones en el organismo de la persona si es sometida constantemente a ellas. Por su parte las ecografías utilizan señales de ultrasonido, es decir, las frecuencias utilizadas están por debajo de los 20MHz, lo que implica que el individuo no estará en riesgo cuando se le hacen estudios con éste método; además, éste proceso es el que mejor presenta relación costo – beneficio de todos los citados con anterioridad. La ecografía es un procedimiento que consiste en enviar señales provenientes de un transductor que viajan a través del organismo y cuando encuentran un obstáculo parte de la energía es reflejada y captada por el transductor. Con este proceso se entiende que cuando se tiene energía reflejada es porque se ha encontrado un objeto másico en el área de estudio. Este método presenta errores en la impresión de las imágenes debido al ruido que se tiene, por lo cual se hace necesaria la disminución de él. Existen varios métodos de filtrado de imágenes para mejorar la calidad de las mismas y reducir el Speckle (ruido granular). En este proyecto de grado se evaluaron tres de los más importantes métodos de filtrado, denominados Filtro Adaptativo, Filtro por Difusión Anisotrópica y Filtro por Wavelet; y se estableció cuál fue el más efectivo en el aumento de SNR y CNR.

ANTECEDENTES Y ESTADO DEL ARTE

El filtrado de imágenes con propósitos médicos ha tenido un desarrollo relevante en las últimas décadas. Esto debido en gran manera al incremento en la aparición de casos de tumores y cánceres en la actualidad; con lo cual se ha hecho necesario mejorar la calidad de las tomas para detectar con menos esfuerzo anomalías en los tejidos u órganos de las personas.

Actualmente existen muchos métodos para el filtrado de imágenes. Filtros Adaptativos, Filtros por Difusión Anisotrópica, Wavelet, Filtros por Deconvolución [1]. La optimización de estos tipos de filtros ha sido consecuente con el progreso de la bioingeniería del siglo XXI mostrando características variadas en cada implementación propuesta.

Filtros Adaptativos

Algunos de los artículos desatacados para este tipo de filtros se mencionan a continuación:

En 1986, J. C. Bamber y C. Daft realizaron un filtro adaptativo con intenciones de reducir las características del speckle obtenido en imágenes ultrasónicas. Este filtro realizaba una ventana para suavizar el ruido granular con fundamentos estadísticos [2].

V. Dutt en asociación con J. Greenleaf publicaron su metodología para el desarrollo de un filtro adaptativo para ruido granular en el año de 1996. Este filtro de compresión logarítmica fue implementado en imágenes tipo B encontrando cuantitativamente valores para el speckle y así poderlo reducir [3].

S. Contreras, T. Chiu y M. Fox en el año 2011 propusieron un diseño especial para la ventana local de filtrado. Dicha ventana posee una forma hexagonal y

evalúa el pixel intermedio en ella lo que les arrojó resultados bastante buenos de relación señal a ruido y relación contraste a ruido cuando compararon las imágenes filtradas con sus respectivas originales [4].

Filtros por Difusión Anisotrópica

Este tipo de filtros se implementan también para la mejora en la calidad de imágenes. Investigaciones e implementaciones de ellos son:

En el año de 1990 P. Perona y J. Malik muestran una variación en la forma de definir las escalas del espacio, realizando una mejora a la propuesta introducida por Witkin y su núcleo Gaussiano. Ellos proponen algoritmos para realizar el cálculo del coeficiente de difusión variándolo espacialmente para ir suavizando las variaciones de pixeles debido al ruido granular [5].

Yongjian Yu y Scott Acton propusieron tomaron los estudios realizados de SRAD e implementaron un filtro de difusión Anisotrópica para aplicaciones con imágenes de ultrasonido y obtenidas por radar [6].

En el año 2007 fue publicado un documento que describe propiedades numéricas de la realización de filtros de difusión Anisotrópica, con este modelo, Krissian y sus colaboradores toman a la imagen como una matriz de contornos. En ese modelo se toma también una relación entre la intensidad de varianza direccional local y la geometría de la imagen [7].

Filtros por Wavelet

En 1994 fue publicado por H. Guo y sus coinvestigadores un artículo introductorio acerca de la reducción del ruido granular utilizando el coeficiente wavelet de una imagen transformada logarítmicamente. Este método ha sido importante en el avance de la resolución del speckle ya que mantiene no altera en gran manera las características de contraste, nitidez y calidad de la imagen original [8].

A. Bruce que en el año de 1996 publicó un texto en el cual expone la teoría de recomposición de señales con elementos diferentes a las clásicas sinusoides propuestas por Fourier [9].

Zong, Laine y Geiser presentaron en el año de 1998 un artículo en el cual presentan un algoritmo para reducción de speckle y mejora de contraste de imágenes ecocardiográficas utilizando análisis multiescala de la transformada Wavelet. En él se enseñan técnicas para la eliminación del ruido que no afecten en gran manera la nitidez ni el contraste de las imágenes [10].

En el año 2002, S. Kouro y R. Musalem realizaron un texto en el cual presentan los fundamentos de la teoría de Wavelet para el filtrado de imágenes. Este trabajo tomó las bases propuestas por A. Bruce¹ y presentó mejoras en cuanto a la presentación del teorema, sus propiedades, sus aplicaciones y su similitud con el análisis de Fourier para el estudio de señales. En este artículo se presentan también ejemplos con la herramienta de Wavelet que posee el programa MatLab[®] 6.0 [11].

Gupta, Chauhan y Sexana introducen un método basado en el suavizado del ruido granular de imágenes médicas de ultrasonido con la modificación y el manejo de los coeficientes de Wavelet [12]. Los resultados obtenidos y publicados en su artículo muestran cuantitativamente las buenas características del algoritmo implementado, lo que constituyó un avance notorio en la mejora del filtrado, más claridad en las ecografías y por ende un mejor diagnóstico médico. Este método aparece luego de realizar una variación al propuesto por Chang and Vetterli [13].

En el año 2006 fue presentada una forma de supresión de ruido en imágenes ultrasónicas que utiliza fundamentos de Wavelet con difusión. Este método utiliza las técnicas de esparcimiento y multiresolución de Wavelet, también mejora las características de presencia de ruido con la difusión no-linear. Yue, Croitoru, Zwischenberger y Clark presentaron un diseño para suavizar speckle en las

¹A. Bruce, «Wavelet analysis,» 1996.

imágenes detectando los ejes en ellas, utilizando módulos de wavelet normalizados; con lo cual, el algoritmo implementado por ellos no requiere etapa de preprocesamiento para las imágenes y el speckle es suprimido debido a las múltiples iteraciones de la difusión multiescala en los coeficientes de wavelet. Esta técnica fue probada con imágenes de ultrasonido obtenidas por simulación y en imágenes adquiridas por transductores para ecografías [14].

Comparación Entre Los Filtros

Algunos autores han realizado comparaciones entre diferentes tipos de filtros con objeto de determinar cuál de ellos es mejor en cuanto a brindar calidad de imagen, contraste y nitidez, algunos de las publicaciones hechas son:

S. Finn, M. Glavin y E. Jones realizaron en enero del año 2011 una serie variada de filtros para un ecocardiograma en especial. En ese documento se muestra la metodología para la implementación de los mismos. Los filtros efectuados fueron de tipo difusión anisotrópica, wavelet y adaptativo [15].

En el año 2012 sale la publicación realizada por S. Contreras y colaboradores en la cual se presenta una revisión en de las metodologías implementadas para la realización de filtros digitales para imágenes de ultrasonido. En este documento se menciona el proceso a seguir para el desarrollo del filtrado. Inicialmente se introduce acerca de la problemática tenida en la adquisición de las imágenes por medio de transductores, posteriormente se mencionan las técnicas de preprocesamiento para la preparación del filtrado y las de postprocesamiento más efectivas para la mejora de la calidad de las imágenes [1].

PLANTEAMIENTO DEL PROBLEMA

DESCRIPCIÓN DEL PROBLEMA

La ecografía es un estudio de adquisición de imágenes que se realiza para determinar posibles traumas en los órganos internos de las personas². Éste método utiliza un transductor que dispara una señal de ultrasonido que se propaga a través del organismo y se refleja hacia la fuente si encuentra objetos que interfieran en su trayectoria; de esta manera el equipo capta la presencia de órganos y los muestra en dichas imágenes.

El transmisor tiene un foco en el cual las características de la señal ultrasónica son aptas para la medida. Mientras la señal viaja entre los tejidos se encuentra con partículas que son de menor tamaño que su longitud de onda, y como consecuencia de ello, las imágenes ecográficas presentan ruido granular que le quita fidelidad a la imagen [1]. Por esta razón se hace necesaria la implementación de mecanismos para la reducción de este ruido y disminuir, de esa manera, los errores en los diagnósticos médicos.

PREGUNTAS DE INVESTIGACIÓN

¿En qué medida puede mejorarse la calidad de las imágenes de ultrasonido en SNR y CNR empleando filtros digitales adaptativos por difusión anisotrópica y por wavelet?

²<http://www.encyclopediasalud.com/definiciones/ecografia/>

HIPÓTESIS

La implementación de filtros en imágenes ultrasónicas suaviza el ruido tenido por los errores y/o problemas en la adquisición e impresión de las mismas. Es viable la implementación de un software en la plataforma MatLab para realizar filtrado digital a imágenes de ultrasonido con los métodos Adaptativo, Wavelet y por Difusión Anisotrópica y mejorar cuantitativamente los parámetros SNR y CNR de ellas.

OBJETIVOS

OBJETIVO GENERAL

Implementar tres técnicas de procesamiento de imágenes de ultrasonido, denominadas filtro Adaptativo, filtro basado en Wavelet y filtro por Difusión Anisotrópica en la plataforma MatLab, y comparar los parámetros SNR y CNR de las imágenes filtradas con las originales.

OBJETIVOS ESPECÍFICOS

- Revisar el estado del arte sobre técnicas de Procesamiento para Imágenes de Ultrasonido.
- Implementar las técnicas de filtrado denominadas Filtro Adaptativo, Filtro por Wavelet y Filtro por Difusión Anisotrópica y probar su desempeño en 20 imágenes de ultrasonido.
- Realizar un estudio estadístico comparativo de los resultados de SNR y CNR obtenidos con el filtrado.
- Realizar una interfaz gráfica para los filtros implementados.

JUSTIFICACIÓN

La adquisición de imágenes por ultrasonido tiene una mejor relación costo-beneficio en comparación con los métodos de resonancia magnética y tomografía computacional. Esto se debe a que los equipos utilizados en esas tecnologías y su mantenimiento tienen precios más elevados, además que su entorno de trabajo debe tener características especiales.

La implementación de filtros para imágenes ecográficas podría reducir la probabilidad de error en los diagnósticos médicos y ampliar las aplicaciones del ultrasonido en medicina e investigación biomédica.

ALCANCES DE LA INVESTIGACIÓN

En el desarrollo y culminación del presente proyecto se alcanzaron los siguientes logros:

- Se escogieron los tres mejores métodos de filtrado de imágenes ultrasónicas de acuerdo a la información bibliográfica recolectada.
- Con la información recolectada, se realizó la implementación de los filtros seleccionados en una interfaz gráfica hecha en el MatLab v2010a.
- Se evaluaron las imágenes filtradas y se compararon con las originales con los parámetros SNR y CNR.

2 MARCO DE REFERENCIA

2.1 Procesamiento Digital de Imágenes

El procesamiento de señales de imágenes es el desarrollo de muchas técnicas implementadas para el mejoramiento de la calidad y de las características de las imágenes, ya sea para imágenes fijas o en movimiento (videos). Esta aplicación de ingeniería utiliza conceptos físicos con fundamentos matemáticos para realizar las intervenciones y poder manipular las imágenes con las que se trabaja. [16].

Una imagen digital es una matiz numérica de dos o más dimensiones (de acuerdo a cómo se trabaje), en la cual los valores de ella representan los colores, ya sea si está en escala de grises o si se tiene combinaciones de Rojo, Verde y Azul para realizar los diferentes colores conocidos [17].

Aplicaciones del Procesamiento Digital de Imágenes

El procesamiento digital de imágenes se ha convertido hoy día en una herramienta de apoyo en muchos campos de trabajo, ya que facilita la captación y visualización de eventos que son complicados de observar a simple vista. Ejemplos de estos campos de aplicación son:

- Imágenes televisivas

Las imágenes proyectadas en la televisión han tenido un mejoramiento notorio en el transcurrir de las últimas décadas, hoy en día no se tiene el molesto parpadeo visible en la pantalla al presentar cambios bruscos en las intensidades de los colores, también se cuenta con una gama de colores más variada, desaparecieron las imágenes fantasmas y el movimiento notorio entre líneas.

- Videoconferencias

Hoy en día es posible realizar video llamadas desde computadores y hasta celulares con ayuda de internet o con servicio de plan de datos. El realizar videoconferencias tuvo un duro inicio, debido a la gran cantidad de ancho de banda requerido para el envío de las imágenes en movimiento y el sonido. Hoy en día, gracias al procesamiento de imágenes se realizan algoritmos de empaquetamiento y compresión de las mismas; lo cual favorece el envío, puesto que el equipo host (por medio de un software) es el encargado de reconstruir dichas imágenes, permitiendo un mayor y más veloz flujo de datos por la red.

- Procesamiento de Imágenes Médicas

Gracias a las nuevas tecnologías de adquisición y visualización de imágenes se puede observar fácil y claramente tumores presentes en tejidos humanos en especial los cerebrales que durante mucho tiempo fueron difícilmente captables. Uno de estos métodos, es el que entrega imágenes de ultrasonido (Ver 2.2), que junto con los avances de los filtros digitales modernos (Sección 2.3) han permitido a la comunidad médica encontrar malformaciones y lesiones en las personas desde hace décadas.

2.2 Ultrasonido

Las Señales de Ultrasonido son aquellas que se encuentran en una frecuencia por encima de los 20KHz [18], razón por la cual se le da su nombre, ya que el oído humano en promedio escucha el rango de frecuencias 20Hz - 20KHz. Este tipo de señales es utilizado en la actualidad principalmente en medicina y en la milicia.

2.2.1 Aplicaciones del Ultrasonido

Utilizar señales de ultrasonido ha sido de mucha utilidad con el transcurrir de los tiempos. Este tipo de onda posee aplicaciones en mucho de los campos de investigación existentes. A continuación se presentan algunos ejemplos:

- Dispositivos Sonar

Existen herramientas de medición de grietas en los suelos. Estos, son utilizados especialmente en construcción de vías terrestres de transporte masivo. También existen este tipo de detectores para materiales metálicos para encontrar posibles fallas en su estructura.

- Metalurgia

Se utiliza ultrasonido en operaciones de refinado de algunos granos cristalinos durante el enfriamiento de coladas fundidas industriales. Ya que estos eliminan oxido superficial de algunos materiales soldados.

- Milicia

Las aplicaciones más relevantes del ultrasonido están en labores de la milicia. Una de estas es, por ejemplo, la labor de sensado y sondeo aplicado en los submarinos militares para rastreo de posibles amenazas o posibles interferencias en el fondo del mar.

- Medicina

En el campo de la medicina, el ultrasonido se utiliza para la adquisición de imágenes y toma de muestra de órganos internos (Sección 2.2.2). Aunque existen otras aplicaciones, como por ejemplo la *Desintegración de células por ultrasonido*; método en el cual puede desintegrarse fibrosa, materias celulósicas en partículas finas y se rompen las paredes de la estructura de la célula. Este proceso es útil en algunos casos de tratamiento de enfermedades específicas.

2.2.2 Imágenes de Ultrasonido

El procesamiento digital de imágenes se ha convertido hoy día en una herramienta de apoyo en muchos aspectos. Las imágenes de ultrasonido o ultrasónicas son aquellas que se han adquirido por métodos que utilizan señales que se encuentran por encima del rango de las audibles. Este tipo de imágenes son tomadas por transductores que envían dicha señal hacia un medio establecido y reciben la energía que algún objeto o interferencia (ubicado en este canal) refleje hacia él.



Figura 1. Imagen ultrasónica simulada de un feto humano.

Parámetros de las Imágenes de Ultrasonido

Las imágenes de ultrasonido presentan características particulares. Por tal razón, pueden ser fácilmente reconocibles. A continuación se presentan algunos conceptos importantes:

- *Ruido Granular - Speckle*

El Speckle es un tipo especial de ruido que afecta en gran manera la calidad de las imágenes de ultrasonido. Éste altera la forma en que se ven los objetos en las imágenes, en especial a los pequeños detalles; por tal razón el speckle es un problema relevante en la medicina actual.

El origen de este ruido en las imágenes de ultrasonido se debe a variados errores en la adquisición de la señal ultrasónica por parte del transductor. Las causas principales son acumulación de ecos en fase generados en los centros de difusión de cada uno de ellos al momento de ser enviada la señal hacia el medio. Otros fenómenos que provocan degradación en las imágenes de ultrasonido son:

- Fenómeno de Absorción: Cuando la señal de ultrasonido va a ingresar de un medio a otro, parte de ella es reflejada (si no hay un debido acople de impedancias). La parte posterior a este reflejo resulta oscura, ya que la amplitud de las ondas ultrasónicas que ingresan al segundo medio son débiles [19].
- Fenómeno de Reflexión: Las señales después de ser enviadas son reflejadas en varias partes del medio antes de regresar al transductor [19].
- Fenómeno de Difusión: Los objetos que tienen un tamaño comparable con la longitud de la onda ultrasónica hacen que se emita en diversas partes la onda incidente. Esta es una de las causas principales de la aparición de Speckle en las imágenes adquiridas [19].

- *Brillo*

Para modificar el brillo de una imagen en escala de grises, se debe adicionar o sustraer una constante cualquiera a la matriz de los colores de la imagen. El resultado final de la variación no debe estar fuera de los límites establecidos por los bits de la escala de grises (rango 0-255).

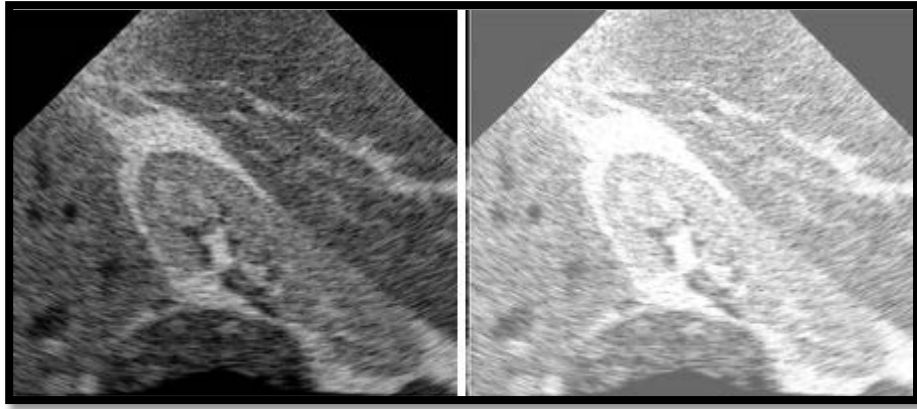


Figura 2. Variación de brillo en imagen ultrasónica simulada de riñón

- *Contraste*

Si a la recta que representa la escala de grises se le modifica la pendiente de su inclinación (se modifica a un valor diferente de 45°), se variará el contraste de la imagen debido a la inclinación que se tendrá en la escala de grises.

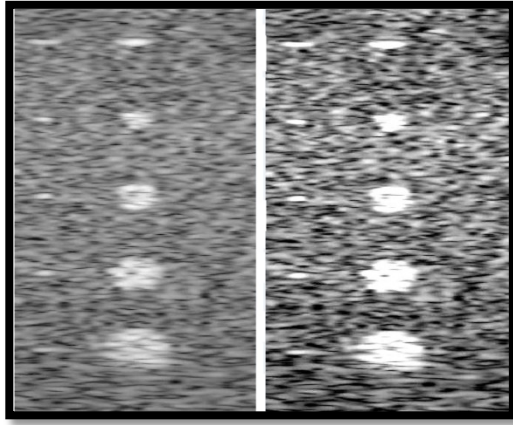


Figura 3. Variación de Contraste en Imagen ultrasónica de prototipo de lesión.

2.3 Filtros Digitales

Un filtro digital es aquel que funciona en términos de señales de tipo discreto y que han sido cuantizadas. Este tipo de filtros no admite el paso de señales cuyos parámetros no están definidos en los rangos de permisión. Se utilizan en aplicaciones en las cuales el flujo de datos es reducido y se puede captar fácilmente un ruido que esté afectando el sistema.

En imágenes, la utilización de este tipo de filtros mejora la calidad de las mismas, suavizando el ruido presente en ellas. Por esta razón este tipo de filtros es implementado en aplicaciones en donde se capturan imágenes de ultrasonido, ya que en estas es usual que se tenga ruido granular debido a falencias en la adquisición. Algunos de los filtros más importantes para imágenes de ultrasonido son:

- Filtro Adaptativo
- Filtro por Difusión Anisotrópica
- Filtro por Wavelet

2.3.1 Filtros Adaptativos

Utilizan conceptos estadísticos para suavizar el ruido granular presente en las imágenes de ultrasonido. Realiza un barrido por secciones de toda la imagen, comparando las características de cada segmento con unas supuestas ideales. Si estas están muy alejadas de las requeridas entonces se reemplaza el pixel por uno nuevo obtenido con el promedio de la ventana de estudio [1]. De esta manera se obtiene una mejora notoria en la calidad de la imagen procesada.

Se escoge manualmente la ventana de Speckle para realizar los cálculos de varianza del mismo. Figura 4.

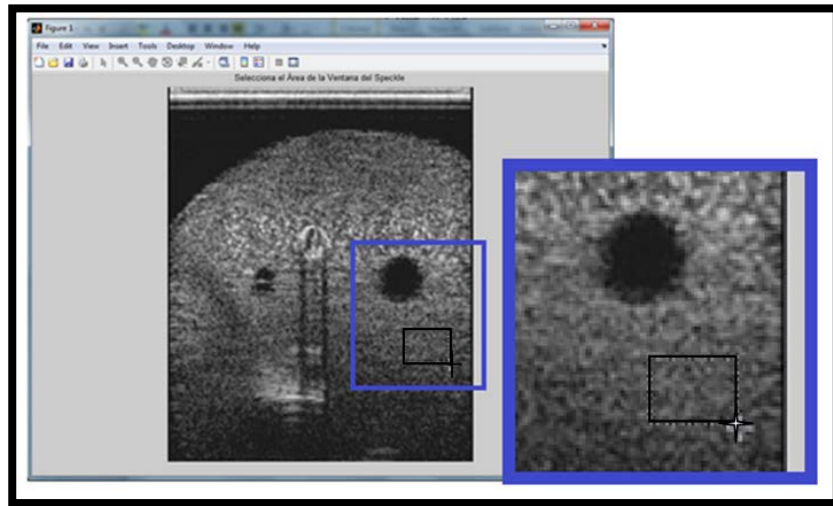


Figura 4. Selección de la Ventana de Speckle.

Se encuentra la constante Adaptativa del filtro y se reasigna el pixel de acuerdo con las expresiones de las ecuaciones (1), (2), (3) y (4). Como se muestra en la Figura 5.

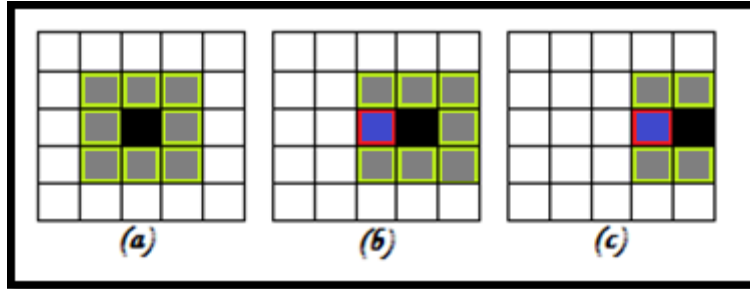


Figura 5. Secuencialidad del Filtrado de un pixel.

- (a). Ventana de área Local-Gris-, Pixel de trabajo -Negro-.
- (b). Ventana de área Local- Gris -, Pixel Anterior Filtrado-Azul-, Pixel de Trabajo -Negro-
- (c). Ventana de área Local- Gris -, Pixel Anterior Filtrado-Azul-, Pixel de Trabajo -Negro-

La expresión matemática que representa la aparición de un pixel que reemplaza a otro para realizar filtrado adaptativo varía de acuerdo con la relación de varianza del Speckle con respecto a la varianza local; a esta relación se le conoce con el nombre de Coeficiente Adaptativo [7]. El coeficiente adaptativo se calcula como se muestra a continuación:

$$k = \frac{\sigma_s^2}{\sigma_v^2} \quad (1)$$

k es el coeficiente adaptativo, σ_s^2 representa a la varianza estadística de la ventana de Speckle (seleccionada manualmente) y σ_v^2 es la varianza de la ventana de análisis.

La imagen filtrada es conseguida debido a la matriz de pixeles corregidos. Si al pixel que será reemplazado se le llama " x " y al filtrado se le llama " y ", la obtención de " y " varía de acuerdo al coeficiente adaptivo k , como sigue:

- $y = \frac{\sum_{i=1}^N x_i}{N}; k \leq 0$ (2)

En donde y reemplazará al actual pixel de estudio x ; $\sum_{i=1}^N x_i$ representa a la suma aritmética de cada uno de los N pixeles en la ventana de análisis alrededor de x , para este caso $N = 49$ (ventana de 7×7); k representa al coeficiente adaptativo.

