



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

PROGRAMA DE MAESTRÍA Y DOCTORADO EN
INGENIERÍA

FACULTAD DE INGENIERÍA

**MÉTODO DE PROTECCIÓN CON “MARCA DE AGUA”
PARA IMÁGENES DIGITALES, UTILIZANDO UNA TÉCNICA
DE REPRESENTACIÓN ORIENTADA GEOMÉTRICAMENTE:
CONTOURLET**

T E S I S

QUE PARA OPTAR POR EL GRADO DE:

MAESTRO EN INGENIERÍA

INGENIERÍA ELÉCTRICA – TELECOMUNICACIONES
P R E S E N T A :

ING. CARLOS ALBERTO BALBUENA CAMPUZANO

TUTOR:

DR. FRANCISCO GARCÍA UGALDE

2007





Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

JURADO ASIGNADO:

Presidente: Dra. Lucia Medina Gómez

Secretario: Dr. Boris Escalante Ramírez

Vocal: Dr. Francisco García Ugalde

1^{er} Suplente: Jorge Alberto Márquez Flores

2^{do} Suplente: María Elena Martínez Pérez

Lugar o lugares donde se realizó la tesis:

México, DF.

TUTOR DE TESIS:

Dr. Francisco García Ugalde

FIRMA

Agradecimientos

Quiero agradecer el apoyo del Dr. Francisco García Ugalde y al jurado en la elaboración de este trabajo.

A la Universidad Nacional Autónoma de México por darme una formación de posgrado, y por las facilidades otorgadas para la realización de este trabajo, así como a CONACYT por su apoyo económico.

A mis padres, hermano y a mi novia que me apoyaron a lo largo de este tiempo.

Este trabajo fue realizado con apoyo parcial de los proyectos:
UNAM-PAPIIT-IN101305 y SEP-CONACyT-41069.

INDICE GENERAL

INDICE DE FIGURAS.....	5
INDICE DE TABLAS.....	9
INDICE DE GRAFICAS.....	10
INTRODUCCIÓN.....	11
HIPOTESIS.....	12
OBJETIVOS.....	12
Objetivo Principal.....	12
Objetivos Específicos.....	12
JUSTIFICACIÓN.....	12
MÉTODO.....	13
CAPITULO 1: MARCA DE AGUA.....	14
1.1. ENCUBRIR LA INFORMACIÓN.....	15
1.1.1. INTRODUCCIÓN.....	15
1.2. MARCA DE AGUA.....	16
1.2.1. Clasificación de las marcas de agua.....	16
1.2.2. Requerimientos para Marca de Agua.....	20
1.3. ESQUEMA DE LA MARCA DE AGUA.....	20
1.3.1. Proceso de codificación.....	20
1.3.2. Proceso de decodificación.....	21
1.4. CARACTERÍSTICAS DE LAS MARCAS DE AGUA.....	22
1.5. APLICACIONES DE LAS MARCAS DE AGUA.....	23
1.6. RESEÑA DE LOS CAPÍTULOS POSTERIORES.....	25
CAPITULO 2: TRANSFORMADA CONTOURLET.....	26
2.1. INTRODUCCIÓN.....	27
2.2. WAVELETS.....	28
2.2.1. Traslaciones y Dilataciones.....	31
2.2.2. Transformada <i>wavelet</i> discreta (DWT).....	33
2.2.3. La transformada <i>wavelet</i> discreta para señales bidimensionales....	35
2.2.4. Organización de los coeficientes wavelet.....	35
2.3. CURVELET.....	37
2.4. CONTOURLET.....	39
2.4.1. Pirámide Laplaciana.....	39
2.4.2. Descomposición de Multiescala.....	43
2.4.3. Descomposición direccional.....	45
2.4.4. Direcciones Múltiples.....	49
2.4.5. Multiescala y descomposición direccional	52
2.5. LA CONTOURLET REDUNDANTE.....	56
2.5.1. Algoritmo de marcado de agua # 1 ([1]).....	57
2.5.1.1. Generación de la marca de agua e incrustación.....	57
2.5.1.2. Detección de la marca.....	59
2.5.2. Algoritmo de marcado de agua # 2 (propuesto).....	62
2.5.2.1. Creación de la marca de agua.....	62
2.5.2.2. Obtención de T y E.....	65

CAPITULO 3: RESULTADOS.....	68
3.1. INTRODUCCIÓN.....	69
3.2. PARÁMETROS DE LAS IMÁGENES.....	70
3.2.1. Histograma.....	70
3.2.2. Varianza.....	70
3.2.3. Entropía.....	70
3.2.4. Relación señal a ruido pico (PSNR).....	72
3.3. RESULTADOS ENFOCADOS A CAPAS MARCADAS.....	72
3.4. RESULTADOS RESPECTO A MARCAS DE AGUA NO ORIGINALES..	76
3.5. RESULTADOS DE LAS PRUEBAS DE ROBUSTEZ.....	77
3.5.1. Prueba de Ruido.....	77
3.5.2. Prueba de filtrado de mediana.....	79
3.5.3. Prueba de resistencia a la compresión JPEG.....	80
3.5.4. Pruebas Geométricas.....	82
3.6. RESULTADOS DE PRUEBAS CON OTRAS IMÁGENES.....	83
CAPITULO 4: CONCLUSIONES.....	89
REFERENCIAS.....	93

INDICE DE FIGURAS

Capítulo 1

Figura 1.1. Técnicas para ocultar información.....	15
Figura 1.2. Imagen Lena con marca de agua invisible.....	18
Figura 1.3. Imagen con marca de agua visible.	18
Figura 1.4. Marca de Agua Pseudo-aleatoria.....	19
Figura 1.5. Técnicas para marca de agua.	19
Figura 1.6. Codificador.....	20
Figura 1.7. Codificador con llave.	21
Figura 1.8. Decodificador para I.	21
Figura 1.9. Decodificador para X.	22

Capítulo 2

Figura 2.1. Esquema del análisis <i>wavelet</i>	29
Figura 2.2. <i>Wavelets</i> importantes.	30
Figura 2.3. a) Ejemplo de función <i>wavelet</i> . b) <i>Wavelet</i> desplazada.	31
Figura 2.4. Función <i>wavelet daubechies</i> con diferentes valores de escala 'a'. a) <i>Wavelet</i> con 'a'=1. b) <i>Wavelet</i> con 'a'=1/2. c) <i>Wavelet</i> con 'a'=1/4.	32
Figura 2.5. Ejemplos de diferente tipo de análisis. En tiempo, frecuencia y escala.....	32
Figura 2.6. Diferencia entre la transformada Fourier y <i>Wavelet</i>	33
Figura 2.7. Proceso de descomposición (análisis).	34
Figura 2.8. Proceso de descomposición y reconstrucción (síntesis).	34
Figura 2.9. Descomposición <i>wavelet</i> de primer nivel.	35
Figura 2.10. a) Algoritmo de pirámide de las <i>wavelets</i> de dos niveles. b) Esquema de la organización de los coeficientes <i>wavelets</i> . c) Pirámide de coeficientes <i>wavelets</i>	36

Figura 2.11. a) Imagen original de Bárbara. b) Organización de los Coeficientes <i>wavelet</i> en dos niveles c) Árboles de descendencia <i>wavelets</i>	37
Figura 2.12. Pirámide Gaussiana.....	39
Figura 2.13. Primeras cuatro imágenes de la pirámide Gaussiana (línea superior) y pirámide Laplaciana (línea inferior).....	40
Figura 2.14. Pirámide Laplaciana.....	41
Figura 2.15. Dos transformadas para imágenes con contornos suaves. a) Transformada curvelet: transformadas de bloque ridgelet son aplicadas a las subbandas de la imagen. b) Transformada Contourlet: una imagen es descompuesta por una estructura con doble banco de filtros, donde el primero captura los puntos de eje y el segundo enlaza estos ejes con los contornos. Las áreas grises en los cuadros representan los soportes espaciales de los filtros.....	42
Figura 2.16: Esquema de la pirámide laplaciana. a) Análisis: las salidas son una aproximación robusta <i>c</i> y una diferencia <i>d</i> entre la señal original y la predicción. El proceso puede ser iterado por descomposición sucesiva de la versión robusta. b) Esquema propuesto de la reconstrucción para la pirámide.....	43
Figura 2.17. Subespacios generados por la pirámide laplaciana.	44
Figura 2.18: a) Primer nivel de filtros en forma de ventilador. b) Segundo nivel con filtros cuadrados. c) Combinación de los dos niveles. d) Banco de filtros direccional particionando las frecuencias, donde $n = 3$ y hay $2^3 = 8$ bandas frecuenciales.....	46
Figura 2.19. a) Enrejado de muestreo romboidal. b) Partición del espectro bidimensional usando el banco de filtros con soporte en forma de rombo, con filtros en forma de hélice. Las regiones negras representan la frecuencia ideal soportada por cada filtro. c) Primeros dos niveles de un DFB.....	47
Figura 2.20. Ejemplo de una operación de remuestreo que es usado efectivamente como una operación de rotación para la descomposición con un banco de filtros direccionales DFB. a) La imagen “cameraman” b) La imagen “cameraman” después del remuestreo.....	47
Figura 2.21. Respuestas al impulso de 32 filtros equivalentes para la primera mitad de los canales de un DFB de 6 niveles los cuales usan filtros Haar. Los cuadros negros y grises corresponden a valores +1 y -1 respectivamente. Ya que las funciones de base se asemejan a “líneas locales”, reciben el nombre de Radonlets.....	48
Figura 2.22: Subespacios multidireccionales generados por un banco de filtros direccionales (DFB).	49

Figura 2.23. Subespacios generados multidireccionalmente por el PDFB.	50
Figura 2.24: Banco de filtros direccional piramidal. a) Bloques. Primero, una descomposición estándar multiescala, en 8 bandas, donde el canal pasa bajas es submuestreado mientras que el pasa altas no. Después, una descomposición direccional con un banco de filtros DFB es aplicado a cada canal pasa altas. b) División de frecuencias resultante, donde el número de direcciones es incrementado con la frecuencia.	52
Figura 2.25: Ilustración de imágenes de base para una contourlet que satisfacen la relación del escalamiento de la curva. Desde la línea superior a la inferior, la escala es reducida por cuatro mientras el número de direcciones es duplicado.	53
Figura 2.26. Cuadrículas incrustadas de la aproximación en el dominio espacial para un banco de filtros direccional piramidal PDFB que implementan la transformada curvelet. Éstos son 4 subespacios ilustrativos burdos y finos $W_{f,k}$ que representan escalas y direcciones básicamente horizontales y verticales. Cada subespacio es recorrido por la función de la curvelet dada en (2.23)	54
Figura 2.27. Ejemplos de la transformada contourlet de las imágenes de “Peppers” y “Bárbara”. Para una visualización más clara, cada imagen esta solamente descompuesta en dos niveles de la pirámide, los cuales están a su vez descompuestos en 4 y 8 subbandas direccionales. Pequeños valores de coeficientes son mostrados en negro mientras que grandes valores son mostrados en blanco.....	55
Figura 2.28. Comparación de las transformadas: a) Tercer nivel de 2-D Wavelet <i>daubechies</i> y b) Tercer nivel de la transformada Contourlet.....	56
Figura 2.29: Representación frecuencial de una descomposición contourlet con 3 niveles de resolución ($L = 2$) y 4 direcciones de frecuencia ($D = 4$). La aproximación burda de la imagen C_3 no es descompuesta direccionalmente.	57
Figura 2.30. Diagrama de Flujo del sistema propuesto en [1].....	58
Figura 2.31: La imagen Lena marcada y su diferencia absoluta con la imagen original, magnificada por un factor de 8.	60
Figura 2.32: Subbandas contourlet de la imagen Lena.....	60
Figura 2.33: Imagen burda.....	61
Figura 2.34. Las 4 mascararas de peso direccionales, respectivamente, M1, M2, M3, M4 (de arriba abajo y de izquierda a derecha) calculado a partir de la contourlet redundante de Lena con $L = 3$ y $D = 4$	61

Figura 2.35: Diagrama de Flujo del algoritmo propuesto.	63
Figura 2.36: Diagrama de flujo para la inserción de la marca, por medio de la mascara de peso direccional.....	64
Figura 2.37. Vecindad incompleta en una ventana de 3x3, en los márgenes de la imagen.	65
Figura 2.38. Píxeles faltantes reemplazados por el valor del píxel más cercano, en los márgenes de la imagen.	66

Capítulo 3

Figura 3.1. a) Imagen Original “Lena 512 x 512”. b) Histograma c)Parámetros importantes.....	71
Figura 3.2. a) Imagen Marcada en las cuatro primeras capas con el algoritmo ref. [1]. b) Imagen Marcada en las cuatro primeras capas con el algoritmo propuesto.....	73
Figura 3.3. a) Imagen Marcada en todas las capas con el algoritmo ref [1]. b) Histograma de la imagen.	74
Figura 3.4. a) Imagen Marcada en todas las capas con el algoritmo propuesto. b) Histograma de la imagen.	74
Figura 3.5 Ejemplos de ataques geométricos.....	83
Figura 3.6. a) Imagen Original “Bárbara” b) Histograma de “Bárbara”.....	84
Figura 3.7 a) Imagen “Bárbara” marcada en 12 capas con algoritmo #2 b) Histograma de a).....	85
Figura 3.8 a) Imagen “Bárbara” marcada en 12 capas con algoritmo #1 b) Histograma de a).....	85
Figura 3.9 a) Imagen “Fishing boat” b) Histograma de “Fishing boat”.....	87
Figura 3.10 a) Histograma de “Fishing boat” con marcado en 12 capas con algoritmo #1 b) Histograma de “Fishing boat” ” con marcado en 12 capas con algoritmo #2.....	87
Figura 3.11 a) Imagen “Fishing boat” marcada en 1 capas con algoritmo #1 y con factor de compresión JPEG 100 b) Imagen “Lena” marcada en 1 capas con algoritmo #1 y con factor de compresión JPEG 100.....	88

INDICE DE TABLAS

Capítulo 1

Tabla 1.1. Marca Visible y Marca invisible.....	17
---	----

Capítulo 2

Tabla 2.1. Ejemplos de funciones madre	30
--	----

Capítulo 3

Tabla 3.1. Parámetros obtenidos en la imagen marcada al insertar la marca de agua en una capa.....	72
--	----

Tabla 3.2. Parámetros obtenidos en la imagen marcada al insertar la marca de agua en cuatro capas.....	73
--	----

Tabla 3.3. Parámetros obtenidos en la imagen marcada al insertar la marca de agua en todas las capas (12 capas).....	73
--	----

INDICE DE GRAFICAS

Capítulo 3

Gráfica 3.1. Grafica de los valores PSNR de ambos algoritmos.....	75
Gráfica 3.2. Grafica de los valores R' y T del algoritmo ref. [1].....	75
Gráfica 3.3. Grafica de los valores R' y T del algoritmo propuesto.....	76
Grafica 3.4. Detección de marcas de agua no originales.....	77
Gráfica 3.5. Resultado a la prueba de ruido.....	78
Gráfica 3.6. Resultado a la prueba de ruido.....	78
Gráfica 3.7. Resultado de la prueba de filtrado de mediana para el algoritmo ref.[1].....	79
Gráfica 3.8. Resultado de la prueba de filtrado de mediana para el algoritmo Propuesto.....	80
Grafica 3.9. Resultado de la resistencia a la compresión JPEG para el algoritmo ref. [1]	81
Grafica 3.10. Resultado de la resistencia a la compresión JPEG para el algoritmo propuesto	81
Gráfica 3.11. Valores de detección de la imagen “Bárbara” (algoritmo #2).....	84
Gráfica 3.12. Valores de detección de la imagen “Fishing boat” (algoritmo #2)...	86
Gráfica 3.13. PSNR obtenidos al analizar las imágenes “Bárbara“ y “Fishing boat” (algoritmo #2).....	86

INTRODUCCIÓN

En la actualidad existe una gran cantidad de intercambio de información digital, por ejemplo a través del Internet, donde no hay un control estricto de distribución, debido a esto es importante tener un control de derechos de autor en cuestión de copias ilegales.

La llamada “marca de agua digital” es una herramienta que está siendo utilizada para detectar la autenticidad de imágenes digitales. Existen diferentes tipos de marca de agua y diferentes métodos de inserción.

Las principales herramientas que se utilizan en el marcado de agua, explicadas en este trabajo son: la transformada *wavelet* discreta, la cual normalmente se utiliza para la codificación de señales; la transformada *contourlet*, desarrollada por Minh N. Do y Martin Vetterli [2], que puede capturar en buena medida los contornos unidireccionales en las imágenes con pocos coeficientes; y la transformada *contourlet* redundante, que usa información redundante para una mejor manipulación de dicha transformada. Dichas transformadas son la base para la realización del algoritmo propuesto por Nadia Baaziz [1] (basado en la inserción de la marca de agua en el dominio de la transformada *contourlet* redundante, y el cual utiliza un umbral para la detección de la marca, determinado por medidas que tienen que ver con el sistema visual humano HVS). Con el objetivo de mejorar la relación señal pico a ruido (PSNR) de la imagen marcada, en este trabajo se realizó una comprobación de ese trabajo publicado, y se agregaron algunas modificaciones al mismo.

Los cambios que se realizan al algoritmo [1] permiten obtener una menor cantidad de información en la marca de agua, por medio de la obtención de la transformada *contourlet* redundante de la marca de agua misma. Sin embargo, como existe una relación directa entre los parámetros PSNR y la robustez del sistema, dicho cambio hace que efectivamente la razón PSNR de la imagen marcada se incremente, pero haciendo que los parámetros utilizados en la detección de la marca de agua sean menos eficientes.

En el desarrollo del documento, son mostrados los resultados obtenidos después de realizar pruebas para analizar el comportamiento del algoritmo propuesto con respecto al algoritmo original [1]. Concretamente, se hicieron comparaciones en relación al número de capas marcadas y de la robustez del algoritmo a diferentes ataques contra la marca de agua, como son la prueba de inserción de ruido, compresión JPEG, y ataques geométricos.

Los resultados de dichas pruebas demuestran que efectivamente en el esquema propuesto, el valor de la razón PSNR aumenta en todos los casos de marcado en comparación con el algoritmo [1], y que a pesar de que los valores referentes a la detección de la marca de agua, generan un menor intervalo entre el valor obtenido y el umbral establecido, siguen permitiendo alcanzar el resultado correcto al decidir si existe, o no, la marca de agua.

Hipótesis:

Debido a las características de la transformada contourlet redundante y al algoritmo utilizado en [1], el valor de PSNR de la imagen marcada puede ser mejorado realizando cambios al diagrama de flujo, los cuales en contraparte disminuirán la información de la marca de agua. En consecuencia, la detección de ésta será más difícil, sin embargo se deberá poder detectar la marca de agua aun cuando se apliquen algunos ataques a la misma.

Objetivo:

El principal objetivo de esta tesis es proponer un nuevo algoritmo, que permita insertar una marca de agua binaria invisible en una imagen digital de escala de grises usando la transformada contourlet, esto buscando mantener la mejor razón señal pico a ruido PSNR de la imagen marcada, con respecto a otros algoritmos los cuales usan la misma transformada.

Objetivos Específicos:

- El trabajo publicado en [1] es tomado como referencia, en este trabajo de tesis. Se busca mantener, o mejorar, la razón PSNR de la imagen marcada con respecto a la imagen original.
- Lo anterior sin afectar la robustez del algoritmo, contra los ataques expuestos en ese mismo trabajo y contra otros conocidos en la literatura.

Justificación:

El intercambio de información digital se incrementa cada día, y con ello el comercio de dicha información.

Debido a que es importante encontrar el medio para identificar al propietario de la información digital original, un medio que se ha desarrollado, consiste en técnicas de inserción de marcas de agua digitales, las cuales consisten en códigos de identificación, sellos, firmas, etc. El principal objetivo de dichas marcas es la identificación de un posible infractor al encontrar alguna copia no autorizada, y así poder reclamar la propiedad intelectual de la información digital genuina.

Actualmente se requieren nuevos métodos y/o mejoramiento de inserción y detección de marcas de agua, hay que tomar en cuenta que en ciertos métodos su mejor robustez es el no detectar que existe marca de agua, por lo que se pretende mejorar un algoritmo para tener una mejor invisibilidad -existen técnicas que aprovechan las propiedades del sistema de visión humano, lo cual es indispensable para una buena invisibilidad de la marca de agua.- y que no pierda su robustez contra algunos ataques conocidos en la literatura especializada.

Método:

En el procesamiento digital de imágenes el uso de transformaciones que permiten representar la información en otro dominio es de fundamental importancia. Desde principios de los años 80's unas nuevas funciones conocidas como *wavelets* permitieron una mejor representación de la información en el plano "tiempo (o espacio) – frecuencia (o escala)". Para propósitos de análisis de la información contenida en una imagen digital, a principios de los 90's, estas funciones han sido complementadas con otras llamadas *contourlets*, las cuales permiten separar la información de una imagen digital en varias orientaciones espaciales. Tomando en cuenta las propiedades del sistema de percepción visual estas transformaciones tienen mucha relevancia. A partir de esto, se decidió estudiar en esta tesis su uso para cubrir uno de los requerimientos mayores de las ciencias de la información, concerniente a la protección intelectual de la información digital por medio de "marcas de agua". Se valora en esta tesis un método que ya fue publicado por otros autores para tomarlo como punto de partida, y se propone una modificación al mismo, cuyo desempeño es evaluado en términos de la razón señal pico a ruido PSNR y de la robustez respecto a algunos ataques conocidos en la literatura científica especializada.

1.1. Encubrir la Información

1.1.1. Introducción

La producción y distribución de multimedia en nuestros días en su mayoría digital. Esto da ventajas como: el procesamiento de dicha información y la posibilidad de controlar el ruido de los canales de transmisión, la utilización de *software* en lugar de *hardware* y, la configuración de sistemas con mayor facilidad, las cuales favorecen el uso de dicha tecnología digital. Por tal motivo, la transferencia de información por medios electrónicos, es cada día mayor. Sin embargo, una gran desventaja de dicha distribución digital es que existe la posibilidad de copiar los datos digitales, haciendo una copia ilegal sin pérdida de calidad o fidelidad. Lo cual implicaría una considerable pérdida financiera.

Por lo tanto son necesarios medios para proteger dicha información, ya sea ocultarla, o marcarla para la autenticación de los derechos de autor y, con ello la remuneración por la información.

En la figura 1.1 es mostrado un desglose de las técnicas para ocultar información resaltando donde se encuentra la marca de agua invisible usada en este trabajo.

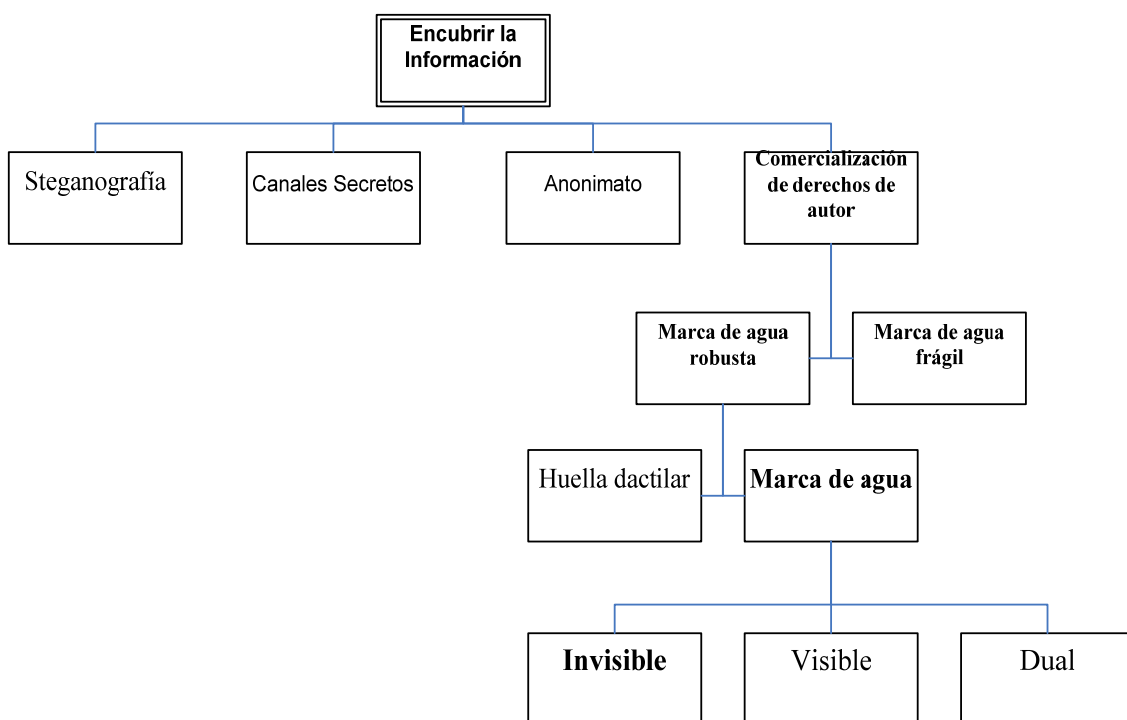


Figura 1.1. Técnicas para ocultar información.

1.2. Marca de Agua

Las marcas de agua han sido usadas para este propósito, permitiendo identificación de la fuente, autor, propietario, distribuidor y consumidor; en imágenes digitales, grabaciones de audio o video. La principal ventaja de estos sistemas consiste en que la marca es inseparable del contenido del archivo.

Haciendo referencia al tema principal de este trabajo, las marcas de agua para imágenes, la definición más apropiada es la siguiente:

La marca de agua digital es un patrón o una señal digital insertada en una imagen y sus características pueden y / o deben ser:

- Identificar al receptor previsto (único a cada copia).
- Identificar la fuente del documento (común para múltiples copias).
- La información marcada (imagen) es una información transformada.
- La información (imagen) queda intacta, reconocible.
- Quedando igual en visualización, impresión, retransmisión y diseminación, en el caso de imágenes.
- Contrastando con la huella dactilar y cifrado de datos.
 1. En una huella dactilar digital, un nuevo archivo es creado para describir el archivo original. (*e.g., checksum in Tripwire*).
 2. El cifrado transforma una imagen en una información irreconocible.

1.2.1. Clasificación de las Marcas de Agua

Las marcas de agua se pueden clasificar de acuerdo a la técnica utilizada y al resultado físico (si la marca de agua se puede identificar a simple vista o no).

A. Técnica utilizada.

- Marcado en el dominio espacial.
 1. Llevar al orden de bit menor del píxel escogido.
 2. Sobreponer una marca.
 3. Separación de color: marca de agua en una sola banda de color.
- Marcado en el dominio de frecuencia.
 1. Transformada de Fourier: Una de las transformadas utilizadas en las marcas de agua debido a sus características como lo es la robustez ante manipulaciones geométricas, escalamiento, compresión e imperceptibilidad. La transformada discreta de Fourier bidimensional ésta dada por la ecuación (1.1)

$$I'(u, v) = \frac{1}{MN} \sum_{i=1}^{M-1} \sum_{j=1}^{N-1} I(i, j) \exp(-j2\pi(\frac{ui}{M} + \frac{vj}{N})) \quad (1.1)$$

2. Transformada Wavelet: Nueva herramienta en el campo de procesamiento de imágenes la cual es explicada en el capítulo 2.
3. Transformada Contourlet.: Herramienta que toma como base la transformada wavelet, la cual es explicada en el siguiente capítulo y usada en el algoritmo desarrollado en éste trabajo.