- $y = \bar{x} + k(x - \bar{x}); \quad 0 \leq k \leq 1$ (3)

En donde y representa el pixel nuevo calculado que reemplazará al actual pixel de estudio x ; \bar{x} es el valor promedio de las intensidades de los pixeles de la ventana de filtrado establecida de 7×7 que se encuentran rodeando al pixel x ; k representa al coeficiente de filtrado adaptativo con el cual se determina la variación que tendrá el pixel x para tener suavizado y ser reemplazado por y .

- $y = x; \quad k \geq 1$ (4)

El pixel nuevo y es igual al de estudio x cuando el coeficiente adaptativo k es superior a 1.

El filtro Adaptativo reconstruye la imagen que se quiere corregir con argumentos estadísticos, con lo cual reemplaza un pixel por otro calculando la probabilidad que el pixel de estudio tenga su intensidad debida al ruido granular.

2.3.2 Filtros por Difusión Anisotrópica

Este tipo de filtros se basa en la descripción de la forma como se distribuyen las partículas de una zona de mayor concentración a otra de menor. Al igual que los

filtros adaptativos utiliza la estadística para realizar cambios por ventanas de estudio que varíen de una ideal; la diferencia entre ellos es la manera de remplazar al viejo pixel por uno nuevo. El coeficiente de difusión se calcula en términos de la variación de los pixeles y el divergente del gradiente de los mismos [1].

La expresión matemática que describe el comportamiento del filtro por difusión anisotrópica y la corrección de un pixel por uno nuevo es mostrada en la ecuación (5).

$$y = x + (1 - k).(\bar{x} - x) \quad (5)$$

Si se llama c al término $(1 - k)$ y se calcula el promedio de las cuatro ventanas directas en imagines en segunda dimensión, la expresión $(\bar{x} - x)$ puede ser remplazada por una aproximación del operador laplaciano [7]. Con lo cual la ecuación no lineal-diferencial-parcial que modela el suavizado de una imagen bidimensional en el dominio del tiempo se muestra más adelante.

Para realizar el cálculo de la difusión anisotrópica es necesario expresar el fenómeno isotrópico matemáticamente, Ecuación (6).

$$\frac{\partial x}{\partial t} = \text{div}[c * \nabla x] ; x(t = 0) = x_0 \quad (6)$$

Por lo tanto, la expresión matemática para filtrado por difusión anisotrópica está en la ecuación (7).

$$\frac{\partial x}{\partial t} = c. \text{div}[\nabla x] + \nabla c. \nabla x \quad (7)$$

En donde ∇ representa al operador de gradiente, *div* es el operador de divergencia, x_0 es la imagen inicial a la que se le va a realizar filtrado y c es el coeficiente de difusión anisotrópico que se calcula como sigue [20, 21]:

$$c^2 = \frac{\frac{1}{2}|\nabla x|^2 \cdot \frac{1}{16}(\nabla^2 x)^2}{x + \frac{1}{4}(\nabla^2 x)^2} \quad (8)$$

El filtro por difusión anisotrópica reduce el ruido granular en la imagen con argumentos matemáticos y estadísticos, teniendo en cuenta cómo se difunden partículas de speckle en zonas de mayor concentración a menor concentración.

2.3.3 Filtros por Wavelet

Utilizan la transformada Wavelet para la descomposición de las señales como una combinación lineal escalonada. La imagen se desintegra en varias de alta y baja frecuencia para realizar de esta manera un mejor filtrado, puesto se le va quitando partes a la imagen con las frecuencias no deseadas. Al final se hace una recomposición de ella con los segmentos requeridos para que la imagen esté suavizada y ausente de speckle [11].

El análisis por medio de Wavelet es utilizado especialmente para eliminar ruido debido a transitorios y variaciones temporales de las señales. La transformada wavelet también es usada para analizar la respuesta de las armónicas de la señal total. La diferencia de la transformada de Fourier es que Wavelet es capaz de realizar tanto el análisis temporal como espectral sin perder precisión [20, 22].

Actualmente existe variedad de familias para cálculo de Wavelet, ejemplo de estas son: Daubechies, Coiflets, la Symmlet, Biorntonormales, entre otras. El tipo de

Wavelet escogido para este trabajo de grado es el último mencionado; el análisis con Wavelet que permite analizar a la señal en términos temporales en aquellos segmentos de ella en donde haya componentes de baja frecuencia, además, permite analizar en términos espectrales a las componentes de la misma imagen que tengan alta frecuencia [11]; de esta manera Wavelet permite la descomposición de la imagen.

Se realiza filtrado cuando las componentes de alta frecuencia de los segmentos en los que se divide la imagen son retiradas de ella y se reconstruye dicha señal solamente con las componentes de baja frecuencia utilizando la transformada inversa de Wavelet.

La expresión matemática que permite calcular la transformada Wavelet de una señal se muestra en la ecuación (9).

$$F_W(a, b) = \int_{-\infty}^{\infty} f(t) * \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) dt \quad (9)$$

En donde $\psi(x)$ representa a la Wavelet madre – primera descomposición -. Con lo cual, $\psi\left(\frac{x-b}{a}\right)$ es el corrimiento y escalonamiento de la Wavelet madre; a es el escalonamiento de dilatación de la wavelet con $a = \frac{f}{f_0}$, siendo f_0 la frecuencia central y b el corrimiento de traslación temporal.

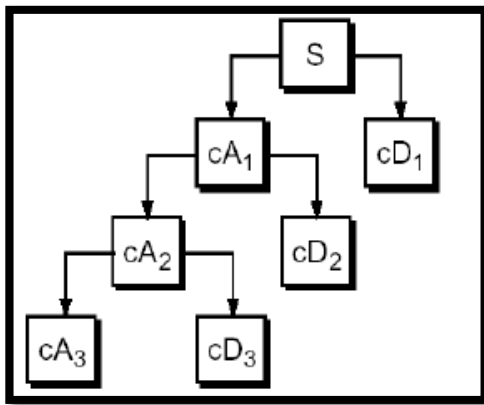


Figura 6. Diagrama de bloques de Descomposición con Wavelet.

S (Señal), Aproximaciones -A- (Baja frecuencia) y Detalles -D-(Alta frecuencia)³

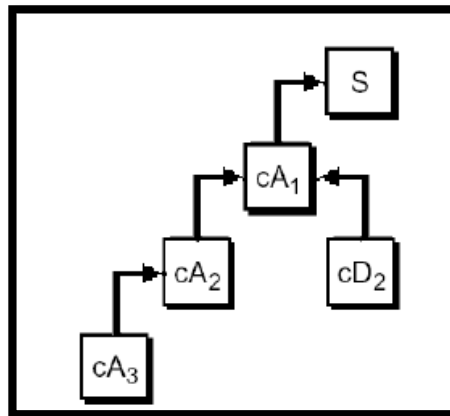


Figura 7. Diagrama de bloques de la Transformada Inversa de Wavelet.

S (Señal), Aproximaciones -A- (Baja frecuencia) y Detalles -D-(Alta frecuencia)

Los diagramas de bloques de la Figura 6 y de la Figura 7 representan esquemáticamente la desintegración y restauración de señales con la teoría Wavelet. En ellas se ve cómo la señal S es descompuesta en Aproximaciones (baja frecuencia) y Detalles (alta frecuencia) que van siendo removidas de ella

³ Kuoro, S y Musalem, R. (2002). *Tutorial Introductorio a la Teoría Wavelet*

para realizar filtrado. Posteriormente se arma nuevamente la señal con las señales que se quiere dejar y se tienen una imagen filtrada.

2.4 Evaluación de la Calidad de una Imagen

El análisis cuantitativo de los resultados obtenidos por parte del filtrado de imágenes es relevante si se pretende conocer en qué medida están siendo corregidas las mismas. Éste sirve para tener una visión general del comportamiento de los filtros y las limitaciones que presentan.

Hoy día, se utiliza un gran número de parámetros para evaluaciones métricas de la calidad de las imágenes.

Ejemplo de estos son:

- Error Medio Cuadrático.
- Raíz del Error Medio Cuadrático.
- Error de Suma en la forma métrica de Minkowski.
- Error Medio Geométrico.
- Relación Señal a Ruido.
- Relación Señal a Ruido Pico.
- Índice de Calidad.
- Índice de Similitud Estructural.
- Relación Contraste Ruido

En este proyecto se evaluaron las imágenes con los parámetros SNR y CNR debido a que el objetivo planteado es conocer en qué medida los filtros eliminan ruido granular y cómo se ve afectado el contraste de las imágenes al éstas ser filtradas. Dichos parámetros se calculan como sigue:

- **Relación Señal a Ruido:**

$$SNR = \frac{S_v}{N_s} \quad (10)$$

$$S_v = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (x_{i,j}) \quad (11)$$

$$N_s = \sigma_s \quad (12)$$

En donde, SNR es la Relación Señal a Ruido, S_v es una porción de imagen de señal (órgano, quiste, objeto de estudio), M es la cantidad de Filas de ventana de Señal, N son las columnas de la ventana de la señal, $x_{i,j}$ representa a cada uno de los pixeles ubicados dentro de la ventana de señal en la fila i y la columna j de la imagen, N_s es el ruido de una ventana de Speckle y σ_s es la varianza de la ventana de Speckle [20, 23, 24, 25].

- **Relación Contraste Ruido:**

$$CNR = \frac{|\bar{x} - \bar{y}|}{\sqrt{0.5(\sigma_x^2 + \sigma_y^2)}} \quad (13)$$

En donde, CNR es la Relación Señal a Ruido, \bar{x} y \bar{y} representan los valores promedios de los pixel de señal original y filtrado respectivamente. σ_x^2 y σ_y^2 son las varianzas de las ventanas de Speckle original y filtrada respectivamente [4, 26, 27].

3 METODOLOGÍA

- Revisión bibliográfica: Se realizó la investigación acerca de los avances realizados en Filtrado de imágenes ultrasónicas. Con lo cual, se determinó los métodos más sobresalientes en dicha área.
- Selección de los diseños reportados en la literatura científica: Se seleccionaron los tres mejores métodos de filtrado ultrasónico obtenidos de la investigación, teniendo en cuenta, además, si era viable su implementación bajo las condiciones establecidas. Los filtros escogidos fueron Adaptativo, por Wavelet y por Difusión Anisotrópica.
- Implementación de los Filtros Seleccionados: Con la información recolectada, se procedió a efectuar la ejecución de los filtros escogidos. Dicha implementación se realizó con códigos de archivos *.m (MatLab[®]v.2010a)
- Ensayos de Filtrado: Se puso a prueba los filtros realizados y se compararon los resultados obtenidos. El análisis matemático de pruebas se hizo teniendo en cuenta la SNR y la CNR.
- Interfaz Gráfica: Una vez concluidas la construcción y prueba de los filtros, se hizo el diseño de una interfaz gráfica (GUIDE) para un mejor manejo por parte de los usuarios. Dicha interfaz se implementó con la herramienta GUI del Software MatLab[®]v.2010a.

Al término de la etapa de elección de los filtros se procedió a realizar la implementación de los mismos. Los filtros escogidos son: Filtro Adaptativo (Ver 2.3.1), Filtro por Difusión Anisotrópica (Ver 2.3.2) y Filtro por Wavelet (Ver 2.3.3).

El algoritmo del filtro adaptativo implementado fue adoptado del desarrollado por Sonia Contreras, PhD. [4]. El filtro por difusión Anisotrópica implementado es el desarrollado por Christos Loizou [28]. El filtro por Wavelet se realizó con los comandos Wavelet de MatLab R2010a [29].

Los filtros implementados tienen diferentes métodos empleados para la disminución del Speckle. A continuación se presenta el resumen del funcionamiento de los mismos con sus respectivos diagramas de flujo:

3.1 Filtro Adaptativo:

Se implementa un código que siga la siguiente secuencia de pasos y cuyo diagrama de flujo se muestra en la Figura 8.

- i. Carga la Imagen ultrasónica que se desea filtrar.
- ii. Se asegura que MatLab tome la imagen como una matriz dos dimensiones.
- iii. Normaliza los valores de los pixeles y los transforma en tipo Double.
- iv. Se especifica la región de Speckle para dar una referencia al filtrado.
- v. Se calcula la varianza del Speckle.
- vi. Empieza el recorrido desde la parte izquierda superior de la imagen hasta la parte inferior derecha.
- vii. Se calcula y reemplaza el pixel medio de la ventana de análisis de acuerdo con las ecuaciones (2), (3) y (4) según sea el caso.
- viii. Se imprime la imagen original y la filtrada.

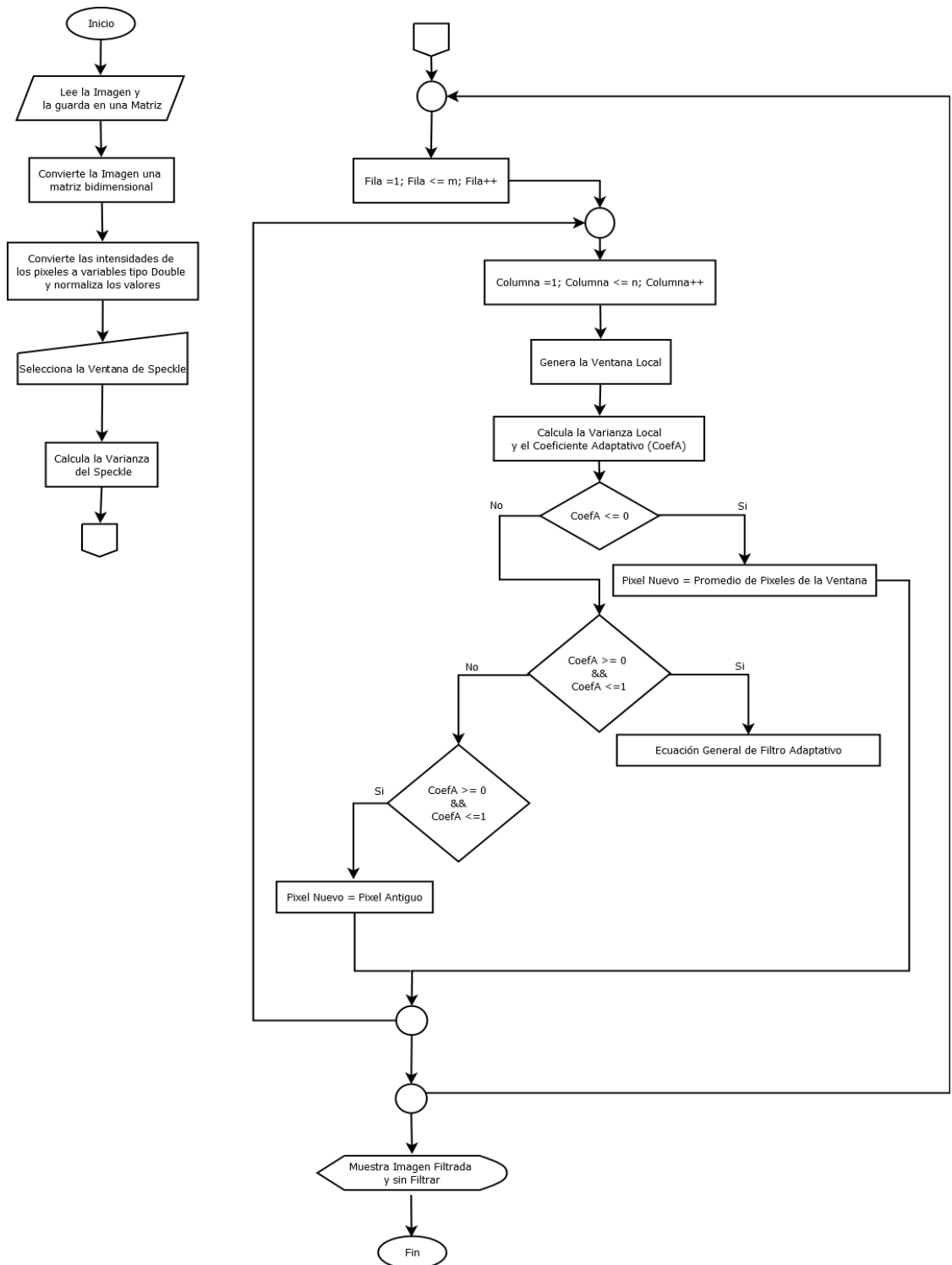


Figura 8. Diagrama de Flujo de Algoritmo para Filtro Adaptativo.

3.2 Filtro por Difusión Anisotrópica:

Se implementa un código que siga la siguiente secuencia de pasos y cuyo diagrama de flujo se muestra en la Figura 9.

- i. Carga la Imagen ultrasónica que se desea filtrar.
- ii. Se asegura que MatLab tome la imagen como una matriz dos dimensiones.
- iii. Normaliza los valores de los pixeles y los transforma en tipo Double.
- iv. Empieza el recorrido desde la parte izquierda superior de la imagen hasta la parte inferior derecha con una ventana de análisis de 3x3.
- v. Se calcula y reemplaza el pixel medio de la venta de análisis de acuerdo con las ecuación (6), (7) y (8).
- vi. Se imprime la imagen original y la filtrada.

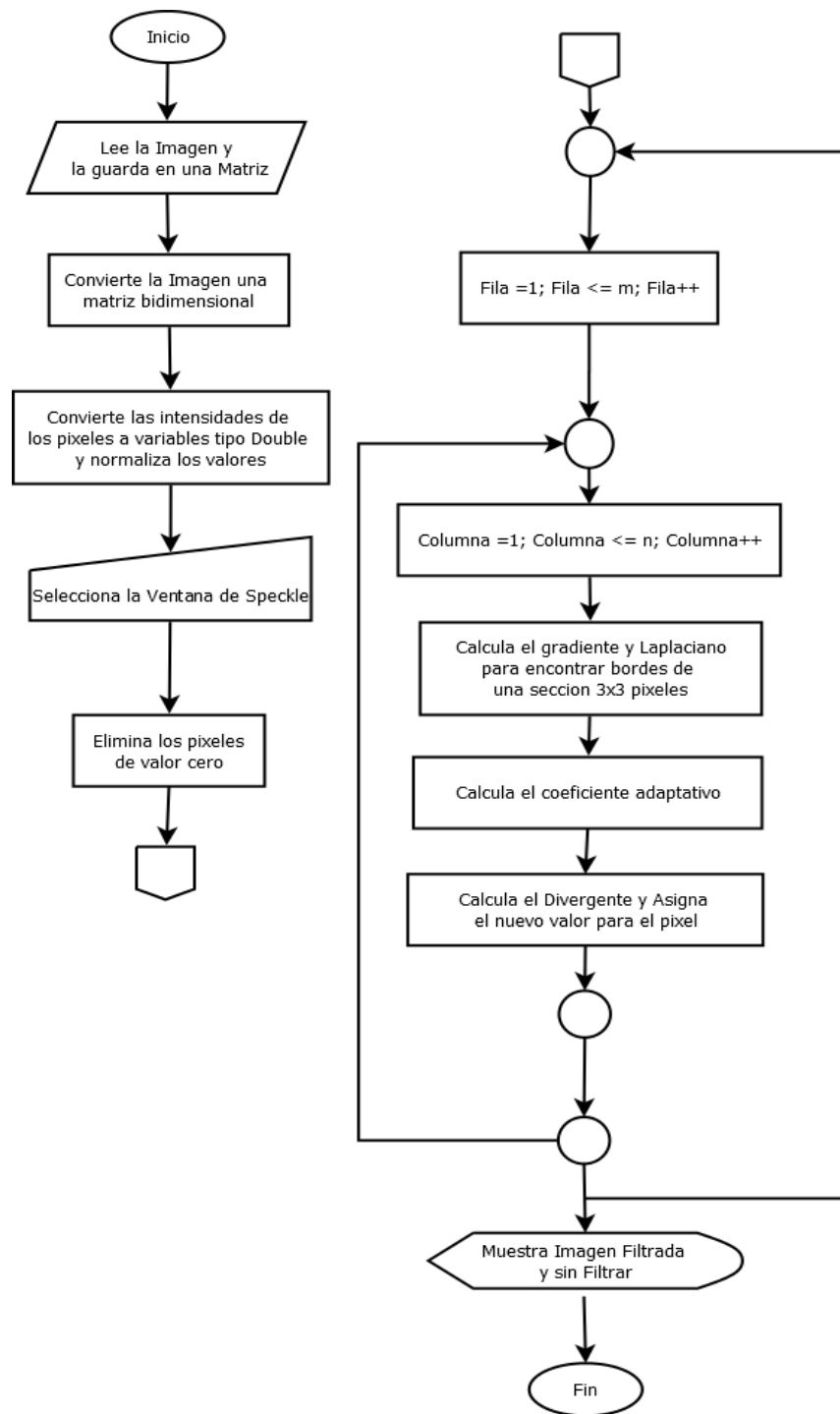


Figura 9. Diagrama de Flujo de Algoritmo para Filtro por Difusión Anisotrópica.

3.3 Filtro por Wavelet:

Se implementa un código que siga la siguiente secuencia de pasos y cuyo diagrama de flujo se muestra en la Figura 10.

- i. Carga la Imagen ultrasónica que se desea filtrar.
- ii. Se asegura que MatLab tome la imagen como una matriz dos dimensiones.
- iii. Normaliza los valores de los pixeles y los transforma en tipo Double.
- iv. Realiza las descomposiciones por Wavelet requeridas utilizando la transformada Wavelet. Ecuación (9).
- v. Realiza la reconstrucción de la imagen sumando las descomposiciones Wavelet de baja frecuencia.
- vi. Imprime la imagen original y la imagen filtrada.

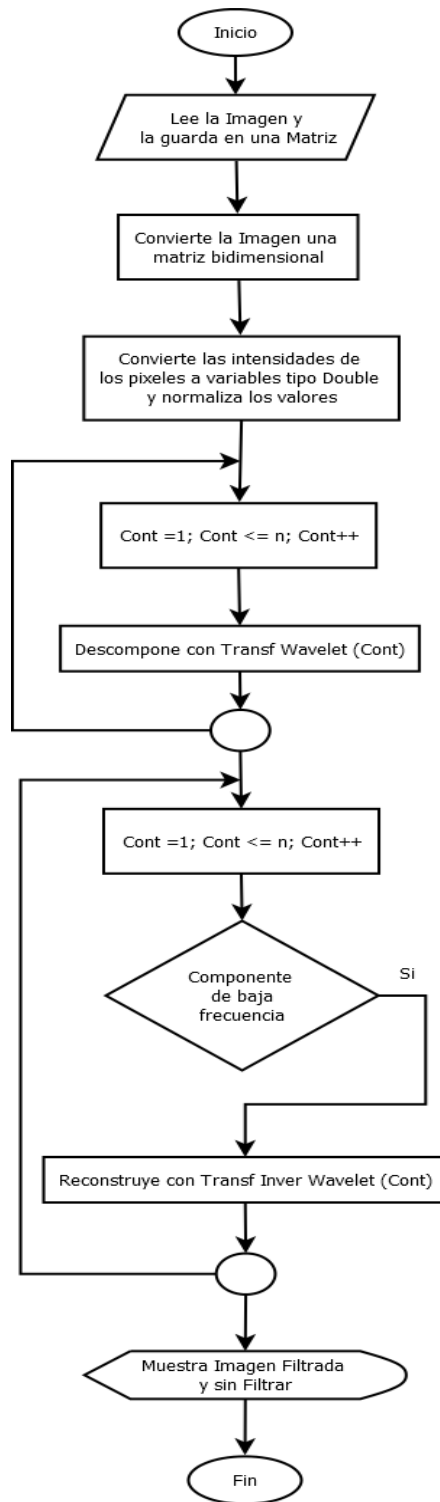


Figura 10. Diagrama de Flujo de Algoritmo para Filtro por transformada Wavelet.

Se realizó una interfaz gráfica en MatLab v2010a en la cual se escoge la imagen que se desea filtrar, se selecciona el tipo de filtro que se desea realizar y entrega las imágenes original y filtrada.

Las imágenes que se dispongan a filtrar se pueden seleccionar desde la base de datos de la interfaz o pueden ser cargadas desde una ubicación cualquiera del computador del usuario. Dichas imágenes deben estar en formatos Archivo de Mapa de Bits (*.bmp); JPEG (*.jpg;*.jpeg;*.jpe;*.jfif); GIF (*.gif); TIFF (*.tiff;*.tif) o PNG (*.png) para que sean reconocidas por la *Guide*. La interfaz se muestra en la **Figura 11**.

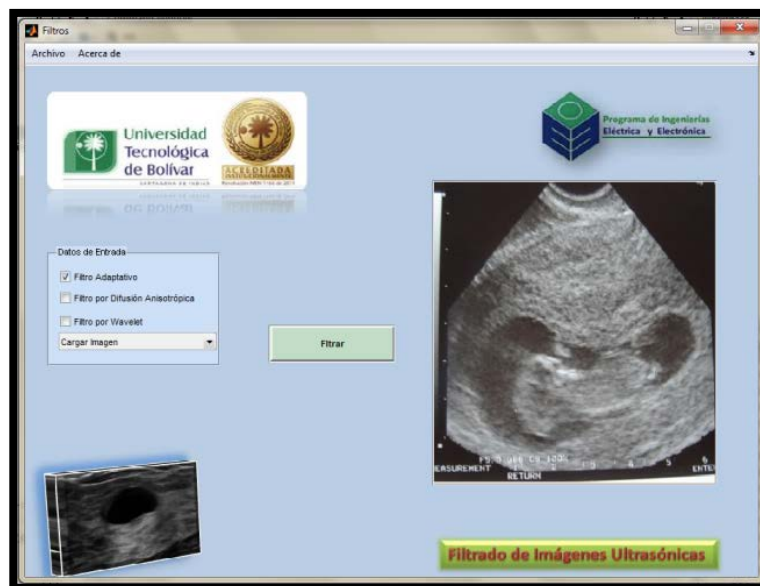


Figura 11. Interfaz gráfica para Filtrado de Imágenes Ultrasónicas

Se escoge la imagen desde el **menú Archivo** o desde el menú desplegable **Cargar Imagen** que se encuentra en la sección **Datos de Entrada**. Se elige el tipo de filtro que se desea implementar y se presiona el botón **Filtrar**.

Se debe garantizar que MatLab trabaje con las imágenes como Matrices de dos dimensiones, puesto que es usual que para dibujos de tipo JPEG los desintegre en más de dos dimensiones. Esto se hace con el comando

$$K = K(:, :, 1);$$

Siendo K la matriz en la cual se ha guardado la Imagen.

El proceso de corrección de las imágenes varía de acuerdo al tipo de filtro seleccionado. Las técnicas van ligadas a las expresiones matemáticas de cada uno de ellos.

El documento adjunto, [Guía de Usuario.pdf](#), entrega la metodología que se debe utilizar para el buen uso del programa, así como también la información básica para su correcta instalación.

El código implementado para la realización de la interfaz gráfica y los filtros se encuentra en los [Anexos](#).

La evaluación del desempeño de los filtros se hizo teniendo en cuenta los parámetros SNR y CNR (Sección 2.4). Con lo cual se verificó matemáticamente mejora en la relación entre Señal a Ruido y Contraste Ruido de la imagen filtrada con respecto a la original. Se hizo con la intención de ver la mejora de la calidad de las señales con respecto al ruido que presentan en las imágenes ultrasónicas.

4 RESULTADOS Y DISCUSIÓN

Se realizó filtrado a las veintidós imágenes ultrasónicas que se encuentran en la base de datos de la interfaz y se calculó los parámetros SNR y CNR a las diez que representan quistes u órganos humanos (resultados se muestran en los [Anexos](#)) y se pudo establecer claramente qué filtro arroja mejores resultados.

En cada imagen tomó cinco muestras de las relaciones SNR y CNR, con lo cual se calculó el valor promedio de las muestras junto con sus desviaciones estándar. Esto se hizo para las imágenes originales y las filtradas con los diferentes métodos (Ver tablas en [Anexos](#)).

La Figura 12 muestra las imágenes resultantes al realizar los filtros mencionados a una imagen real de quiste mamario. En ella se observa claramente que el filtro que realiza mayor suavizado es el Filtro por Difusión Anisotrópica, seguido por el Filtro Adaptativo y por último en Filtro por Wavelet.

En las Tabla 1, Tabla 2, Tabla 3 se observan los resultados de los cálculos de las SNR y CNR obtenidos con la realización de los Filtros Adaptativos, por Difusión Anisotrópica y Wavelet respectivamente. Para este caso, el filtro adaptativo entregó una mejora en la SNR del 321% y 256% de CNR; el filtro por difusión anisotrópica arroja resultados mejorados de 511% en SNR y 466% en CNR y por último el filtro wavelet entregó mejoras de 206% y 196% para SNR y CNR respectivamente.

En las Gráfica 1 y Gráfica 2 se observa proporcionalmente las mejoras en los parámetros evaluados de los filtros. En ellas se evidencia el predominio del Filtro por Difusión Anisotrópica en cuanto al suavizado de ruido granular.

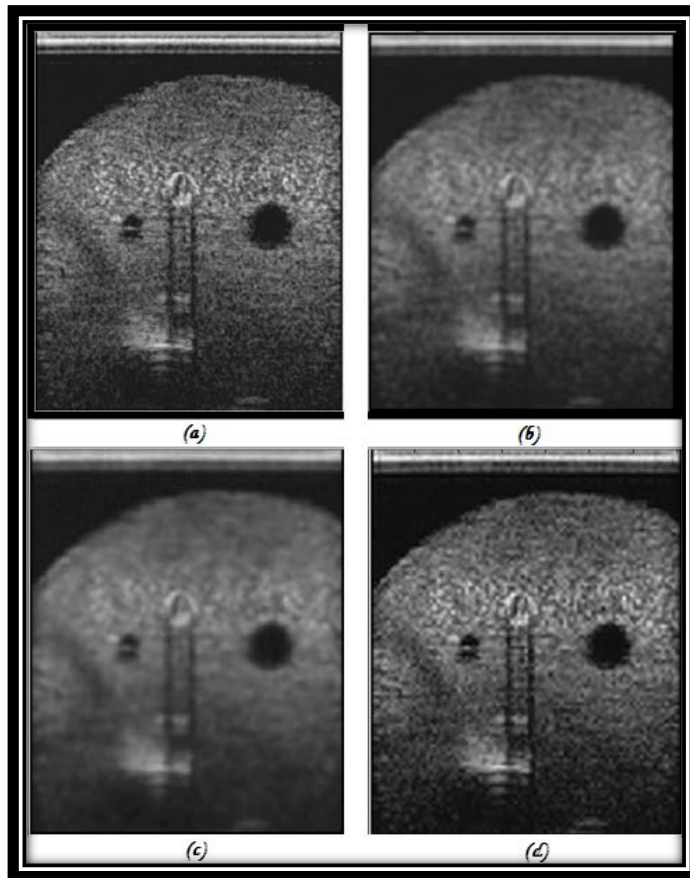


Figura 12. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #1.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
 (c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 1. SNR y CNR para Quiste Mamario. Ejemplo#1. Filtro Adaptativo.

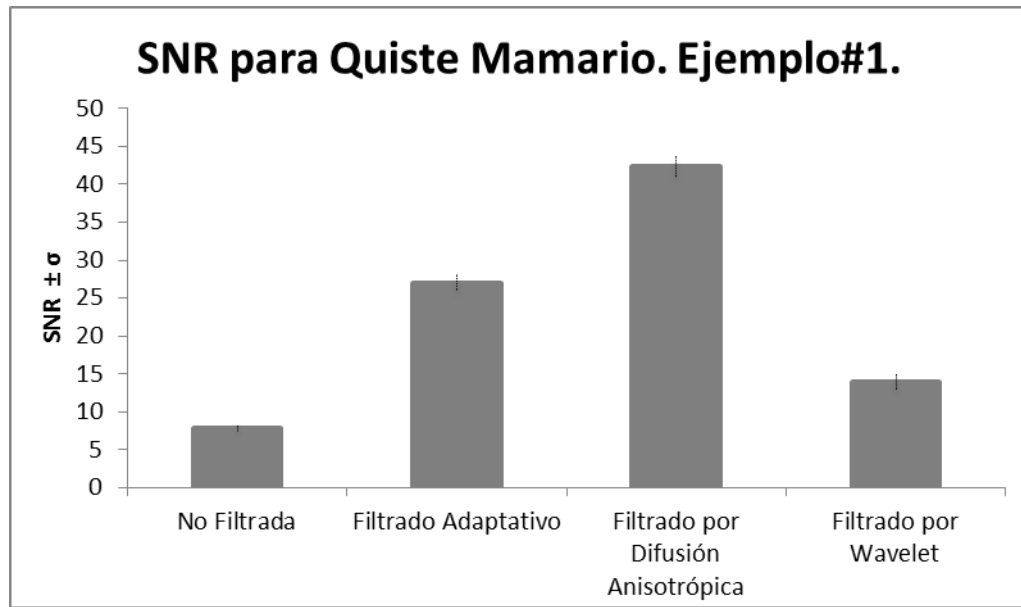
		Original		Filtrada	
MUESTRA		SNR	CNR	SNR	CNR
Ejemplo#1. Quiste Mamario. Filtro Adaptativo	M1	8,5131	4,4608	27,8424	11,6544
	M2	8,3607	4,2972	25,9969	9,8934
	M3	8,8245	4,7506	27,8975	12,9359
	M4	8,3396	4,1249	26,3045	9,0092
	M5	8,0544	3,7188	27,0717	11,2135
	PROMEDIO	8,41846	4,27046	27,0226	10,94128
	σ	0,28103381	0,38503092	0,86713046	1,5323298

Tabla 2. SNR y CNR para Quiste Mamario. Ejemplo#1. Filtro por Difusión Anisotrópica.

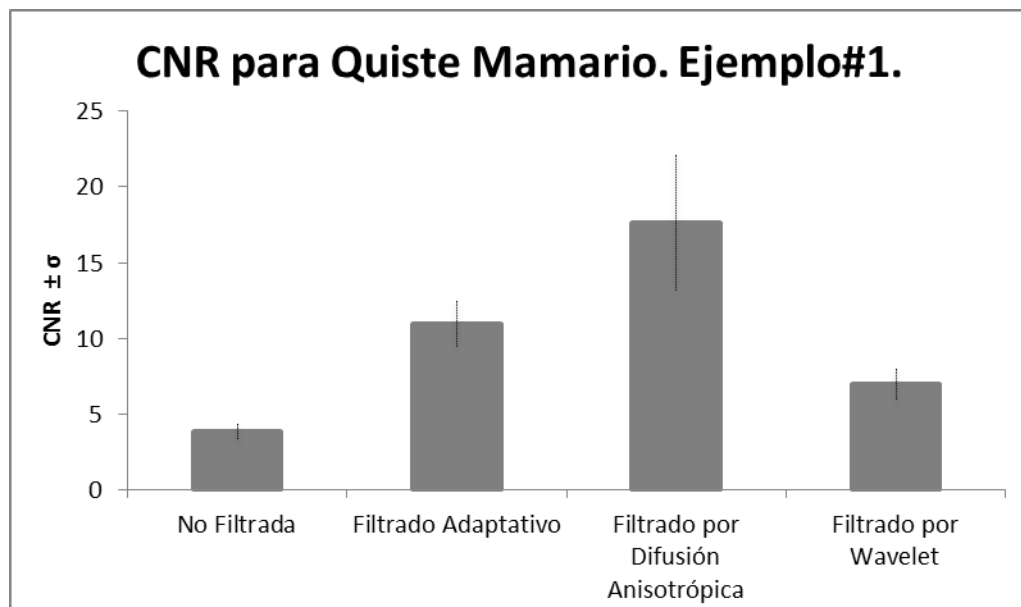
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#1. Quiste Mamario. Filtro por Difusión Anisotrópica	M1	8,5273	4,3272	43,6345	18,6454
	M2	7,9782	3,3853	40,7395	15,2913
	M3	8,7542	4,1715	43,5521	23,8313
	M4	8,0278	3,5811	41,0953	11,7338
	M5	8,0803	3,4725	42,3387	18,6646
	PROMEDIO	8,27356	3,78752	42,27202	17,63328
	σ	0,3465486	0,43079081	1,34468307	4,49422391

Tabla 3. SNR y CNR para Quiste Mamario. Ejemplo#1. Filtro por Wavelet.

	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#1. Quiste Mamario. Filtro por Wavelet	M1	7,3122	4,3013	15,0477	8,605
	M2	6,198	3,0567	12,9565	6,6608
	M3	7,0571	3,6182	14,3232	6,283
	M4	6,419	3,1508	12,9903	6,1838
	M5	6,8834	3,6434	14,3832	7,1789
	PROMEDIO	6,77394	3,55408	13,94018	6,9823
	σ	0,45814141	0,49508807	0,9272904	0,98764769



Gráfica 1. Comparación de SNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#1.



Gráfica 2. Comparación de CNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#1.

- Las gráficas anexadas demuestran que el filtro por Difusión Anisotrópica es que mejor desempeño tiene, ya que en todas las pruebas realizadas éste es que mejor entrega SNR y CNR. Éste entregó mejoras que van desde 320 a 752% en SNR y 193 a 557% en CNR.
- El segundo filtro con mejor trabajo es el Adaptativo con un rango que está de 216 a 389% en SNR y 166 a 698% en CNR.
- El filtro con el desempeño menos óptimo es el Wavelet que entregó mejoras en la calidad de la imagen que van del rango 133 a 247% y 139 a 195%.
- En todos los casos de filtrado se superó a las relaciones SNR y CNR de las imágenes sin filtrar de acuerdo a las gráficas mostradas con anterioridad.
- Se realizó una confrontación matemática con diez de las imágenes no filtradas y filtradas de acuerdo al tipo de filtro y se determinó que independientemente de la forma de adquisición de la imagen, el filtro que mejor se desempeña en la suavización de Speckle es el Filtro por Difusión Anisotrópica. No obstante, este filtro presentó en muchos casos variaciones notables de valores de SNR y CNR con respecto a sus medias, esto se debió a los errores presentes en los bordes de las estructuras.
- Se colocan las imágenes en los anexos en los cuales se puede observar el suavizado del ruido granular de las imágenes filtradas versus las originales.

5 CONCLUSIONES y RECOMENDACIONES

- Con la literatura consultada y las ecuaciones desarrolladas se logró construir los filtros de imágenes ultrasónicas propuestos en el presente trabajo.
- Se realizó el filtro Adaptativo implementando un código que editara pixel a pixel cada uno de los segmentos de la imagen y suavizara el ruido de las imágenes con argumentos estadísticos.
- Se efectuó un filtro por Wavelet utilizando la herramienta Toolbox Wavelet de MatLab con el cual se utiliza la transformada Wavelet para realizar dos aproximaciones a la imagen. Posteriormente se realiza la transformada inversa de Wavelet y se reconstruye la imagen sin los detalles queridos.
- Se hizo un filtro por Difusión Anisotrópica que suavizara el ruido granular con argumentos estadísticos así como también con la resolución de una ecuación diferencial parcial para el cálculo de sus argumentos.
- Los filtros implementados degradaron satisfactoriamente el Speckle. A pesar de que hubo diferencias en los resultados de SNR y CNR para cada filtro, todos ellos mejoraron dichos parámetros con respecto a las imágenes originales.
- El filtro que mejores resultados entregó es el filtro por Difusión Anisotrópica, seguido del filtro Adaptativo y por último el Wavelet en su segunda aproximación.
- Se efectuó una interfaz gráfica que permite leer imágenes ultrasónicas y aplica el tipo de filtro seleccionado.

- Para futuros trabajos de investigación, se recomienda aumentar el número de ensayos, realizando réplicas que permitan profundizar el análisis en imágenes no ultrasónicas.
- Se sugiere emplear una etapa de preprocesamiento de las imágenes para tener un ajuste más preciso la disminución de Speckle.
- Se invita a realizar más de dos aproximaciones Wavelet para ese tipo de filtros y observar cuánto mejora su filtrado.
- Se recomienda evaluar la degradación de Speckle empleando otros métodos de evaluación de la calidad de las imágenes, como por ejemplo MSE, RMSE, Err, GAE, PSNR, Q index, SSIN, C index, entre otros.
- En trabajos posteriores, se puede comparar los filtros Adaptativos, por Difusión Anisotrópica y Wavelet implementados en este proyecto con otros existentes de uso no exclusivo en imágenes ultrasónicas.

6 BIBLIOGRAFÍA

- [1] S. H. Contreras Ortíz, T. Chiu y M. D. Fox, «Ultrasound image enhancement: A review,» *Biomedical Signal Processing and Control*, vol. 7, nº 5, pp. 419-428, 2012.
- [2] J. C. Bamber y C. Daft, «Adaptive filtering for reduction of speckle in ultrasonic pulse-echo images,» *Ultrasonics*, vol. 24, nº 1, pp. 41-44, 1986.
- [3] V. Dutt y J. F. Greenleaf, «Adaptive Speckle Reduction Filter for Log-Compressed B-Scan Images,» *Medical Imaging, IEEE Transactions on*, vol. 15, nº 6, pp. 802-813, 1996.
- [4] S. H. Contreras Ortíz, T. Chiu y M. D. Fox, «Hexagonal adaptive filtering on compound ultrasound images,» de *Annual International Conference of the IEEE*, 2011.
- [5] P. Perona y J. Malik, «Scale-Space and Edge Detection Using Anisotropic Diffusion,» *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 12, nº 7, pp. 629- 639 , 1990.
- [6] Y. Yu y S. T. Acton, «Speckle Reducing Anisotropic Diffusion,» *Image Processing, IEEE Transactions on*, vol. 11, nº 11, pp. 1260- 1270 , 2002.
- [7] K. Krissian, C.-F. Westin, R. Kikinis y K. G. Vosburgh, «Oriented Speckle Reducing Anisotropic Diffusion,» *Image Processing, IEEE Transactions on*, vol. 16, nº 5, pp. 1412- 1424 , 2007.
- [8] H. Guo, J. E. Odegard, M. Lang, R. A. Gopinath, I. W. Selesnick y C. S. Burrus, «Wavelet based speckle reduction with application to SAR based

- ATD/R,» *Image Processing, IEEE* , vol. 1, nº 1, pp. 75-79, 1994.
- [9] A. Bruce, D. Donoho y H. Y. Gao, «Wavelet analysis,» *Spectrum, IEEE*, vol. 33, nº 10, pp. 26-35, 1996.
- [10] X. Zong, A. F. Laine y E. A. Geiser, «Speckle reduction and contrast enhancement of echocardiograms via multiscale nonlinear processing,» *Medical Imaging, IEEE Transactions on*, vol. 17, nº 4, pp. 532-540, 1998.
- [11] S. Kouro R. y R. Musalem M., «Tutorial introductorio a la Teoría de Wavelet,» *Técnicas Modernas en Automática*, 2002.
- [12] S. Gupta, R. C. Chauhan y S. C. Sexana, «Wavelet-based statistical approach for speckle reduction in medical ultrasound images,» *Medical and Biological Engineering and Computing*, vol. 42, nº 2, pp. 189-192, 2004.
- [13] S. G. Chang, B. Yu y M. Vetterli, «Adaptive wavelet thresholding for image denoising and compression,» *Image Processing, IEEE Transactions on*, vol. 9, nº 9, pp. 1532-1546, 2000.
- [14] Y. Yue, M. M. Croitoru, A. Bidani, J. B. Zwischenberger y J. W. Clark, «Nonlinear multiscale wavelet diffusion for speckle suppression and edge enhancement in ultrasound images,» *Medical Imaging, IEEE Transactions on*, vol. 25, nº 3, pp. 297-311, 2006.
- [15] S. Finn, M. Glavin y E. Jones, «Echocardiographic Speckle Reduction,» *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, vol. 58, nº 1, pp. 82-101, 2011.
- [16] B. Escalante Ramirez, «Procesamiento Digital de Imágenes,» 2006.

- [17] R. C. Gonzalez y P. Woods, Digital image processing, Prentice-Hall, 2002.
- [18] J. Nuez, J. M. Benitez, L. Falero y P. Guerra, «Las Ondas y el Sonido,» [En línea]. Available: <https://sites.google.com/site/lasondasyelsonido/efectos-sonoros/ultrasonidos-e-infrasonidos>. [Último acceso: 19 Noviembre 2012].
- [19] M. Bianculli y A. Duffour, «Imágenes de Ultrasonido,» Montevideo, 2005.
- [20] C. P. Loizou, C. S. Pattichis, C. I. Christodoulou, R. S. H. Istepanian, M. Pantziaris y A. Nicolaidis, «Comparative evaluation of despeckle filtering in ultrasound imaging of the carotid artery,» *Ultrasonics, Ferroelectrics and Frequency Control, IEEE Transactions on*, vol. 52, nº 10, pp. 1653-1669, 2005.
- [21] Y. Yongjian y S. T. Acton, «Speckle reducing anisotropic diffusion,» *IEEE Trans. Image Process*, vol. 11, nº 11, pp. 1260-1270, 2002.
- [22] D. L. Donoho, «Denoising by soft thresholding,» *IEEE Trans. Inform theory*, vol. 41, pp. 613-627, 1995.
- [23] M. Noda, E. González y C. Cabal, «Análisis de las relaciones señal-ruido y contraste - ruido en imágenes de resonancia magnética a tres intensidades de campo magnético,» *Revista Cubana de Física*, vol. 26, nº 2A, pp. 144-149, 2009.
- [24] D. Sakrison, «On the role of observer and a distortion measure in image transmission,» *IEEE Trans. Comm*, vol. 25, pp. 1251-1267, November 1977.

- [25] J. C. Gatenby, J. C. Hoddinott y S. Leeman, «Phasing out speckle,» *Physics in Medicine and Biology*, vol. 34, pp. 1683-1689, 1989.
- [26] V. Dutt, «Statistical analysis of ultrasound echo envelope,» de *PhD. dissertation, Mayo Graduate School, Rochester, MN, 1995.*
- [27] J. C. Dainty, *Laser speckle and related phenomena*, Berlin: Springer-Verlag, 1974.
- [28] C. P. Loizou y C. S. Pattichis, *Despeckle Filtering Algorithms and Software for Ultrasounds Imagins*, Morgan & Claypool Publishers, 2008.
- [29] M. Misiti, Y. Misiti, G. Oppenheim y J. M. Poggi, «Wavelet Toolbox. User's Guide,» MatLab, 2012.
- [30] J. V. Barchiesi, *Introducción al Procesamiento Digital de Señales*, Valparaíso – Chile: Ediciones Universitarias de Valparaíso. Pontificia Universidad Católica de Valparaíso, 2008.
- [31] A. V. Oppenheim y A. S. Willsky, *Señales y Sistemas*, Cambridge, Massachusetts, 1998.
- [32] Misiti, Michael; Misiti, Yves; Oppenheim, Georges; Poggi, Jean Michael;, *Wavelet Toolbox™ User's Guide*, 2012.
- [33] J. Arendt Jensen, *Linear description of ultrasound imaging systems*, Lyngby, Dinamarca, 1999.
- [34] J. Arendt Jensen, *Users' guide for the Field II program*, Lyngby, Dinamarca, 2011.
- [35] «Aprenda MatLab com si estuiera en primero».

- [36] M. V. Lapuerta González , *Introducción a MatLab*, 2007.
- [37] Rao, Min; Varghese, Tomy; Zagzebski, James A.; «Simulation of ultrasound two-dimensional array transducers using a frequency domain model,» *Medical Physics, The University of Wisconsin—Madison, 1300 University Avenue, 1530 MSC, Madison, Wisconsin 53706*, vol. 35, nº 7, 2008.
- [38] A. García, *Procesamiento y análisis de imágenes digitales II*, Centro de Estudios Moleculares de la Célula, FONDAP, Facultad de Medicina, Universidad de Chile.
- [39] R. Nava, A. Gallego y G. Cristóbal, «Una nueva herramienta para la evaluación de la calidad perceptible en imágenes».
- [40] A. R. Gallego Aguilar, *Modelos Visuales en el Análisis de la Calidad de Imagen*. Proyecto de Fin de Carrera, Madrid: Universidad de Madrid. Escuela Técnica Superior de Ingenieros de Telecomunicación, Febrero 2006.
- [41] H. Y. Gao, «Wavelet Shrinkage Denoising Using the Non-Negative Garrote,» Seattl, WA 98109, USA, 1997.
- [42] I. The MathWorks, «Mathworks. Accelerating the pace of engineering and science,» MatLab, 1994-2012 . [En línea]. Available: <http://www.mathworks.com>. [Último acceso: 14 Diciembre 2012].
- [43] Universidad de Valladolid, «Departamento de Matemática Aplicada,» [En línea]. Available: http://www.ma.uva.es/~antonio/Industriales/Apuntes_05-06/LabM/MatLab-en-pantalla.pdf. [Último acceso: 14 Diciembre 2012].

- [44] «LWP - La Web del Programador -», 2000-2012. [En línea]. Available: <http://www.lawebdelprogramador.com>. [Último acceso: 14 Diciembre 2012].
- [45] «LiveJournal,» LiveJournal, Inc., 1999. [En línea]. Available: <http://nibotlab.livejournal.com>. [Último acceso: 14 Diciembre 2012].
- [46] I. e. A. y. C. Industrial, «Ingeniería en Automatización y Control Industrial,» Departamento de Ciencia y Tecnología, [En línea]. Available: http://iaci.unq.edu.ar/iaci_2.htm. [Último acceso: 18 Diciembre 2012].
- [47] E. d. V. Cuenca, *MATLAB: Introducción al procesamiento de imágenes - Visión por Computador -*.
- [48] T. J. Chen, K. S. Chuang, J. Wu, S. C. Chen, I. M. Hwang y M. L. Jan, «A novel Image quality index using Moran I Statistics,» *Phys. Med. Biol.*, vol. 48, pp. 131-137, 2003.
- [49] S. Winkler, «Vision models and quality metrics for image processing applications,» PhD Dissertation , University of Lausanne , Lausanne, Switzerland, December 21, 2000.
- [50] R. Gonzalez y R. Woods, *Digital image processing*, MA, USA: Addison-Wesley Longman Publish Co. 2 Ed, 2002.
- [51] Z. Wang, A. Bovik, H. Sheikh y E. Simoncelli, «Image quality assessment: From error measurement to structural similarity,» *IEEE Trans. Image Process*, vol. 13, nº 4, pp. 600-612, 2004.
- [52] Z. Wang y A. Bovik, «An universal quality index,» *IEEE Signal Process*, vol. 9, nº 3, pp. 81-84, 2002.