B. Resultado físico.

- Marca de agua visible.
 1. Similar a las contrapartes gráficas (estampas digitales).
- Marca de agua invisible.
 1. Usadas para identificar la fuente, autor, propietario, distribuidor o consumidor autorizado.
 2. Es permanente, y la marca no altera la imagen.
 3. También se pueden utilizar para dar un seguimiento a las imágenes en el proceso de su distribución ilícita. Esto es con una marca de agua única para cada comprador.

En la tabla 1.1 son mostrados los principales propósitos de las marcas de agua visibles e invisibles, considerando si dicho propósito es primario o secundario o en dado caso que no aplique para cada tipo de marca.

Tabla 1.1. Marca visible y marca invisible.

Propósito	Visible	Invisible
Validación para el receptor previsto, inicial.	NA	Primario
Transmisión no-rechazable.	NA	Primario
Disuasión contra hurto.	Primario	Secundario
Disminuir el valor comercial sin utilidad.	Primario	Primario
Desalentar la duplicación desautorizada.	Primario	Secundario
Notarización digital y autenticación.	Primario	Secundario
Identificación de la fuente.	Primario	Secundario

La figura 1.2 muestra la imagen Lena con una marca de agua invisible mientras que en la figura 1.3 se puede ver una marca de agua visible con la leyenda demo sobre un paisaje de un lago.



Figura 1.2 Imagen Lena con marca de agua invisible.



Figura 1.3. Imagen con marca de agua visible.

En la figura 1.4 es mostrado un ejemplo de marca pseudo-aleatoria, comúnmente usada como marca de agua invisible.

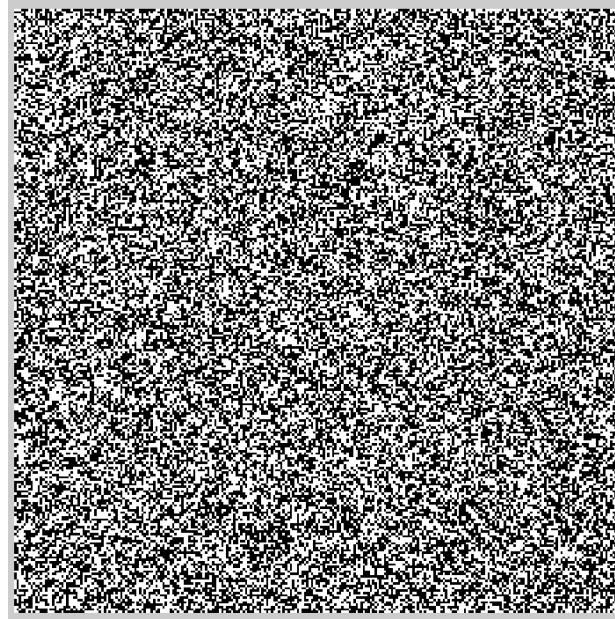


Figura 1.4. Marca de Agua Pseudo-aleatoria.

Como resumen y para una mejor comprensión en la figura 1.5 se clasifican los diferentes tipos de marcas de agua, donde ésta se divide en el dominio de trabajo, al tipo de documento, a la percepción humana y a la aplicación y las subdivisiones de cada una de ellas.

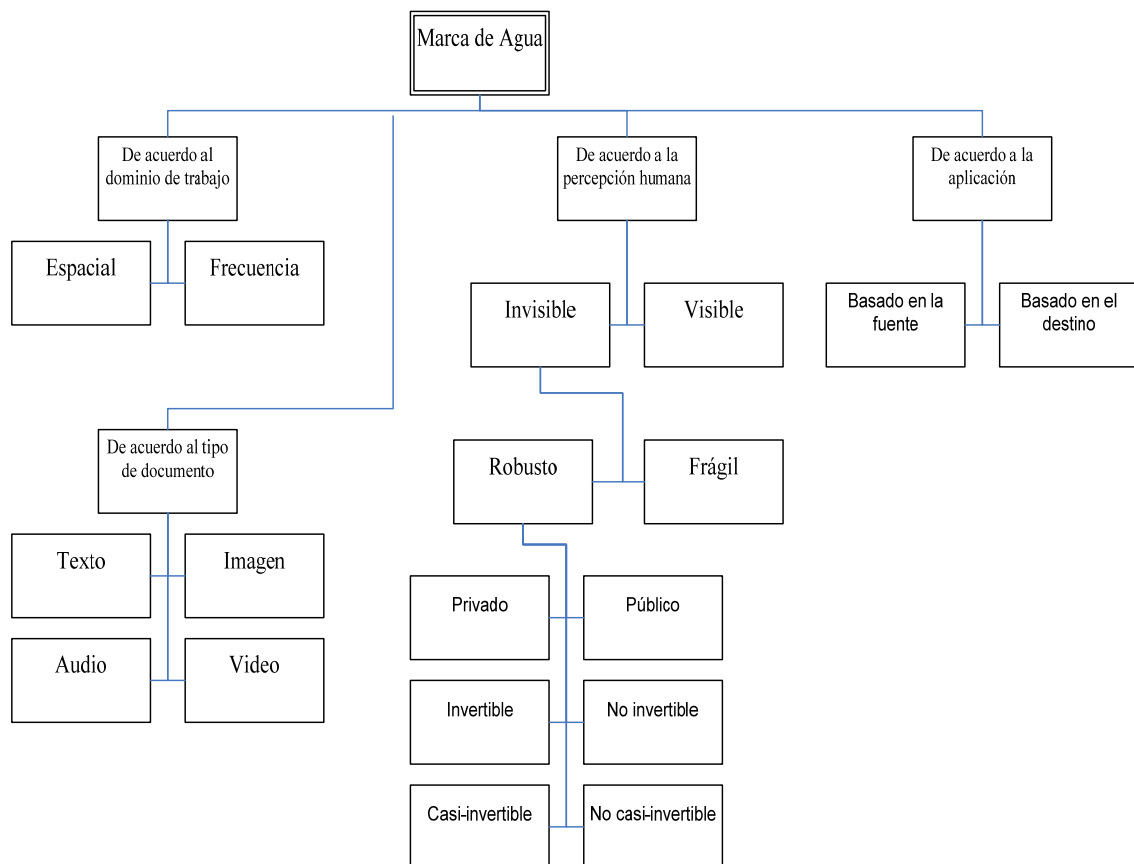


Figura 1.5. Técnicas para marca de agua.

1.2.2. Requerimientos para Marca de Agua

- Para proteger propiedad intelectual
 1. La marca de agua debe de ser difícil o imposible de remover, al menos sin degradación de la información original.
 2. La marca de agua debe sobrevivir sobre modificaciones de la imagen.
 3. Una marca de agua invisible debe ser imperceptible para no afectar la experiencia de la visión.
 4. Las marcas de agua deben ser fácilmente detectables por su propio autor.

1.3. Esquema de la marca de agua

Ya que la marca de agua es insertada directamente en el contenido de un archivo multimedia, este sistema está compuesto por dos módulos principales: proceso de codificación (o inserción) de la marca de agua y decodificación (o extracción) de la misma.

1.3.1. Proceso de codificación.

El módulo codificador mostrado en la figura 1.6 y descrito por las ecuaciones (1.2 y 1.3) realiza la inserción de la marca de agua X en la información original I para crear la información marcada I' , que debe ser visualmente similar a I .

$$I'(i, j) = I(i, j) + X(i, j) \quad (1.2)$$

$$I'(i, j) = f(I(i, j), X(i, j)) \quad (1.3)$$

Donde:

- $I'(i, j)$: Imagen marcada de dimensiones $i \times j$.
- $I(i, j)$: Imagen original de dimensiones $i \times j$.
- $X(i, j)$: Marca de agua de dimensiones $i \times j$.

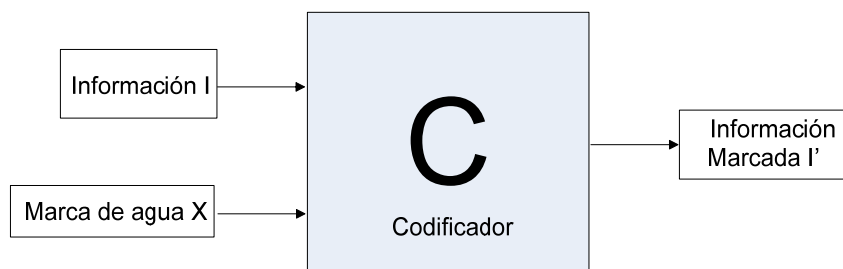


Figura 1.6. Codificador

La marca de agua, normalmente puede recibir un tratamiento antes de la inserción en la información I , obteniendo un X_2 que depende de la llave K , donde K es una información que controla la operación del algoritmo criptográfico. Un ejemplo de ésta codificación es la figura 1.7, y ecuaciones (1.4 y 1.5).

1. Dependier de una llave K .

$$X_2(i, j) = f(X(i, j), K) \quad (1.4)$$

2. Dependier de una llave K y de la información I .

$$X_2(i, j) = f(X(i, j), K, I(i, j)) \quad (1.5)$$

Donde:

$X(i, j)$: Marca de agua original de dimensiones $i \times j$.

$X_2(i, j)$: Marca de agua tratada de dimensiones $i \times j$.

$I(i, j)$: Imagen original de dimensiones $i \times j$.

K : Llave.

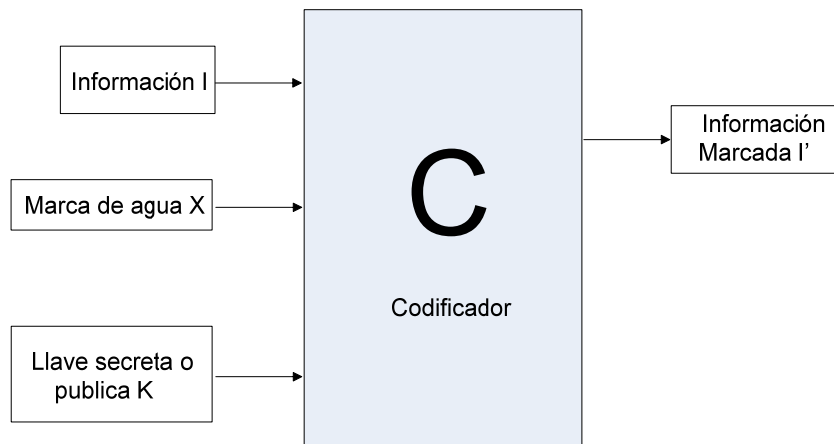


Figura 1.7. Codificador con llave.

1.3.2. Proceso de decodificación

El módulo decodificador realiza en primer lugar la extracción de la marca extraída denominada X^* de una información, cuyos derechos de propiedad se desean probar, I^* , posiblemente manipulada o distorsionada, haciendo uso o no (esto depende de la técnica de extracción) de la información original I . A continuación establece el parecido entre la marca extraída X^* y la marca original X , calculando el valor de un índice de similitud entre ambas. Finalmente utiliza una función de comparación (e.g., un umbral T , un valor de relación señal a ruido) para determinar si la información *prueba* I^* es una versión marcada de la información original I . Las figuras 1.8 y 1.9 son ejemplos de decodificadores.

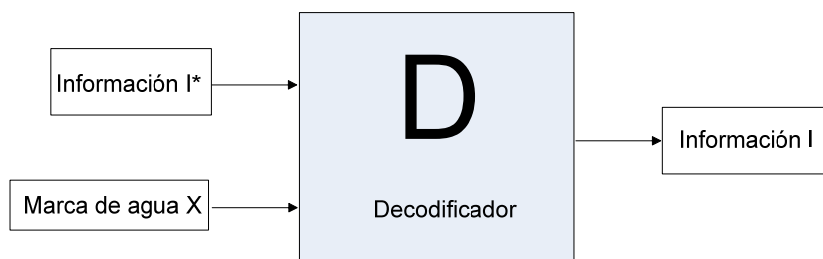


Figura 1.8. Decodificador para I .

Recuperación de la marca:

1. Usando la llave y la información original

$$X'(i, j) = g(I'(i, j), K, I(i, j)) \quad (1.6)$$

2. Sin usar la información original

$$X'(i, j) = g(I'(i, j), K) \quad (1.7)$$

Donde:

$X'(i,j)$: Marca de agua recuperada de dimensiones $i \times j$.

$I'(i,j)$: Imagen marcada de dimensiones $i \times j$.

$I(i,j)$: Imagen original de dimensiones $i \times j$.

K : Llave.

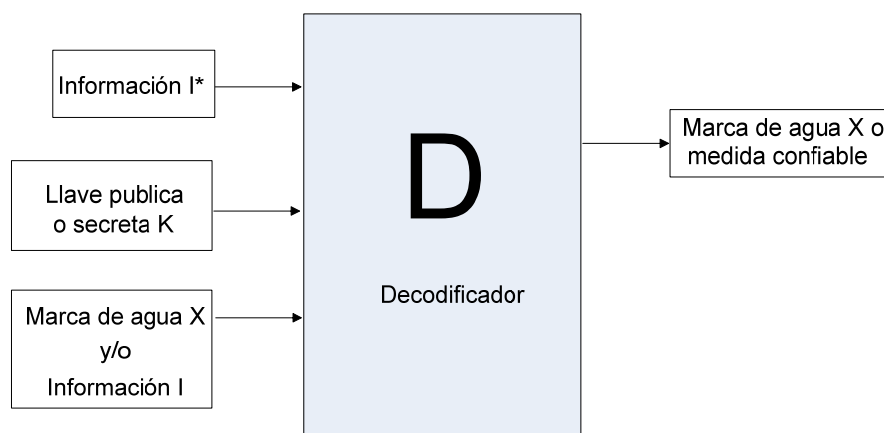


Figura 1.9. Decodificador para X.

1.4. Características de las marcas de agua

Ya que los sistemas de seguridad no están basados en la no publicación de los algoritmos, sino en su fortaleza y en la seguridad de la clave, los algoritmos usados para las marcas de agua deben de cumplir con las siguientes propiedades:

1. Robustez

Todos los archivos digitales son o pueden ser expuestos a modificaciones: pérdidas por compresión, cambios producidos por el mejoramiento de imágenes, la amplificación de las señales de audio, etc. Una marca de agua es considerada robusta si después de estas modificaciones se conserva. En el caso de imágenes y video, debe persistir también, después de transformaciones geométricas tales como recortado, rotación y escalado.

2. Resistencia a manipulaciones

Este punto tiene relación con el aspecto de seguridad ya que se refiere a la resistencia de los ataques basados en el conocimiento de los algoritmos

utilizados y de la información marcada. Quedando sólo como incógnita la clave o llave utilizada en el proceso. Los ataques básicos son:

- a) Ataques activos: El objetivo es eliminar la marca de agua o hacerla indetectable, dañino para algunas aplicaciones. Entre los ataques activos pueden citarse los siguientes:
 - Ataques por cálculo de un promedio estadístico.
 - Ataques al detector de la marca.
 - Ataques al dispositivo que inserta la marca.
 - Ataque por confabulación.
 - Falsificación.
- b) Ataque pasivos: Para este tipo de ataque sólo se pretende detectar la marca, lo cual no tiene gran trascendencia para las aplicaciones comunes.

3. Imperceptibilidad e indetectabilidad (marca invisible)

La imperceptibilidad es un concepto el cual tiene relación con el sistema perceptual humano. Si la degradación, causada por insertar la marca de agua en los datos, no es fácil de apreciar cuando el sistema cumple con esta característica.

En cambio la indetectabilidad refiere al modelo estadístico antes y después de la inserción de la marca de agua, lo cual es necesario para que no se detecte que existe una marca de agua en la información.

4. Viabilidad del sistema

Como todo sistema en ingeniería, este tipo de sistema necesita tener un balance entre costos, tiempo de procesamiento y características para el sistema, según la aplicación en la que se esté interesado.

1.5. Aplicaciones de las marcas de agua

1. Marcas de agua como firmas

Las marcas de agua pueden ser utilizadas como firma para derechos de autor o propiedad. Para una simple identificación de la información sólo se aplica una marca visible con la firma de propiedad. O bien se puede realizar una prueba de propiedad con la marca de agua incrustada.

2. Marcas de agua para huellas digitales (*fingerprinting*)

Esta marca de agua es utilizada para saber quién es el usuario o comprador de la información y así saber quien hace distribución ilegal de dicha información.

3. Marcas de agua para autenticación

La marca para autenticación es usada para aplicaciones en las cuales es crucial saber si la información es veraz, ya que en la transmisión de dicha información, ésta pudo ser modificada, la marca después de ser insertada, sufre un cambio si la información es modificada y si es posible, se detecta dónde es que sufrió el cambio efectuado.

4. Monitoreo de las transmisiones de radiodifusión

Al haber usado las marcas de agua en las anteriores aplicaciones, se pueden usar sistemas que monitoricen dichas marcas, en transmisiones de radiodifusión y así tener pleno conocimiento de quién usa y/o transmite la información en la que se está interesado.

5. Control de copias

Dicha marcas deben contener información determinada por su propietario, como reglas de uso y copiado de archivos.

6. Comunicaciones secretas

En esta aplicación la marca es incrustada con información secreta sin que nadie externo a la comunicación sepa que existe.

1.6. Reseña de los capítulos posteriores.

En el *segundo capítulo* son explicados en forma simple las principales características de las transformaciones relacionadas con el mecanismo de marcado utilizado en esta tesis, (transformada *wavelet*, transformada *wavelet* discreta, curvelet, ridgelets).

También se define la principal transformada en este trabajo de investigación la cual es usada para la aplicación de la marca de agua: La Transformada Contourlet, para la cual se explica en detalle su construcción: descomposición en multiescala, descomposición direccional, y la conjunción de ambos, para después definir la “Transformada Contourlet Redundante”.

La transformada contourlet redundante es explicada según el trabajo realizado por Nadia Baaziz [1], referido en ese mismo capítulo.

También son explicados los dos algoritmos para marcar una imagen, los cuales utilizan la transformada contourlet redundante, el primero, desarrollado en [1], antes mencionado, precedente para el segundo algoritmo, propuesto en esta tesis.

Son presentadas imágenes las cuales ilustran las explicaciones tanto de las transformadas relacionadas, como de la transformada contourlet y contourlet redundante.

En el *tercer capítulo* se expondrán los resultados obtenidos.

Los resultados comprenden una comparación entre ambos algoritmos (algoritmo de la referencia [1] y el algoritmo propuesto), basada en las medidas más importantes del procesamiento de imágenes, como son: relación señal pico a ruido (PSNR), histograma de las imágenes, varianza, así como comparación en robustez de los algoritmos por medio de diferentes pruebas de ataques simulados.

También son mostradas gráficas con los diferentes parámetros que pueden ser modificados en los diferentes algoritmos, así como imágenes que muestran los resultados obtenidos, para tener así una comparación visual de los mismos.

Para las pruebas de robustez en las imágenes generadas es usado el programa *StirMark Benchmark 4* [17] del cual se hace referencia en la respectiva parte del capítulo tercero.

A partir de los resultados obtenidos y de los resultados esperados, debido a las modificaciones del algoritmo original [1], y con respecto a la teoría explicada anteriormente, se argumentan las *conclusiones*.

2.1. Introducción

Para ciertas aplicaciones en procesamiento de la información que existen en la actualidad, la representación de ésta es muy importante y debe ser bastante eficiente, es decir, que el error sea mínimo en los procesamientos de dicha información. Éstos pueden ser: compresión, filtrado, extracción, transmisión, y otros.

Para señales unidimensionales con segmentos lisos, como son las líneas en una imagen, la principal herramienta considerada es la transformada *wavelet*, ya que da una aproximación en el plano espacio-escala, óptima de dichas señales.

La transformada *contourlet* [2], es una extensión de la transformada *wavelet* en dos dimensiones usando filtros direccionales y no separables. Una de las características más importantes, es la de capturar en buena forma los contornos unidireccionales en las imágenes con pocos coeficientes. Minh N. Do y Martín Vitterli [2] proponen dicho sistema, el cual provee un esquema multiresolución, local y direccional.

Una particularidad de la transformada *contourlet* es que su construcción empieza en el dominio discreto usando un banco de filtros iterados, los cuales pueden ser diseñados con relación a las curvas y dicho dominio converge al dominio continuo, esto ofrece una descomposición rápida y estructurada la cual puede capturar la estructura geométrica de la información de una imagen digital vía una expansión en multiresolución, la cual es muy útil para análisis y procesamiento de imágenes.

2.2. Wavelets

Las transformaciones matemáticas son aplicadas a señales con el objetivo de obtener más información de la que se puede obtener en el dominio espacial, una de las transformadas mas importantes es la de Fourier, la cual descompone una señal en sus componentes sinusoidales de diferentes frecuencias.

La transformada Fourier esta definida por (2.1a) y como se menciono permite determinar las frecuencias de la señal, sin embargo, se pierde la información de tipo temporal de la señal. Debido a esto, Dennis Gabor adopto la transformada de Fourier para poder analizar una pequeña sección de la señal, en un determinado tiempo la cual es conocida como transformada Gabor o STFT (Short Time Fourier Transform) (2.1b).

$$X(\omega) = \int f(t)e^{-i\omega t} dt \quad (2.1a)$$

$$STFT \{x\} = X(\tau, \omega) = \int f(t)\omega(t - \tau)e^{-i\omega t} dt \quad (2.1b)$$

La STFT provee información promedio únicamente acerca de cuando y a que frecuencia de una señal ocurre un determinado evento en la señal, un inconveniente es que siempre se conserva el mismo tamaño de ventana de tiempo lo cual da una resolución limitada, y debido a que existen diferentes tipos de señales que requieren diferentes tamaños de ventana para determinar con mayor precisión el tiempo o la frecuencia, lo cual refiere a un acercamiento más flexible. El problema de resolución en la STFT se puede explicar con el principio de incertidumbre ecuación (2.2) [3]: a mayor resolución espacial menor resolución espectral. Un ejemplo de bloques de Heisenberg de átomos de *wavelets* está mostrado en la figura 2.1.

$$\Delta t \cdot \Delta \omega \geq \pi \quad (2.2)$$

donde:

t representa tiempo

ω representa frecuencia

La STFT alcanza la máxima resolución espacial sacrificando la resolución temporal, mientras que las *wavelets* dan información simultánea de tiempo y frecuencia (t y ω), o espacio-escala.

En el plano tiempo-escala, los análisis por medio de *wavelets* emplean regiones de tamaño variable, esto es útil para señales que requieren largos intervalos de tiempo y bajas frecuencias, así como pequeñas regiones temporales donde la información necesita altas frecuencias.

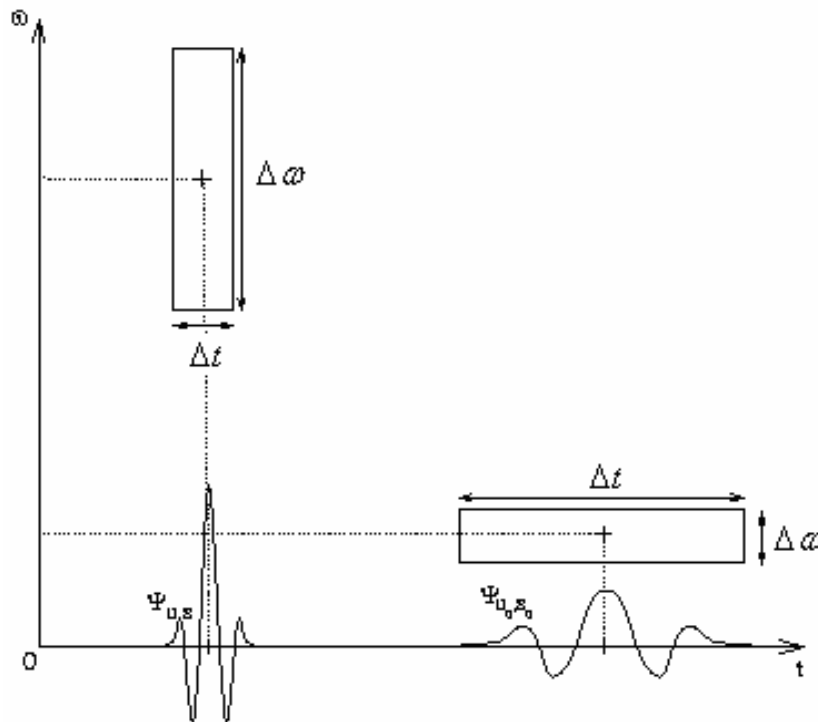


Figura 2.1. Esquema del análisis *wavelet*.

Como es mostrado en la figura 2.1 largos intervalos de tiempo caracterizan bajas frecuencias y cortos intervalos de tiempo caracterizan altas frecuencias.

El análisis *wavelet* es capaz de mostrar aspectos de la señal que otras técnicas no logran encontrar.

Así, la definición para una transformación lineal y continua para una función $s(t)$, está dada por:

$$S(a, \tau) = \int_{-\infty}^{\infty} \Psi_a^*(t - \tau) \cdot s(t) \cdot dt \quad (2.3a)$$

donde:

Ψ_a^* Indica complejo conjugado de la función madre.

a establece la dilatación de la wavelet (se verá mas adelante que es una dilatación).

τ establece la traslación de la wavelet (se verá mas adelante que es una traslación).

La transformada *wavelet* consiste en descomponer la señal por medio de ciertas funciones *wavelet*, las cuales se obtienen a partir de una función *wavelet* madre.

La descomposición permite obtener unos coeficientes que son susceptibles de interpretación y posterior manipulación.

En la tabla 2.1 están las funciones madre de transformadas importantes: *Fourier*, *Gabor* y *wavelet*.

Tabla 2.1. Ejemplos de funciones madre ¹

	BASE
FOURIER	$e^{i \cdot t/a}$
GABOR	$e^{i \cdot t/a} \cdot w(t - \tau)$
WAVELET	$\frac{1}{\sqrt{a}} \Psi\left(\frac{t - \tau}{a}\right)$

El análisis por medio de *wavelets* da información sobre el espectro de frecuencias en función del tiempo. La resolución espectral de una frecuencia f es: $\Delta f \propto f$, y la resolución temporal de esta frecuencia es: $\Delta t \propto 1/f$ ($\Rightarrow \Delta t \cdot \Delta f = cte.$). Realizando una transformada discreta de *wavelets* obtenemos una serie de coeficientes que podemos interpretar gráficamente como se muestra en la figura 2.5:

De entre las diferentes familias, algunas de las *wavelets* más importantes son:

1. *Wavelet Haar*: Es la primer *wavelet* conocida y fue propuesta por Alfred Haar en 1909. Es la *wavelet* más simple posible y su principal desventaja es que no es continua y por lo tanto no es diferenciable.
2. *Wavelet de Daubechie*: Son una familia de *wavelets* ortogonales que definen una transformada wavelet discreta. Con cada wavelet de esta clase, existe una función de escalamiento que genera un análisis ortogonal de multiresolución.