- [53] O. Asaustre García y C. Gómez Barba, Escritores, *Compresión con Wavelet*. [Performance]. 2002.
- [54] D. O. Barragán Guerrero, «MANUAL DE INTERFAZ GRÁFICA DE USUARIO EN MATLAB,» Matpic, 2008. [En línea]. Available: www.matpic.com. [Último acceso: 6 Enero 2013].
- [55] P. Christensson, «TechTerms.com,» 2013. [En línea]. Available: <http://www.techterms.com/>. [Último acceso: 16 Marzo 2013].

ANEXOS

GRÁFICAS Y TABLAS

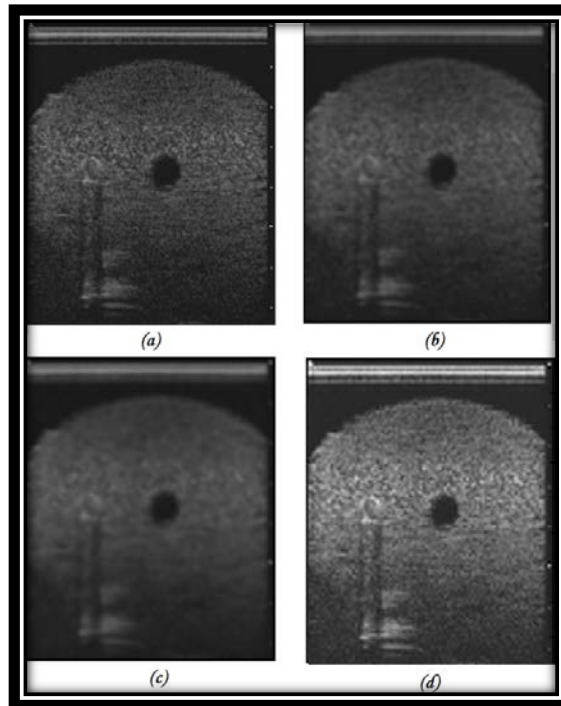


Figura 13. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #2.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
 (c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 4. SNR y CNR para Quiste Mamario. Ejemplo#2. Filtro Adaptativo.

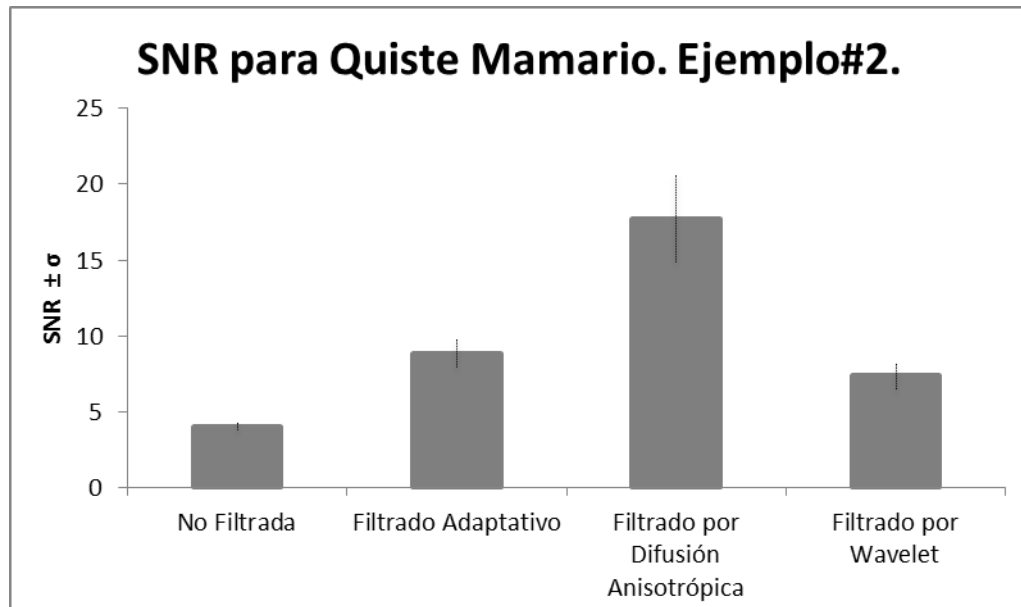
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#2. Quiste Mamario. Filtro Adaptativo	M1	4,1197	4,5303	9,0884	7,684
	M2	3,7135	5,4418	8,002	10,4118
	M3	3,6924	5,5394	8,0136	10,8126
	M4	3,8981	4,8591	8,843	8,9586
	M5	4,1091	4,4278	10,2507	4,2509
	PROMEDIO	3,90656	4,95968	8,83954	8,42358
	σ	0,20594863	0,511345	0,9269528	2,64125532

Tabla 5. SNR y CNR para Quiste Mamario. Ejemplo#2. Filtro por Difusión Anisotrópica.

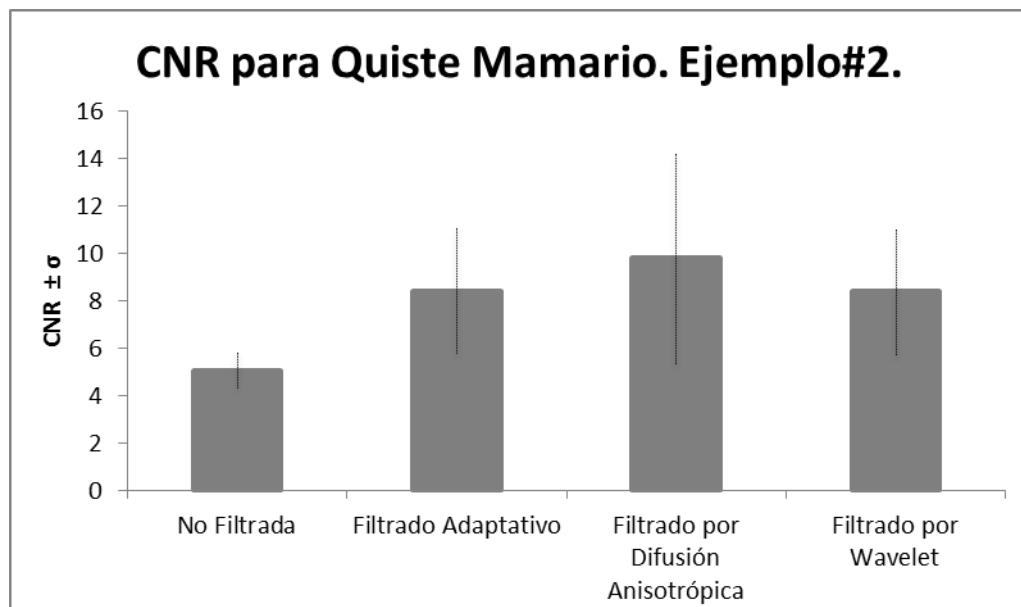
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#2. Quiste Mamario. Filtro por Difusión Anisotrópica	M1	3,9747	4,7063	17,7246	9,0843
	M2	3,5168	5,5459	15,3855	10,0198
	M3	3,5987	5,5379	15,2833	15,5328
	M4	3,6652	5,1726	17,8268	11,1753
	M5	4,1409	4,1852	22,4254	3,2177
	PROMEDIO	3,77926	5,02958	17,72912	9,80598
	σ	0,26621285	0,58372036	2,89565182	4,43405793

Tabla 6. SNR y CNR para Quiste Mamario. Ejemplo#2. Filtro por Wavelet.

	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#2. Quiste Mamario. Filtro por Wavelet	M1	4,3862	5,4033	7,3528	8,6638
	M2	4,0327	6,1752	6,7729	10,0847
	M3	4,2325	5,8398	6,9044	10,0947
	M4	4,1588	5,5903	7,0729	9,3742
	M5	4,918	3,2168	8,7451	3,823
	PROMEDIO	4,34564	5,24508	7,36962	8,40808
	σ	0,34455855	1,17014372	0,79887034	2,63046958



Gráfica 3. Comparación de SNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#2.



Gráfica 4. Comparación de CNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#2.

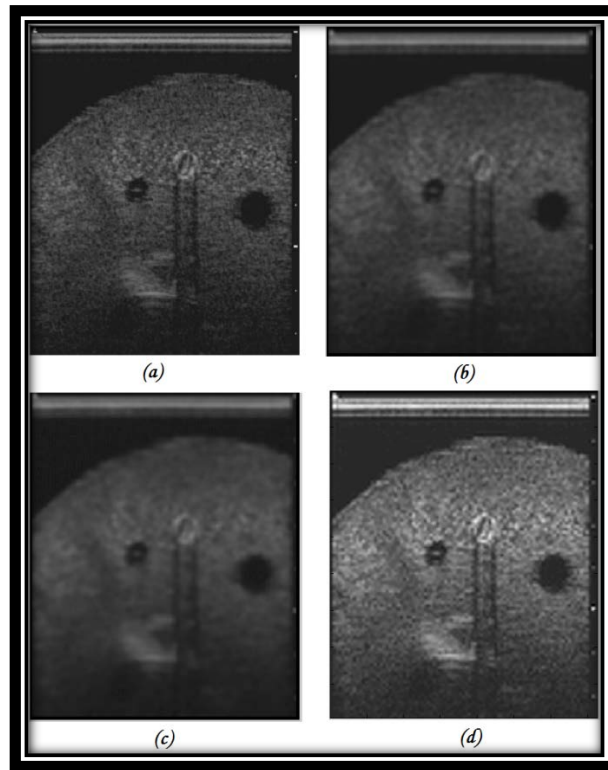


Figura 14. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #3.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
 (c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 7. SNR y CNR para Quiste Mamario. Ejemplo#3. Filtro Adaptativo.

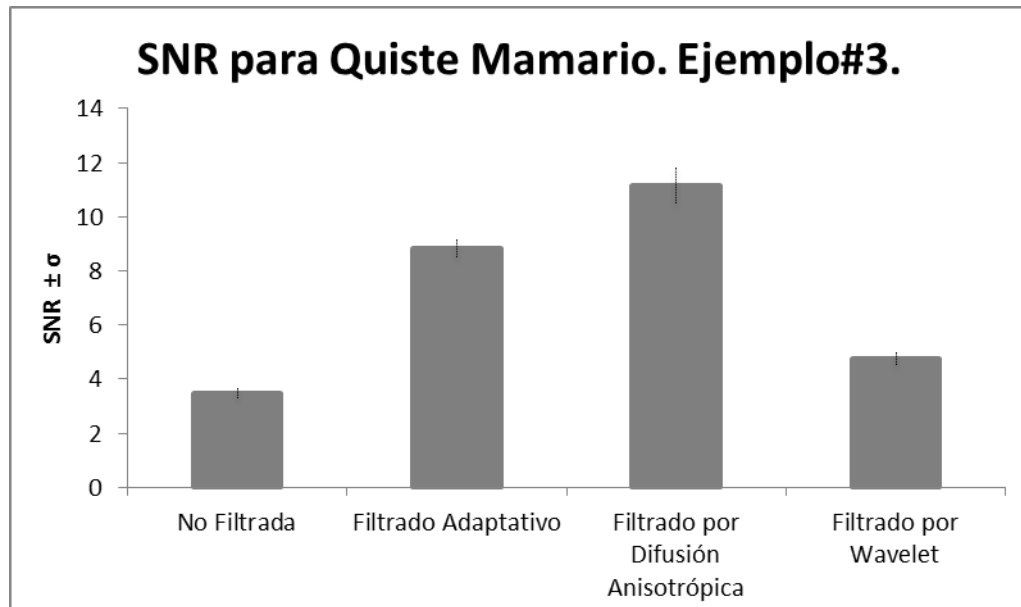
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#3. Quiste Mamario. Filtro Adaptativo	M1	3,5053	5,2518	8,7893	12,4284
	M2	3,397	5,5442	8,7307	13,3128
	M3	3,4021	5,5558	8,6339	13,7526
	M4	3,833	4,0845	9,3525	8,9064
	M5	3,5032	5,2929	8,7001	13,3031
	PROMEDIO	3,52812	5,14584	8,8413	12,34066
	σ	0,17830246	0,60951921	0,29121041	1,97906301

Tabla 8. SNR y CNR para Quiste Mamario. Ejemplo#3. Filtro por Difusión Anisotrópica.

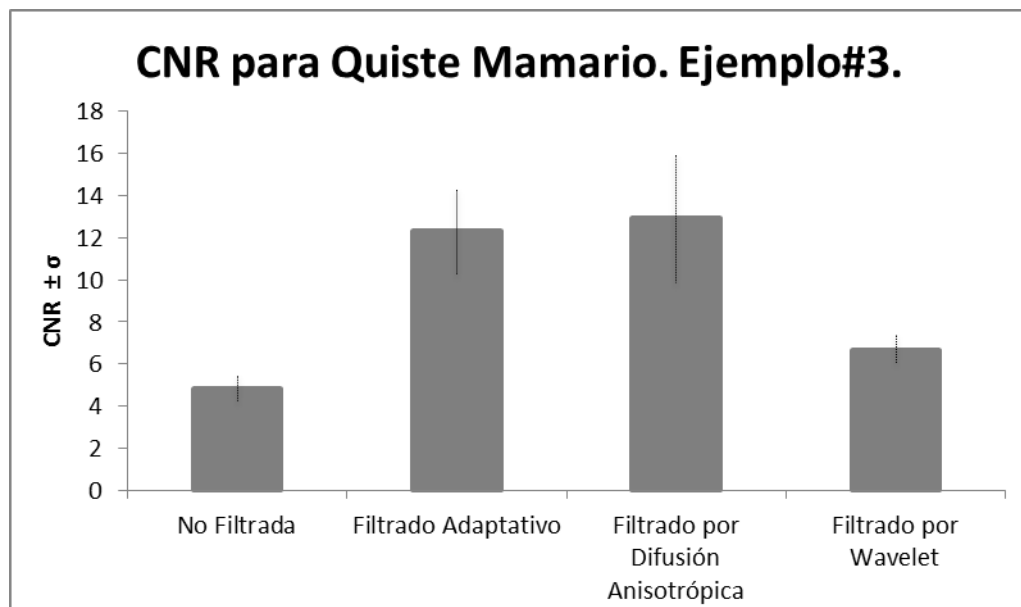
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#3. Quiste Mamario. Filtro por Difusión Anisotrópica	M1	3,0498	4,3367	11,1355	12,3845
	M2	2,9588	4,4814	10,8395	15,0059
	M3	2,957	4,5084	10,4558	15,4271
	M4	3,368	3,2671	12,2217	7,9067
	M5	3,0613	4,2607	11,1648	13,8961
	PROMEDIO	3,07898	4,17086	11,16346	12,92406
	σ	0,16883347	0,51544268	0,65696047	3,04175752

Tabla 9. SNR y CNR para Quiste Mamario. Ejemplo#3. Filtro por Wavelet.

	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#3. Quiste Mamario. Filtro por Wavelet	M1	3,8902	5,3378	4,6445	7,0179
	M2	3,7404	5,6364	4,638	7,1863
	M3	3,7427	5,6226	4,6063	7,2149
	M4	3,9981	4,1081	4,7144	5,9479
	M5	3,9853	4,8762	5,1231	5,9663
	PROMEDIO	3,87134	5,11622	4,74526	6,66666
	σ	0,12560276	0,64226575	0,21487355	0,65212775



Gráfica 5. Comparación de SNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#3.



Gráfica 6. Comparación de CNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#3.

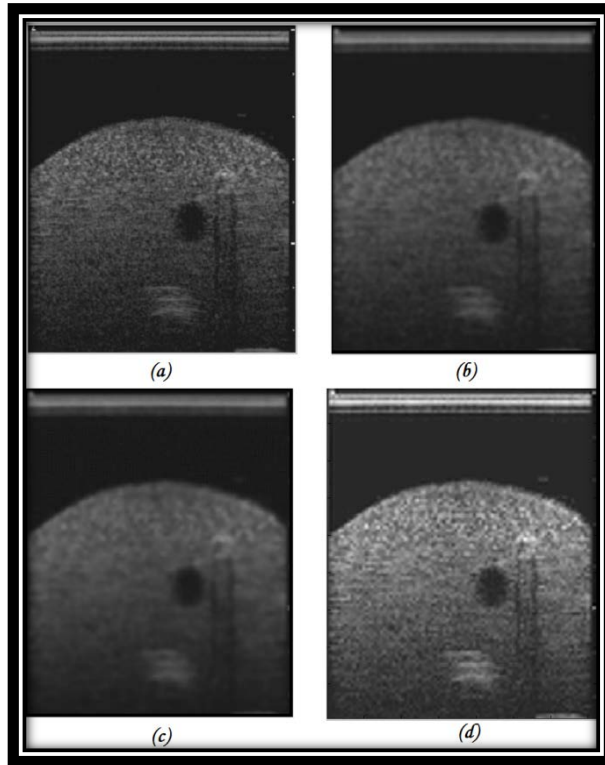


Figura 15. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #4.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 10. SNR y CNR para Quiste Mamario. Ejemplo#4. Filtro Adaptativo.

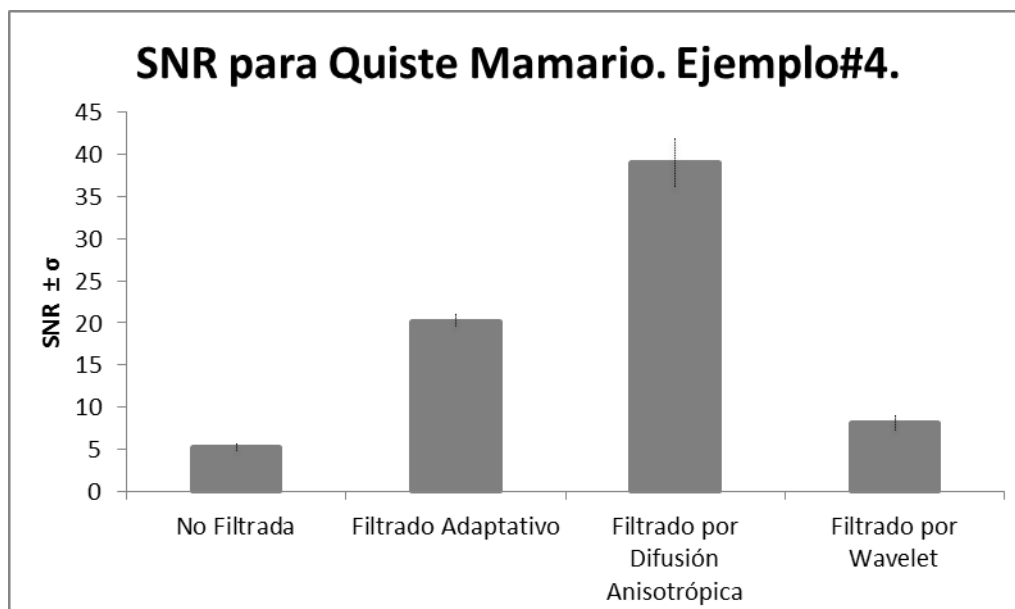
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#4. Quiste Mamario. Filtro Adaptativo	M1	5,136	5,857	19,3521	15,9416
	M2	5,55093	7,4852	19,4549	23,899
	M3	5,9396	7,0749	20,5797	21,8337
	M4	5,6702	7,5119	20,4269	20,812
	M5	5,85552	6,8454	21,2324	16,8864
	PROMEDIO	5,63045	6,95488	20,2092	19,87454
	σ	0,31567963	0,67513735	0,79611411	3,36565057

Tabla 11. SNR y CNR para Quiste Mamario. Ejemplo#4. Filtro por Difusión Anisotrópica.

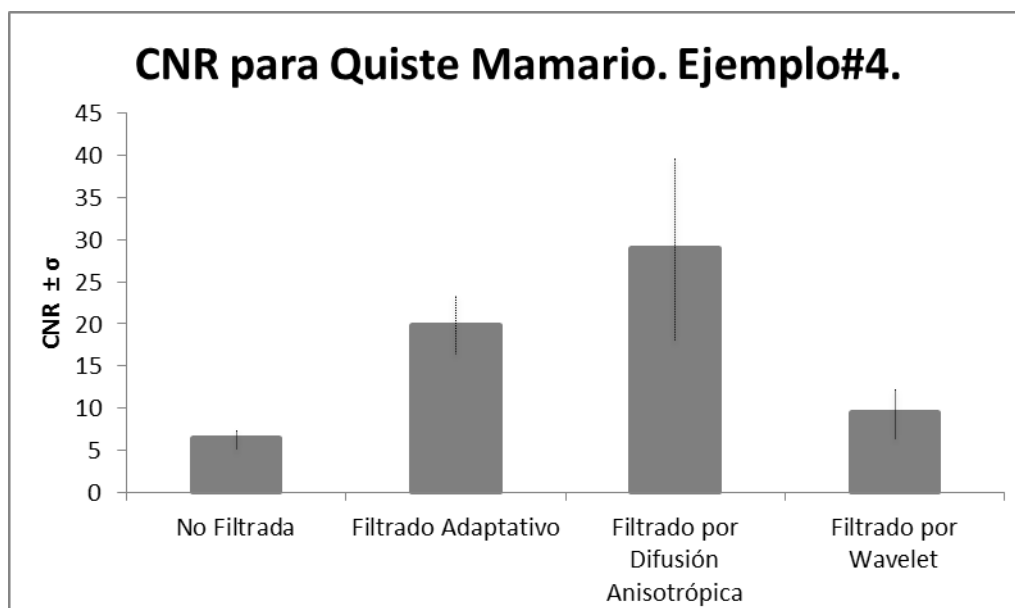
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#4. Quiste Mamario. Filtro por Difusión Anisotrópica	M1	5,2288	4,6797	35,1846	17,6277
	M2	5,652	8,0437	38,4097	42,2763
	M3	6,2037	7,1289	39,3243	33,0681
	M4	5,9089	7,9405	39,2974	33,4832
	M5	6,0479	7,2251	43,1975	18,3823
	PROMEDIO	5,80826	7,00358	39,0827	28,96752
	σ	0,38213859	1,36243011	2,85850303	10,6649912

Tabla 12. SNR y CNR para Quiste Mamario. Ejemplo#4. Filtro por Wavelet.

	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#4. Quiste Mamario. Filtro por Wavelet	M1	4,823	3,1751	6,4299	4,3228
	M2	3,8685	5,8253	8,309	11,0354
	M3	4,02	5,6677	8,4655	10,7909
	M4	4,0289	5,5462	8,3812	10,5686
	M5	4,0111	5,5287	8,6783	10,4694
	PROMEDIO	4,1503	5,1486	8,05278	9,43742
	σ	0,38178175	1,10957782	0,91772285	2,86746128



Gráfica 7. Comparación de SNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#4.



Gráfica 8. Comparación de CNR para Imágenes de Quiste Mamario con y sin filtrado Ejemplo#4.

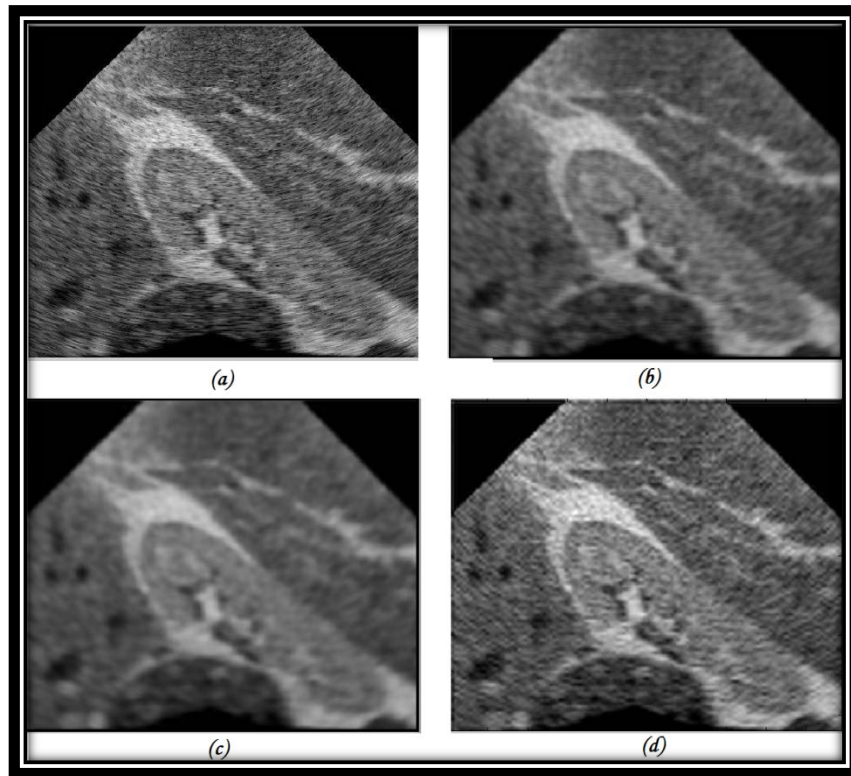


Figura 16. Imágenes ultrasónicas sin filtrar y filtradas de Riñón. Ejemplo #5.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
 (c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 13. SNR y CNR para Riñón. Ejemplo#5. Filtro Adaptativo.

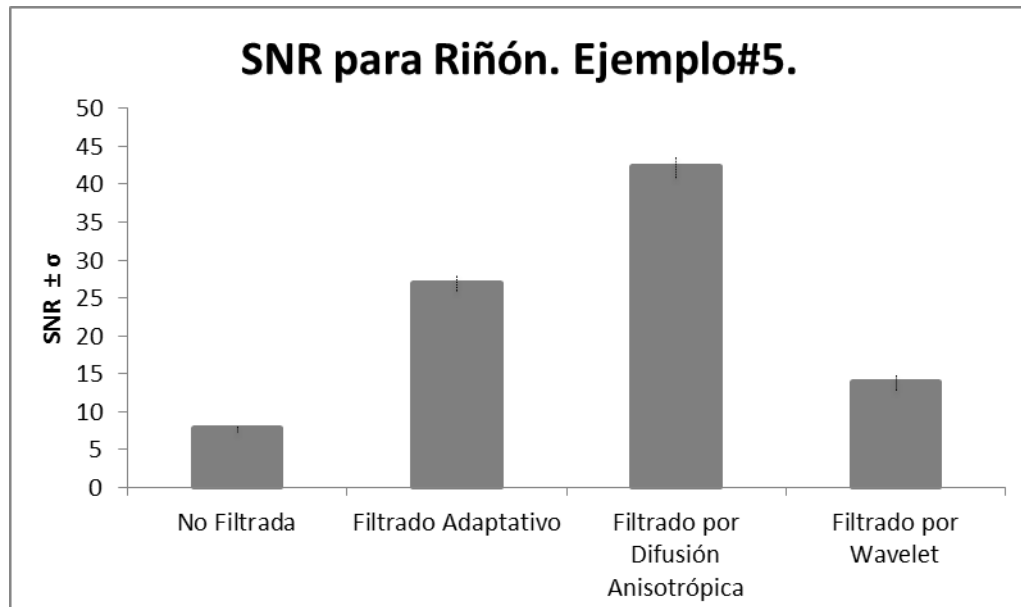
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#5. Riñón Filtro Adaptativo	M1	8,5131	4,4608	27,8424	11,6544
	M2	8,3607	4,2972	25,9969	9,8934
	M3	8,8245	4,7506	27,8975	12,9359
	M4	8,3396	4,1249	26,3045	9,0092
	M5	8,0544	3,7188	27,0717	11,2135
	PROMEDIO	8,41846	4,27046	27,0226	10,94128
	σ	0,281033811	0,385030918	0,86713046	1,5323298

Tabla 14. SNR y CNR para Riñón. Ejemplo#5. Filtro por Difusión Anisotrópica.

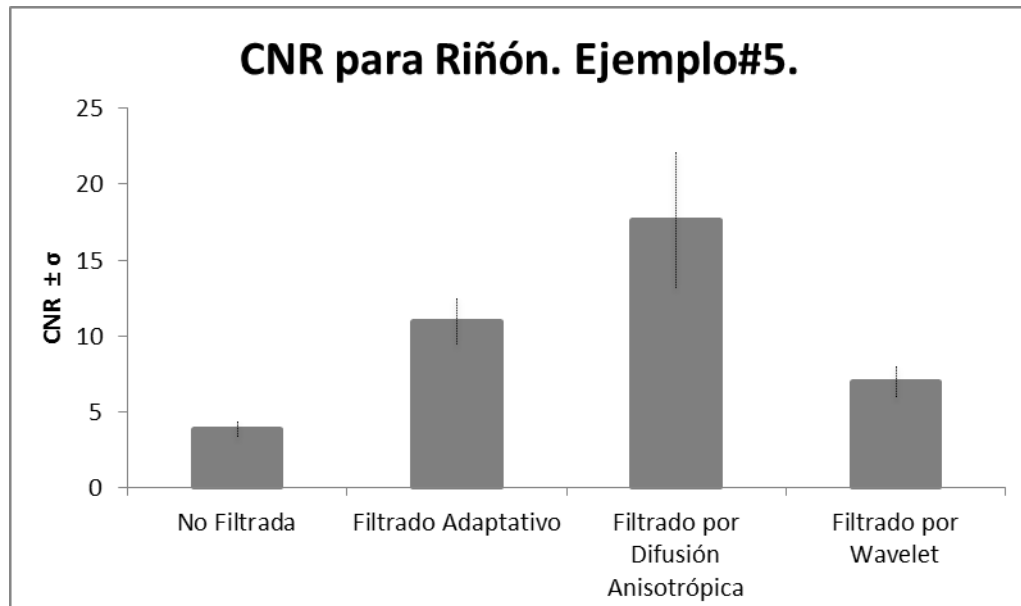
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#5. Riñón Filtro por Difusión Anisotrópica	M1	8,5273	4,3272	43,6345	18,6454
	M2	7,9782	3,3853	40,7395	15,2913
	M3	8,7542	4,1715	43,5521	23,8313
	M4	8,0278	3,5811	41,0953	11,7338
	M5	8,0803	3,4725	42,3387	18,6646
	PROMEDIO	8,27356	3,78752	42,27202	17,63328
	σ	0,346548601	0,43079081	1,344683075	4,494223914

Tabla 15. SNR y CNR para Riñón. Ejemplo#5. Filtro por Wavelet.

	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#5. Riñón Filtro por Wavelet	M1	7,3122	4,3013	15,0477	8,605
	M2	6,198	3,0567	12,9565	6,6608
	M3	7,0571	3,6182	14,3232	6,283
	M4	6,419	3,1508	12,9903	6,1838
	M5	6,8834	3,6434	14,3832	7,1789
	PROMEDIO	6,77394	3,55408	13,94018	6,9823
	σ	0,45814141	0,49508807	0,9272904	0,98764769



Gráfica 9. Comparación de SNR para Imágenes de Riñón con y sin filtrado Ejemplo#5.



Gráfica 10. Comparación de CNR para Imágenes Riñón con y sin filtrado Ejemplo#5.

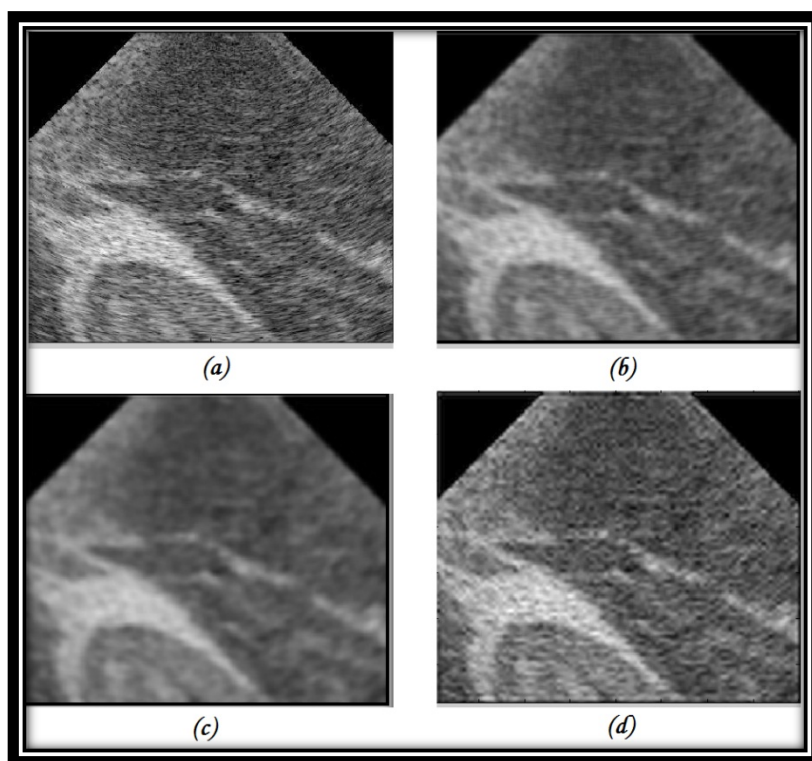


Figura 17. Imágenes ultrasónicas sin filtrar y filtradas de segmento de Riñón. Ejemplo #6.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 16. SNR y CNR para Riñón. Ejemplo#6. Filtro Adaptativo.

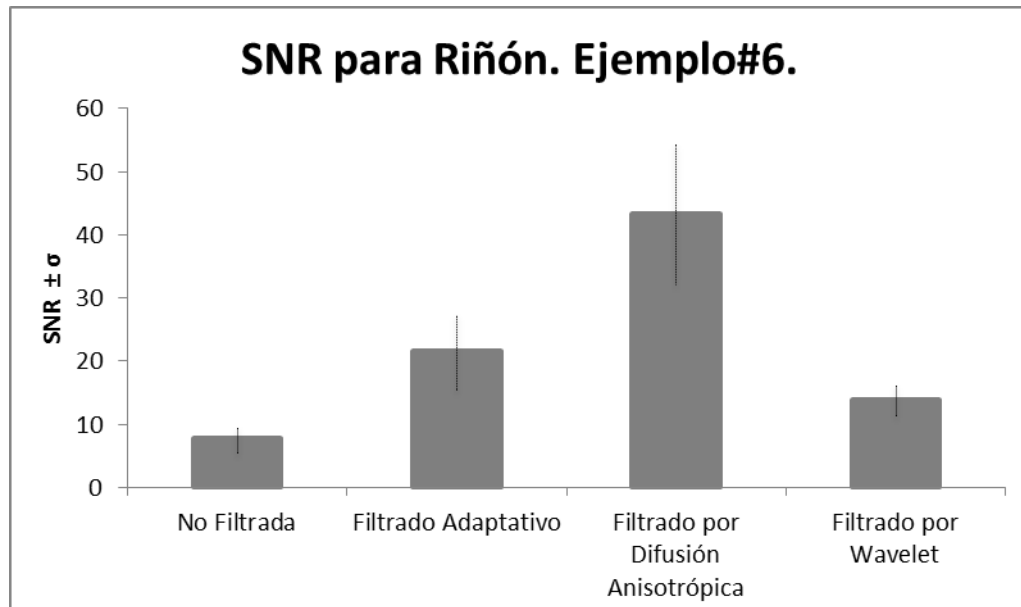
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#6. Riñón Filtro Adaptativo	M1	9,3202	2,1996	23,7034	6,2016
	M2	4,3719	1,9904	11,1686	5,776
	M3	10,3138	3,0566	24,8417	6,3411
	M4	9,7834	3,0581	24,2397	6,8892
	M5	9,416	2,8048	23,554	7,311
	PROMEDIO	8,64106	2,6219	21,50148	6,50378
	σ	2,418185789	0,497447605	5,79837515	0,60158

Tabla 17. SNR y CNR para Riñón. Ejemplo#6. Filtro por Difusión Anisotrópica.

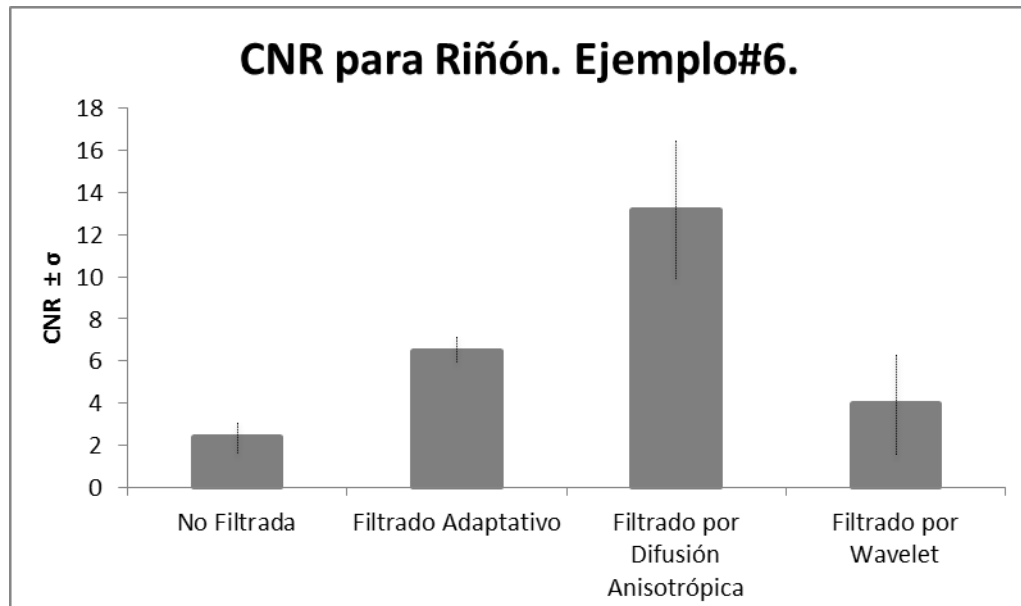
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#6. Riñón Filtro por Difusión Anisotrópica	M1	9,638	2,3732	47,2648	13,0149
	M2	4,3763	2,1399	23,6428	7,6742
	M3	10,3298	2,7148	49,6003	13,682
	M4	9,9334	2,8901	48,5228	15,5648
	M5	9,7648	2,8484	47,5845	15,8946
	PROMEDIO	8,80846	2,59328	43,32304	13,1661
	σ	2,49132146	0,32478611	11,03914463	3,303348824

Tabla 18. SNR y CNR para Riñón. Ejemplo#6. Filtro por Wavelet.

	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#6. Riñón Filtro por Wavelet	M1	5,4264	1,6911	13,8881	3,7891
	M2	6,3106	2,8373	15,507	5,823
	M3	6,2603	2,5143	15,2236	3,9047
	M4	3,9479	0,0035724	9,8799	0,23245
	M5	6,204	2,3631	15,3033	6,035
	PROMEDIO	5,62984	1,88187448	13,96038	3,95685
	σ	1,00758172	1,1301318	2,3688992	2,32933672



Gráfica 11. Comparación de SNR para Imágenes de Riñón con y sin filtrado Ejemplo#6.



Gráfica 12. Comparación de CNR para Imágenes de Riñón con y sin filtrado Ejemplo#6.

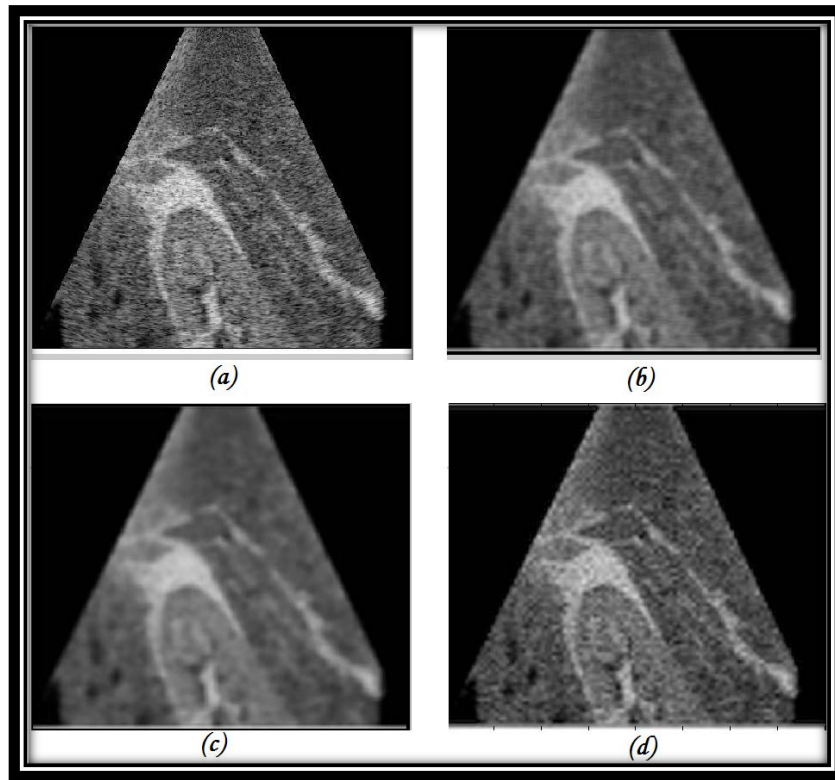


Figura 18. Imágenes ultrasónicas sin filtrar y filtradas de segmento de Riñón. Ejemplo #7.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 19. SNR y CNR para Riñón. Ejemplo#7. Filtro Adaptativo.

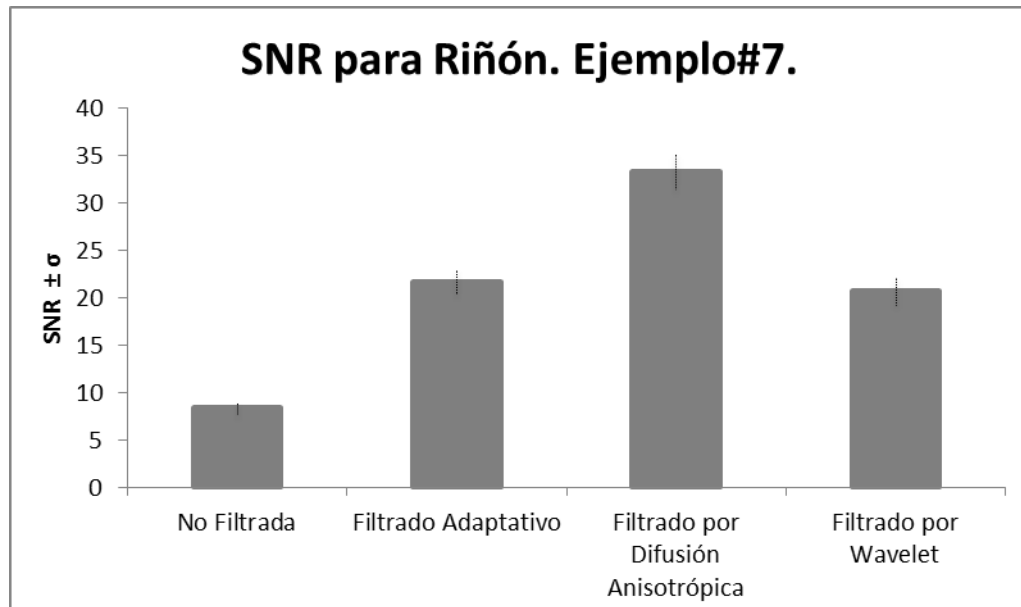
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#7. Riñón Filtro Adaptativo	M1	8,4097	3,0462	21,4601	8,963
	M2	8,7914	3,6477	22,3945	10,3636
	M3	8,4926	3,204	22,6424	11,5419
	M4	7,4969	2,0838	19,7966	7,2793
	M5	8,5502	3,2124	22,375	8,6053
	PROMEDIO	8,34816	3,03882	21,73372	9,35062
	σ	0,496632543	0,578849499	1,17269683	1,64498892

Tabla 20. SNR y CNR para Riñón. Ejemplo#7. Filtro por Difusión Anisotrópica.

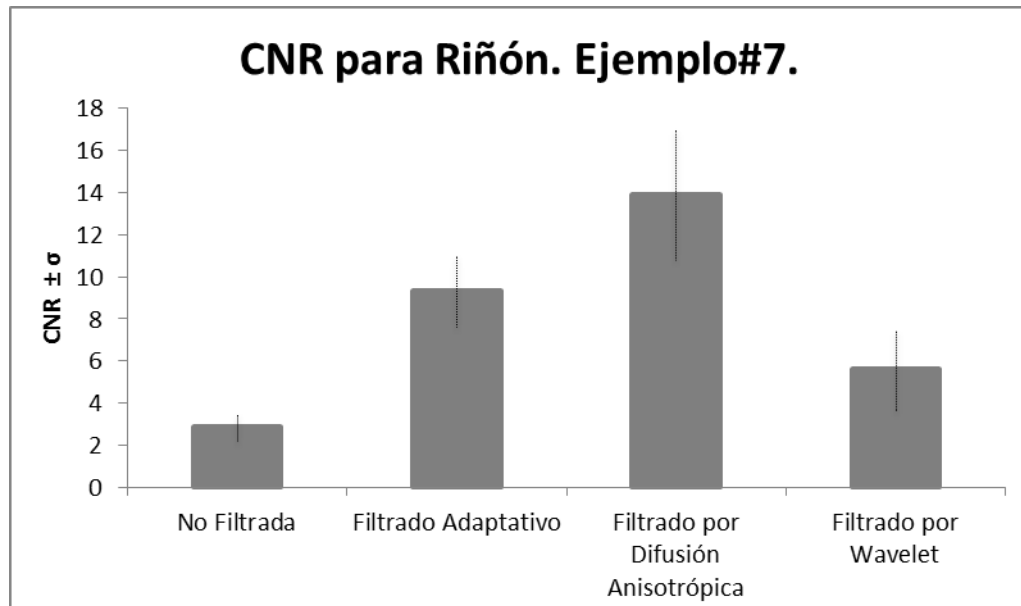
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#7. Riñón Filtro por Difusión Anisotrópica	M1	8,4601	3,0487	32,808	12,94
	M2	9,0156	3,8139	34,2261	13,3001
	M3	8,5729	3,3262	34,9568	19,1667
	M4	7,6836	2,2335	30,3952	13,0602
	M5	8,5615	3,2057	34,168	11,124
	PROMEDIO	8,45874	3,1256	33,31082	13,9182
	σ	0,483355214	0,5749281	1,805229426	3,058978291

Tabla 21. SNR y CNR para Riñón. Ejemplo#7. Filtro por Wavelet.

	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#7. Riñón Filtro por Wavelet	M1	7,8589	1,7162	19,7336	4,0857
	M2	8,85	3,1387	21,5464	6,7
	M3	8,7519	2,7498	21,7092	6,4344
	M4	7,5121	1,6148	18,8377	3,2236
	M5	8,9568	3,0339	21,7102	7,55
	PROMEDIO	8,38594	2,45068	20,70742	5,59874
	σ	0,65508149	0,73163278	1,33764687	1,84723421



Gráfica 13. Comparación de SNR para Imágenes de Riñón con y sin filtrado Ejemplo#7.



Gráfica 14. Comparación de CNR para Imágenes de Riñón con y sin filtrado Ejemplo#7.

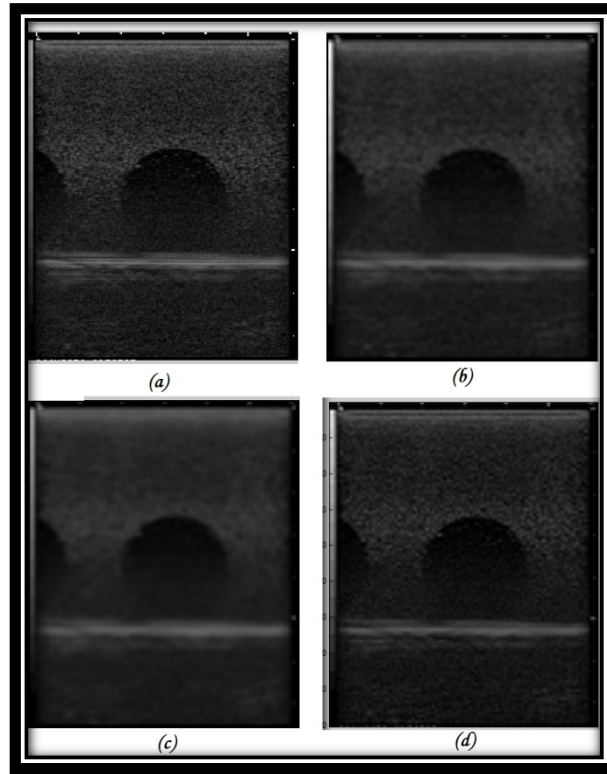


Figura 19. Imágenes ultrasónicas sin filtrar y filtradas de Quiste. Ejemplo #8.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
 (c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 22. SNR y CNR para Quiste. Ejemplo#8. Filtro Adaptativo.