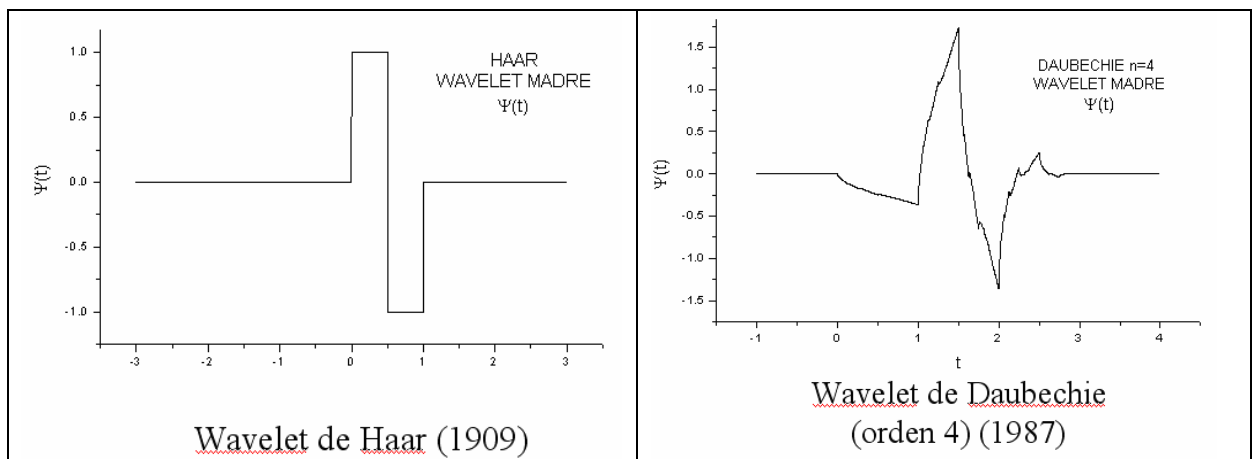


Figura 2.2. *Wavelets* importantes.

En la actualidad, el número de funciones *wavelets* existentes es muy grande. En general conviene usar aquella cuya forma se adecue mejor al tipo de señal con la que se trabaja. Hay *wavelets* continuas o discretas, con o sin soporte compacto, suaves con o sin discontinuidades, ortogonales / biortogonales, etc.

¹ w es una función de peso (ventana) generalmente gaussiana.

El coeficiente $1/\sqrt{a}$ es un factor de normalización.

2.2.1. Traslaciones y Dilataciones

Tal como se definió, ecuación (2.3a), una transformada con *wavelets* de una función $s(t)$ está dada por:

$$S(a, \tau) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{a}} \Psi^* \left(\frac{t - \tau}{a} \right) \cdot s(t) \cdot dt \quad (2.3b)$$

En la cual, el término ' τ ' define las traslaciones o desplazamientos y el término ' a ' las dilataciones de la función *wavelet*.

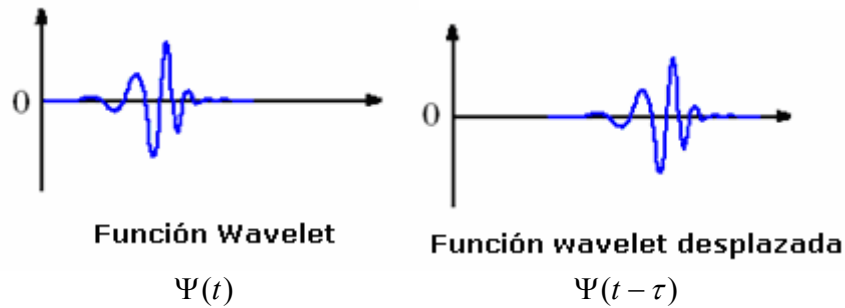
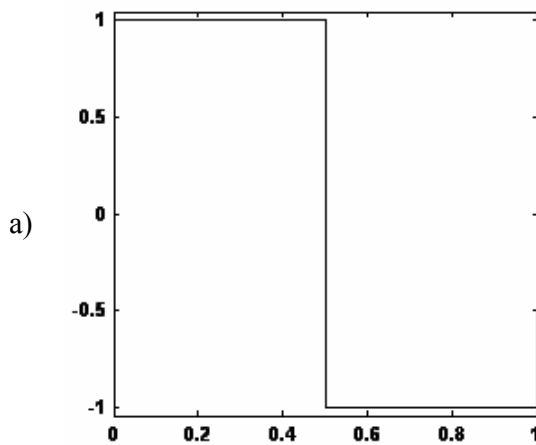
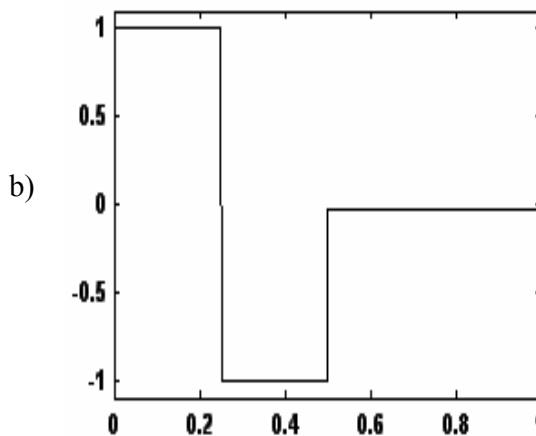


Figura 2.3. a) Ejemplo de función *wavelet*. b) *Wavelet* desplazada.



$$f(t) = \Psi(t); \quad a = 1$$



$$f(t) = \Psi(2t); \quad a = 1/2$$

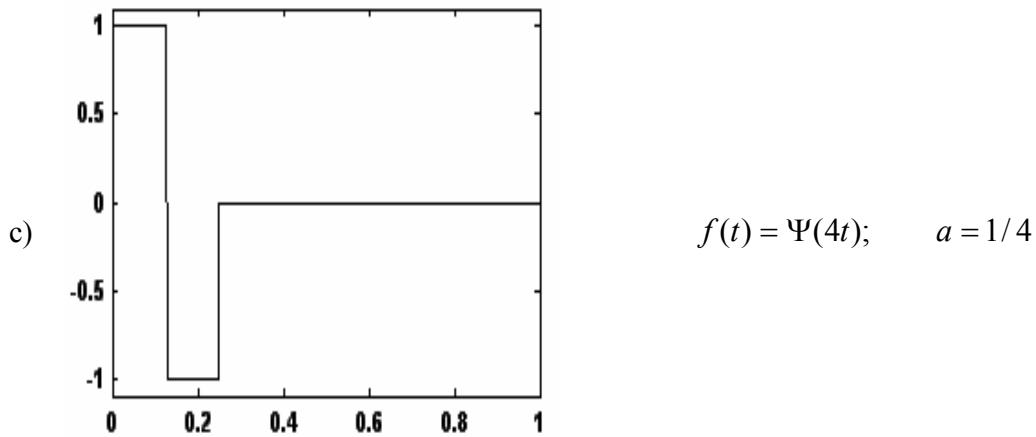


Figura 2.4. Función *wavelet daubechies* con diferentes valores de escala ‘a’
 a) *Wavelet* con ‘a’=1. b) *Wavelet* con ‘a’=1/2. c) *Wavelet* con ‘a’=1/4.

Por lo tanto, la señal se muestrea empleando versiones de funciones *wavelets* madres las cuales están dilatadas y/o trasladadas, estudiando punto a punto, para medir cuáles son las más similares a dichas *wavelets* madre.

La figura 2.5 muestra los diferentes análisis de tiempo y frecuencia mencionados con anterioridad, en 2.5a) los parámetros de la grafica son amplitud y tiempo, por lo tanto se pierden otros parámetros de la señal, por otra parte en 2.5b) , Fourier permite determinar sus frecuencias, pero a costa de perder la información de tipo temporal sobre la señal, en 2.5c) los intervalos de tiempo y frecuencia son iguales lo cual nos da una información rudimentaria sobre el orden temporal en el que se dan las frecuencias y finalmente en 2.5d) los intervalos tanto en tiempo como en frecuencia (escala) incrementan y disminuyen respectivamente como el valor de los mismos aumenta.

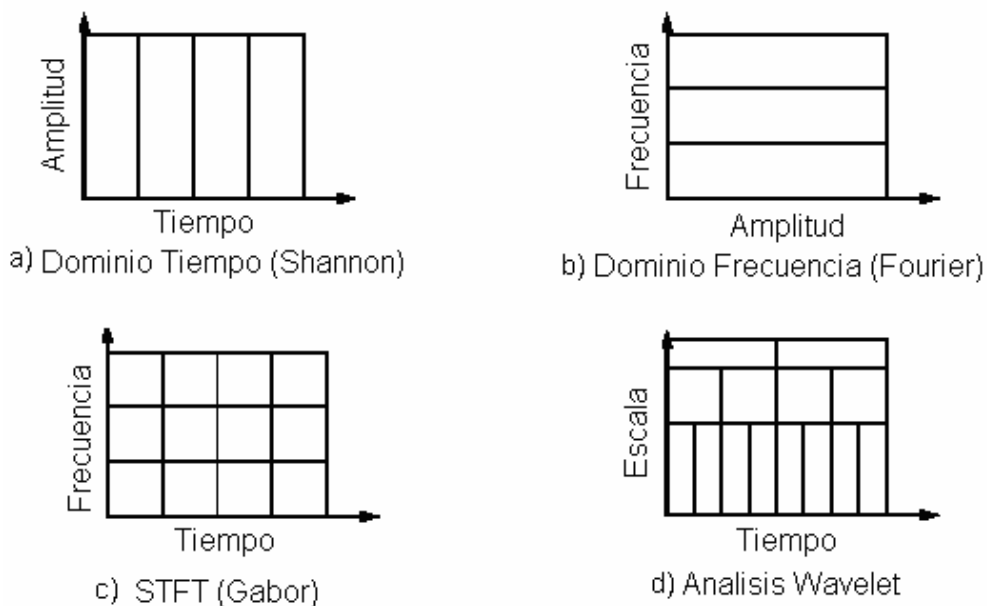


Figura 2.5. Ejemplos de diferente tipo de análisis. En tiempo, frecuencia y escala.

La frecuencia de la señal estudiada está relacionada con la escala de la *wavelet*, además, el hecho de que el análisis sea local, le da a esta transformada sus propiedades de amplio interés, diferentes de las de otras transformadas, como la transformada Fourier.

En la figura 2.6, se ve como la transformada Fourier genera coeficientes con diferentes frecuencias pero también un número diferente de ciclos mientras que la transformada wavelet conserva el número de ciclos a pesar de la variación de la frecuencia.

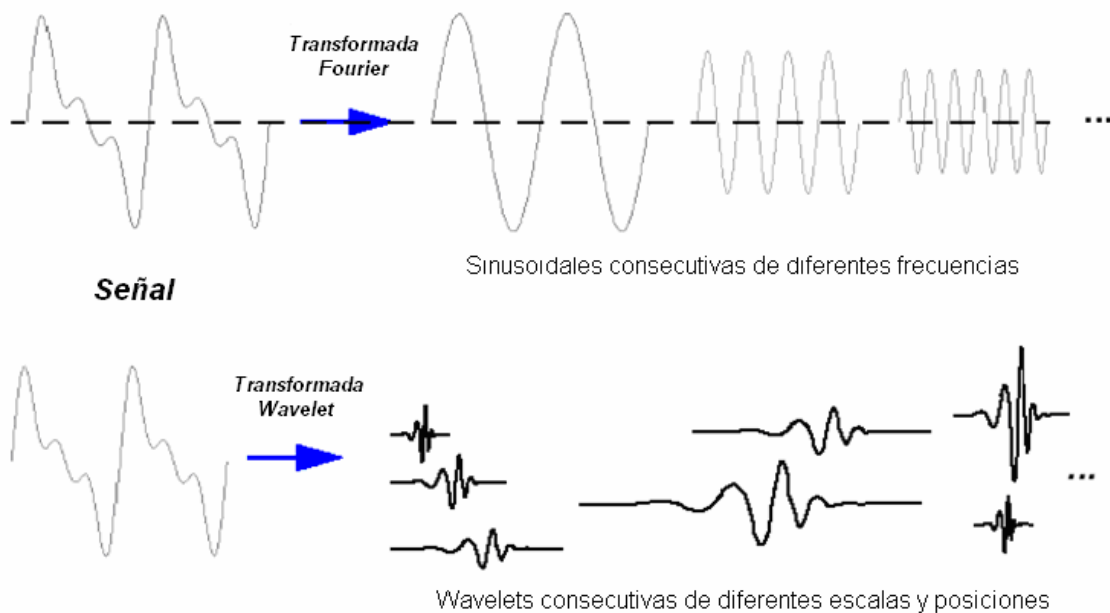


Figura 2.6. Diferencia entre la transformada Fourier y *Wavelet*.

2.2.2. Transformada *wavelet* discreta (DWT)

El cálculo de la transformada *wavelet* para todas las posibles escalas supone una gran cantidad de información. Escoger sólo aquellas escalas y posiciones que resultan interesantes, o apropiadas, para ciertos estudios, o ciertas aplicaciones podría ser una tarea muy ardua. Si se escogen aquellas escalas y posiciones basadas en potencias de dos, los resultados serán más eficaces desde el punto de vista computacional. Esta manera de hacer el análisis se denomina *transformada wavelet discreta (DWT)*.

Para lograr establecer la transformación *wavelet* discreta, debido a su eficiencia computacional y a su uso en señales discretas, se discretiza la transformada para la escala $a = 2^1$, quedando la ecuación (2.3) como:

$$S(\tau) = \sum_{m,n=-\infty}^{\infty} (f, \Psi_{m,n}) \Psi_{m,n}(t) \quad (2.4)$$

Donde $\Psi_{m,n} = Q_0^{-m/2} \Psi(a^{-m}t - nb_0)$ con $Q_0 = 2$, $a > 1$ y $b_0 > 0$.

De manera general, las escalas bajas se refieren a frecuencias altas y escalas altas a frecuencias bajas.

En la mayor parte de las señales, la información más importante desde el punto de vista semántico es encontrada sobre las frecuencias bajas, y en las altas frecuencias se encuentran los detalles de la señal. Un ejemplo clásico, es el de la voz. Al eliminar las altas frecuencias, aunque el sonido es diferente, es posible entender el significado del mensaje, en su contraparte, si se eliminan las bajas frecuencias, el mensaje no se puede entender.

Por tal motivo la transformada *wavelet* puede descomponer una señal en aproximaciones y detalles, proceso el cual se nombra análisis.

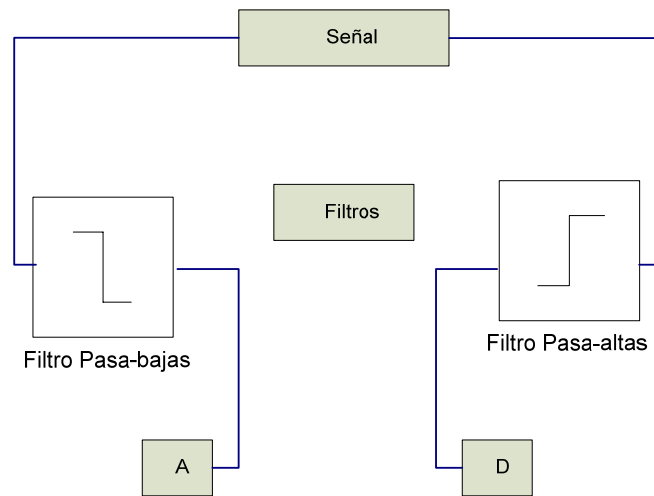


Figura 2.7. Proceso de descomposición (análisis).

El proceso de reconstrucción, también denominado síntesis, se encarga de la obtención de la señal a partir de los detalles y aproximaciones. Este proceso se lleva a cabo con la transformada *wavelet* discreta inversa.

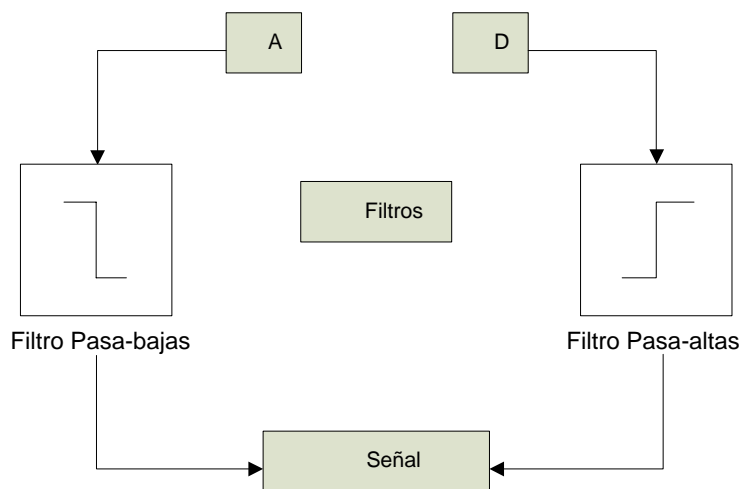


Figura 2.8. Proceso de reconstrucción (síntesis).

Desde este punto de vista la transformada *wavelet* se puede llevar a cabo con un conjunto de filtros. La elección de estos filtros (*wavelets*) influye notablemente en los resultados finales.

2.2.3. La transformada *wavelet* discreta en 2-D

La transformada *wavelet* discreta (DWT) aplicada a imágenes proporciona una matriz de coeficientes, conocidos como coeficientes *wavelet*. En un nivel, al aplicar la DWT son obtenidos cuatro coeficientes, identificados como: aproximaciones, detalles horizontales, detalles verticales y detalles diagonales.

La aproximación es la que conserva la mayor parte de la energía total, la información que puede ser reconocida más fácilmente por un observador, mientras que los detalles son valores cercanos a cero, lo cuales como ya se menciono dan los detalles de orientación debido al la filtración que se le hace a la imagen original, dicho filtrado se vera en el siguiente punto.

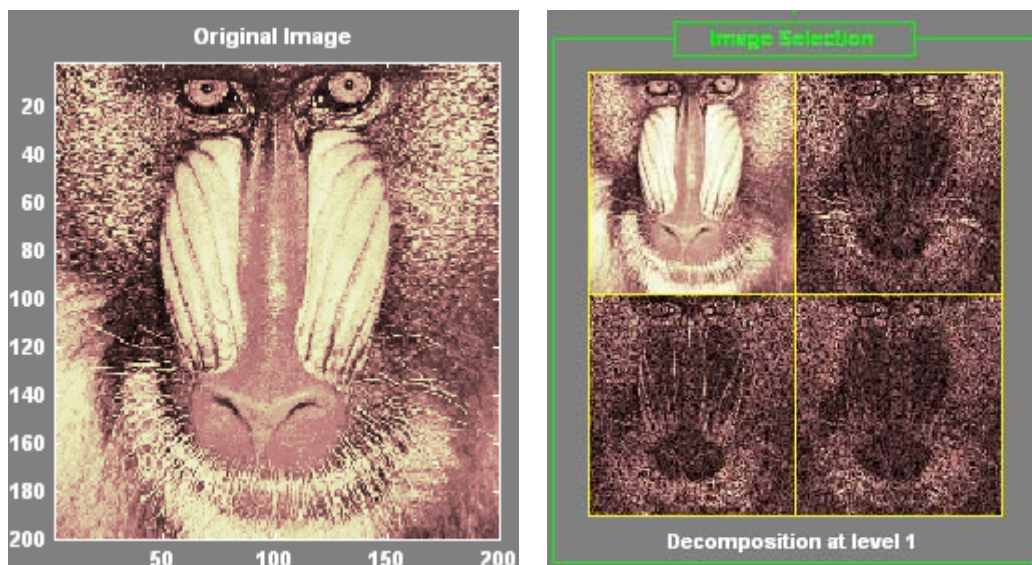


Figura 2.9. Descomposición *wavelet* de primer nivel.

2.2.4. Organización de los coeficientes *wavelet*

Como ya se mencionó la mayor energía de las imágenes está concentrada en las frecuencias bajas, por lo tanto, una imagen tiene un espectro que disminuye de amplitud con el incremento de las frecuencias. Esta propiedad de las imágenes es reflejada en la transformada *wavelet* discreta de la imagen.

Haciendo la descomposición en varios niveles, descomponiendo sucesivamente los coeficientes de las aproximaciones, se construye una descomposición multiresolución. En esta pirámide de coeficientes, los niveles más bajos de codificación corresponden a las bandas de alta frecuencia del espectro.

En la figura 2.10 a), se muestra como la señal (imagen) es descompuesta a través de filtros, primeramente en cuatro coeficientes y después uno de ellos es descompuesto en otros cuatro coeficientes.

En 2.10 b), el primer nivel¹ (V1, D1, H1, donde V, D y H representan dirección vertical, diagonal y horizontal respectivamente) representa la banda de más alta frecuencia y el nivel más fino de resolución. A la inversa, el último nivel (n) de descomposición, en este caso $n = 3$, corresponde con la banda de frecuencia más baja y la resolución más burda. Al desplazarse de los niveles más altos a los más bajos, es decir, de baja resolución a alta resolución, se observa una disminución de la energía contenida en las capas así recorridas. Y en 2.10 c), se puede ver la pirámide de los coeficientes *wavelets* obtenidos por el algoritmo de 2.10 a).

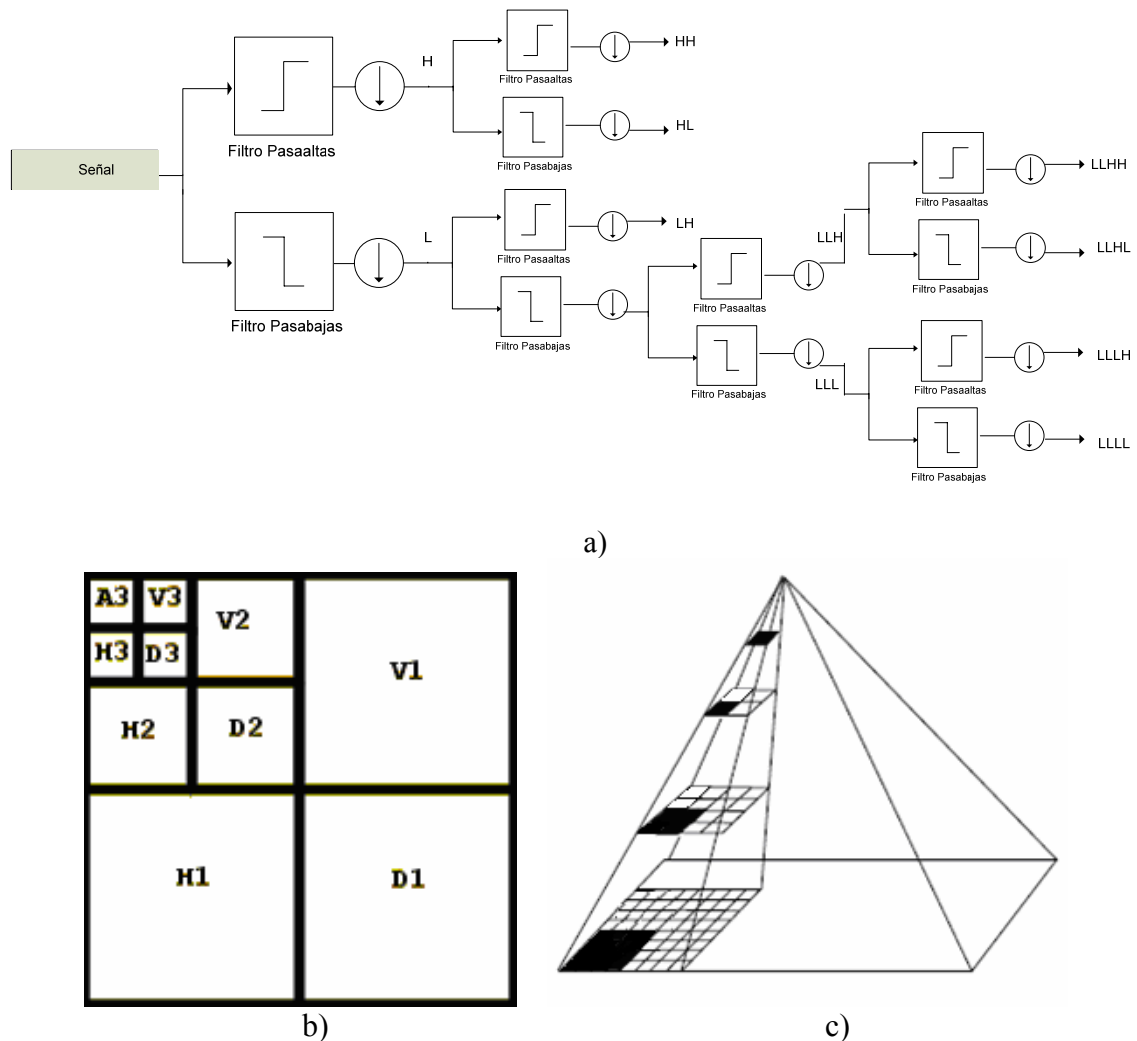


Figura 2.10. a) Algoritmo de pirámide de las *wavelets* de dos niveles. b) Esquema de la organización de los coeficientes *wavelets*. c) Pirámide de coeficientes *wavelets*.

¹Los coeficientes del primer nivel V, D y H son utilizados para referir las direcciones, sin embargo, en 2.10 a) son utilizados HH, HL y LH respectivamente, donde H y L representan el filtrado pasaaltas y pasabajas respectivamente.

Adicionalmente, se puede comprobar que existen correlaciones espaciales a través de las capas wavelets.

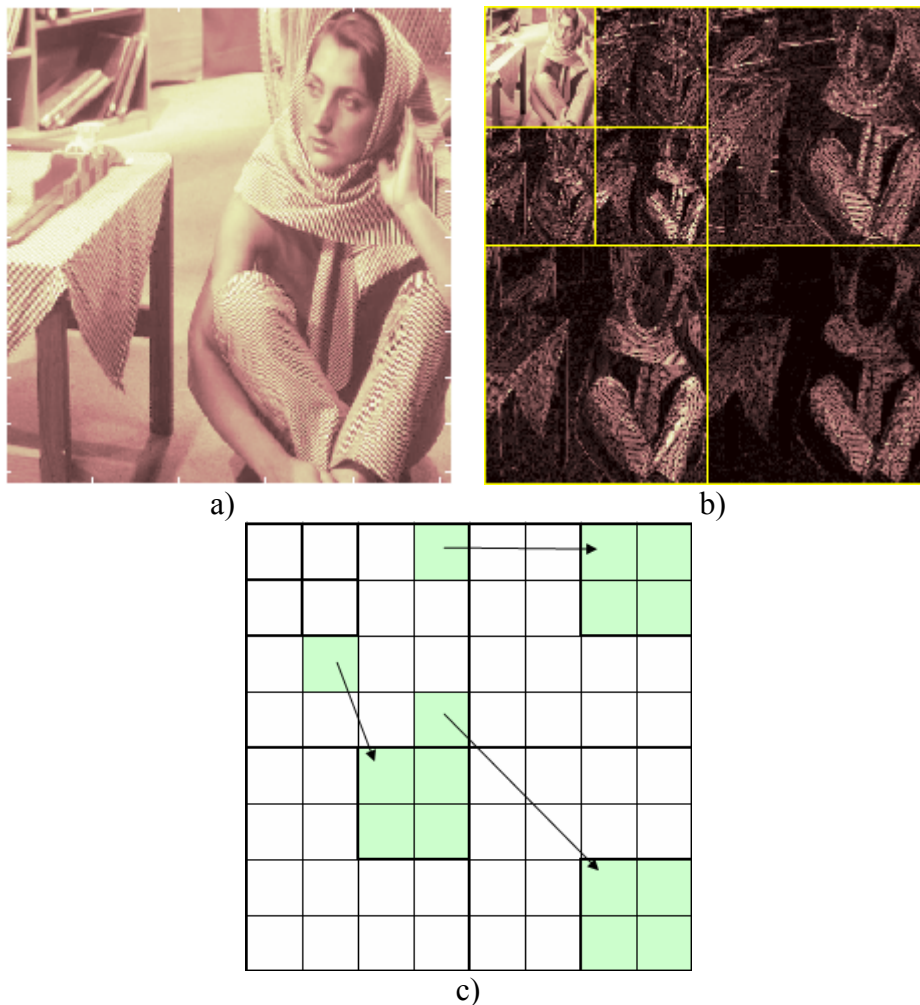


Figura 2.11. a) Imagen original de Bárbara. b) Organización de los coeficientes *wavelet* en dos niveles. c) Árboles de descendencia *wavelets*.