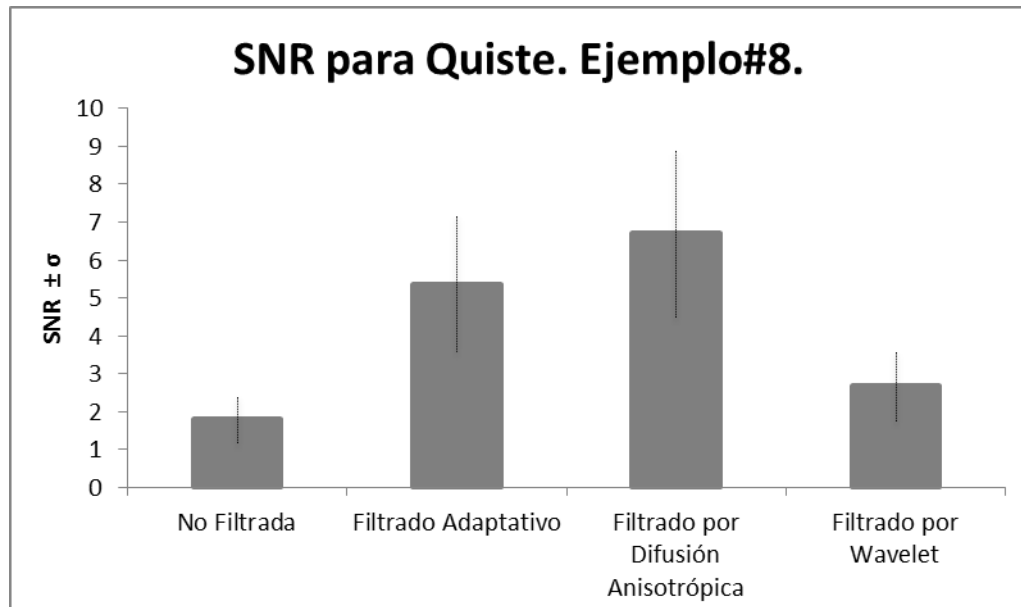
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#8. Quiste Filtro Adaptativo	M1	0,92066	5,0265	2,8206	14,7262
	M2	1,8182	3,8522	5,2359	11,482
	M3	1,8903	3,6719	5,5895	10,4379
	M4	1,8256	4,0492	5,2767	10,9009
	M5	2,7214	2,8478	7,8987	9,1732
	PROMEDIO	1,835232	3,88952	5,36428	11,34404
	σ	0,637407055	0,782967596	1,800587988	2,07292356

Tabla 23. SNR y CNR para Quiste. Ejemplo#8. Filtro por Difusión Anisotrópica.

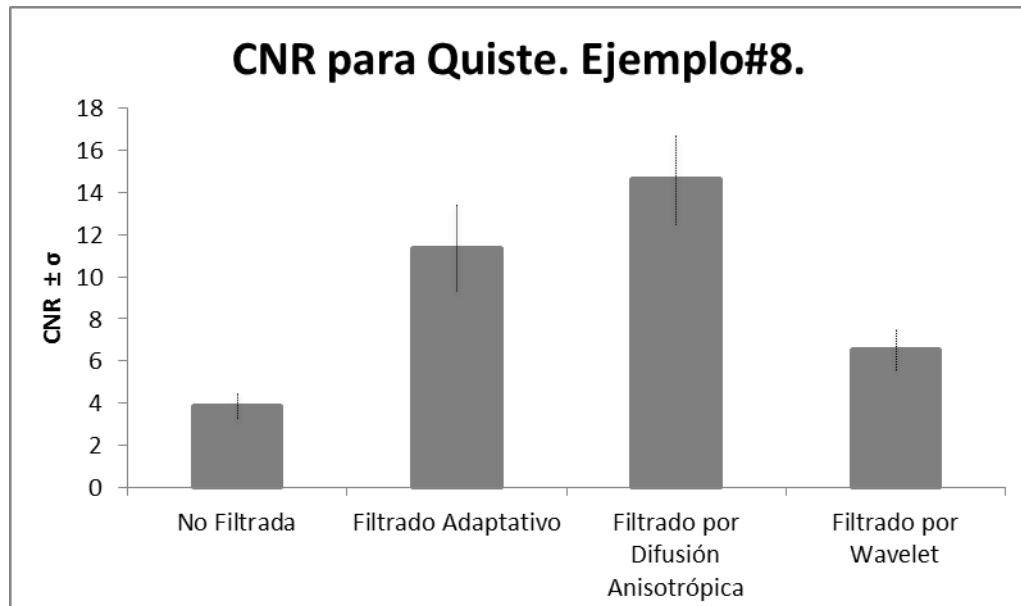
		Original		Filtrada	
MUESTRA		SNR	CNR	SNR	CNR
Ejemplo#8. Quiste Filtro por Difusión Anisotrópica	M1	1,0093	4,2732	3,6766	17,8098
	M2	1,6439	4,0756	6,2999	13,9621
	M3	1,7447	3,7901	7,0557	13,7332
	M4	1,7464	4,0905	6,765	15,4511
	M5	2,6523	2,9341	9,8089	12,1529
	PROMEDIO	1,75932	3,8327	6,72122	14,62182
	σ	0,586013167	0,53122458	2,184878877	2,13139413

Tabla 24. SNR y CNR para Quiste. Ejemplo#8. Filtro por Wavelet.

		Original		Filtrada	
MUESTRA		SNR	CNR	SNR	CNR
Ejemplo#8. Quiste Filtro por Wavelet	M1	0,9758	4,305	1,263	8,1112
	M2	1,9516	3,8002	3,0251	5,9267
	M3	1,7969	3,8841	2,7425	6,3429
	M4	1,8074	4,0471	2,6875	6,5554
	M5	2,4147	3,345	3,6808	5,6665
	PROMEDIO	1,78928	3,87628	2,67978	6,52054
	σ	0,519527205	0,353937872	0,884949986	0,954594245



Gráfica 15. Comparación de SNR para Quiste con y sin filtrado Ejemplo#8.



Gráfica 16. Comparación de CNR para Quiste con y sin filtrado Ejemplo#8.

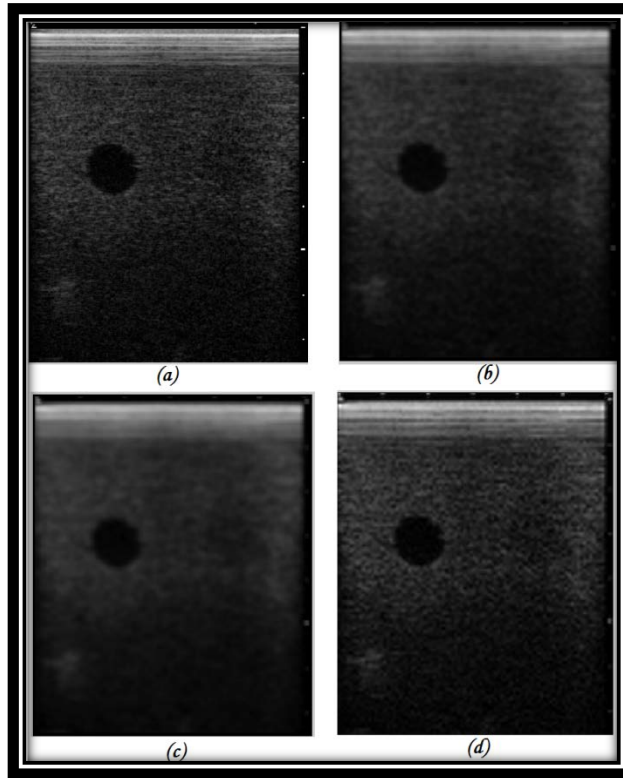


Figura 20. Imágenes ultrasónicas sin filtrar y filtradas de Quiste. Ejemplo #9.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 25. SNR y CNR para Quiste. Ejemplo#9. Filtro Adaptativo.

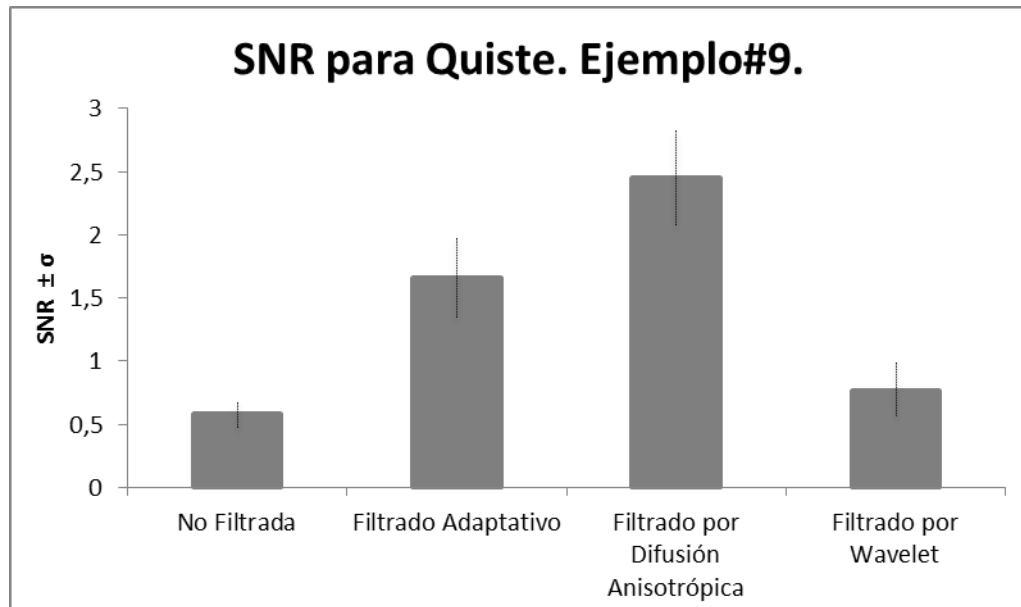
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#9. Quiste Filtro Adaptativo	M1	0,4773	3,5937	1,4055	9,495
	M2	0,58995	3,3831	1,5725	9,3408
	M3	0,51345	3,7337	1,489	9,7822
	M4	0,52596	3,6362	1,6202	9,409
	M5	0,67013	3,351	2,1947	8,4016
	PROMEDIO	0,555358	3,53954	1,65638	9,28572
	σ	0,075965112	0,165827751	0,311885439	0,52206093

Tabla 26. SNR y CNR para Quiste. Ejemplo#9. Filtro por Difusión Anisotrópica.

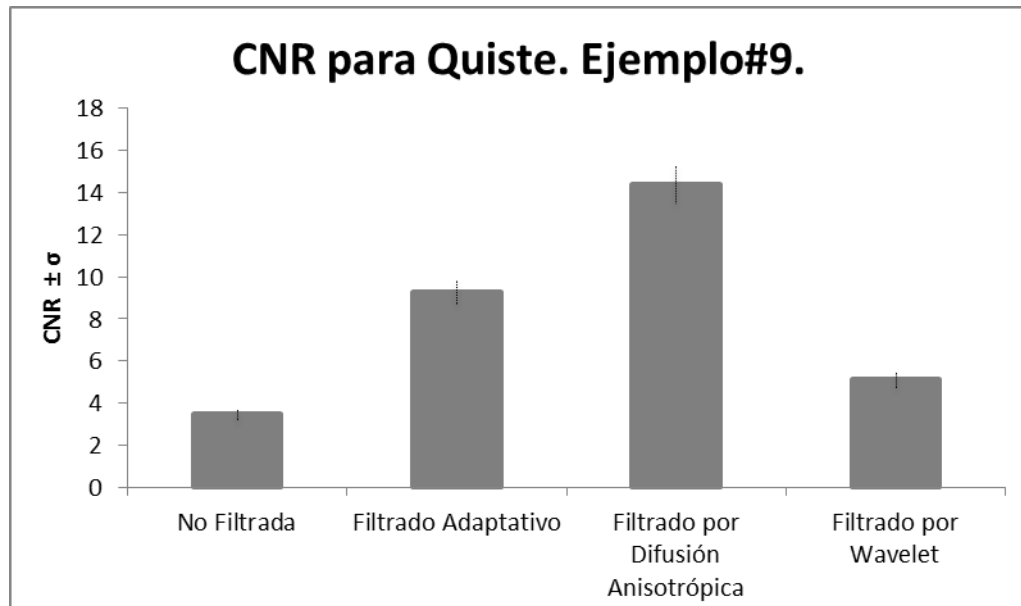
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#9. Quiste Filtro por Difusión Anisotrópica	M1	0,65585	3,6453	2,218	15,086
	M2	0,71668	3,6614	2,187	14,2597
	M3	0,60208	4,1764	2,187	15,4106
	M4	0,61354	4,0977	2,6306	13,2523
	M5	0,87976	3,5353	3,0308	13,9378
	PROMEDIO	0,693582	3,82322	2,45068	14,38928
	σ	0,113356525	0,29189662	0,374868206	0,872046729

Tabla 27. SNR y CNR para Quiste. Ejemplo#9. Filtro por Wavelet.

	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#9. Quiste Filtro por Wavelet	M1	0,39672	3,2059	0,57739	5,1434
	M2	0,5565	2,9687	0,96791	5,2045
	M3	0,37022	3,4105	0,55443	5,6463
	M4	0,51831	3,0742	0,73235	5,1685
	M5	0,5791	2,9851	1,0078	4,6592
	PROMEDIO	0,48417	3,12888	0,767976	5,16438
	σ	0,094922308	0,183413555	0,212534282	0,349828697



Gráfica 17. Comparación de SNR para Quiste con y sin filtrado Ejemplo#9.



Gráfica 18. Comparación de CNR para Quiste con y sin filtrado Ejemplo#9.

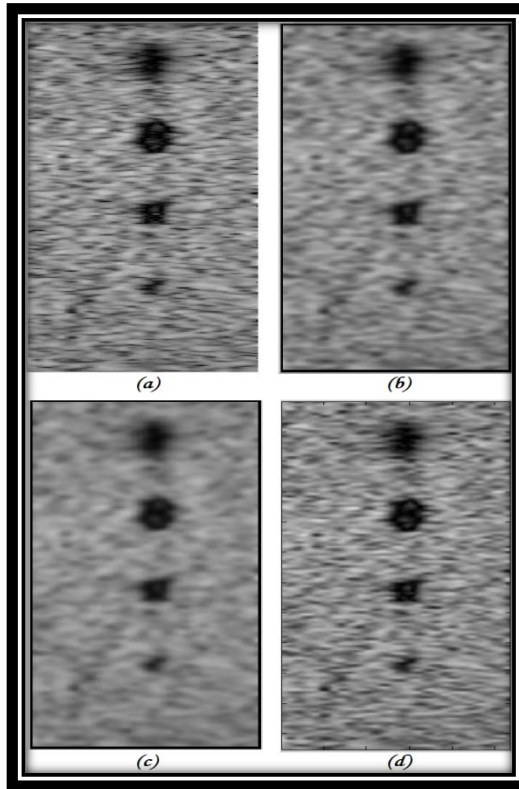


Figura 21. Imágenes ultrasónicas sin filtrar y filtradas de Quistes. Ejemplo #10.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet.

Tabla 28. SNR y CNR para Quistes. Ejemplo#10. Filtro Adaptativo.

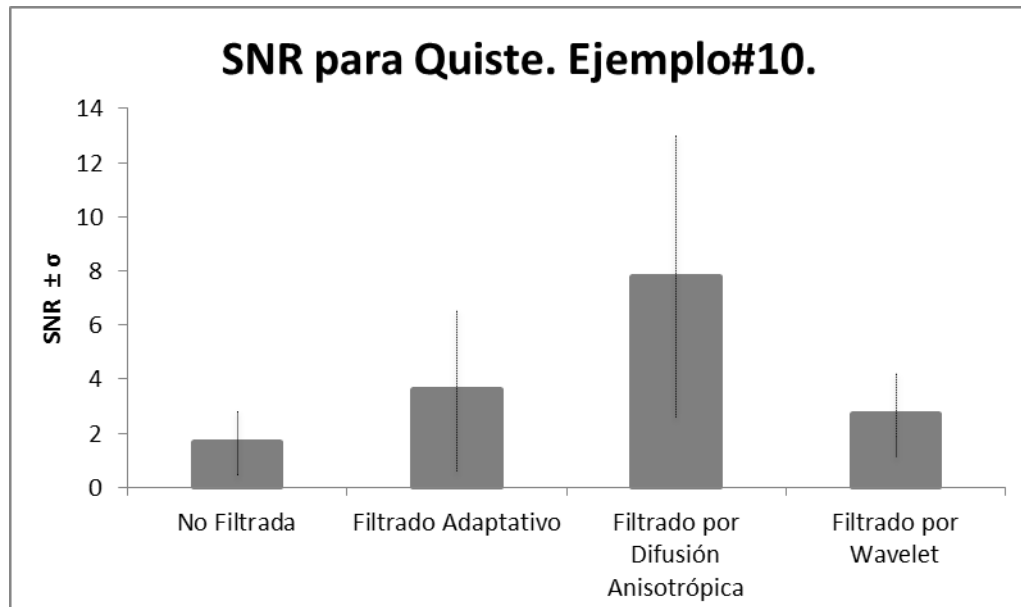
	Original		Filtrada		
	MUESTRA	SNR	CNR	SNR	CNR
Ejemplo#10. Quistes Filtro Adaptativo	M1	1,0093	7,7674	2,6685	20,4158
	M2	0,99644	7,714	2,1037	18,7759
	M3	0,39515	10,1187	1,4771	22,348
	M4	0,96264	7,5357	3,0262	14,3362
	M5	2,9974	5,1114	8,7646	8,073
	PROMEDIO	1,272186	7,64944	3,60802	16,78978
σ	0,998313468	1,772703538	2,941739636	5,7006318	

Tabla 29. SNR y CNR para Quistes. Ejemplo#10. Filtro por Difusión Anisotrópica.

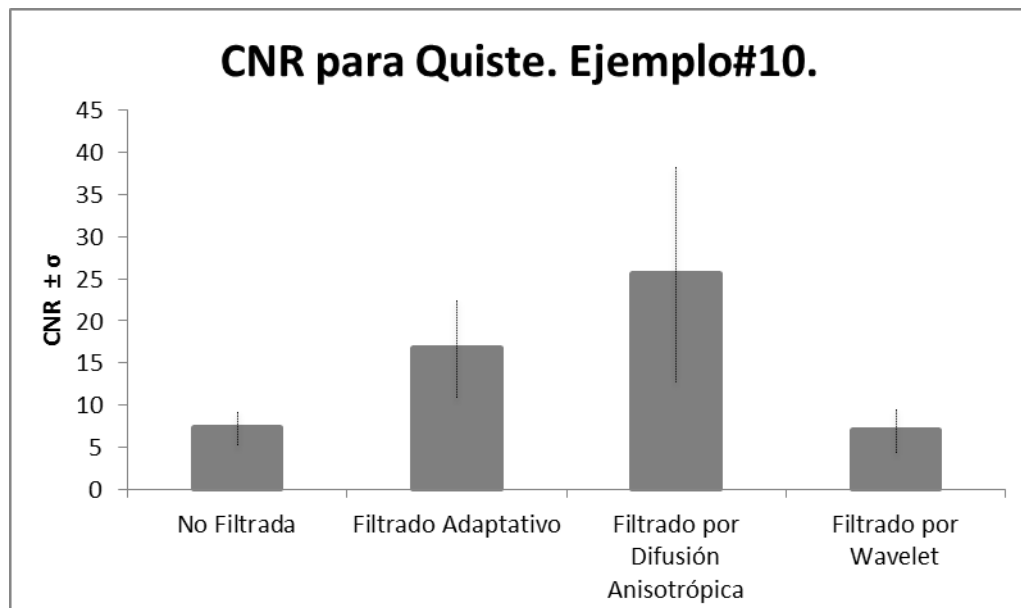
	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#10. Quistes Filtro por Difusión Anisotrópica	M1	1,047	8,0157	6,2011	33,2112
	M2	0,78211	8,6895	5,3391	32,6199
	M3	0,46472	10,4162	3,1083	38,1768
	M4	1,157	7,087	7,7861	15,0751
	M5	3,5538	4,9385	16,5937	8,8457
	PROMEDIO	1,400926	7,82938	7,80566	25,58574
	σ	1,232785959	2,02326929	5,195039741	12,81468232

Tabla 30. SNR y CNR para Quistes. Ejemplo#10. Filtro por Wavelet

	MUESTRA	Original		Filtrada	
		SNR	CNR	SNR	CNR
Ejemplo#10. Quistes Filtro por Wavelet	M1	1,9799	6,683	2,334	7,2507
	M2	1,2098	8,3933	1,2837	9,6458
	M3	1,2547	8,4811	1,3263	9,4019
	M4	3,5093	4,1589	4,1262	4,1644
	M5	3,7847	4,5789	4,4159	4,9046
	PROMEDIO	2,34768	6,45904	2,69722	7,07348
	σ	1,22872147	2,043637499	1,500436102	2,511800336



Gráfica 19. Comparación de SNR para Quiste con y sin filtrado Ejemplo#10.



Gráfica 20. Comparación de CNR para Quiste con y sin filtrado Ejemplo#10.

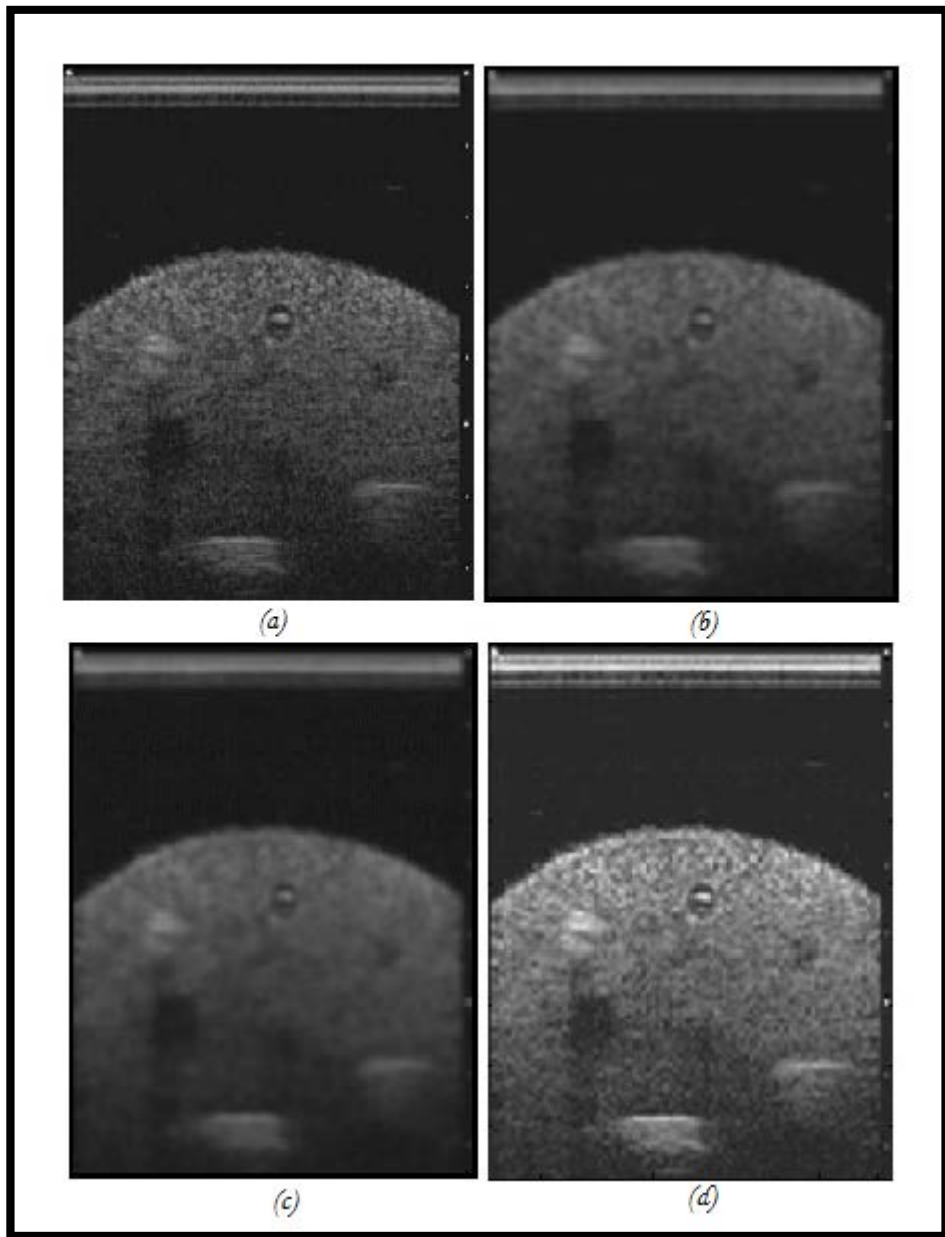


Figura 22. Imágenes ultrasónicas sin filtrar y filtradas de quiste mamario. Ejemplo #11.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