En la figura 2.11 se puede observar los contornos de Bárbara en los distintos niveles y cómo son más bastos en el primer nivel de descomposición, además de cierta similitud entre los distintos niveles. Lo cual lleva a la definición del algoritmo llamado *EZW* (*Embedded Zerotree Wavelet*) [4] para ciertas aplicaciones, donde se aprovechan dichas similitudes, como se muestra en la figura 2.11 c).

Como se mencionó en la introducción de este capítulo, debido al interés de realizar una eficiente expansión lineal para señales de dos dimensiones, como son las discontinuidades a través de curvas suaves, Candès y Donoho [5] definen una transformada llamada *curvelet*. La construcción original de la transformada *curvelet* fue pensada para las funciones definidas en el espacio continuo \mathbb{R}^2 .

2.3. Curvelet

Así como la transformada *wavelet*, la transformada *curvelet* es obtenida mediante un sistema de filtrado, seguido por la aplicación a cada imagen filtrada de una transformada

llamada *ridgelet*¹ en intervalos, en este caso con un filtro pasa-banda [5,6]. En \mathbb{R}^2 , los *ridgelets* son constantes a lo largo de las líneas $x_1 \cos(\theta) + x_2 \sin(\theta) = cte$ y son funciones *wavelets* con una escala s a lo largo de la dirección ortogonal. En el dominio de la frecuencia, la función *ridgelet* es esencialmente localizada en la curva (2.5) y alrededor del ángulo θ .

$$|\omega| \in [2^s, 2^{s+1}] \quad (2.5)$$

En resumen, los pasos para la construcción de la transformada *curvelet* son:

1. Descomponer la imagen en un conjunto de capas.
2. Poner en intervalos cada capa dentro de bloques de tamaño apropiado, dependiendo de su frecuencia central.
3. Aplicar la transformada *ridgelet* a estos bloques.

La transformada *curvelet* plantea problemas en su versión discreta. Primeramente, puesto que es una transformada basada en bloques, de la misma forma que la transformada coseno discreta³, las imágenes aproximadas tienen efectos de bloque, o bien se tienen que utilizar ventanas de traslape y así aumentar la redundancia². En segundo lugar, el uso de la transformada *ridgelet*, la cual está definida en coordenadas polares, realiza la implementación de la transformada *curvelet* para imágenes discretas sobre las coordenadas rectangulares.

Sin embargo, en compensación, comparando el esquema de las *wavelets* con el esquema de las *curvelets*, para ciertas aplicaciones es notable el mejoramiento que se puede obtener con las *curvelets*.

¹La teoría de *ridgelets* fue desarrollada en la tesis de Ph.D. de Emmanuel Candès (1998) [6]. En ese trabajo, Candès demostró que uno podría desarrollar un sistema del análisis basado en funciones de máximos y mínimos (*ridge*) el cual es un método de análisis conveniente para objetos con discontinuidades a través de líneas rectas. Función del *ridge*:

$$\psi_{a,b,\theta}(x_1, x_2) = a^{-1/2} \psi((x_1 \cos(\theta) + x_2 \sin(\theta) - b) / a)$$

² Una versión de la transformada *curvelet* discreta en [7] es un ejemplo que tiene una redundancia de $16J+1$ donde J es el número de niveles en la multiescala.

³ La Transformada de coseno discreta es una [función](#) lineal e invertible $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ (donde \mathbb{R} es el conjunto de números reales), o, de forma equivalente, una [matriz cuadrada](#) $n \times n$.

De una secuencia $u[m, n], 0 \leq m, n \leq N-1$ se define

$$v[k, l] = \alpha(k) \alpha(l) \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} u[m, n] \cos\left[\frac{(2m+1)k\pi}{2N}\right] \cos\left[\frac{(2n+1)l\pi}{2N}\right] \quad \text{donde } \alpha(\psi) (\psi = k, j)$$

$$\text{toman los valores } \alpha(\psi) = \begin{cases} \sqrt{\frac{1}{N}} & \psi = 0 \\ \sqrt{\frac{2}{N}} & \psi = 1, \dots, N-1 \end{cases}$$

2.4. Contourlet

Se desarrollará la transformada *contourlet* considerando las diferentes etapas para su realización, donde se podrá ver que es una versión discreta de la transformada curvelet.

2.4.1. Pirámide Laplaciana

Una forma de obtener una descomposición multiescala es utilizar una *Pirámide Laplaciana* (de sus siglas en ingles, LP^1):

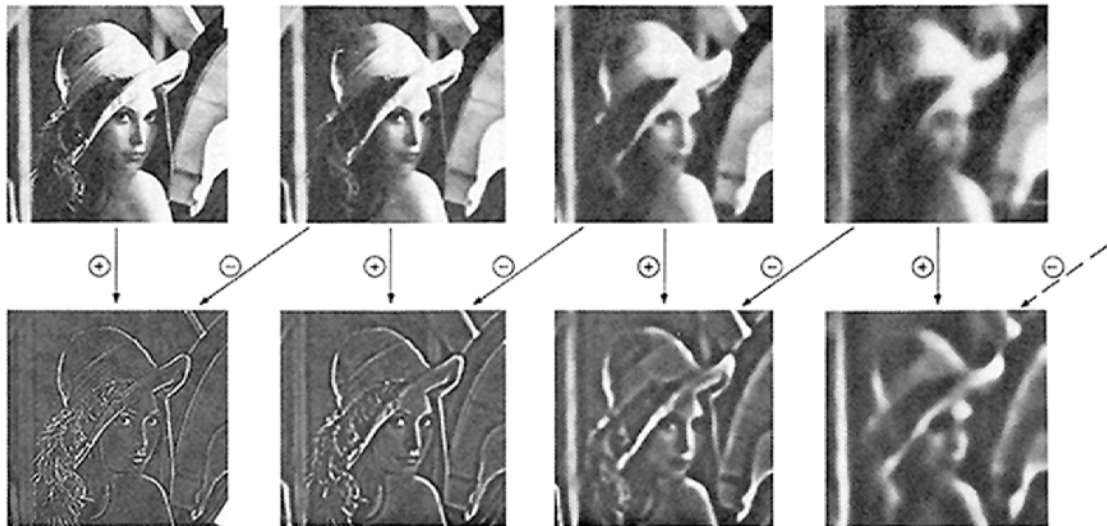
Como es conocido en la literatura [8], lo primero que se realiza para la codificación de una pirámide laplaciana, es filtrar una imagen, es decir, filtrar la imagen original G_0 en un filtro pasa-bajas obteniendo una imagen G_1 . Debido al filtrado, tal imagen obtenida G_1 será una versión de la imagen G_0 reducida en resolución y en densidad de puntos. Luego, es obtenida una imagen G_2 , la cual será la imagen reducida correspondiente a la imagen G_1 y, así sucesivamente.

A partir de la teoría de sistemas lineales, el proceso de filtrado se basa en el procedimiento equivalente al de una convolución con una familia de funciones. El kernel de esta familia es una función Gaussiana, por lo que la secuencia de imágenes G_0, G_1, G_2, \dots es conocida como Pirámide Gaussiana.



(Burt, P.J. et al. 1983.)

Figura 2.12. Pirámide Gaussiana.



(Burt, P.J. et al. 1983.)

Figura 2.13. Primeras cuatro imágenes de la pirámide Gaussiana (línea superior) y pirámide Laplaciana (línea inferior).

En codificación, un propósito al construir la imagen reducida G_1 es que pueda servir como predicción para los valores de los píxeles en la imagen original G_0 . Para la obtención de una representación comprimida, se codifica la imagen del error que existe cuando se amplía G_1 y se resta de G_0 . Dado que una función laplaciana se obtiene restando dos gaussianas, esta imagen de error se convierte en el nivel inferior de la pirámide laplaciana. El nivel siguiente es generado por la codificación de G_1 de la misma forma. Por lo tanto, la pirámide laplaciana es una secuencia de errores L_0, L_1, L_2, \dots siendo cada una, la diferencia entre dos niveles sucesivos de la Pirámide Gaussiana.

Una desventaja de la Pirámide Laplaciana es su sobre-muestreo implícito. Sin embargo, para el muestreo crítico del esquema *wavelet*, la pirámide Laplaciana tiene la característica distintiva de que cada nivel de la pirámide genera solamente una imagen pasa-banda (incluso para los casos multiresolución) como se muestra en la figura 2.13, lo cual no implica que las frecuencias se encuentren en desorden.

¹En [9], Minh N. Do y Martín Vitterli, analizan el LP usando la teoría de marcos (frames) y sobre muestreo de los bancos de filtros. Demuestran que la LP con filtros ortogonales (es decir, $h[n] = g[-n]$ y $g[n]$ ortogonal a sus translaciones con respecto a la cuadrícula del sub-muestreo) es un marco ajustado con los límites del marco iguales a 1.

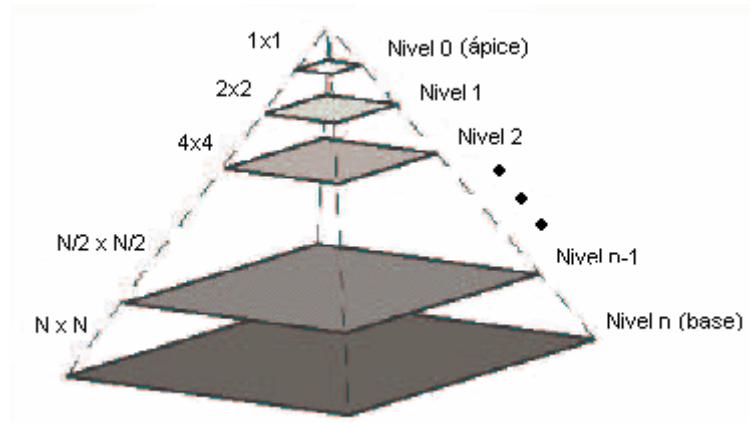
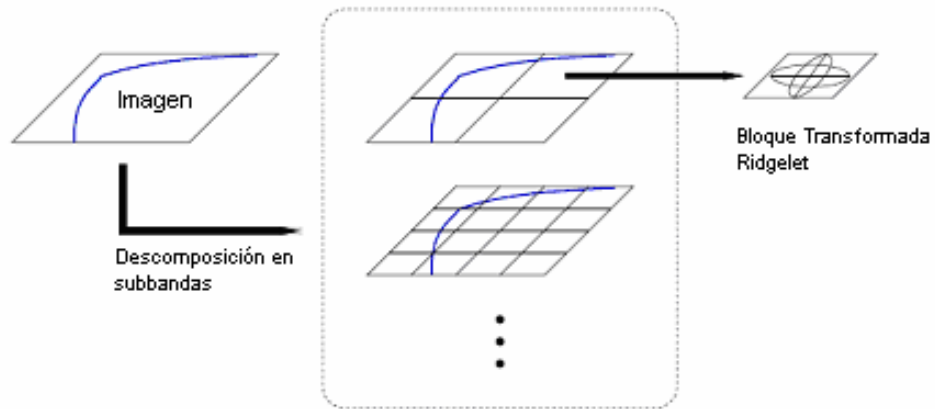


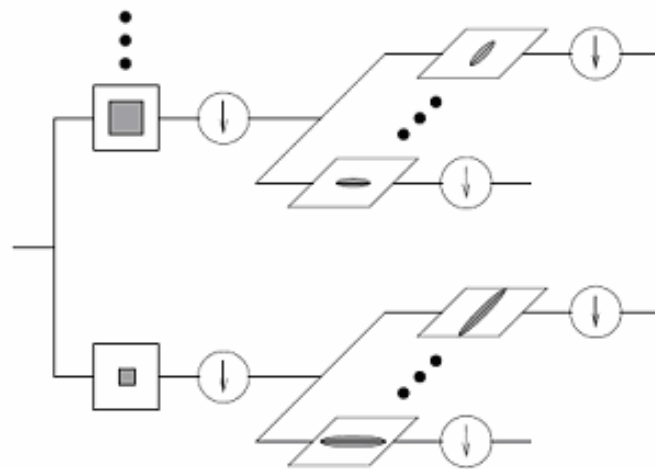
Figura 2.14. Pirámide Laplaciana

Dadas las figuras 2.10 c) y 2.14 se puede ver la diferencia entre la pirámide generada por las *wavelets* y por la pirámide Laplaciana, donde en el primer caso de la imagen es dividida en cuatro por los filtros y en la pirámide Laplaciana solo resulta una imagen reducida en tamaño por un factor de 2.

En las siguientes figuras 2.15 y 2.16 se muestran los esquemas de principio de la transformada curvelet la cual es obtenida con una descomposición en capas y la aplicación de un bloque *ridgelet* y de la transformada *contourlet* donde se aplica la pirámide laplaciana y después una descomposición direccional.



a) Transformada Curvelet



b) Transformada Contourlet

Figura 2.15. Dos transformadas para imágenes con contornos suaves. a) Transformada *curvelet*: transformadas de bloque *ridgelet* son aplicadas a las subbandas de la imagen. b) Transformada *contourlet*: una imagen es descompuesta por una estructura con doble banco de filtros, donde el primero captura los puntos de eje y el segundo enlaza estos ejes con los contornos. Las áreas grises en los cuadros representan los soportes espaciales de los filtros

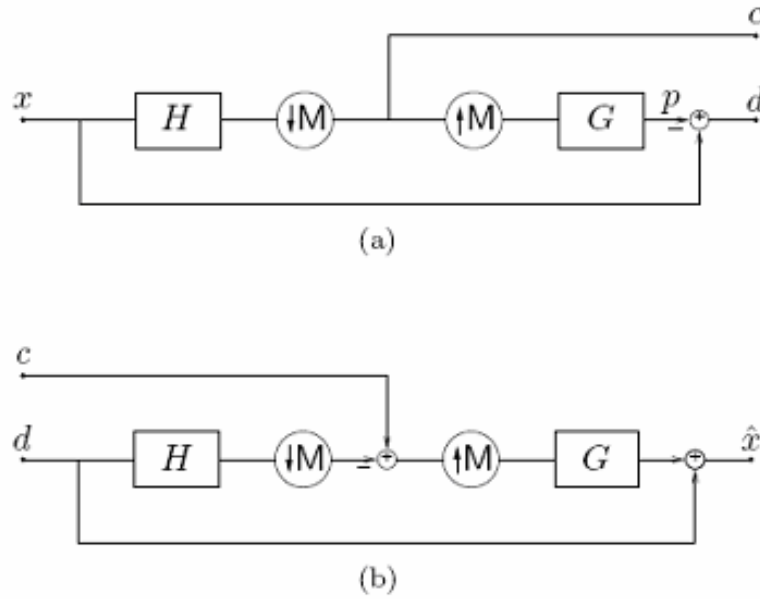


Figura 2.16. Esquema de la pirámide laplaciana. a) Análisis: las salidas son una aproximación robusta c y una diferencia d entre la señal original y la predicción. El proceso puede ser iterado por descomposición sucesiva de la versión robusta. b) Esquema propuesto de la reconstrucción para la pirámide laplaciana.

2.4.2. Descomposición de Multiescala

Esta descomposición se obtiene aplicando la pirámide laplaciana en el banco de filtros direccional piramidal (PDFB) filtros ortogonales y submuestreando con orden dos en cada dimensión. Para ciertas aplicaciones en tiempo real, el filtro pasa-bajas g en la pirámide laplaciana define únicamente una función de escalamiento ortogonal $\phi(t) \in L^2(\mathfrak{R}^2)$.

$$\phi(t) = 2 \sum_{n \in \mathfrak{Z}^2} g[n] \phi(2t - n) \quad (2.6)$$

En donde

$$\phi_{j,n} = 2^{-j} \phi\left(\frac{t - 2^j n}{2^j}\right), \quad j \in \mathfrak{Z}, n \in \mathfrak{Z}^2 \quad (2.7)$$

Donde \mathfrak{Z} representa a los números enteros y la familia $\{\phi_{j,n}\}_{n \in \mathfrak{Z}^2}$ entonces es una base ortonormal del espacio V_j para toda $j \in \mathfrak{Z}$. La secuencia de subespacios jerarquizados satisfacen las condiciones siguientes:

$$\text{Invarianza al corrimiento } f(t) \in V_j \Leftrightarrow f(t - 2^{-1}t) \in V_j, \forall j \in \mathfrak{J}, k \in \mathfrak{Z}^2 \quad (2.8)$$

$$\text{Invarianza a la escala } f(t) \in V_j \Leftrightarrow f(2^{-1}t) \in V_{j+1}, \forall j \in \mathfrak{J}. \quad (2.9)$$

V_j es un subespacio definido sobre una repartición uniforme de intervalos $2^j \times 2^j$ lo cual caracteriza la aproximación de la imagen a la resolución 2^j . Los detalles necesarios para incrementar la resolución de la imagen aproximada son preservados por la diferencia entre las imágenes de la pirámide laplaciana. Sea W_j el complemento ortogonal de V_j en V_{j-1} , se satisface:

$$V_{j-1} = V_j \oplus W_j \quad (2.10)$$

Dicha ecuación (2.10) representada por la figura 2.17, donde ω_0 y ω_1 representan las frecuencias en cada una de las direcciones.

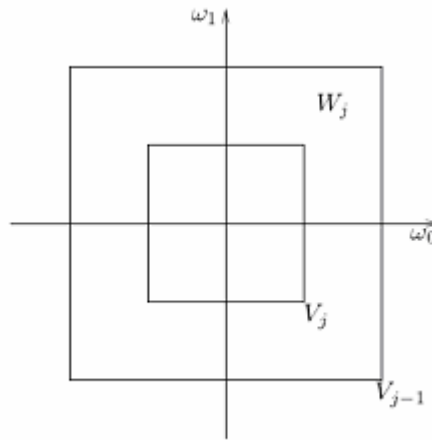


Figura 2.17. Subespacios generados por la pirámide laplaciana.

La pirámide laplaciana es también obtenida con un banco de filtros con submuestreo de sus salidas, donde cada componente polifásico de la diferencia de la señal está en un canal separado del banco de filtros. $F_i(z)$, $0 \leq i \leq 3$, es el banco de filtros de síntesis para estos componentes polifásicos. $F_i(z)$, es un filtro pasa-altas. Similarmente para el banco de filtros de la transformada *wavelet*, a cada uno de estos filtros se asocia una función continua $\psi^{(i)}(t)$, [2,9]. y esta a su vez es igual a la ecuación (2.11).

$$\mu_{j,2n+k_i}(t) = \psi_{j+1,n}^{(i)} = \sum_{m \in \mathfrak{Z}^2} f_i[m] \phi_{j,n+m}(t) \quad (2.11)$$

Originando la familia $\{\mu_{j,n}\}_{n \in \mathfrak{Z}^2}$ es un marco de W_{j+1} y se ve como una reja uniforme en R^2 de intervalos $2^j \times 2^j$.

2.4.3. Descomposición direccional

Considerando la importancia que tienen las orientaciones en varios problemas de visión, en 1992 fue introducido el concepto de banco de filtros direccionales (de sus siglas en inglés, DFB), el cual puede ser decimado al máximo para alcanzar una reconstrucción perfecta. Dicho banco de filtros, a través de un árbol bien estructurado de n – niveles es implementado eficientemente, lo cual produce 2^n capas con una partición de frecuencias como la mostrada en la figura 2.18 a). Como se puede ver en esa imagen, una característica para las regiones frecuenciales de las capas, es que no siguen un orden¹.

En sistemas en varias dimensiones y tasas, operaciones de muestreo son definidas en enrejados. Una estructura en d dimensiones es representada por una matriz M invertible de enteros de $d \times d$.

$$LAT(M) = \{m : m = Mn, n \in \mathfrak{Z}^d\} \quad (2.12)$$

Matrices Q_0 y Q_1 son ejemplos de matrices que representan sub-enrejados romboidales bidimensionales mostrados en la figura 2.19 a).

$$Q_0 = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}, \quad Q_1 = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \quad (2.13)$$

Las siguientes matrices son usadas en el DFB para establecer las operaciones de rotación.

$$R_0 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad R_1 = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}, \quad R_2 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \quad R_3 = \begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix} \quad (2.14)$$

¹Para obtener la partición deseada de la frecuencia, una regla del árbol implicado tiene que ser seguida (véase [10,11] para los detalles).

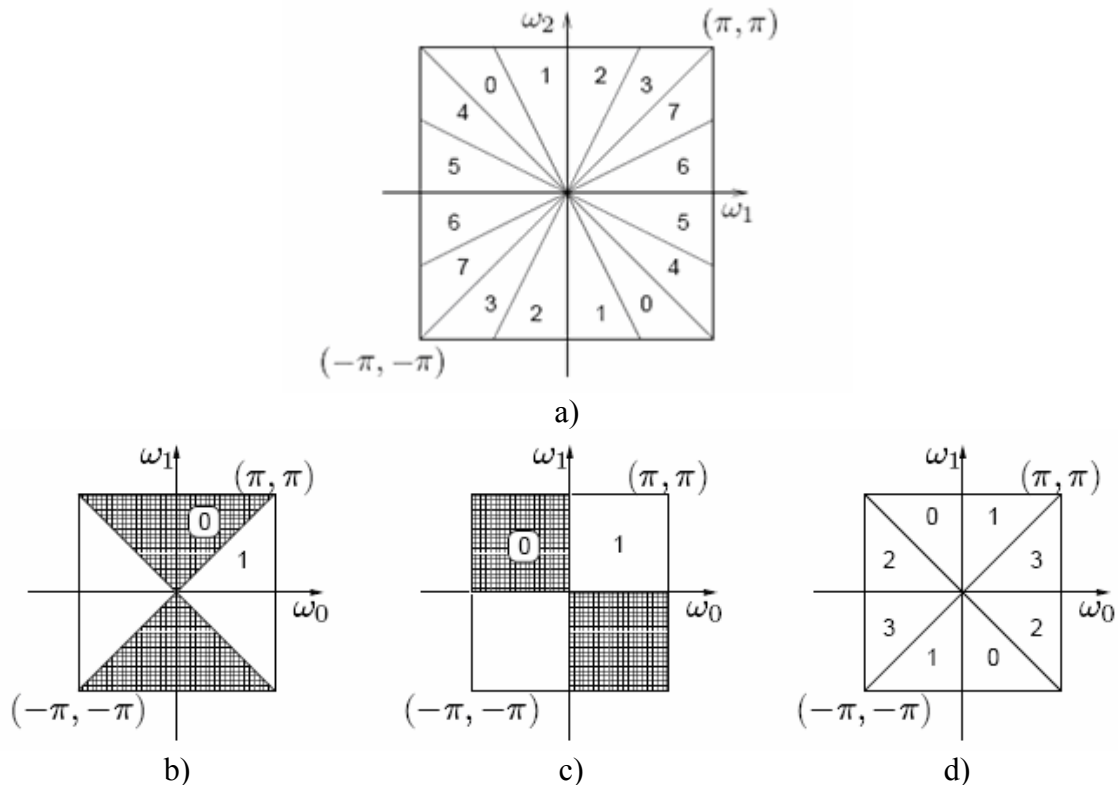


Figura 2.18: a) Primer nivel de filtros en forma de ventilador. b) Segundo nivel con filtros cuadrados. c) Combinación de los dos niveles. d) Banco de filtros direccional particionando las frecuencias, donde $n=3$ y hay $2^3 = 8$ bandas frecuenciales.

Para obtener una partición direccional de frecuencia de cuatro, la primera partición es mostrada en la figura 2.19 b). Matrices de muestreo en el primer nivel y segundo nivel son Q_0 y Q_1 respectivamente, después de dos niveles $Q_0 Q_1 = 2I_2$, o sub-muestreo con un factor de dos en cada dimensión.

Se pueden intercambiar los filtros del segundo nivel en la figura 2.19 b) con la matriz de muestreo Q_0 . Dado intercambio genera un filtro cuadrado. Combinando los filtros en forma de ventilador resultan las cuatro direcciones como se ve en la figura 2.18 d).

Otro esquema para el banco de filtros direccionales, se basa solamente en filtros con soporte en forma de rombo (de sus siglas en ingles, quincunxes QFB).

La partición del plano de frecuencia por el banco de filtros direccionales (DFB) es realizada por una combinación apropiada de la frecuencia direccional del QFB con filtros en forma de hélice y operaciones de rotación hechas por re-muestreo, como se ilustran en las figuras 2.19 c) y 2.20, respectivamente.

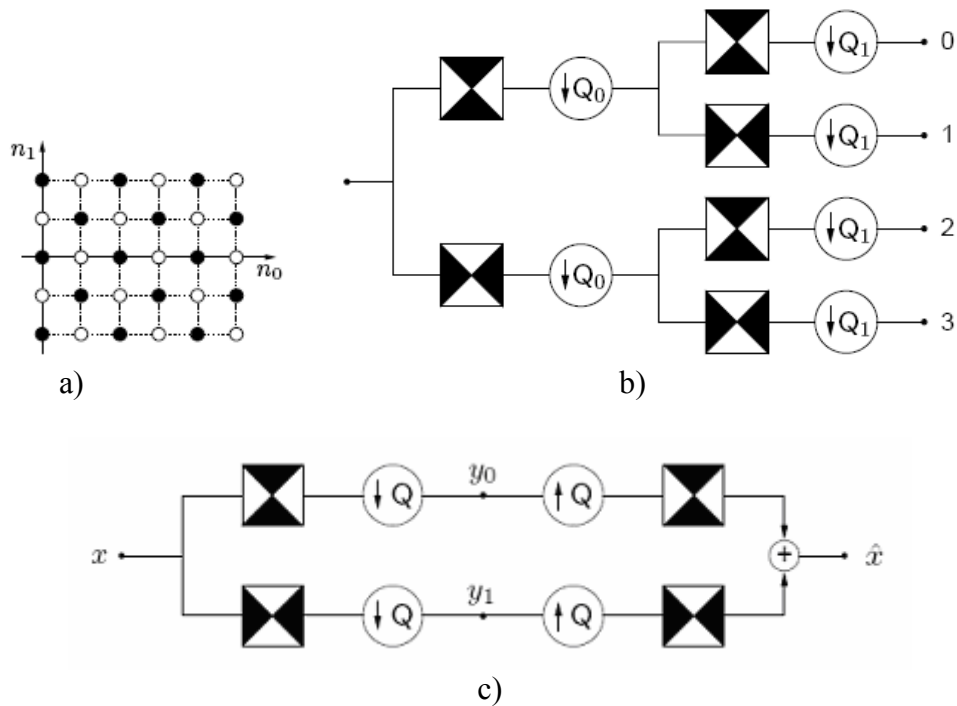


Figura 2.19. a) Enrejado de muestreo romboidal. b) Partición del espectro bidimensional usando el banco de filtros con soporte en forma de rombo, con filtros en forma de hélice. Las regiones negras representan la frecuencia ideal soportada por cada filtro. c) Primeros dos niveles de un DFB.



Figura 2.20. Ejemplo de una operación de remuestreo que es usado efectivamente como una operación de rotación para la descomposición con un banco de filtros direccionales DFB. a) La imagen “camera man” b) Imagen “camera man” después del remuestreo.

Usando el concepto de multitasa, podemos transformar un n^{esimo} - nivel del árbol estructurado en el banco de filtros direccionales en una estructura paralela de 2^n canales, con filtros equivalentes y todas aquellas matrices del muestreo.

Cabe señalar que estos filtros sintetizados equivalentes a: $G_k^{(n)}, 0 \leq k \leq 2^n$ corresponden a las capas de la figura 2.18 a). Las matrices sobremuestreadas tienen una forma diagonal dada por:

$$S_k^{(n)} = \begin{cases} \text{diag}(2^{n-1}, 2) & \text{for } 0 \leq k < 2^{n-1} \\ \text{diag}(2, 2^{n-1}) & \text{for } 2^{n-1} \leq k < 2^n \end{cases} \quad (2.15)$$

Estas matrices corresponden a las capas esencialmente horizontales y verticales, respectivamente. Con esto, es fácil ver que la familia de la siguiente ecuación (2.16), se obtiene trasladando la respuesta al impulso de los filtros de síntesis $G_k^{(n)}$ sobre el enrejado muestreado $S_k^{(n)}$, lo cual representa una base para las señales discretas en $n^2(Z^2)$. Esta base exhibe características direccionales y de localización.

$$\left\{ g_k^{(n)} \left[l - S_k^{(n)} m \right] \right\}_{0 \leq k < 2^n, m \in \mathbb{Z}^2} \quad (2.16)$$

La figura 2.21 muestra este hecho mostrando las respuestas al impulso de los filtros equivalentes, lo cual constituye un ejemplo de banco de filtros direccionales. Estas funciones base tienen propiedades lineales en el espacio y cubren todas las direcciones. Por lo tanto se asemejan a una transformada local tipo Radon [2], por lo que las funciones base se refieren como *radonlets*.

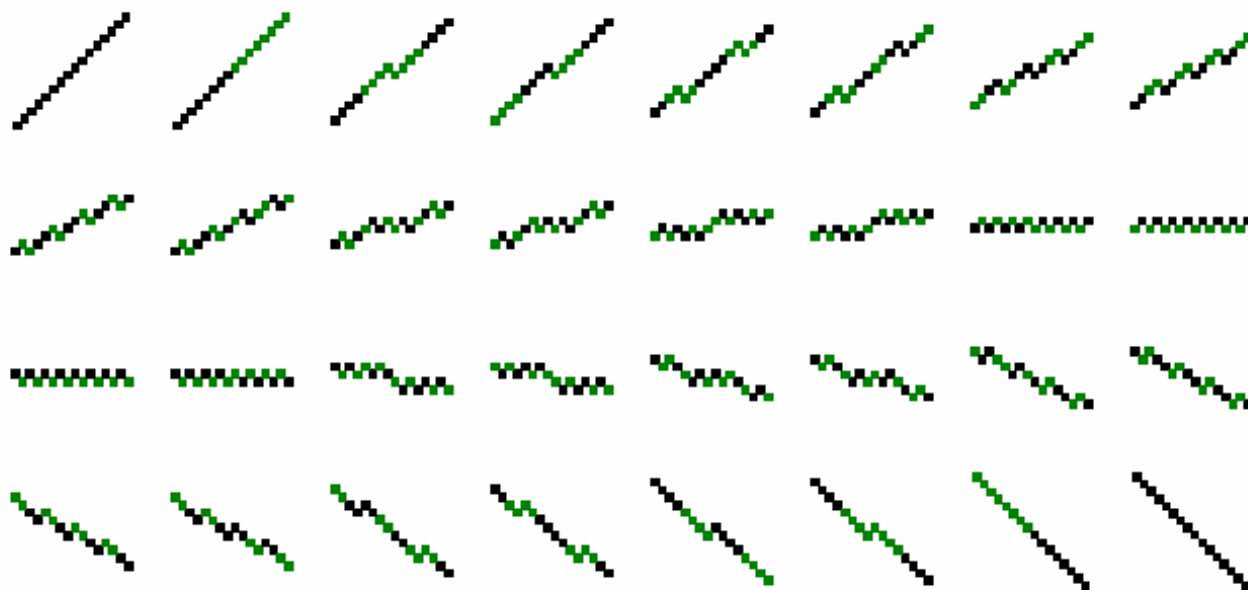


Figura 2.21. Respuestas al impulso de 32 filtros equivalentes para la primera mitad de los canales de un DFB de 6 niveles, los cuales usan filtros Haar. Los cuadros negros y grises corresponden a valores +1 y -1 respectivamente. Ya que las funciones de base se asemejan a “líneas locales”, reciben el nombre de *radonlets*.

2.4.4. Direcciones Múltiples.

Usando filtros ortogonales en el banco de filtros direccional piramidal (de sus siglas en inglés, PDFB) la base discreta $\left\{ g_k^{(l)}[n - S_k^{(l)}m] \right\}_{0 \leq k < 2^l, m \in \mathbb{Z}^2}$ del banco de filtros direccional (DFB) puede representar un cambio de base para el subespacio continuo de la descomposición multiescala.

Aplicando el DFB al subespacio multiresolución V_j , se tiene.

$$\theta_{j,k,n}^{(l)}(t) = \sum_{m \in \mathbb{Z}^2} g_k^{(l)}[m - S_k^{(l)}n] \phi_{j,m}(t) \quad (2.17)$$

Donde la familia $\left\{ \theta_{j,k,n}^{(l)} \right\}_{n \in \mathbb{Z}^2}$ es una base ortonormal del subespacio direccional $V_{j,k}^{(l)}$ para cada $k=0, \dots, 2^l - 1$. Estos subespacios son ortogonales con:

$$V_{j,k}^{(l)} = V_{j,2k}^{(l+1)} \oplus V_{j,2k+1}^{(l+1)} \quad \text{y} \quad (2.18)$$

$$V_j = \bigoplus_{k=0}^{2^l-1} V_{j,k}^{(l)} \quad (2.19)$$

La figura 2.22 muestra el subespacio dividido “bi-direccional” obtenido con el banco de filtros direccionales (DFB) en el dominio de la frecuencia. En el dominio espacial $V_{j,k}^{(l)}$ corresponde a un subespacio a escala 2^j y una dirección k para todas 2^l direcciones.

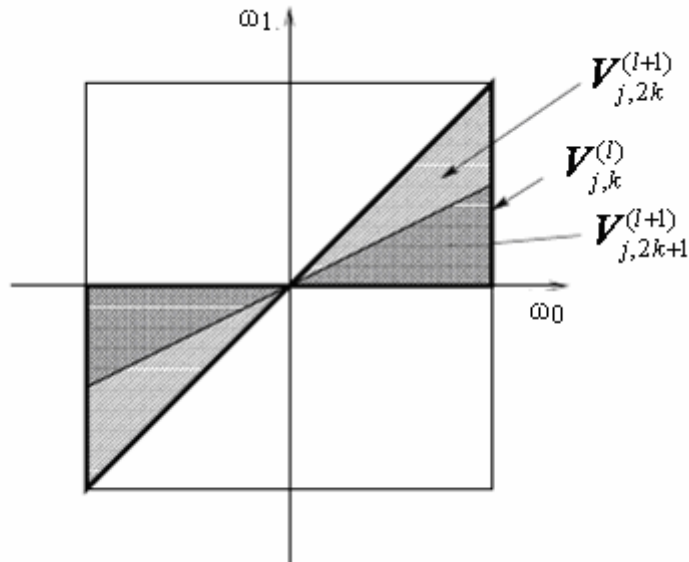


Figura 2.22: Subespacios multidireccionales generados por un banco de filtros direccionales (DFB).

¹Prueba ref. [2]: Este resultado es probado por inducción en el número de los niveles l de la descomposición del DFB, más o menos de la misma manera para las bases de los paquetes de la wavelet [12,13].

Aplicando la descomposición direccional por la familia $\left\{ \mathbf{g}_k^{(l)} [n - \mathcal{S}_k^{(l)} m] \right\}_{0 \leq k < 2^l, m \in \mathfrak{Z}^2}$ sobre el detalle del subespacio W_{j+1} según lo obtenido con el banco de filtros direccional piramidal (PDFB), se obtiene el siguiente un resultado similar.

Definiendo:

$$\rho_{j,k,n}^{(l)}(t) = \sum_{m \in \mathfrak{Z}^2} \mathbf{g}_k^{(l)} [m - \mathcal{S}_k^{(l)} n] \mu_{j,m}(t) \quad (2.20)$$

La familia $\left\{ \rho_{j,k,n}^{(l)} \right\}_{n \in \mathfrak{Z}^2}$ es un marco ajustado de un subespacio $W_{j+1,k}^{(l)}$ con fronteras del marco igual a 1, para cada $k = 0, \dots, 2^l - 1$. Estos subespacios son ortogonales con:

$$W_{j+1,k}^{(l)} = W_{j,2k}^{(l+1)} \oplus W_{j,2k+1}^{(l+1)}, \quad y \quad (2.21)$$

$$W_{j+1} = \bigoplus_{k=0}^{2^l-1} W_{j+1,k}^{(l)} \quad (2.22)$$

La figura 2.23 muestra una representación grafica de los subespacio, visto en el dominio de la frecuencia. La razón para que $\left\{ \rho_{j,k,n}^{(l)} \right\}_{n \in \mathfrak{Z}^2}$ sea un sistema sobrecompleto (overcomplete) para $W_{j+1,k}^{(l)}$ es porque utiliza la misma cuadrícula de muestreo, que el subespacio más grande $V_{j,k}^{(l)}$.

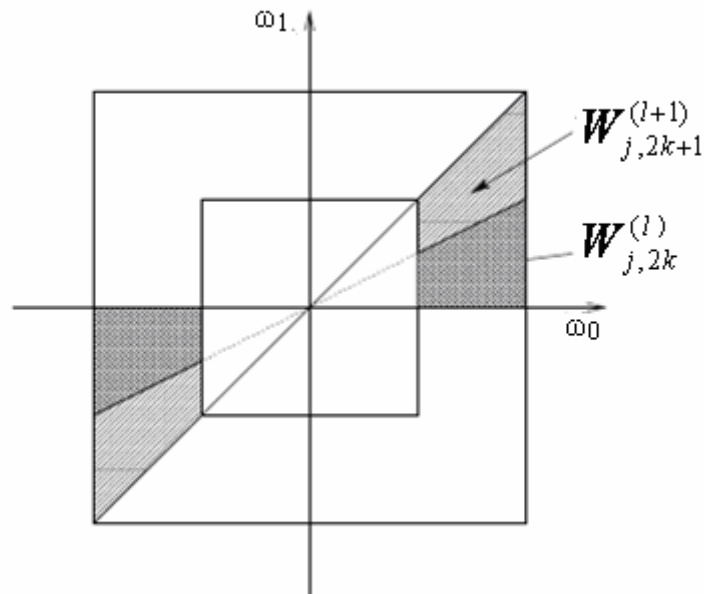


Figura 2.23. Subespacios generados multidireccionalmente por el PDFB.

Se puede señalar que W_{j+1} no es invariante al corrimiento, pero el resultado siguiente establece que sus subespacios $W_{j+1,k}^{(l)}$ son generados por una sola función prototipo.

Entonces para obtener dicho resultado, sea:

$$\rho_{j,k}^{(l)}(t) = \sum_{m \in \mathfrak{I}^2} g_k^{(l)}[m] \mu_{j,m}(t) \quad (2.23)$$

Para $l \geq 2$

$$\rho_{j,k,n}^{(l)}(t) = \rho_{j,k}^{(l)}(t - 2^j S_k^{(l)} n) \quad (2.24)$$

Consecuentemente, el subespacio $W_{j+1,k}^{(l)}$ satisface la siguiente propiedad de invarianza al corrimiento:

$$f(t) \in W_{j+1,k}^{(l)} \Leftrightarrow f(t - 2^j S_k^{(l)} n) \in W_{j+1,k}^{(l)}, \quad \forall n \in \mathfrak{I}^2 \quad (2.25)$$

Esto establece que los subespacios direccionales $W_{j+1,k}^{(l)}$ de la multiescala son definidos en una cuadrícula rectangular con intervalos $2^{j+l-1} \times 2^{j+1}$, (o $2^{j+1} \times 2^{j+l-1}$, dependiendo de si es esencialmente horizontal o vertical).

Substituyendo (2.12) en (2.23), podemos escribir la función prototipo $\rho_{j,k}^{(l)}(t)$ directamente como una combinación lineal de la función de escala $\phi_{j,m}(t)$ de tal manera que:

$$\begin{aligned} \rho_{j,k}^{(l)}(t) &= \sum_{i=0}^3 \sum_n g_k^{(l)}[2n + k_i] \left(\sum_{m \in \mathfrak{I}^2} f_i[m] \phi_{j,n+m} \right) \\ &= \sum_{m \in \mathfrak{I}^2} \underbrace{\left(\sum_{i=0}^3 \sum_{n \in \mathfrak{I}^2} g_k^{(l)}[2n + k_i] f_i[m - n] \right)}_{c_k^{(l)}[m]} \phi_{j,m}(t) \end{aligned} \quad (2.26)$$

Donde k_i son los puntos representativos del enrejado de sub-muestreo dados por: $k_0 = (0,0)^T, k_1 = (1,0)^T, k_2 = (0,1)^T, k_3 = (1,1)^T$ usados para satisfacer la invarianza al corrimiento para W_{j+1} con $F_i(z) = z^{-k_i} F(z)$

¹ Prueba ref. [2]: por sustitución directa y cambio de variable

La secuencia $c_k^{(l)}[m]$ se asemeja a una adición de convoluciones entre $g_k^{(l)}[m]$ y $f_i[m]$, por lo tanto es un filtro pasa altas y direccional. La ecuación (2.26) revela el aspecto *contourlet* de la función prototipo $\rho_{j,k}^{(l)}(t)$, donde se ve como un agrupamiento de elementos de detección de bordes en una escala j y a lo largo de una dirección k .

2.4.5. Multiescala y descomposición direccional

El banco de filtros direccional (DFB) se diseña para capturar las componentes de alta frecuencia, las cuales representan direccionalidad en las imágenes. Por lo tanto, los componentes de baja frecuencia son manejados de manera incorrecta por este banco de filtros. De hecho, con la partición de la frecuencia mostrada en la figura 2.15, las frecuencias bajas se fragmentarían en varias capas direccionales, por lo tanto, el banco de filtros direccional no proporciona una representación dispersa para imágenes. Para mejorar la situación, las frecuencias bajas deben ser removidas antes del filtrado direccional. Esto proporciona otra razón para combinar el banco de filtros direccionales con un esquema del multiresolución.

Con esto en mente, la pirámide laplaciana permite que la descomposición direccional de capas sea aplicada en sus imágenes pasa-banda. Esas imágenes pasa-banda pueden ser alimentadas dentro de un banco de filtros direccional para poder capturar la información direccional eficientemente. El esquema se puede iterar en varias ocasiones en la imagen pasa-bajas (figura 2.24). El resultado final es una estructura doblemente iterada del banco de filtros, llamada *banco de filtros direccional piramidal (PDFB)*, que descompone imágenes en subbandas direccionales en múltiples escalas. El esquema es flexible puesto que permite un diverso número de direcciones en cada escala.

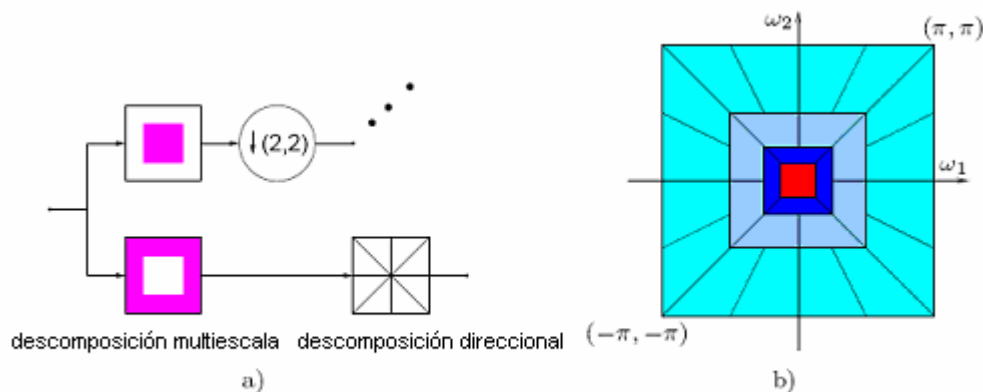


Figura 2.24: Banco de filtros direccional piramidal. a) Bloques. Primero, una descomposición estándar multiescala, en 8 bandas, donde el canal pasa-bajas (pirámide laplaciana) es sub-muestreado mientras que el pasa-altas no. Después, una descomposición direccional con un banco de filtros DFB es aplicado a cada canal pasa-altas. b) División de frecuencias resultante, donde el número de direcciones es incrementado con la frecuencia.

Con una reconstrucción perfecta de la pirámide laplaciana y un banco de filtros direccionales, el banco de filtros direccional piramidal tiene una reconstrucción perfecta, y es así un operador de marco para las señales en dos dimensiones. El banco de filtros direccional piramidal tiene la misma redundancia que la pirámide laplaciana.

Respecto al banco de filtros de la *wavelet*, el PDFB iterado se puede asociar a un sistema del dominio continuo, que conocemos con el nombre de *contourlet* [14, 15]. Las características son direcciones múltiples y una combinación con el aspecto multiescala.

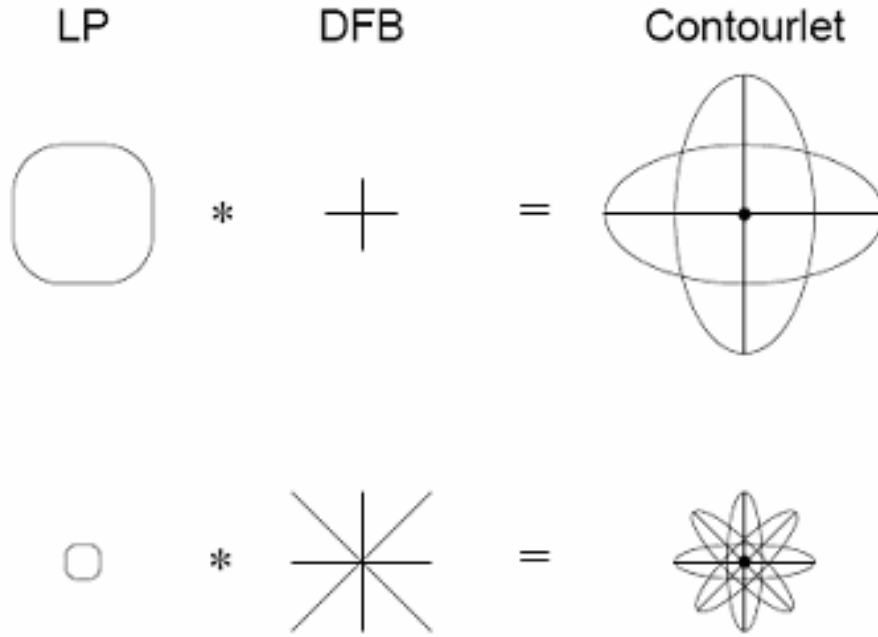


Figura 2.25: Ilustración de imágenes de base para una *contourlet* que satisfacen la relación del escalamiento de la curva. Desde la línea superior a la inferior, la escala es reducida por cuatro mientras el número de direcciones es duplicado.

Finalmente, integrando sobre las escalas (redundantes) tenemos el resultado siguiente para los marcos de la *contourlet* en el espacio $L^2(\mathbb{R}^2)$.

Para una secuencia de enteros finitos positivos $\{l_j\}_{j \leq j_0}$ la familia $\{\phi_{j_0,n}(t), \rho_{j,k,n}^{(l_j)}(t)\}_{j \leq j_0, 0 \leq k \leq 2^l j-1, n \in \mathfrak{T}^2}$ (2.27) es un marco ajustado de $L^2(\mathbb{R}^2)$. Para la secuencia de enteros finitos positivos $\{l_j\}_{j \in \mathfrak{T}}$ la familia $\{\rho_{j,k,n}^{(l_j)}(t)\}_{j \in \mathfrak{T}, 0 \leq k \leq 2^l j-1, n \in \mathfrak{T}^2}$ es un marco ajustado direccional de $L^2(\mathbb{R}^2)$ ¹.

¹ Dicho resultado es obtenido en ref. [2] aplicando lo propuesto en la página 50, a la siguiente descomposición de $L^2(\mathbb{R}^2)$ dentro de subespacios ortogonales mutuos [2].

$$L^2(\mathbb{R}^2) = V_{j_0} \oplus \left(\bigoplus_{j \leq j_0} \right), \quad y \quad L^2(\mathbb{R}^2) = \bigoplus_{j \in \mathfrak{T}} W_j$$

El marco ajustado en la ecuación (2.27) proporciona una *curvelet* como su extensión cuando el número de direcciones es duplicado en cada una de las escalas más finas.

Esto significa que si en la escala 2^{j_0} comenzamos con un nivel de descomposición l_{j_0} para el banco de filtros direccionales (que tiene direcciones $2^{l_{j_0}}$, entonces en la escala más fina 2^j ; $j < j_0$, el número de los niveles de descomposición obtenidos con el DFB deberá ser:

$$l_j = \lceil l_{j_0} - (j - j_0) / 2 \rceil, \quad \text{para } j \leq j_0 \quad (2.28)$$

Así la cuadrícula incrustada de la aproximación para la expansión *curvelet* del banco de filtros direccional piramidal a la escala 2^j es $2^{\lceil n_0 + j / 2 \rceil} \times 2^j$ para direcciones básicamente horizontales y $2^j \times 2^{\lceil n_0 + j / 2 \rceil}$ para direcciones verticales, donde $n_0 = l_{j_0} - j_0 / 2 + 2$. La figura 2.26 ilustra este patrón de muestreo para diferentes escalas y direcciones. El punto principal a notar aquí es que en el proceso de refinamiento, una dimensión espacial está refinada dos veces respecto a la otra dimensión espacial.

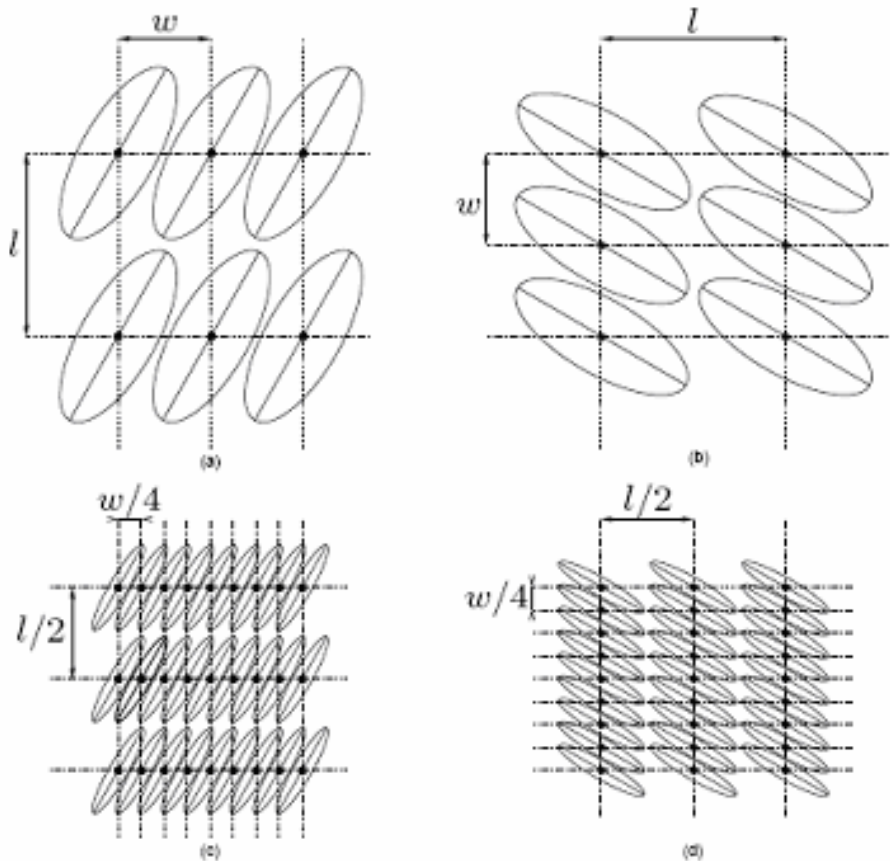


Figura 2.26. Cuadrículas incrustadas de la aproximación en el dominio espacial para un banco de filtros direccional piramidal que implementa la transformada *curvelet*. Éstos son 4 subespacios ilustrativos burdos y finos $W_{f,k}$ que representan escalas, y direcciones básicamente horizontales y verticales. Cada subespacio es recorrido por la función de la *curvelet* dada en (2.23)

Los intervalos de muestreo se emparejan con la ayuda de la función prototipo, por ejemplo de anchura W y longitud l , de modo que los cambios cubrirán el plano \mathbb{R}^2 .

Las figuras 2.27 y 2.28 dan una vista completa de la aproximación multiresolucional de la construcción *curvelet* basada en el banco de filtros direccional piramidal. Claramente se muestra un esquema refinado donde la resolución se incrementa en ambos dominios, espacial y en dirección, desde burda hasta fina, y exhibe los elementos básicos de escalas anisotrópicas respecto a curvas.



Figura 2.27. Ejemplos de la transformada *contourlet* de las imágenes “Peppers” y “Bárbara”. Para una visualización más clara, cada imagen está solamente descompuesta en dos niveles de la pirámide, a su vez descompuestos en 4 y 8 capas direccionales. Pequeños valores de coeficientes son mostrados en negro mientras que grandes valores son mostrados en blanco.

A manera de ilustración, en la figura 2.28 se comparan los efectos de las transformaciones *wavelets* y *contourlets* en la imagen Zoneplate.