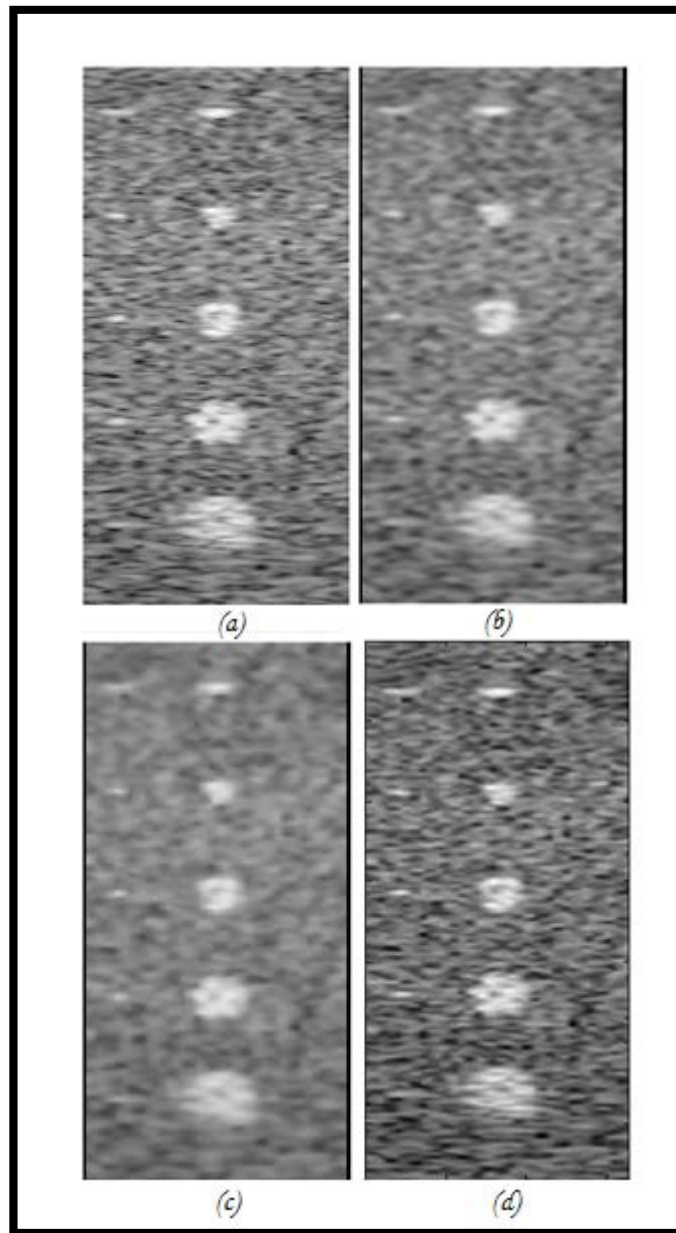


Figura 23. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #12.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

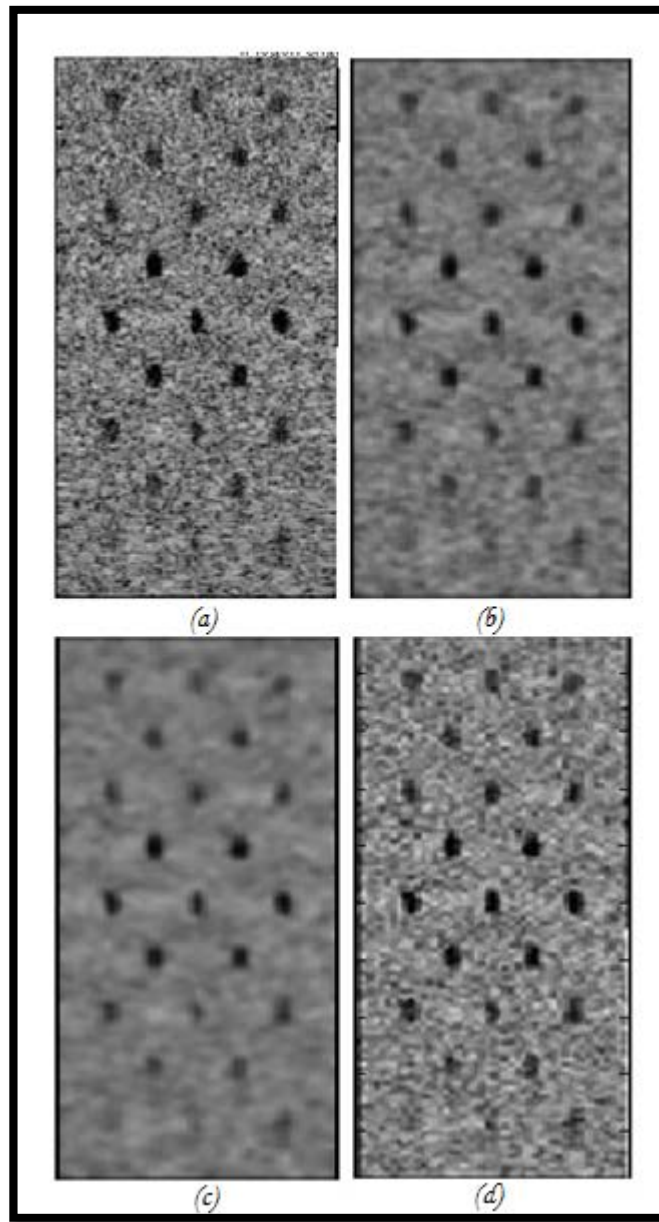


Figura 24. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #13.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

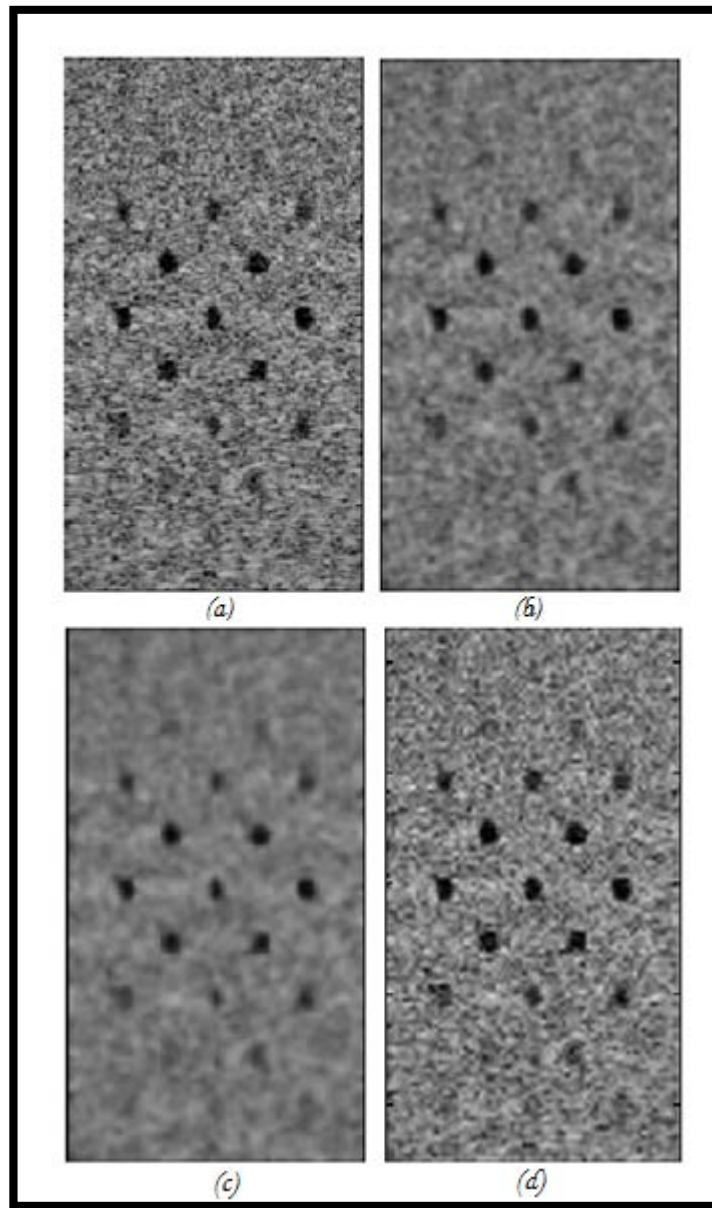


Figura 25. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #14.

- (a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

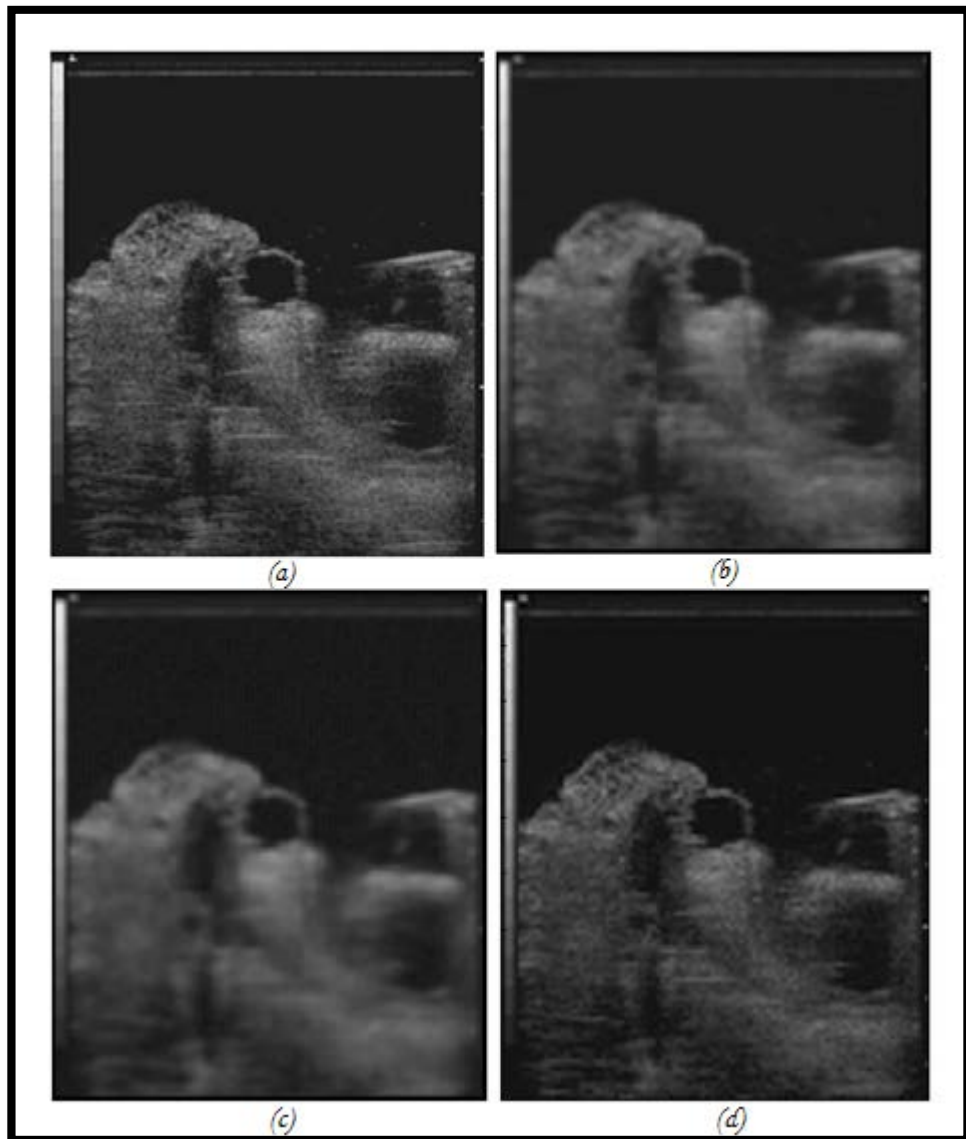


Figura 26. Imágenes ultrasónicas sin filtrar y filtradas. Ejemplo #15.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

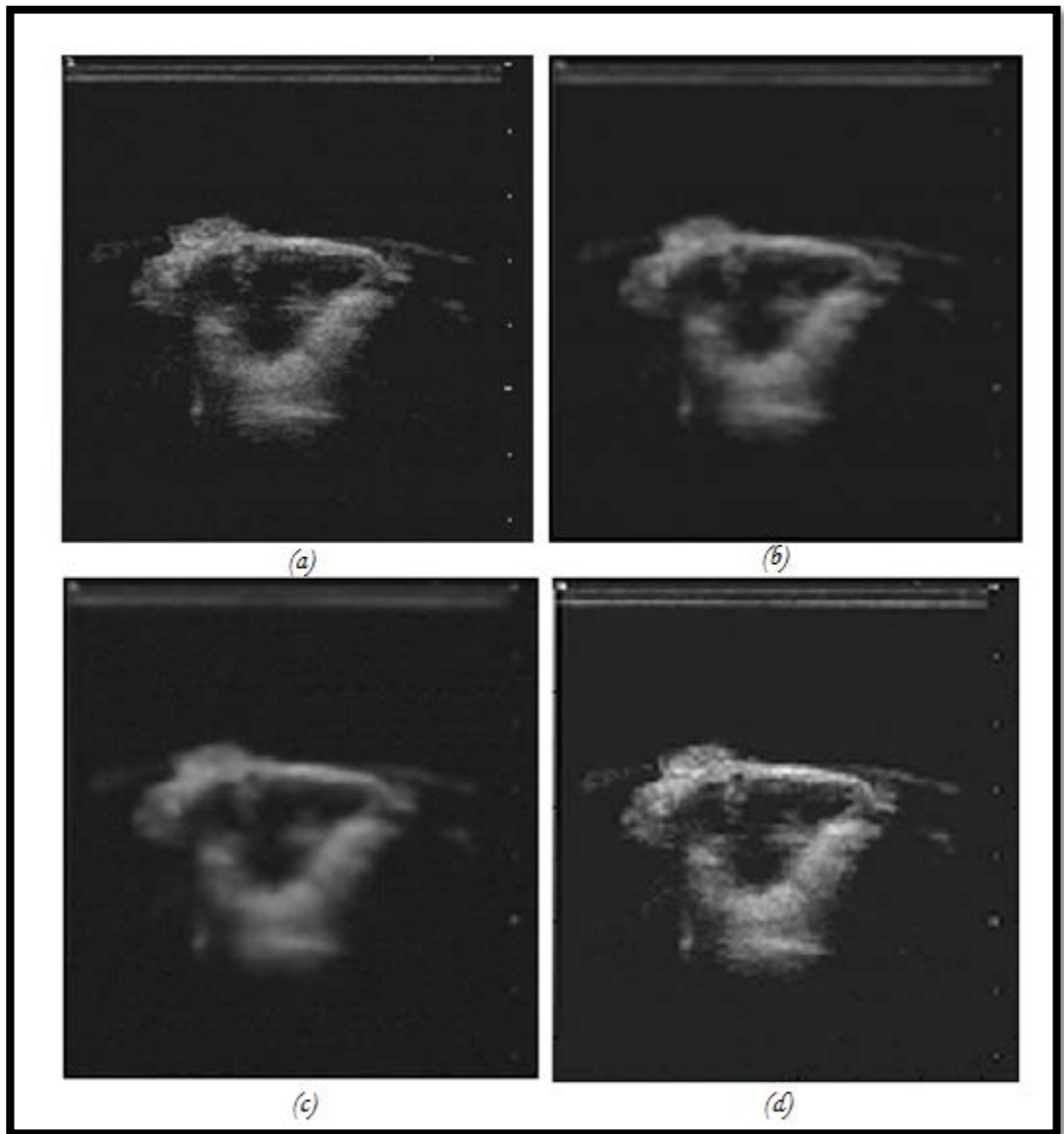


Figura 27. Imágenes ultrasónicas sin filtrar y filtradas de pelvis simulado. Ejemplo #16.

- (a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

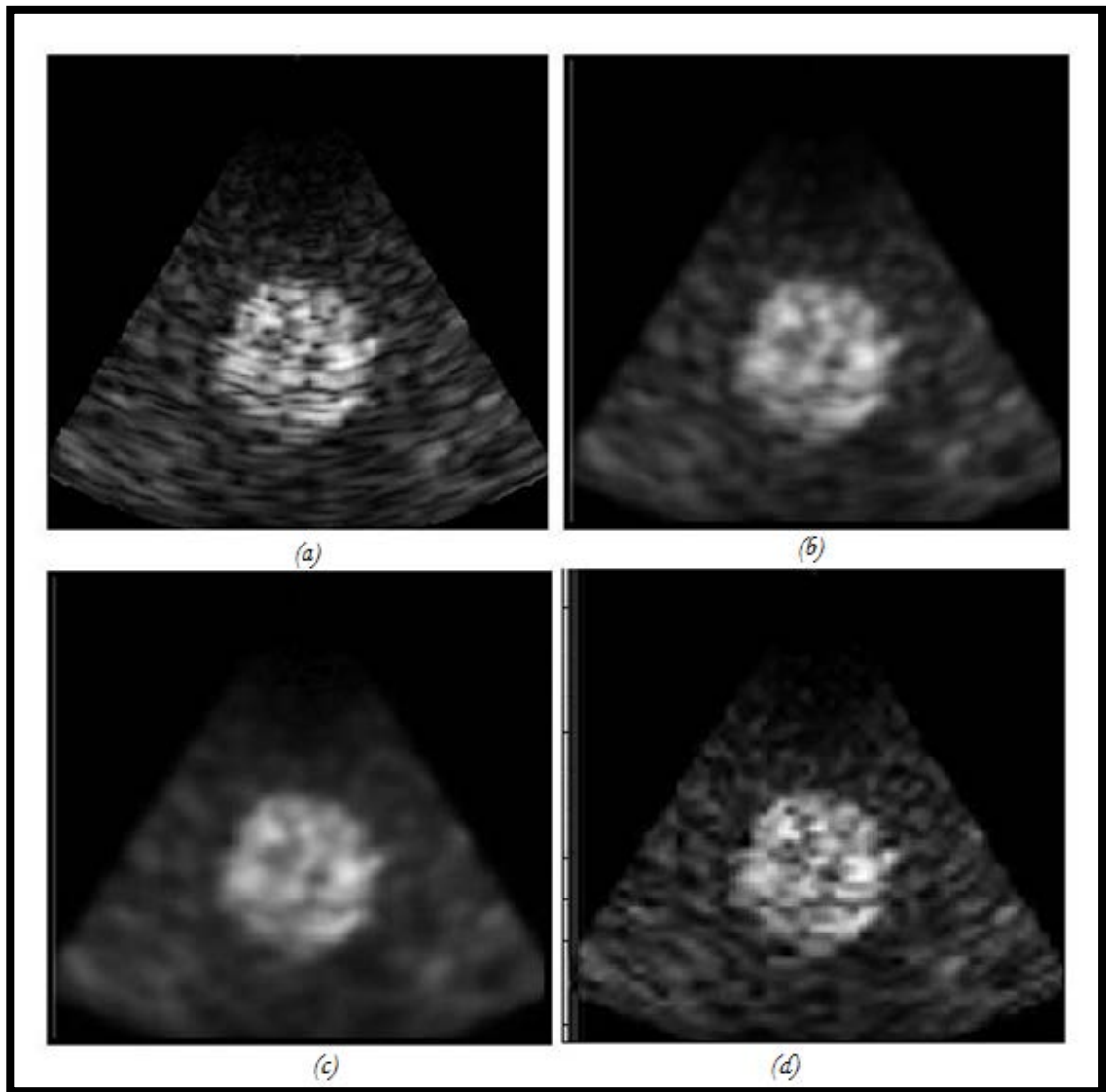


Figura 28. Imágenes ultrasónicas sin filtrar y filtradas de quiste. Ejemplo #17.

- (a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

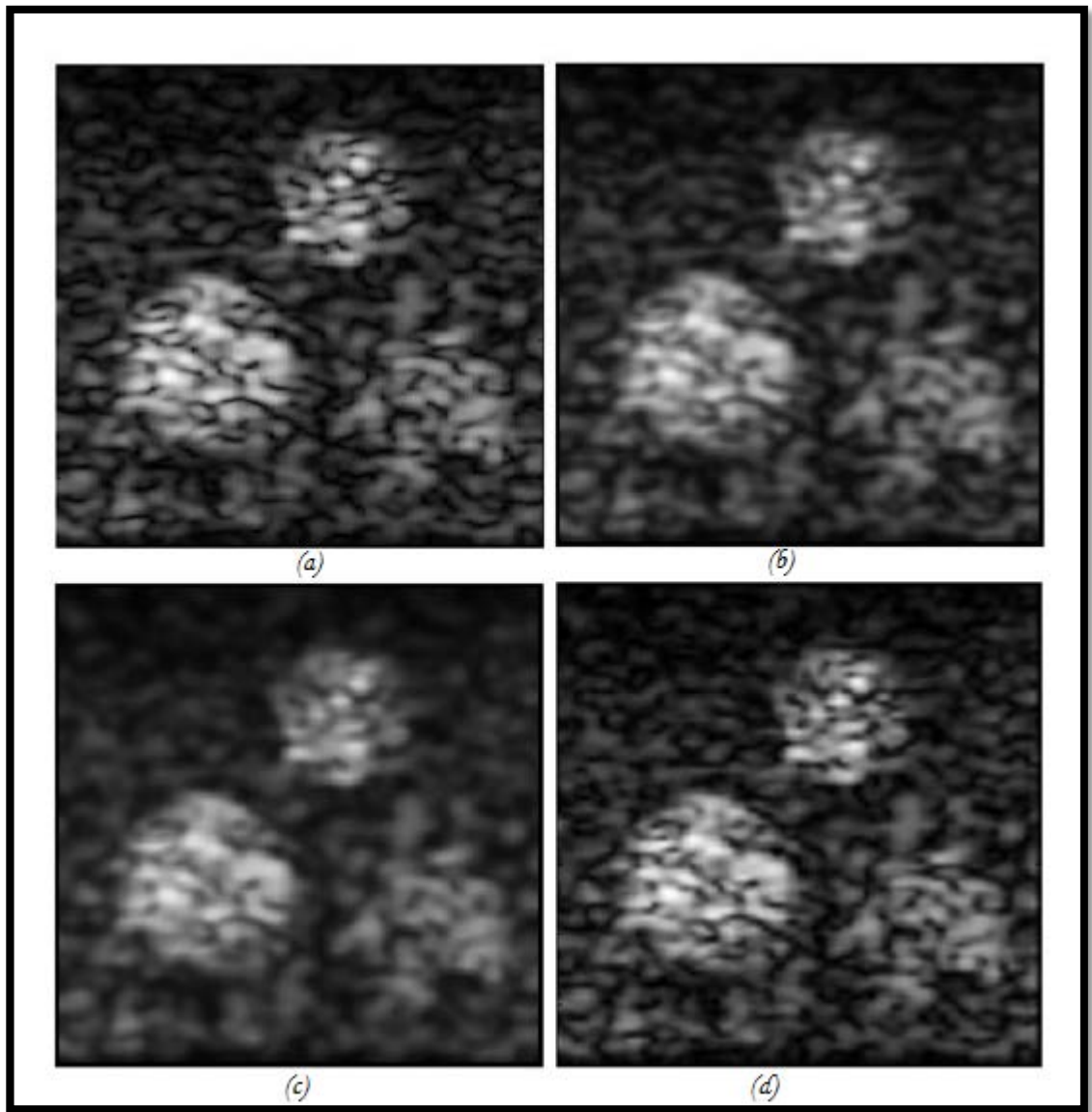


Figura 29. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #18.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

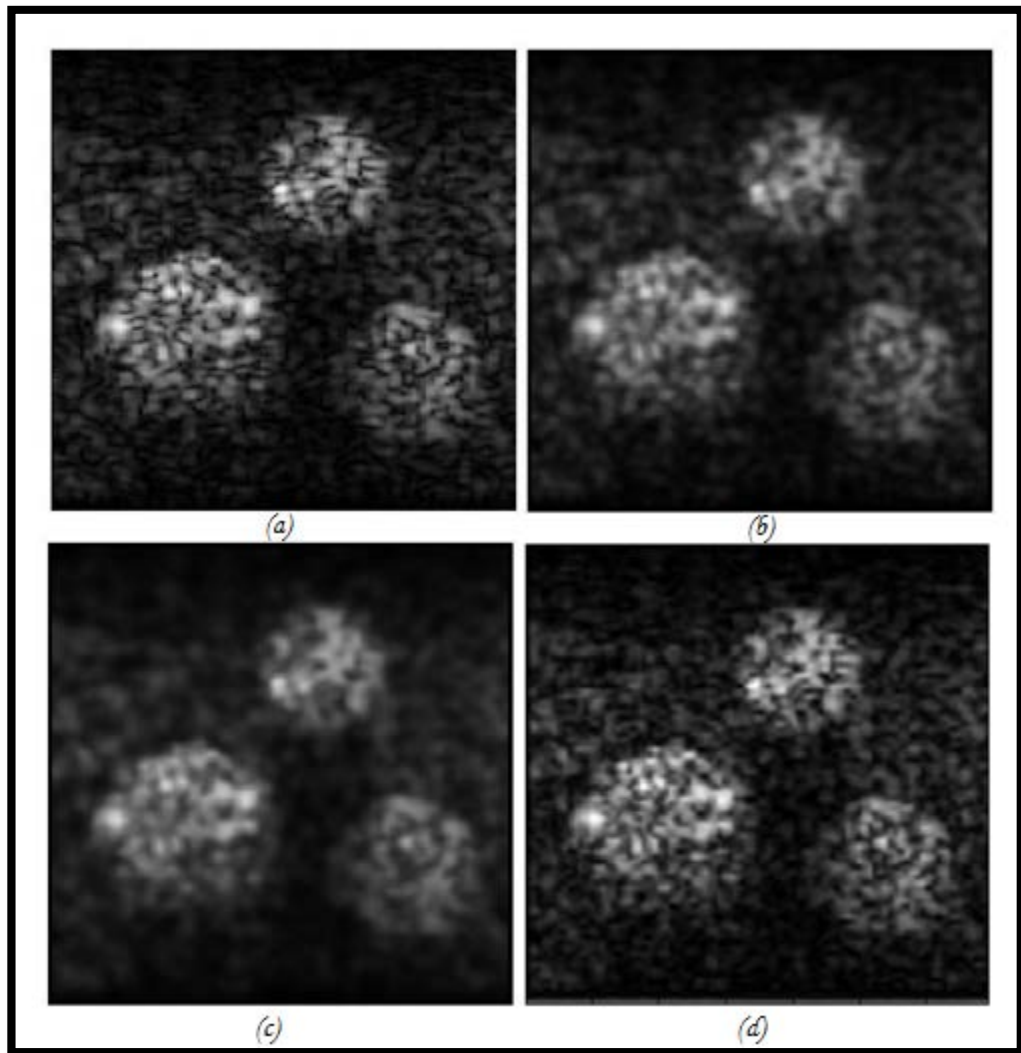


Figura 30. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #19.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

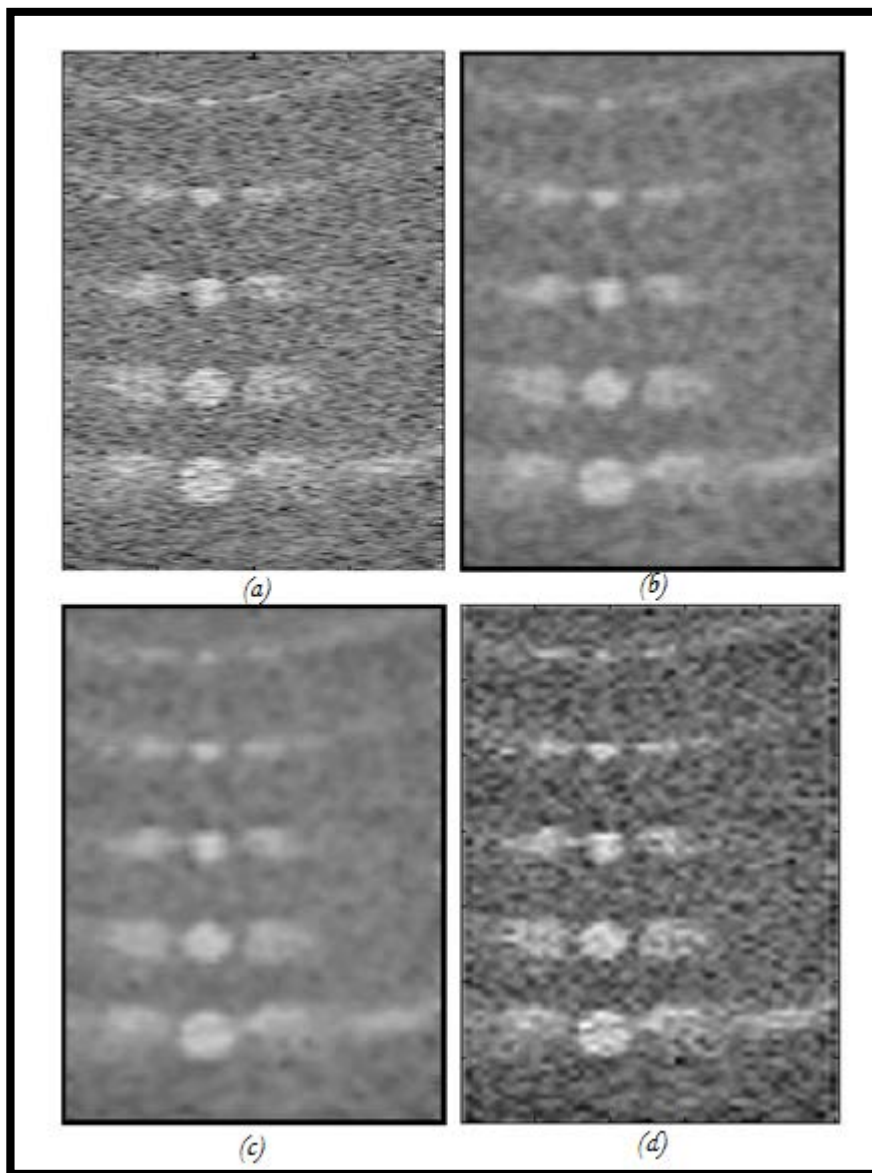


Figura 31. Imágenes ultrasónicas sin filtrar y filtradas de quistes. Ejemplo #20.