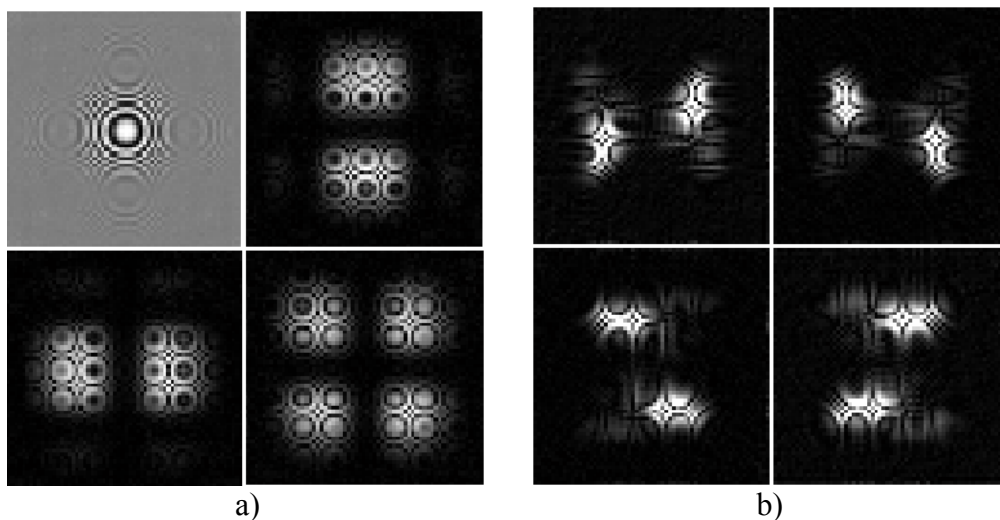


Figura 2.28. Comparación de las transformadas: a) Tercer nivel de 2-D *wavelet daubechies*. y b) Tercer nivel de la transformada *contourlet*.

2.5. La contourlet redundante

Tomando como base el trabajo de N. Baaziz [1], en relación con las imágenes, y sus principales objetivos en cuanto a procesamiento, como es la compresión, es necesario utilizar transformadas muestreadas críticamente, esto es utilizar un muestreo con el menor número de coeficientes que permitan la recuperación de la imagen original. Esto con el teorema de Shannon y la entropía. Sin embargo, para lo que se refiere al análisis de imágenes y sus propósitos primitivos de extracción, la transformada redundante es preferida, la cual no utiliza el muestreo crítico antes mencionado ya que permite un mejor manejo de la información.

Además, debido a que el proceso jerárquico requiere la cooperación o mapeo entre los varios niveles de resolución de las imágenes, por razones prácticas de procesamiento es preferible que todos los niveles sean del mismo tamaño. Esta es la razón por la que es introducida la transformada *contourlet* redundante que proporciona capas direccionales del mismo tamaño.

Baaziz [1], propone alcanzar la redundancia en el esquema de la pirámide laplaciana eliminando cualquier operación de submuestreo, usando L filtros pasa-bajas apropiados, lo cual lleva a la creación de L aproximaciones pasa-bajas de la imagen. La diferencia entre cada aproximación y su versión subsiguiente pasa-bajas es una imagen pasa-bandas, donde el resultado final es una pirámide redundante Laplaciana (por sus siglas en inglés, RLP) con los niveles iguales de tamaño $L+1$, una aproximación burda y L imágenes pasa-bandas de la imagen.

Aplicando la misma descomposición direccional de D -niveles (con muestreo crítico) sobre cada RLP pasa-banda, se obtiene una transformada *contourlet* redundante con

capas direccionales de igual tamaño LD , además de la aproximación burda de la imagen (figura 2.29). Cada capa *contourlet* es representada por una subimagen $\{C_{l,d}(i,j), i=1\dots M, j=1\dots N\}$ donde l es su nivel de resolución, d es su dirección en frecuencia y MN es el número total de sus coeficientes.

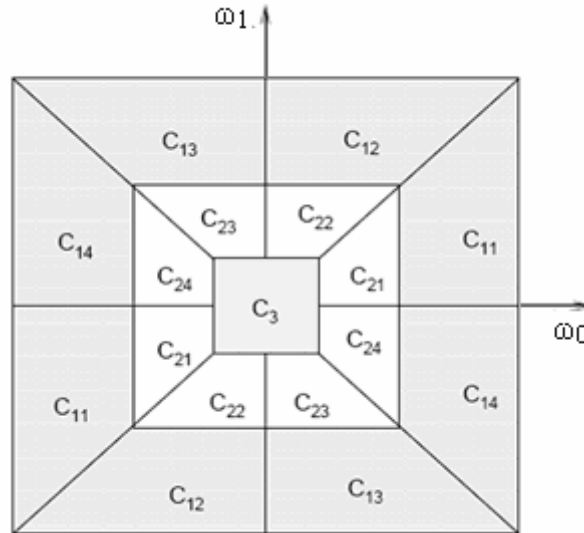


Figura 2.29: Representación frecuencial de una descomposición *contourlet* con 3 niveles de resolución ($L = 2$) y 4 direcciones de frecuencia ($D = 4$). La aproximación burda de la imagen C_3 no es descompuesta direccionalmente.

2.5.1. Algoritmo de marcado de agua # 1 ([1]).

2.5.1.1 Generación de la marca de agua e incrustación

En el algoritmo desarrollado en [1], existe una marca de agua W la cual será incrustada en una imagen. La marca de agua es estructurada como un sistema de una o más matrices $W_{l,d}(i,j)$ con el tamaño MN y valores binarios pseudos-aleatorios $\{-1,1\}$. Los índices l,d indican cuál capa *contourlet* $C_{l,d}$ está siendo marcada por $W_{l,d}$.

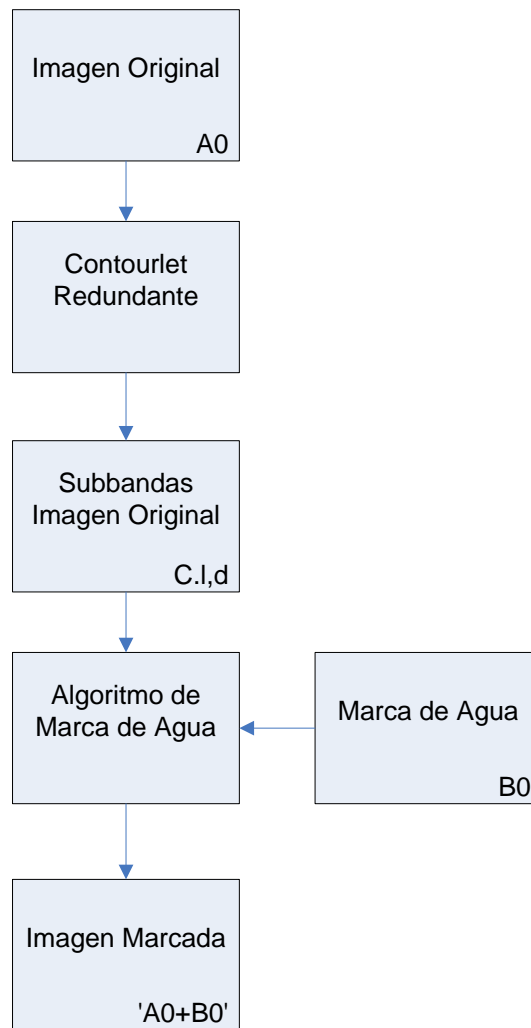


Figura 2.30. Diagrama de Flujo del sistema propuesto en [1].

En el trabajo [1], se utiliza la ecuación siguiente, insertando datos de la marca de agua en las capas *contourlet* de la imagen:

$$C'_{l,d}(i, j) = C_{l,d}(i, j) + \alpha M_d(i, j)W_{l,d}(i, j) \quad (2.29)$$

donde $C_{l,d}$ es una capa *contourlet* marcada con un nivel de resolución l y una dirección frecuencial d , el valor α es un parámetro de control, y M_d es una máscara direccional de peso usada para adaptar el nivel de marcado y de la invisibilidad de la marca de agua según las características locales de la imagen para una dirección frecuencial dada d . En [1] se calcula la máscara peso introduciendo algunas modificaciones al trabajo preliminar propuesto en [16]. Desde el punto de vista perceptual o sistema visual humano (de sus siglas en inglés HVS), la sensibilidad del sistema de visión a las variaciones del ruido y del brillo, así como la presencia de las características significativas de la imagen (texturas, bordes) se toman en consideración. Para una dirección de la frecuencia de la descomposición *contourlet*, M_d es creada de acuerdo a:

$$M_d(i, j) = \frac{1}{2} B(i, j) T(i, j)^{0.2} E(d, i, j)^{0.2} \quad (2.30)$$

Donde:

$$B(i, j) = 1 + \frac{1}{256} C_{L+1}(2i-1, 2j-1) \quad (2.31)$$

$$T(i, j) = Var\{C_{L+1}(2i-1+y, 2j-1+x)\}_{x=0,2; y=0,2} \quad (2.32)$$

$$E(d, i, j) = \frac{1}{4} \sum_{l=1}^L \sum_{x=0,1} \sum_{y=0,1} [C_{l,d}(i+y, j+x)]^2 \quad (2.33)$$

El factor B es proporcional a los valores del brillo de la sub-imagen burda en resolución mientras que T y E capturan respectivamente la actividad de la textura de la vecindad en la sub-banda pasa-bajas y la presencia de los bordes orientados de la imagen en las capas direccionales. Después de la etapa de inserción, se construye la imagen marcada usando el algoritmo *contourlet* redundante.

2.5.1.2 Detección de la marca

El algoritmo de la detección de la marca de agua se basa en la comparación de un valor de correlación R' con un umbral dado T . El valor R' es una medida promedio de la correlación entre los datos de la marca de agua y las capas *contourlet* apropiadas de una imagen dada. Asumiendo una marca de agua W , una imagen dada, y un esquema de marcado con K como el número total de capas marcadas durante el proceso de inserción, el proceso de la detección comienza con la descomposición de la imagen dada en sus capas *contourlet* y cálculo de la correlación R' según las ecuaciones siguientes:

$$R' = \frac{1}{K} \sum_l \sum_d R(l, d) \quad (2.34)$$

Donde:

$$R(l, d) = \begin{cases} \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N C'_{l,d}(i, j) W_{l,d}(i, j) & \text{si } C_{l,d} \text{ es marcado} \\ 0 & \text{otro} \end{cases} \quad (2.35)$$

Si $R' > T$ entonces la marca de agua genuina W está presente en la imagen. Este método de marcado es del tipo “ciego” puesto que el proceso de la detección no requiere el uso de la imagen original (sin marcar). Para determinar el valor del umbral T , se adopta un cómputo a posteriori que se adapta del criterio de Neyman-Pearson [16] a las capas *contourlet* marcadas como sigue:

$$T = 3.97 \sqrt{2\sigma^2} \quad (2.36)$$

Donde:

$$\sigma^2 = \frac{1}{(KMN)^2} \sum_l \sum_d \sum_{i=1}^M \sum_{j=1}^N C'_{l,d}(i,j)^2 \quad (2.37)$$



Figura 2.31: La imagen Lena marcada y su diferencia absoluta con la imagen original, magnificada por un factor de 8.

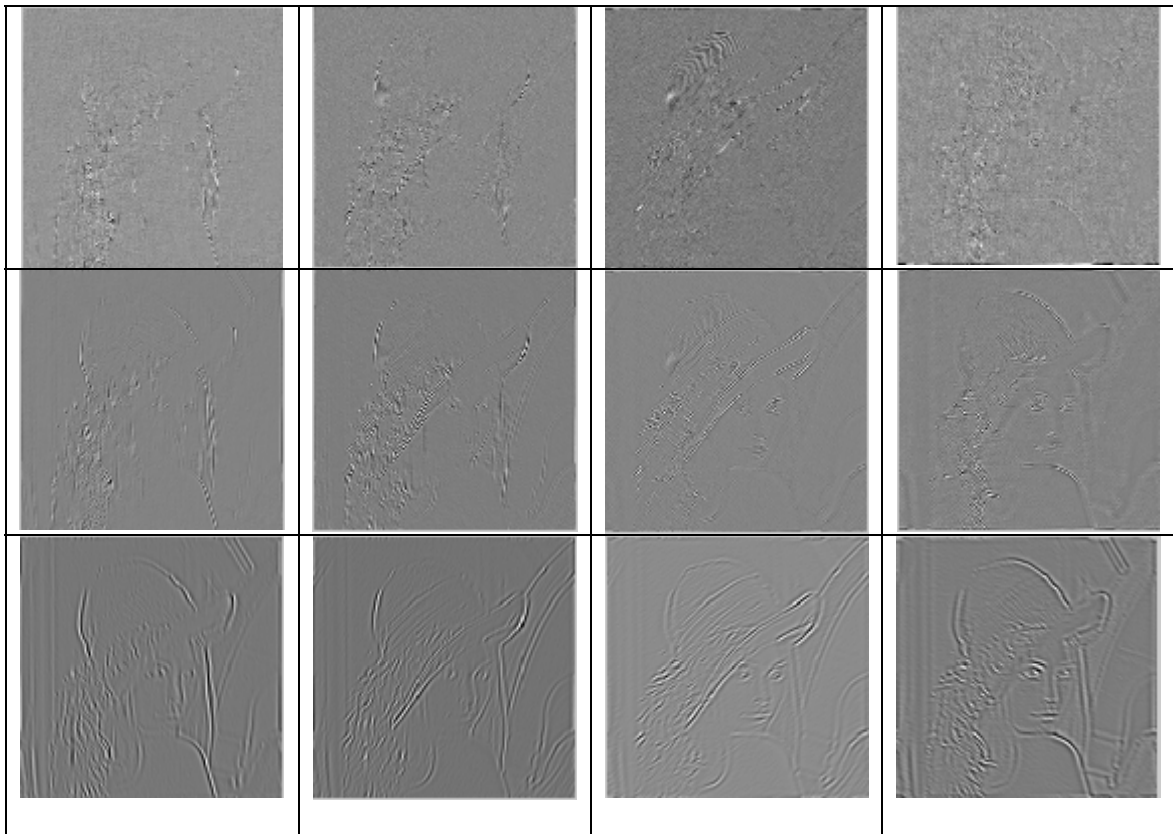


Figura 2.32: Capas *contourlet* de la imagen Lena



Figura 2.33: Imagen obtenida de filtros pasa-bajas.

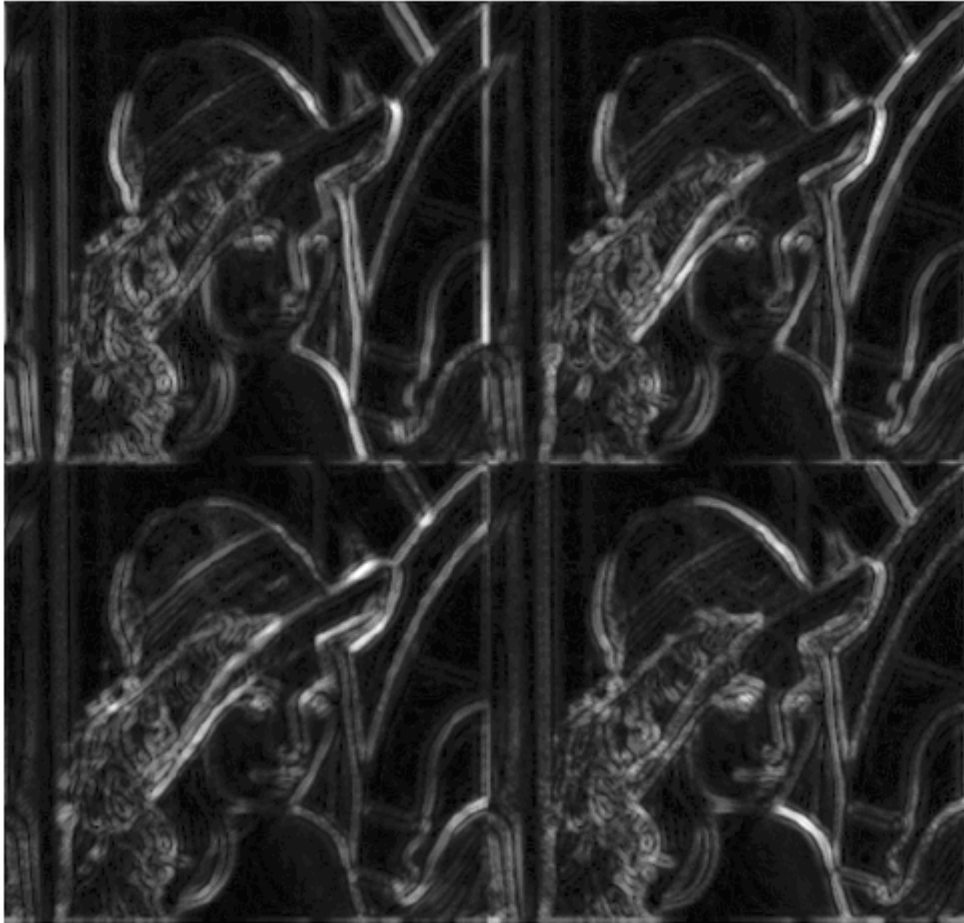


Figura 2.34. Las 4 máscaras de peso direccionales, respectivamente, M1, M2, M3, M4 (de arriba a abajo y de izquierda a derecha) calculado a partir de la *contourlet* redundante de Lena con $L = 3$ y $D = 4$.

2.5.2. Algoritmo de marcado de agua # 2 (propuesto)

Para el sistema de marcado de agua propuesto en esta tesis, la marca de agua será solo una, o varias capas de una imagen W , esta imagen será del tamaño de la imagen a marcar C , la imagen W será de tamaño MN con valores pseudos-aleatorios $\{-1,1\}$ dando lugar a las capas *contourlet* $W_{l,d}^*(i, j)$, estas capas marcarán respectivamente las capas de la imagen $C_{l,d}(i, j)$. Donde los índices l,d indican las capas que serán marcadas, (la capa $C_{l,d}(i,j)$ será marcada por la capa $W_{l,d}^*(i, j)$).

2.5.2.1. Creación de la marca de agua

La mayoría de los generadores de números aleatorios generan una secuencia de números enteros por la repetición siguiente:

$$x_0 = \text{dado}, \quad x_{n+1} = P_1 x_n + P_2 \pmod{N} \quad n = 0,1,2,\dots \quad (*) \quad (2.38)$$

La notación $\text{mod } N$ significa que la expresión en la derecha de la ecuación es dividida por N , y conservando sólo el residuo.

Ejemplo:

$$x_0 = 79, N = 100, P_1 = 263, \text{ y } P_2 = 71$$

Entonces:

$$x_1 = 79*263 + 71 \pmod{100} = 20848 \pmod{100} = 48,$$

$$x_1 = 48*263 + 71 \pmod{100} = 12695 \pmod{100} = 95,$$

$$x_1 = 95*263 + 71 \pmod{100} = 25056 \pmod{100} = 56,$$

$$x_1 = 56*263 + 71 \pmod{100} = 14799 \pmod{100} = 99,$$

Los números subsecuentes son: 8, 75, 96, 68, 36, 39, 28, 35, 76, 59, 88, 15, 16, 79, 48. Luego la secuencia se repite. Esto indica una debilidad en el generador: Si los números aleatorios están entre 0 y 99 entonces se quisiera que cada número entre 0 y 99 fuera un miembro posible de la secuencia.

Los parámetros P_1 , P_2 y N determinan las características del generador de números aleatorios, y la selección de x_0 (la semilla) determina la secuencia particular de números aleatorios que se generará. Si el generador funciona con los mismos valores de los parámetros, y la misma semilla, generará una secuencia idéntica a la anterior. En éste sentido los números generados ciertamente no son aleatorios, sino pseudos-aleatorios.

Para el algoritmo implementado en esta tesis, la creación de la marca de agua se realizó por medio de números pseudos-aleatorios determinados por la siguiente sección del programa:

```

for j = 1 : n
for i = 1 : m
seed = floor ( seed );
seed = mod ( seed, 2147483647 );
if ( seed < 0 )
seed = seed + 2147483647;
end
k = floor ( seed / 127773 );
seed = 16807 * ( seed - k * 127773 ) - k * 2836;
if ( seed < 0 )
seed = seed + 2147483647;
end
a(i,j) = b + ( c - b ) * seed * 4.656612875E-10;
end
end

```

Por lo tanto el diagrama de flujo es el siguiente

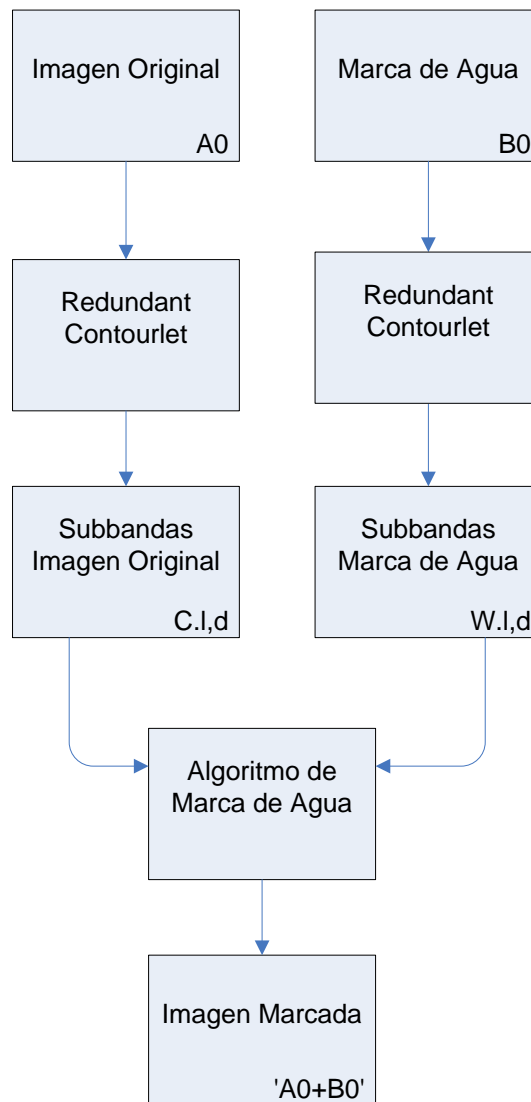


Figura 2.35: Diagrama de flujo del algoritmo propuesto.

La ecuación de marcado será:

$$C'_{l,d}(i, j) = C_{l,d}(i, j) + \alpha M_d(i, j)W^*_{l,d}(i, j) \quad (2.39)$$

Donde, al igual que en el algoritmo #1 [1], $C'_{l,d}$ será una capa *contourlet* marcada con un nivel de resolución l y una dirección frecuencial d , el valor α es un parámetro de control (se usa el mismo valor del sistema anterior, para no hacer diferencias entre ambos), y M_d es una máscara direccional de peso usada para adaptar el nivel de marcado y de invisibilidad de la marca de agua según las características locales de la imagen para una dirección frecuencial dada d .

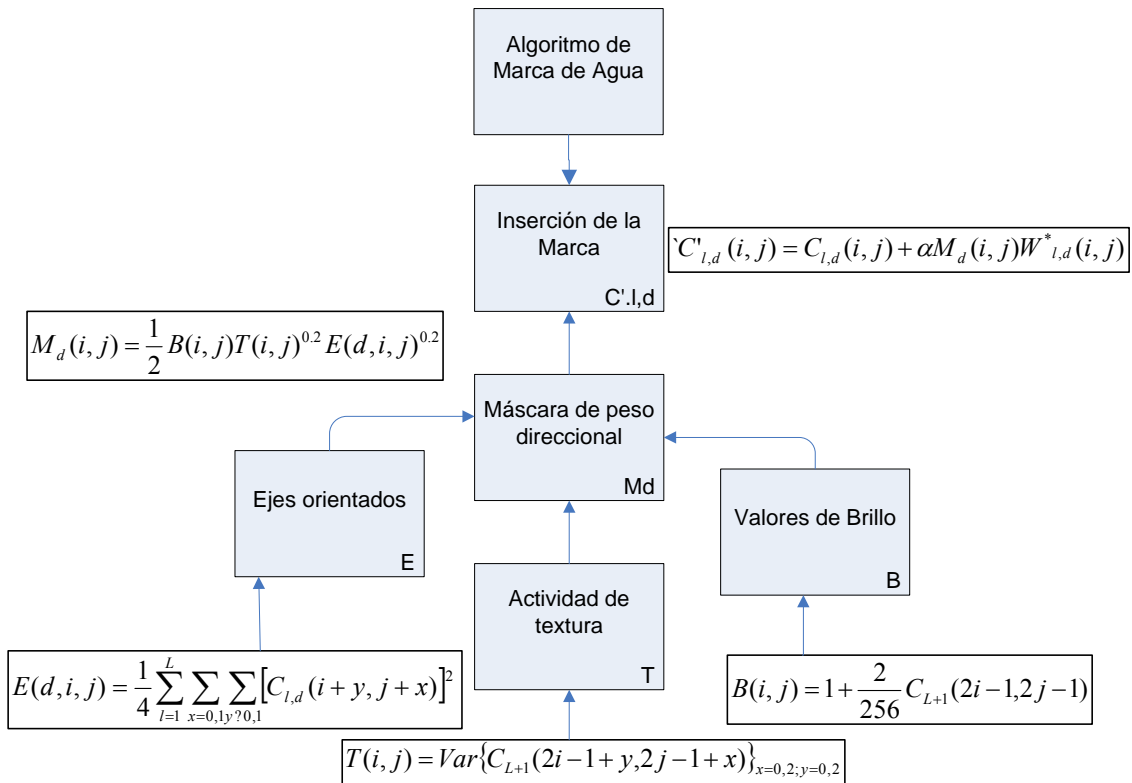


Figura 2.36: Diagrama de flujo para la inserción de la marca, por medio de la máscara de peso direccional.

Donde las variables B , T y E representan lo mismo que en el algoritmo redesarrollado en [1], es decir, B es proporcional a los valores del brillo, T y E refieren respectivamente a la textura de la vecindad en la capa pasa-bajas y a la presencia de los bordes orientados de la imagen en las capas direccionales.

2.5.2.2. Obtención de T y E

Para el cálculo de T y E se tomó en consideración el problema de obtención de la varianza en los márgenes de la imagen.

Al procesar una imagen completa con dimensiones MN de una manera local, cada píxel $C_{l,d}(x, y)$ se procesa alternadamente usando solamente su vecindad inmediata del píxel. Pero en este procedimiento existen dificultades al procesar los píxeles que están en los márgenes de la imagen, ya que estos píxeles tienen vecindades incompletas. La figura (2.34) muestra dicha situación para $f(0,0)$ al usar una ventana 3×3 . Los 5 píxeles dentro de la ventana punteada marcados con “?” se denotan como faltantes, y los 4 restantes se denotan como píxeles disponibles.

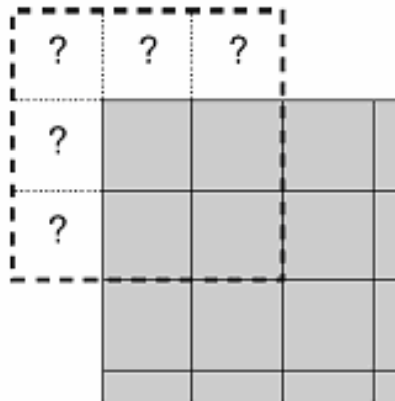


Figura 2.37. Vecindad incompleta en una ventana de 3×3 , en los márgenes de la imagen.

Existen varias formas para manejar esta situación:

- Completar con una constante (*Constant fill*)

En este esquema, los píxeles que faltan se substituyen por un valor constante fijo, más comúnmente cero, o tonos medios de gris.

- Cubrir con su entorno (*Wrap around*)

La imagen se trata como un cilindro, como si la parte superior y la parte inferior, y la parte izquierda y la derecha, fueran espacialmente próximas. Esto se logra manejando las coordenadas de los píxeles con una operación módulo.

- Vecinos más cercanos (*Nearest neighbour*)

Los píxeles que faltan adquieren el valor igual a su vecino disponible más cercano.

- El llenado con un valor promedio (*Average fill*)

Los píxeles faltantes se fijan igual a la media aritmética de los píxeles disponibles.

- Especialización (*Specialization*)

El algoritmo se modifica específicamente para manejar los casos especiales donde la vecindad contiene pocos píxeles.

Los dos primeros métodos de la lista, son métodos claramente inferiores porque inventan los valores de los píxeles y dan poca importancia a los píxeles disponibles. El método de la especialización es deseable, pero aun cuando con una ventana 3x3 existen ocho casos especiales a tratar (cuatro lados más cuatro esquinas), resulta difícil de poner en práctica eficientemente.

La figura 2.38 describe el método del “vecino más cercano”. Esta técnica es simple de poner en ejecución, y posee la capacidad de preservar cierta estructura en la imagen. Por ejemplo, si los píxeles disponibles, cerca del margen pertenecen a dos poblaciones separadas, la vecindad extrapolada también. La desventaja es que algunos valores del píxel, tales como las esquinas, están duplicados más de una vez, y por lo tanto hay una sobre-representación. Esto puede predisponer ciertos resultados, especialmente si los píxeles disponibles son muy ruidosos.

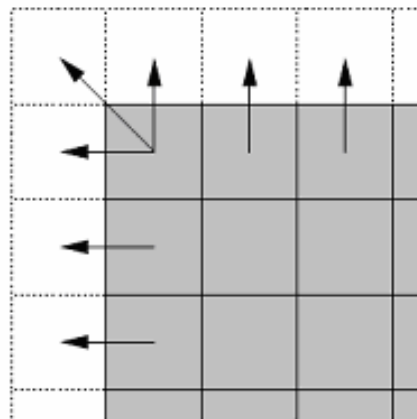


Figura 2.38. Píxeles faltantes reemplazados por el valor del píxel más cercano, en los márgenes de la imagen.

El siguiente código ilustra el procedimiento:

Creación de los márgenes para T

$$Q1 = [Q_i(1,1) \quad Q_i(1,:) \quad Q_i(256,1); \quad Q_i(:,1) \quad Q_i \quad Q_i(:,256); \quad \dots \\ Q_i(1,256) \quad Q_i(256,:) \quad Q_i(256,256)];$$

Creación de los márgenes para E

$$S1 = [S1, S1(:,256); S1(256,:), S1(256,256)]; \\ S2 = [S2, S2(:,256); S2(256,:), S2(256,256)]; \\ S3 = [S3, S3(:,256); S3(256,:), S3(256,256)];$$

El algoritmo de la detección de la marca de agua es el mismo que se utiliza en el algoritmo de marcado #1, una comparación de valor de correlación R' con un umbral dado T .

El único cambio es el que hay que contemplar el cambio en las ecuaciones (2.29) y (2.39). Dando entonces:

$$R' = \frac{1}{K} \sum_l \sum_d R(l, d)$$

Donde:

$$R(l, d) = \begin{cases} \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N C'_{l,d}(i, j) W^*_{l,d}(i, j) & \text{si } C_{l,d} \text{ es marcado} \\ 0 & \text{otro} \end{cases} \quad (2.40)$$

Y si $R' > T$ entonces la marca de agua genuina W^* está presente en la imagen.

$$T = 3.97 \sqrt{2\sigma^2}$$

Donde:

$$\sigma^2 = \frac{1}{(KMN)^2} \sum_l \sum_d \sum_{i=1}^M \sum_{j=1}^N C'_{l,d}(i, j)^2$$

3.1. Introducción

Después de haber realizado diferentes simulaciones con los algoritmos, las cuales comprendieron la inserción de la marca de agua con diferentes parámetros y la detección de dicha marca de agua después de haber realizado algunos cambios a la imagen marcada, se obtuvieron los resultados que se muestran en este capítulo.

Cabe señalar que el procedimiento de marcado de la imagen utilizado para el desarrollo del algoritmo propuesto en esta tesis es similar al algoritmo presentado en [1], esta similitud se expresa en el sentido de la utilización de los coeficientes B, T, y E de las ecuaciones (2.31) – (2.33) (lo cual constituye una modificación del sistema propuesto en [16]). De igual forma se utiliza una transformada *contourlet* redundante para una descomposición de la imagen a marcar, y en el caso de esta tesis, de la marca de agua también.

Sin embargo, a pesar de que existen dichas similitudes, y debido a que en este trabajo existe la descomposición de la marca de agua, por medio de la transformada *contourlet* redundante, se analiza el cambio de los resultados obtenidos con respecto al algoritmo propuesto en [1].

Para la obtención de los resultados, validación y comparación, se utilizó la imagen “lena” con un tamaño de 512 x 512, así como el programa StirMark Benchmark 4 [17], el cual sirve para la realización de diferentes pruebas y ataques en contra de la marca de agua.

3.2. Parámetros de las imágenes

Para la diferenciación entre la imagen original y las imágenes obtenidas por los algoritmos utilizados, son utilizadas las siguientes características para imágenes.

3.2.1. Histograma

Un histograma traza la frecuencia relativa de cada valor del píxel en escala de grises que ocurra en una imagen, y proporciona una gráfica conveniente de las intensidades en una imagen, pero no puede representar cualquier información con respecto a relaciones espaciales entre los píxeles.

3.2.2. Varianza

La varianza de una imagen cualquiera, $Var[f]$, donde f es la imagen de tamaño (X, Y) , da una estimación de la distribución de los valores de los píxeles alrededor de la media de la imagen f . Puede ser calculada de acuerdo a las ecuaciones (3.1), o (3.2). La última tiene la ventaja de requerir solamente un recorrido por la imagen. La desviación estándar es la raíz cuadrada de la varianza $E[f] = \sqrt{Var[f]}$

$$\begin{aligned}
 Var[f] &= E[f - E[f]]^2 \\
 &= \frac{1}{YX} \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} (f(x, y) - E[f])^2 \\
 &= \frac{1}{YX} \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} (f(x, y) - \frac{1}{YX} \sum_{y'=0}^{Y-1} \sum_{x'=0}^{X-1} f(x', y'))^2
 \end{aligned} \tag{3.1}$$

$$Var[f] = E[f^2] - E[f]^2 = \left(\frac{1}{YX} \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} f(x, y)^2 \right) - \left(\frac{1}{YX} \sum_{y=0}^{Y-1} \sum_{x=0}^{X-1} f(x, y) \right)^2 \tag{3.2}$$

3.2.3. La entropía

Normalizando el histograma de la imagen para que el área bajo la curva sea unitaria, se puede considerar una distribución de la probabilidad sobre los valores de los píxeles. Para el caso de una imagen en escala de grises, a partir del histograma de las intensidades se puede calcular la entropía de una imagen, dada por la ecuación 3.3. Al usar logaritmos base 2, las unidades de la entropía son bits.

$$H(Z) = - \sum_{z=1}^Z p(z_i) \log_2 p(z_i) \quad \text{bits} \tag{3.3}$$

Donde:

- $p(z_i) = Pr(Z = z_i)$ la probabilidad de la i -ésima salida de Z .
- z son los valores de la imagen.
- $H(f)$ representa la entropía.

Considerando la entropía como una medida de la incertidumbre, esta tiene un valor máximo cuando todas las intensidades ocurren con igual frecuencia, corresponde al caso equiprobable, con un histograma uniforme. Y tiene un valor mínimo cuando todos los píxeles tienen la misma intensidad (uno de los valores de los píxeles tiene probabilidad de ocurrencia 1), por lo tanto la incertidumbre es nula. Por lo anterior, la entropía es una medida de la cantidad de información de una imagen. Sin embargo, puesto que se calcula a partir del histograma, no puede tomar en consideración factores espaciales.

En la Figura 3.1 son mostrados los parámetros antes mencionados así como la imagen de la cual fueron obtenidos.

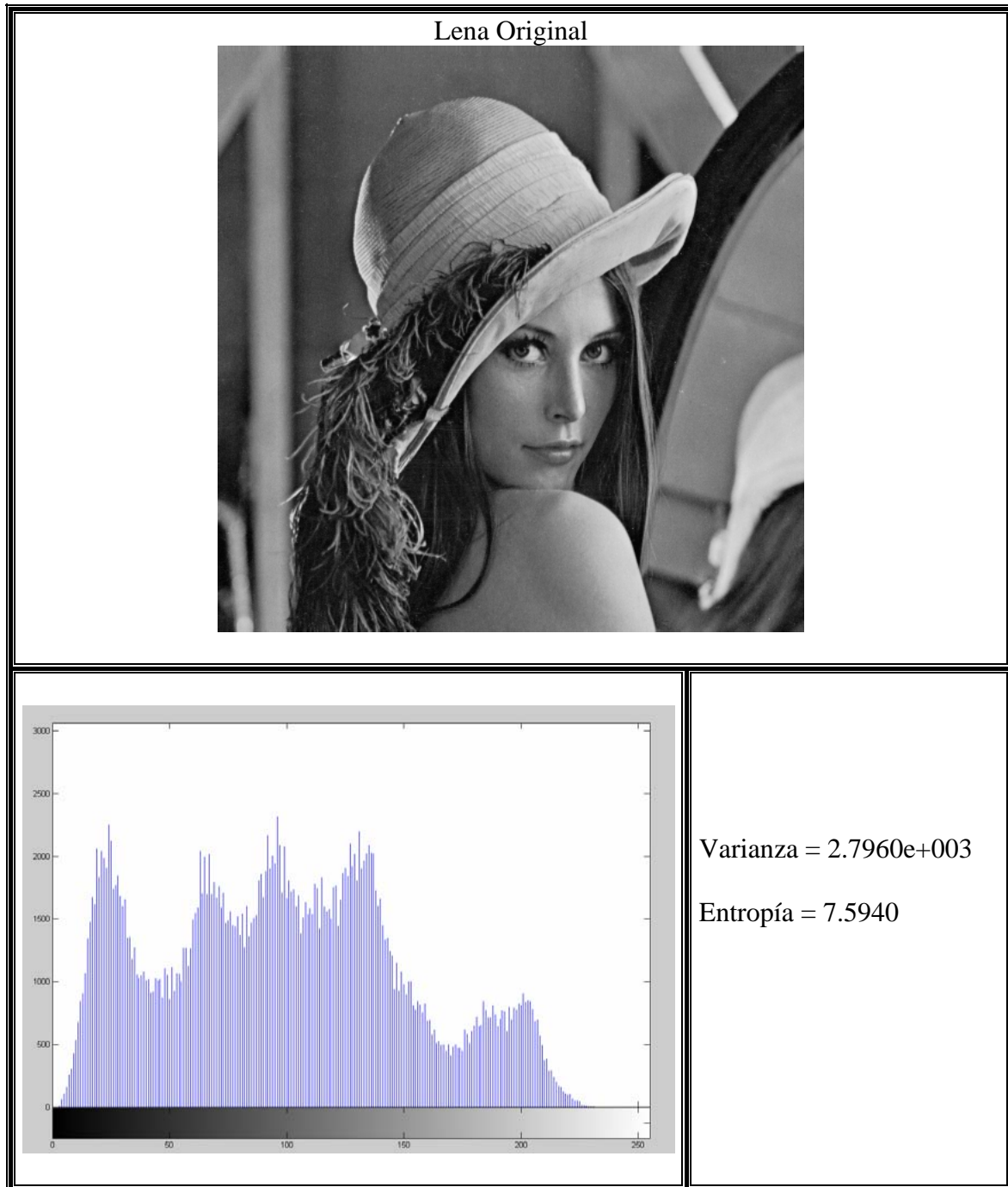


Figura 3.1. a) Imagen Original "Lena 512 x 512". b) Histograma c) Parámetros importantes.

3.2.4. Relación señal a ruido pico (PSNR)

Las medidas señal-a-ruido (SNR) son estimaciones de la calidad de una imagen reconstruida (en este caso imagen con marca de agua) comparada con una imagen original. La idea básica obtener un número que refleje la calidad de la imagen reconstruida. Las imágenes reconstruidas con valores más altos se juzgan como mejores. De hecho, las medidas tradicionales de SNR no se comparan con la opinión subjetiva humana.

Para la obtención del PSNR, primero es necesario tener el valor error cuadrático medio (MSE) de la imagen marcada, ecuación (3.4).

$$MSE = \frac{\sum [I'(i, j) - I(i, j)]^2}{N^2} \quad (3.4)$$

Donde:

$I'(i,j)$: Imagen marcada de dimensiones $i \times j$.

$I(i,j)$: Imagen original de dimensiones $i \times j$.

N : Tamaño de la imagen

La sumatoria se realiza sobre todos los píxeles.

La PSNR en decibeles (dB) es obtenida con la ecuación (3.5)

$$PSNR = 20 \log_{10} \left(\frac{255}{\sqrt{MSE}} \right) \quad (3.5)$$

3.3. Resultados enfocados a capas marcadas

De acuerdo con el capítulo 2, los parámetros más importantes para la detección de la marca de agua son: R' y T , ecuaciones 2.34 y 2.36, respectivamente. En las tablas siguientes son mostrados dichos valores obtenidos de las diferentes pruebas realizadas, así como los parámetros mencionados en el punto anterior. Para estas pruebas se insertó la marca de agua en una, cuatro y doce capas. En el caso de una capa se utilizó la capa $C_{1,1}$, referida en la figura 2.26, ya que solo se utilizó como punto de comparación entre ambos métodos. Para la inserción en cuatro capas se utilizaron las capas del segundo nivel de descomposición, nivel que tiene el mejor rendimiento en el marcado según los resultados obtenidos. Y en el caso del marcado de todas las capas fue con el propósito de comparar los dos métodos con el sistema operando en su máxima capacidad.

Tabla 3.1. Parámetros obtenidos en la imagen marcada al insertar la marca de agua en una capa.

Parámetros	Algoritmo ref [1] (Primera Capa)	Algoritmo Propuesto (Primera Capa)
R'	2.6114	2.4915
T	0.1523	0.1516
PSNR	40.6353	41.9839
Varianza	$2.8020 \cdot 10^3$	$2.8004 \cdot 10^3$
Entropía	7.6088	7.6055

Tabla 3.2. Parámetros obtenidos en la imagen marcada al insertar la marca de agua en cuatro capas.

Parámetros	Algoritmo ref [1] (Cuatro primeras capas)	Algoritmo Propuesto (Cuatro primeras capas)
R'	0.5508	0.3609
T	0.0771	0.0766
PSNR	35.0566	41.3802
Varianza	$2.8159 * 10^3$	$2.7989 * 10^3$
Entropía	7.6271	7.6048

Tabla 3.3. Parámetros obtenidos en la imagen marcada al insertar la marca de agua en todas las capas (12 capas).

Parámetros	Algoritmo ref [1] (Todas las capas)	Algoritmo Propuesto (Todas las capas)
R'	3.3730	0.9415
T	0.0653	0.0503
PSNR	25.5141	35.3167
Varianza	$2.9708 * 10^3$	$2.8132 * 10^3$
Entropía	7.6833	7.6255

En la figura 3.2 son mostradas las imágenes marcadas en cuatro capas por ambos métodos, debido a que la razón señal pico a ruido (PSNR) es mayor de 34db en ambos casos, no es posible percibir la diferencia entre ambas, pero numéricamente existe un diferencial de 6.3236db. Para el caso del marcado en las doce capas, figuras 3.3 y 3.4, se tiene una diferencia de 9.8026db entre un método y otro. Dicha diferencia es fácil de apreciar entre ambas figuras, así como en los histogramas correspondientes: en comparación con el mostrado en la imagen original de la figura 3.1 b), existe mayor diferencia en el relativo a la figura 3.3 b), que en el de la figura 3.4 b).



Figura 3.2. a) Imagen marcada en las cuatro primeras capas con el algoritmo ref [1]. b) Imagen marcada en las cuatro primeras capas con el algoritmo propuesto.

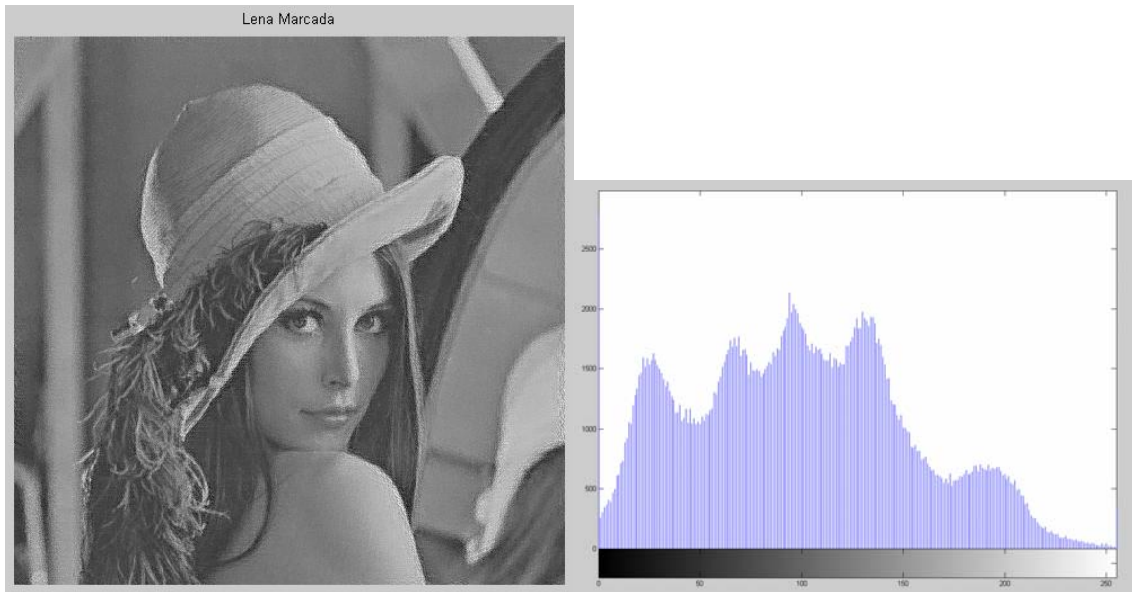


Figura 3.3. a) Imagen marcada en todas las capas con el algoritmo de ref [1].
b) Histograma de la imagen.

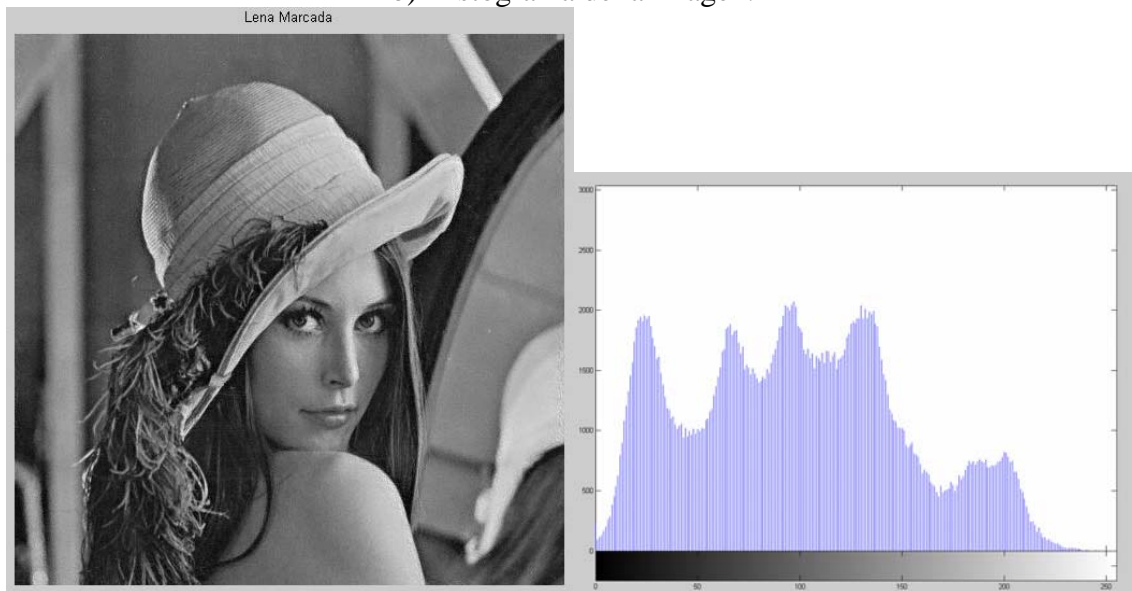
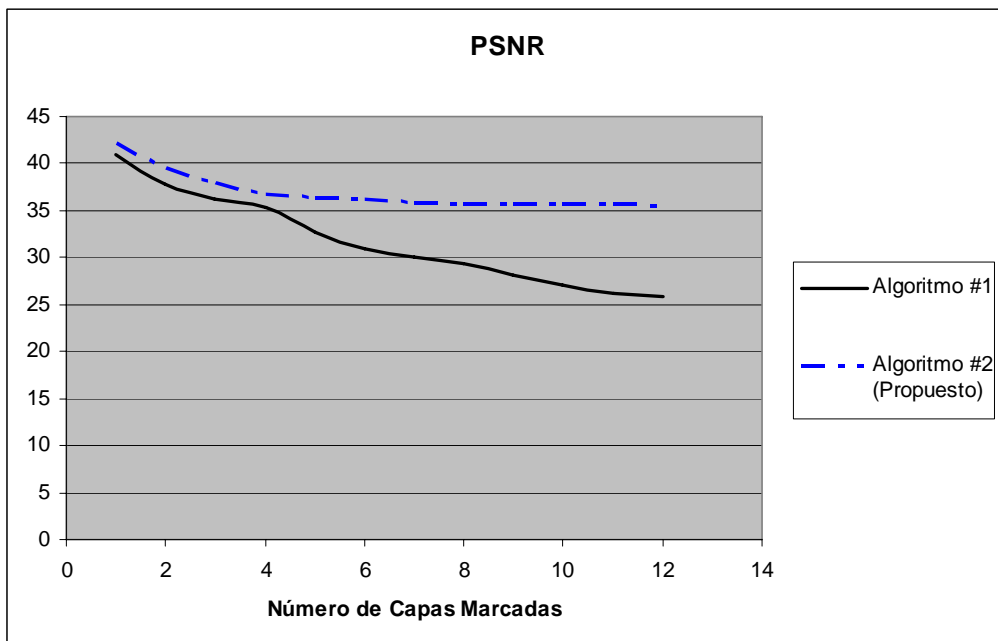


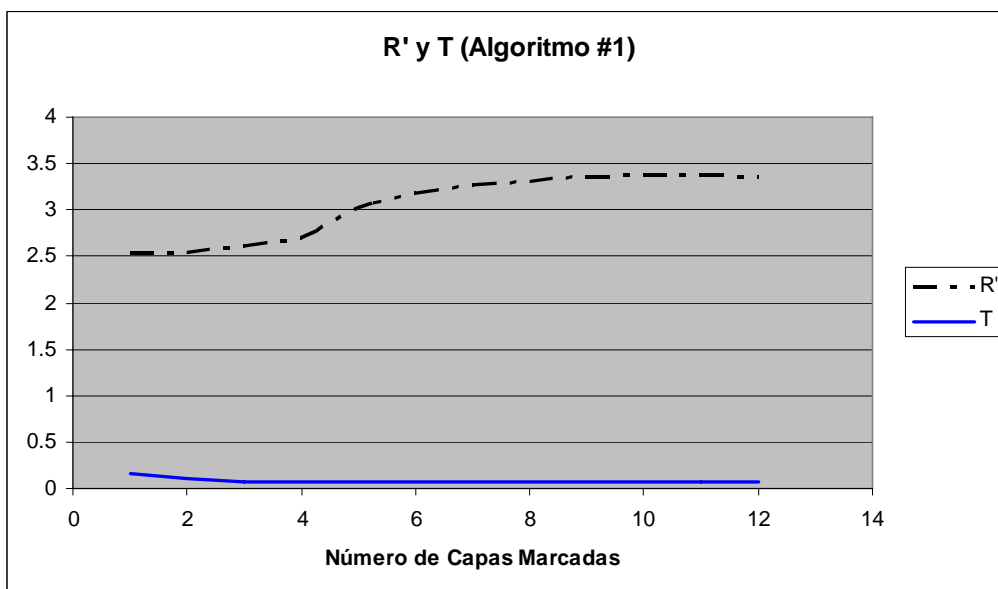
Figura 3.4. a) Imagen Marcada en todas las capas con el algoritmo propuesto.
b) Histograma de la imagen.

En las graficas de razón señal pico a ruido (PSNR) vs. número de capas marcadas, presentadas en las gráficas 3.1, 3.2 y 3.3. Están representadas las características de las imágenes después de la aplicación de la marca de agua por medio de ambos algoritmos.

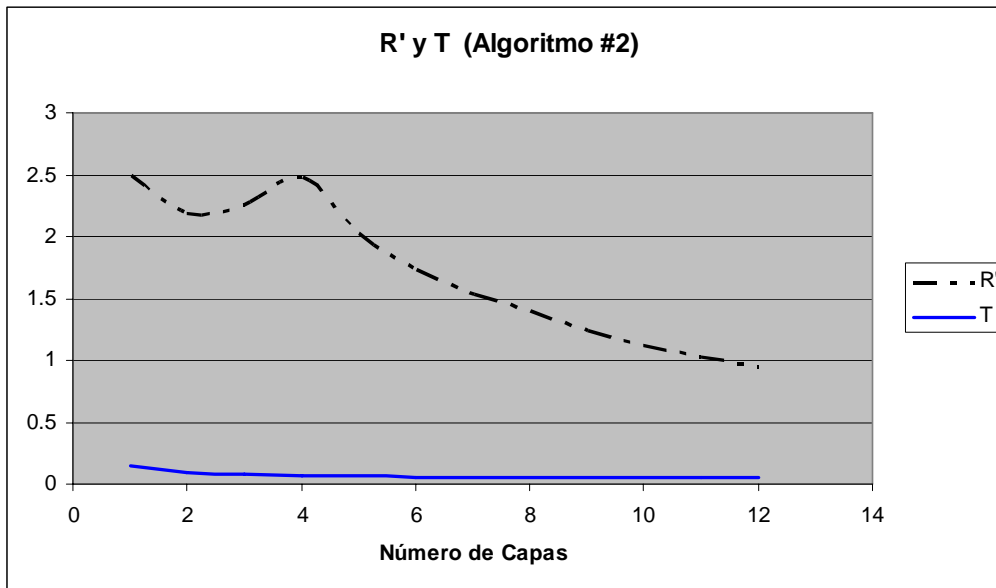


Gráfica 3.1. Grafica de los valores PSNR de ambos algoritmos.

En la gráfica 3.1 se puede observar que la PSNR disminuye conforme el número de capas marcadas es incrementado, sin embargo, el resultado obtenido con el algoritmo propuesto siempre es mejor que con el algoritmo de ref [1]. En la parte izquierda de la gráfica, ambos siguen el mismo patrón de comportamiento (de 1 capa a 4 capas). De la parte media de la grafica, hacia el final, la PSNR sigue un comportamiento diferente, para el algoritmo propuesto dichos valores son bastante consistentes, esto es, sin una gran variación, y manteniéndose por encima del algoritmo ref [1].