(a). Imagen Original (No filtrada). (b). Imagen Suavizada con Filtro Adaptativo.
(c). Imagen Suavizada con Filtro por Difusión Anisotrópica. (d). Imagen Suavizada con Filtro por Wavelet

CÓDIGO DE LA INTERFAZ IMPLEMENTADA

```
%////////////////////////////////////
////
///IMPLEMENTACIÓN Y EVALUACIÓN DE MÉTODOS DE FILTRADO DIGITAL PARA
IMÁGENESULTRASÓNICAS////////////////////////////////////
%////////////////////////////////////
////
//
% HANSEL ARMANDO CASTILLA CABALLERO. INGENIERO ELECTRÓNICO
% DESARROLLADOR
% Universidad Tecnológica de Bolívar
% 2013

function varargout = Filtros(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',  @Filtros_OpeningFcn, ...
                  'gui_OutputFcn',  @Filtros_OutputFcn, ...
                  'gui_LayoutFcn',   [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% --- Executes just before Filtros is made visible.
function Filtros_OpeningFcn(hObject, eventdata, handles, varargin)
clc
global K cont xp yp pos_act filter_option
cont=0;
set(gcf,'CurrentAxes',handles.axes3)
fondo = imread('Data\Design\Ultrasound.jpg'); %Imagen Seleccionada
image(fondo)
axis off

%Centrar la Interfaz en la
Pantalla
scrsz=get(0,'ScreenSize');
pos_act=get(gcf,'Position');
xr=scrsz(3)-pos_act(3);
xp=round(xr/2);
yr=scrsz(4)-pos_act(4);
yp=round(yr/2);
```

```

set(gcf, 'Position', [xp yp pos_act(3) pos_act(4)]);

set(gcf, 'CurrentAxes', handles.axes2)
fondo = imread('Data\Design\Background.jpg'); %Fondo de la GUI
image(fondo)
axis off

set(handles.AdaptiveFilter, 'Value', 1);
set(handles.waveletfilter, 'Value', 0);
set(handles.anisotrofilter, 'Value', 0);
filter_option=1;

handles.output = hObject;
guidata(hObject, handles);
function varargout = Filtros_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%//////// BARRA DE MENÚ //////////
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%////////MENÚ ARCHIVO////////
% -----
function Archivo_menu_Callback(hObject, eventdata, handles)
% hObject     handle to Archivo_menu (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

%Carga la Imagen del Disco Duro que será Filtrada
% -----
function AbrirImagen_Callback(hObject, eventdata, handles)
function Abrir_Imagen_Disco_Callback(hObject, eventdata, handles)

try
    [filename,pathname] =
    uigetfile({'*.bmp;*.jpg;*.jpeg;*.jpe;*.jfif;*.gif;*.tiff;*.tif;*.png', 'To
das los Archivos de Imagen';
            '*.bmp', 'Archivo de Mapa de Bits(*.bmp)';
            '*.jpg;*.jpeg;*.jpe;*.jfif', 'JPEG(*.jpg;*.jpeg;*.jpe;*.jfif)';
            '*.gif', 'GIF(*.gif)';
            '*.tiff;*.tif', 'TIFF(*.tiff;*.tif)';
            '*.png', 'PNG(*.png)';
            '*.*', 'Todos los Archivos'},...
            'Selecciona imagen para abrir');

    global K cont

    if isequal(filename,0)
        %Cargar Imagen Cancelada
    else

```



```

        %disp(['User selected ', fullfile(pathname, filename)])
        K=imread(fullfile(pathname, filename));
        set(gcf, 'CurrentAxes', handles.axes3)
        image(K), colormap(gray);
        axis off
        cont=1;
        %handles.myImage = imread(fullfile(pathname, filename));
    end
    guidata(hObject, handles);
catch
    clc
    msgbox('Error', 'Intenta Cargar la Imagen Nuevamente', 'error', 'modal')
end

%Carga la Imagen de la Base de datos que será Filtrada
% -----
function Abrir_Imagen_Database_Callback(hObject, eventdata, handles)

% -----
function Imagen_1_Callback(hObject, eventdata, handles)
    global K cont
    K=imread('Samples\1.bmp');
    set(gcf, 'CurrentAxes', handles.axes3)
    imshow(K), colormap(gray);
    axis off
    cont=1;

% -----
function Imagen_2_Callback(hObject, eventdata, handles)
    global K cont
    K=imread('Samples\2.bmp');
    set(gcf, 'CurrentAxes', handles.axes3)
    imshow(K), colormap(gray);
    axis off
    cont=1;

% -----
function Imagen_3_Callback(hObject, eventdata, handles)
    global K cont
    K=imread('Samples\3.bmp');
    set(gcf, 'CurrentAxes', handles.axes3)
    imshow(K), colormap(gray);
    axis off
    cont=1;

% -----
function Imagen_4_Callback(hObject, eventdata, handles)
    global K cont
    K=imread('Samples\4.bmp');
    set(gcf, 'CurrentAxes', handles.axes3)
    imshow(K), colormap(gray);

```

```

axis off
cont=1;
% -----
function Imagen_5_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\5.bmp');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_6_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\6.jpg');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_7_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\7.jpg');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_8_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\8.jpg');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_9_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\9.jpg');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_10_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\10.jpg');

```

```

set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_11_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\11.jpg');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_12_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\12.jpg');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_13_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\13.jpg');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_14_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\14.bmp');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_15_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\15.bmp');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----

```

```

function Imagen_16_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\16.bmp');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_17_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\17.bmp');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_18_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\18.png');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_19_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\19.png');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_20_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\20.png');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

% -----
function Imagen_21_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\21.jpg');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off

```

```

cont=1;

% -----
function Imagen_22_Callback(hObject, eventdata, handles)
global K cont
K=imread('Samples\22.jpg');
set(gcf, 'CurrentAxes', handles.axes3)
imshow(K), colormap(gray);
axis off
cont=1;

%Cierra el Programa
% -----
function Cerrar_menu_Callback(hObject, eventdata, handles)
close all
clear all
clc

%//////////MENÚ ACERCA DE//////////
% -----
function Acerca_de_Callback(hObject, eventdata, handles)
msgbox('Filtrado de Imágenes Ultrasonicas
Ing. Hansel Castilla Caballero
hansel.castilla@outlook.com', 'Acerca de...', 'help', 'modal');

%////////// INTERFAZ //////////
%//////////

% --- Ejecuta cuando adaptivefilter está activo.
function AdaptiveFilter_Callback(hObject, eventdata, handles)
global filter_option
set(handles.AdaptiveFilter, 'Value', 1);
set(handles.waveletfilter, 'Value', 0);
set(handles.anisotrofilter, 'Value', 0);
filter_option=1;

% --- Ejecuta cuando waveletfilter está activo.
function waveletfilter_Callback(hObject, eventdata, handles)
global filter_option
set(handles.AdaptiveFilter, 'Value', 0);
set(handles.waveletfilter, 'Value', 1);
set(handles.anisotrofilter, 'Value', 0);
filter_option=2;

% --- Ejecuta cuando anisotrofilter está activo.
function anisotrofilter_Callback(hObject, eventdata, handles)
global filter_option
set(handles.AdaptiveFilter, 'Value', 0);
set(handles.waveletfilter, 'Value', 0);

```

```

set(handles.anisotrofilter, 'Value', 1);
filter_option=3;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% POPUPMENU ABRIR IMAGEN %%%%%%%%%%%%%%%
function Cargar_Imagen_Callback(hObject, eventdata, handles)
Selector=get(handles.Cargar_Imagen, 'Value');

switch Selector
    case 1
        %Carga una Imagen
    case 2

        [filename,pathname] =
uigetfile({'*.bmp;*.jpg;*.jpeg;*.jpe;*.jfif;*.gif;*.tiff;*.tif;*.png', 'To
das los Archivos de Imagen';
        '*.bmp', 'Archivo de Mapa de Bits(*.bmp)';
        '*.jpg;*.jpeg;*.jpe;*.jfif', 'JPEG(*.jpg;*.jpeg;*.jpe;*.jfif)';
        '*.gif', 'GIF(*.gif)';
        '*.tiff;*.tif', 'TIFF(*.tiff;*.tif)';
        '*.png', 'PNG(*.png)';
        '*.*', 'Todos los Archivos'},...
        'Selecciona imagen para abrir');

        global K cont

        if isequal(filename,0)
            %Cargar Imagen Cancelada
        else
            K=imread(fullfile(pathname, filename));
            set(gcf, 'CurrentAxes', handles.axes3)
            image(K), colormap(gray);
            axis off
            cont=1;
        end
        guidata(hObject, handles);

    case 3
        global K cont
        K=imread('Samples\1.bmp');
        set(gcf, 'CurrentAxes', handles.axes3)
        imshow(K), colormap(gray);
        axis off
        cont=1;

    case 4
        global K cont
        K=imread('Samples\2.bmp');
        set(gcf, 'CurrentAxes', handles.axes3)
        imshow(K), colormap(gray);
        axis off
        cont=1;

```

```

case 5
    global K cont
    K=imread('Samples\3.bmp');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 6
    global K cont
    K=imread('Samples\4.bmp');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 7
    global K cont
    K=imread('Samples\5.bmp');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 8
    global K cont
    K=imread('Samples\6.jpg');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 9
    global K cont
    K=imread('Samples\7.jpg');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 10
    global K cont
    K=imread('Samples\8.jpg');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 11

```

```

        global K cont
        K=imread('Samples\9.jpg');
        set(gcf,'CurrentAxes',handles.axes3)
        imshow(K),colormap(gray);
        axis off
        cont=1;

case 12
    global K cont
    K=imread('Samples\10.jpg');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 13
    global K cont
    K=imread('Samples\11.jpg');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 14
    global K cont
    K=imread('Samples\12.jpg');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 15
    global K cont
    K=imread('Samples\13.jpg');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 16
    global K cont
    K=imread('Samples\14.bmp');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 17
    global K cont
    K=imread('Samples\15.bmp');
    set(gcf,'CurrentAxes',handles.axes3)

```



```

imshow(K),colormap(gray);
axis off
cont=1;

case 18
    global K cont
    K=imread('Samples\16.bmp');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 19
    global K cont
    K=imread('Samples\17.bmp');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 20
    global K cont
    K=imread('Samples\18.png');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 21
    global K cont
    K=imread('Samples\19.png');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 22
    global K cont
    K=imread('Samples\20.png');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

case 23
    global K cont
    K=imread('Samples\21.jpg');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

```

```

case 24
    global K cont
    K=imread('Samples\22.jpg');
    set(gcf,'CurrentAxes',handles.axes3)
    imshow(K),colormap(gray);
    axis off
    cont=1;

otherwise
    msgbox('Selecciona una imagen para cargar');
end
set(handles.Cargar_Imagen,'Value',1);

%////////////////// BOTÓN FILTRAR ////////////////////
function Filtrar_KeyPressFcn(hObject, eventdata, handles)
function Filtrar_Callback(hObject, eventdata, handles)
clc

global K y cont xp yp pos_act filter_option m n

if cont==0
    clc
    uiwait(msgbox('Selecciona una Imagen Ultrasónica para
Filtrar','ERROR','error','modal'));
else

    K=K(:,:,1);%Asegura la Imagen en dos dimensiones

    if filter_option==1

        try

            %Normaliza los pixeles para un mejor análisis
            K=double(K);
            mx=max(K(:));
            mn=min(K(:));
            K=(K-mn)/(mx-mn);

            %Normaliza los pixeles para un mejor análisis
            I=double(K);
            mx=max(I(:));
            mn=min(I(:));
            I=(I-mn)/(mx-mn);

            if nargin<4 || isempty(rect)

                import java.awt.Robot;
                mouse = Robot;

```

```

        figure,imshow(I,[],'notruesize'),title('Selecciona el
Área de la Ventana del Speckle'),colormap(gray),...

        set(gcf,'Position',[xp yp pos_act(3) pos_act(4)]);
        mouse.mouseMove((xp+xp+pos_act(3))/2,(yp+yp +
pos_act(4))/2);

        set(gca, 'Position', get(gca, 'OuterPosition') - ...
get(gca, 'TightInset') * [-1 0 1 0; 0 -1 0 1; 0 0 1 0; 0
0 0 1]);
        rect=getrect;
        close(gcf)
    end
    clc

    rect = rect(:); % Transforma la matriz en un vector
    homogvar=var(rect);% varianza del speckle

    hwait=waitbar(0,'Generando Filtrado Adaptativo...');

    % Filtro Adaptativo
    [m,n] = size(I);
    t = 3; % Para un tamaño de máscara de 7x7
    y = zeros(m,n);
    for k=t+1:m-t-1,
        for l=t+1:n-t-1,
            x = I(k,l);%Pixel que se va a estudiar
            x1 = I(k-t:k+t,l-t:l+t);% Genera la ventana local
            x1 = x1(:); % Transforma la matriz en un vector
            localvar = var(x1);% varianza en la ventana de
análisis

            c=1-(homogvar./localvar); %Cálculo del Coeficiente
Adaptativo

            %// Asignación del Pixel Filtrado
            if (c<=0)
                y(k,l)= mean(x1);
            end

            if (c>0 && c<1)
                y(k,l)= mean(x1)+c*(x-mean(x1)); % Expresión del
filtro adaptativo
            end

            if (c>=1)
                y(k,l)= x;
            end
        end
    end
end

```

```

        waitbar(k/(m-t-1),hwait);
    end
    %Imprime la imagen original y la imagen filtrada
    figure,imshow(K),colormap(gray);axis image, title('Imagen
original')
    set(gca, 'Position', get(gca, 'OuterPosition') - ...
get(gca, 'TightInset') * [-1 0 1 0; 0 -1 0 1; 0 0 1 0; 0 0 0
1]);

    figure,imshow(y),colormap(gray); axis image, title('Imagen
filtrada')
    set(gca, 'Position', get(gca, 'OuterPosition') - ...
get(gca, 'TightInset') * [-1 0 1 0; 0 -1 0 1; 0 0 1 0; 0 0 0
1]);

    close(hwait)

    clc
    catch
        clc
        uiwait(msgbox('ERROR -No se puede aplicar Filtrado
Adaptativo-', 'Advertencia', 'warn', 'modal'));
        close(hwait)

        clc
    end

elseif filter_option==2
    try
        hwait=waitbar(0,'Generando Filtrado Wavelet...');
        X=double(K);
        %Normaliza los pixeles para un mejor análisis
        mx=max(X(:));
        mn=min(X(:));
        X=(X-mn)/(mx-mn);

        %load wbarb;
        waitbar(20,hwait)
        %Genera las matrices de coeficientes de Wavelet de
Aproximación 1 (cA1)
        %Horizontal (cH1), Vertical (cV1) y Diagonal (cD1)
        [cA1,cH1,cV1,cD1]=dwt2(X,'bior3.7');

        %Se contruyen las Aproximaciones 1 de la imagen debido a los
coeficientes
        %encontrados
        A1 = upcoef2('a',cA1,'bior3.7',1);
        H1 = upcoef2('h',cH1,'bior3.7',1);
        V1 = upcoef2('v',cV1,'bior3.7',1);
    end
end
end

```

```

D1 = upcoef2('d',cD1,'bior3.7',1);

waitbar(40,hwait)
Xsyn = idwt2(cA1,cH1,cV1,cD1,'bior3.7');
waitbar(60,hwait)
%Transformada Wavelet Multiescala -X- es la imagen original y
2 el nivel de
%descomposición
[C,S] = wavedec2(X,2,'bior3.7');

%Se extrae el nivel de Aproximación 2
cA2 = appcoef2(C,S,'bior3.7',2);
cH2 = detcoef2('h',C,S,2);
cV2 = detcoef2('v',C,S,2);
cD2 = detcoef2('d',C,S,2);
cH1 = detcoef2('h',C,S,1);
cV1 = detcoef2('v',C,S,1);
cD1 = detcoef2('d',C,S,1);

waitbar(80,hwait)
%Reconstrucción El nivel 2 de Aproximación de C
A2 = wrcoef2('a',C,S,'bior3.7',2);
H1 = wrcoef2('h',C,S,'bior3.7',1);
V1 = wrcoef2('v',C,S,'bior3.7',1);
D1 = wrcoef2('d',C,S,'bior3.7',1);
H2 = wrcoef2('h',C,S,'bior3.7',2);
V2 = wrcoef2('v',C,S,'bior3.7',2);
D2 = wrcoef2('d',C,S,'bior3.7',2);

waitbar(100,hwait)
%Imprime la Imagen debido a la segunda descomposición
figure,imagesc(X),colormap(gray);axis image, title('Imagen
original')
set(gca, 'Position', get(gca, 'OuterPosition') - ...
get(gca, 'TightInset') * [-1 0 1 0; 0 -1 0 1; 0 0 1 0; 0 0 0
1]);

figure,colormap(gray), imagesc(wcodemat(A2,192));
title('Imagen Filtrada');axis image
set(gca, 'Position', get(gca, 'OuterPosition') - ...
get(gca, 'TightInset') * [-1 0 1 0; 0 -1 0 1; 0 0 1 0; 0 0 0
1]);

close(hwait)
clc

K=X;

catch
clc

```

```

        uiwait(msgbox('ERROR -No se puede aplicar Filtrado Wavelet-
', 'Advertencia', 'warn', 'modal'));
        close(hwait)
        clc
    end

elseif filter_option==3

    try
        niter=10;%Numero de veces que se hará filtrado
        lambda=1;%Tiempo de Detención

        %Normaliza los pixeles para un mejor análisis
        K=double(K);
        mx=max(K(:));
        mn=min(K(:));
        K=(K-mn)/(mx-mn);

        %Normaliza los pixeles para un mejor análisis
        I=double(K);
        mx=max(I(:));
        mn=min(I(:));
        I=(I-mn)/(mx-mn);

        %Pide el área de Speckle en la Imagen normalizada
        [M,N]=size(I);
        iN=[1,1:M-1];
        iS=[2:M,M];
        jW=[1,1:N-1];
        jE=[2:N,N];

        if nargin<4 || isempty(rect)

            import java.awt.Robot; mouse = Robot;
            figure,imshow(I,[],'notruesize'),title('Selecciona el
Área de la Ventana del Speckle'),colormap(gray),...
            set(gcf,'Position',[xp yp pos_act(3) pos_act(4)]);
            mouse.mouseMove((xp+xp+pos_act(3))/2,(yp+yp +
pos_act(4))/2);

            set(gca, 'Position', get(gca, 'OuterPosition') - ...
            get(gca, 'TightInset') * [-1 0 1 0; 0 -1 0 1; 0 0 1 0; 0
0 0 1]);

            rect=getrect;
            close(gcf)

        end

        clc
    end
end

```

```

%close all

I=exp(I);
hwait=waitbar(0,'Generando Coeficiente de Filtrado...');
%Filtro Anisotrópico
for iter=1:niter          Iuniform=imcrop(I,rect);
    q0_squared=(std(Iuniform(:))/mean(Iuniform(:)))^2;

    dN=I(iN,:)-I;
    dS=I(iS,:)-I;
    dW=I(:,jW)-I;
    dE=I(:,jE)-I;

    %Cálculo de la Magnitud del Gradiente
    G2=(dN.^2+dS.^2+dW.^2+dE.^2)./(I.^2+eps);
    %Cálculo del Laplaciano
    L=(dN+dS+dW+dE)./(I+eps);

    num=(0.5*G2)-((1/16)*(L.^2));
    den=(1+((1/4)*L)).^2;
    q_squared=num./(den+eps);

    %Cálculo de los Coeficientes de difusión
    den=(q_squared-
q0_squared)./(q0_squared*(1+q0_squared)+eps);
    c=1./(1+den);
    cS=c(iS,:);
    cE=c(:,jE);

    %Cálculo del Divergente
    D=(cS.*dS)+(c.*dN)+(cE.*dE)+(c.*dW);

    %// Asignación del Pixel Filtrado
    I=I+(lambda/4)*D;

    waitbar((iter/niter),hwait)
end
close(hwait)

hwait=waitbar(0,'Generando Filtrado Anisotrópico...');
I=log(I);rect = rect(:);
homogvar=var(rect);
[m,n] = size(I);t = 3;y = zeros(m,n);
for k=t+1:m-t-1,for l=t+1:n-t-1,x = I(k,l);x1 = I(k-t:k+t,l-
t:l+t);
        x1 = x1(:);localvar = var(x1);c=1-
(homogvar./localvar);
        if (c<=0)y(k,l)= mean(x1);end

```

```

        if (c>0 && c<1)y(k,l)= mean(x1)+c*(x-mean(x1));end
        if (c>=1)y(k,l)= x;end
    end
    waitbar(k/(m-t-1),hwait);
end

%Imprime la imagen original y la imagen filtrada
figure,imshow(K),colormap(gray);axis image, title('Imagen
original')
set(gca, 'Position', get(gca, 'OuterPosition') - ...
get(gca, 'TightInset') * [-1 0 1 0; 0 -1 0 1; 0 0 1 0; 0 0 0
1]);

figure,imshow(y),colormap(gray); axis image, title('Imagen
filtrada')
set(gca, 'Position', get(gca, 'OuterPosition') - ...
get(gca, 'TightInset') * [-1 0 1 0; 0 -1 0 1; 0 0 1 0; 0 0 0
1]);

close(hwait)

clc
catch
clc
uiwait(msgbox('ERROR -No se puede aplicar Filtrado
Anisotrópico-', 'Advertencia', 'warn', 'modal'));
close(hwait)

clc
end

end

end
clc

clc

%---End program-----

```