Gráfica 3.2. Grafica de los valores R' y T del algoritmo ref [1].



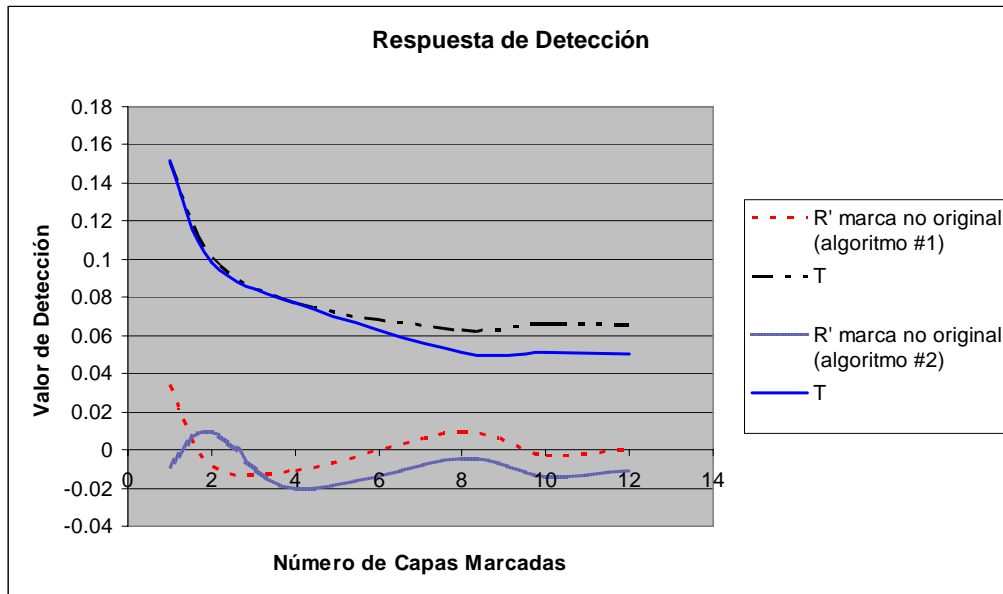
Gráfica 3.3. Grafica de los valores R' y T del algoritmo propuesto.

En las gráficas 3.2 y 3.3 son mostrados los valores de los parámetros R' y T obtenidos con ambos algoritmos. En la gráfica 3.2 se puede observar que la separación entre ambos parámetros se incrementa conforme el número de capas en las cuales se realiza la inserción de la marca de agua es incrementado. Pero en el algoritmo propuesto el intervalo entre dichos parámetros es cada vez menor, a partir del número de capas marcadas cuatro.

Esto se puede interpretar como que el algoritmo propuesto es menos robusto que el algoritmo de referencia. Pero en términos de funcionamiento práctico es positivo, ya que en todo el intervalo de evaluación, siempre el valor R' es mayor que el valor T, lo cual significa que siempre es posible detectar la marca de agua correctamente. Las pruebas de robustez tratadas en el siguiente párrafo demostrarán dicho funcionamiento.

3.4. Resultados respecto a marcas de agua no originales

Se realizaron diferentes pruebas insertando marcas de agua no originales a la imagen. El resultado es muy positivo, dado que las marcas no originales nunca fueron aceptadas como marcas auténticas.



Gráfica 3.4. Detección de marcas de agua no originales.

Como se ilustra en la gráfica 3.4 con ambos métodos siempre que se usa una marca de agua no original, el valor obtenido de R' está por debajo del valor del umbral T, indicando que no se trata de la marca auténtica.

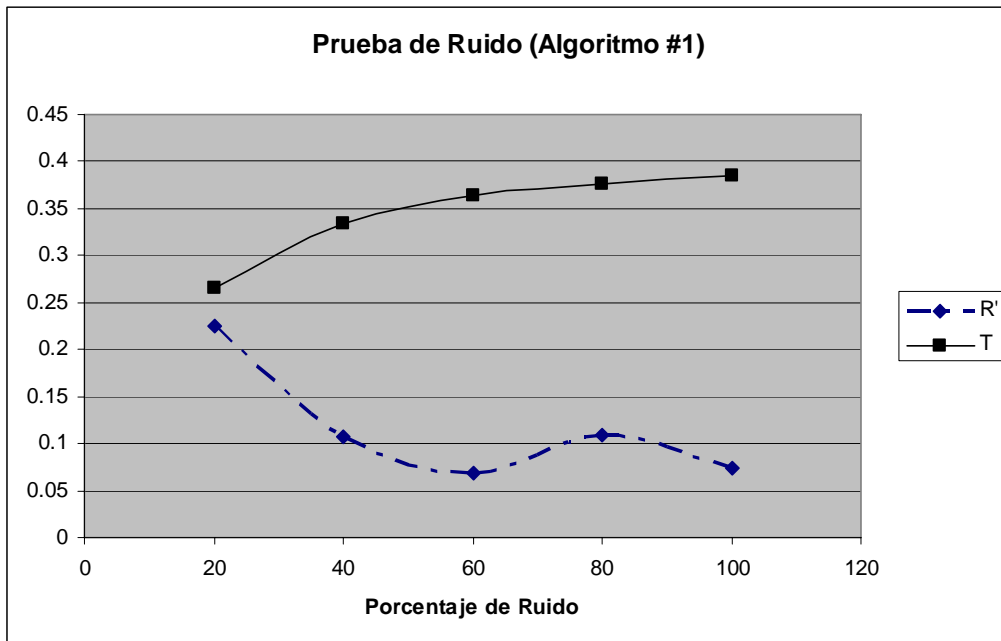
Para la obtención de la gráfica 3.4 fueron realizadas múltiples pruebas, con diferente número de capas marcadas, y en diferentes capas, así como con diferentes marcas de agua no originales, esto para ambos algoritmos de marcado.

3.5. Resultados de las pruebas de robustez

Después de haber realizado la comparación entre los diferentes parámetros obtenidos con los dos algoritmos, las pruebas desarrolladas en este párrafo se refieren a algunos ataques contra la marca de agua, estos fueron producidos con el programa StirMark Benchmark 4 [17], y con ello se pudo determinar la robustez de los algoritmos contra dichos ataques sobre la marca de agua.

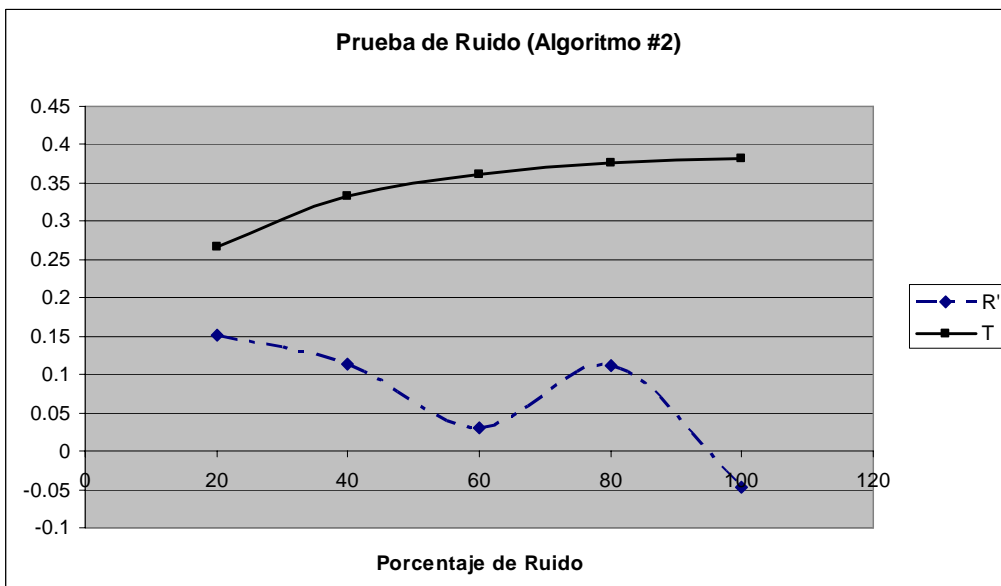
3.5.1. Prueba de Ruido

Esta prueba consiste en insertar ruido aditivo a la imagen marcada con cuatro capas y observar el funcionamiento correspondiente al realizar la detección de la marca y, con ello determinar el nivel de ruido que es capaz de soportar el algoritmo.



Gráfica 3.5. Resultado a la prueba de ruido.

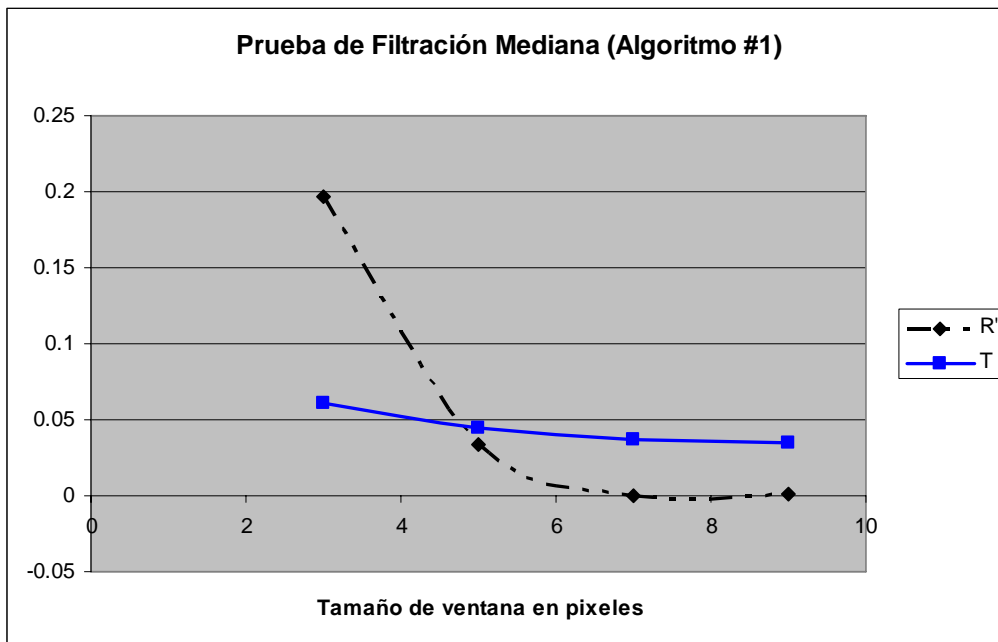
En la gráfica 3.6 se puede observar que al introducir ruido aditivo en la imagen marcada, cuando se realiza la detección de la marca, el valor de T es mayor que el valor R', por lo tanto el algoritmo no tiene una robustez suficiente para tener un buen desempeño con respecto a este ataque. También, en la gráfica 3.5 se pueden obtener los resultados de la misma prueba con el método de ref [1]. El resultado con este otro método es también fallido, aunque los intervalos entre los valores de los parámetros son menores. Sin embargo, la conclusión global acerca de la existencia de marca de agua, es fallida en ambos casos, ya que el valor de R' es menor que el valor del umbral T.



Gráfica 3.6. Resultado a la prueba de ruido.

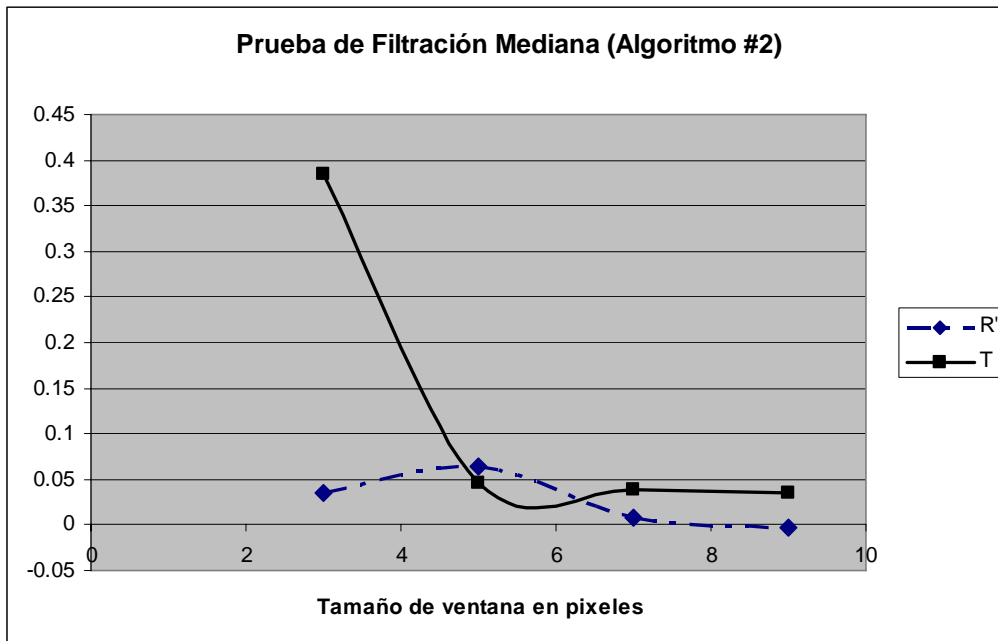
3.5.2. Prueba de filtrado de mediana

Con este filtro se obtiene la mediana de los valores de los píxeles dentro de una ventana y se sustituye el valor del píxel central por el resultado. En seguida se discute el efecto de este tipo de filtrado sobre la detección de la marca de agua, en una imagen marcada en cuatro capas.



Gráfica 3.7. Resultado de la prueba de filtrado de mediana para el algoritmo. ref [1].

En las gráficas 3.7 y 3.8 son mostrados los valores de los parámetros - R' y T- al aplicar un filtrado de mediana con diferentes tamaños de ventanas a las imágenes marcadas con ambos métodos, respectivamente. En ambas pruebas, se puede observar que sólo para una parte de la gráfica el resultado es favorable. Para el algoritmo propuesto, el comportamiento no es asintótico, solamente los valores de tamaño de ventana cinco y seis satisfacen la detección de la marca de agua. Con el algoritmo ref [1], el comportamiento es asintótico y los valores de tamaño de ventana que satisfacen la detección correcta de la marca van de tres a cuatro y medio aproximadamente.



Gráfica 3.8. Resultado de la prueba de filtrado de mediana para el algoritmo de propuesto.

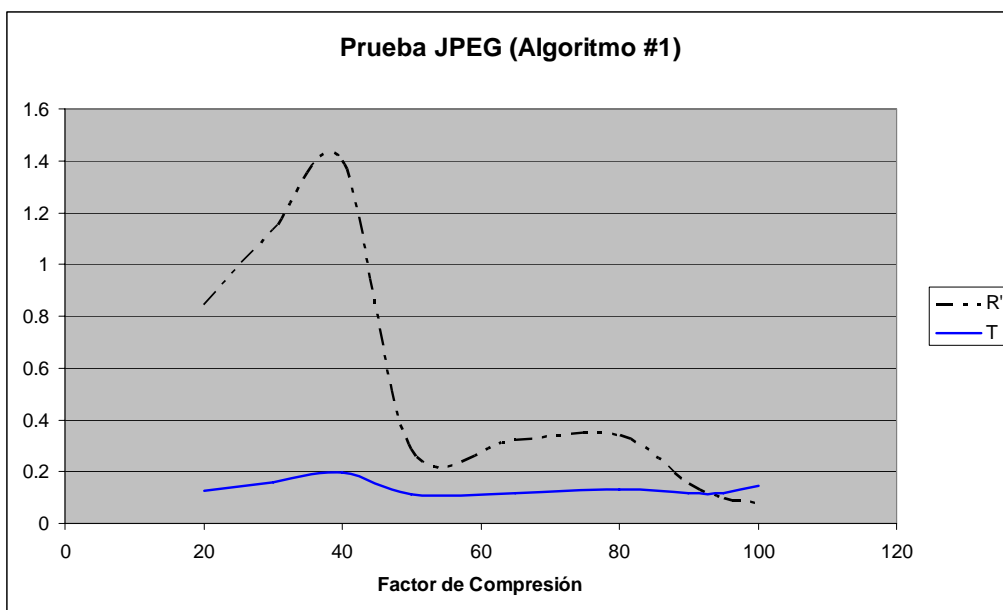
Estos resultados nos indican que este ataque, afecta de manera diferente a cada uno de los algoritmos, pero en general se puede concluir que un ataque de este tipo afectará la detección correcta de la marca de agua con ambos métodos.

3.5.3. Prueba de resistencia a la compresión JPEG

La compresión de imágenes digitales con el estándar JPEG es actualmente una de las más utilizadas, por tal motivo, los sistemas de marca de agua deben ser resistentes a los niveles de compresión mas altos posibles.

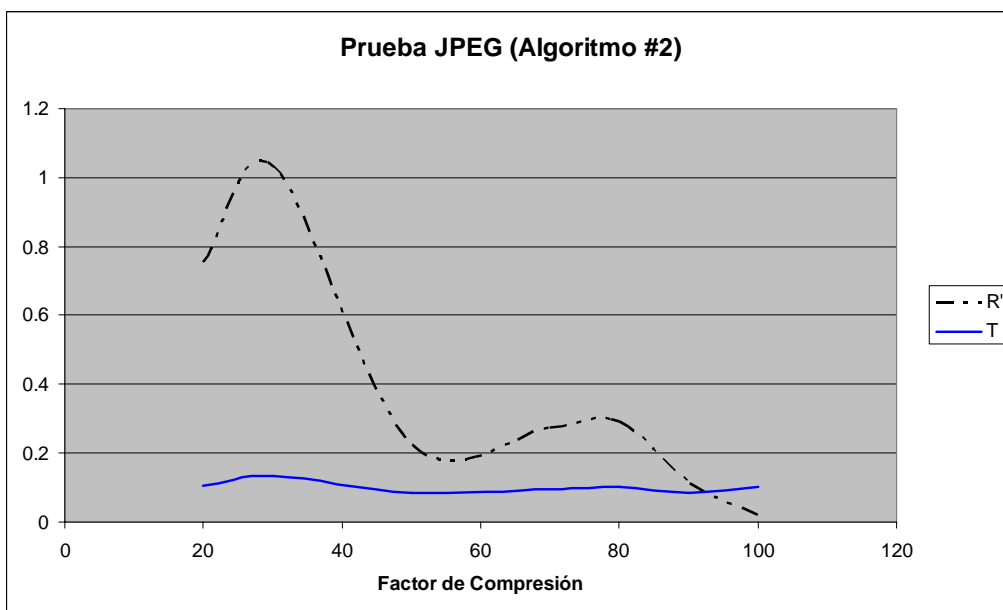
Después de realizar pruebas de compresión JPEG, ambos métodos mostraron una buena resistencia a dicho ataque. Solo cuando el nivel de compresión superó un factor de compresión de 90, fue cuando en ambos algoritmos ocurre una falla en la detección de la marca de agua, dicha marca de agua insertada en cuatro capas.

En las gráficas 3.9 y 3.10 son mostrados los comportamientos obtenidos de los algoritmos en sus parámetros de detección.



Gráfica 3.9. Resultado de la resistencia a la compresión JPEG para el algoritmo. ref [1]

Como puede verse, en ambas graficas el comportamiento en la detección es el mismo, con la diferencia, de que los intervalos de los valores obtenidos del algoritmo propuesto son menores.



Gráfica 3.10. Resultado de la resistencia a la compresión JPEG para el algoritmo propuesto.

Como se puede observar, en ambos algoritmos la detección es fallida a partir de un factor de compresión de poco más de 90%.

3.5.4. Pruebas geométricas

Corrimiento horizontal: muchas imágenes pueden ser recorridas sin perder ningún valor. Sin embargo, muy pocos sistemas de marcado sobreviven a él.

Rotación: la rotación pequeña del ángulo, a menudo conjuntamente con el recorte (cropping), no cambia el valor comercial de la imagen pero puede generalmente hacer la marca de agua indetectable.

Escalamiento: puede ser dividido en dos grupos, escalamiento uniforme y no uniforme. El primero se refiere al igual cambio de dimensiones en las direcciones vertical y horizontal. El segundo usa diferentes dimensiones en ambas direcciones.

Recorte: la imagen puede ser usada solo en alguna de sus partes, por tal motivo la imagen es recortada en la parte que podría ser de utilidad.

Distorsiones geométricas y JPEG: Además de las transformaciones geométricas, éstas pueden ser utilizadas en conjunción con la compresión JPEG.

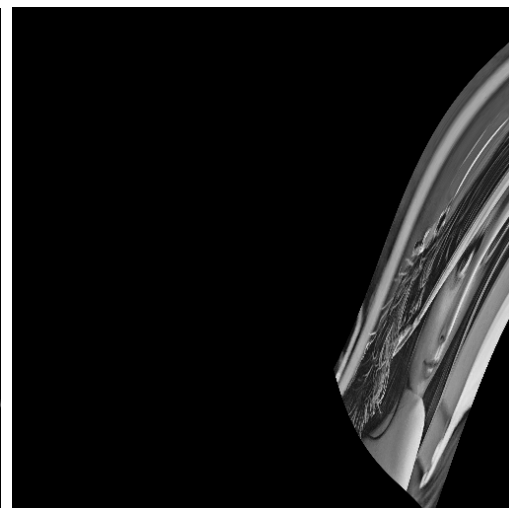




Figura 3.5 Ejemplos de ataques geométricos.

En cuanto a lo que se refiere a este tipo de ataques, se realizaron pruebas para los dos algoritmos con el ataque de rotación, y prácticamente en todas las pruebas no se detectó correctamente la marca de agua, sólo en algunos casos en los cuales la rotación de la imagen fue de menos de 0.25 grados, se pudo detectar en forma correcta.

3.6. Resultados de pruebas con otras imágenes

Después de haber realizado las diferentes pruebas del algoritmo #2 con respecto al #1, con la imagen “Lena”, también se realizaron todas las pruebas con otras imágenes, para así poder corroborar los resultados obtenidos. Las graficas mostradas a continuación refieren al algoritmo #2.

En las gráficas 3.11 - 3.13 se muestran los resultados obtenidos para los valores R' y T (valores de detección de 1 marca de agua), y PSNR para las imágenes “Barbara” y “Fishing Boat”. Estos resultados son muy similares a los obtenidos en los puntos anteriores con la imagen “Lena”. Por lo tanto, los resultados previamente mostrados se pueden estandarizar para cualquier imagen de tipo televisión digital a la que se le aplique una marca de agua con los métodos descritos.

También las figuras 3.7 - 3.10 muestran los resultados al aplicar la marca de agua en la opción de marcado con 12 capas, las cuales tienen los mismos comportamientos en cuanto a nivel de PSNR que la imagen Lena.

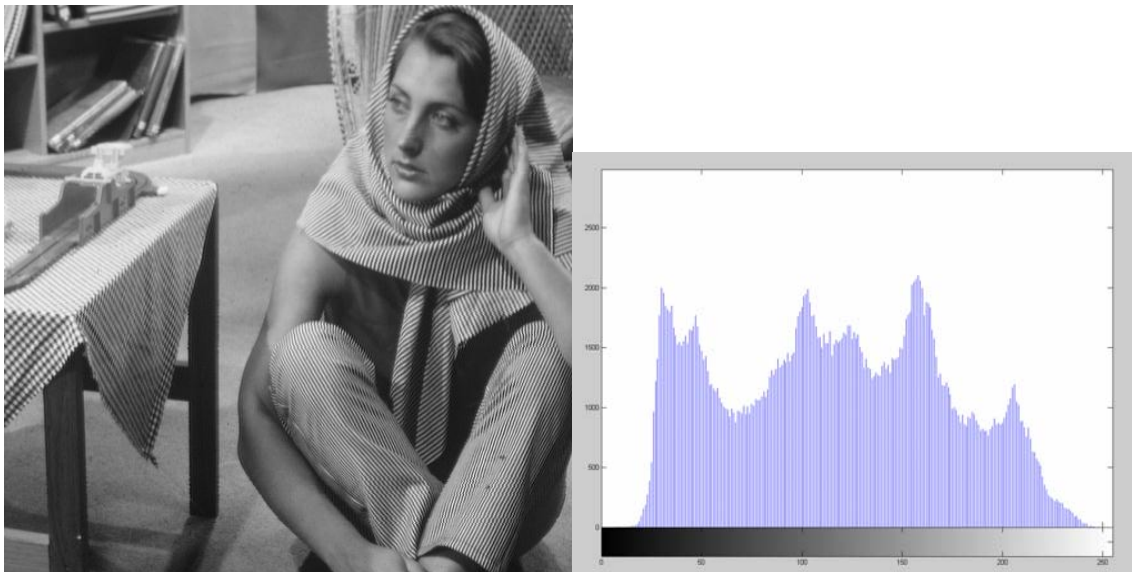
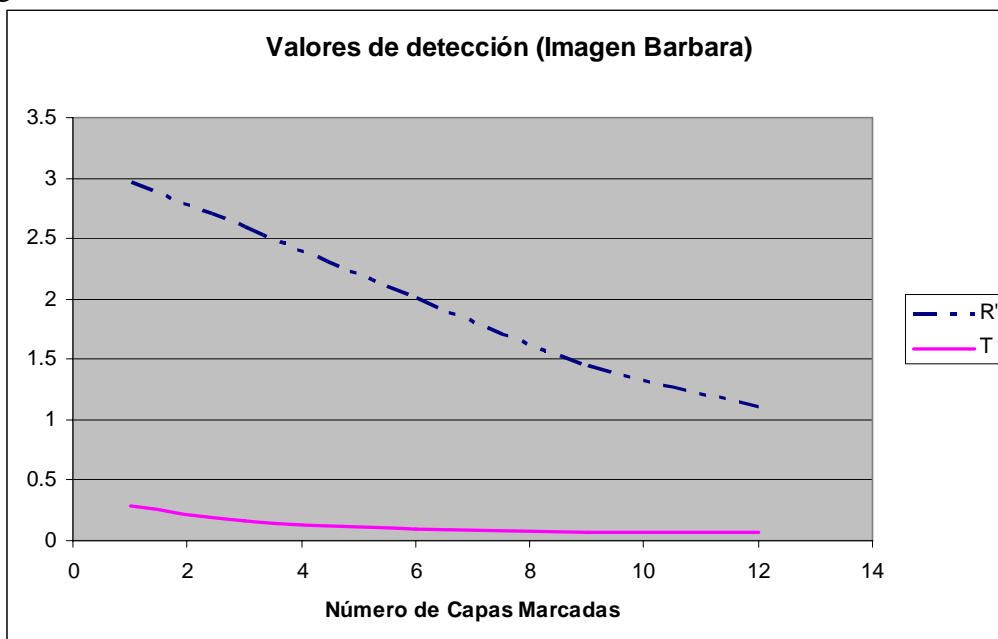


Figura 3.6. a) Imagen Original “Bárbara” b) Histograma de “Bárbara”.

La figura 3.6 muestra la imagen Bárbara y su histograma, los cuales pueden ser comparados con los histogramas mostrados en las figuras 3.7 y 3.8, que refieren a la imagen marcada con los dos diferentes métodos mostrados, algoritmos de marcado #2 y #1 respectivamente.

Con las figuras 3.7 y 3.8, se puede corroborar que con respecto a las figuras de Lena 3.3 y 3.4 los histogramas sufren los mismos efectos de pérdida de detalles y las imágenes sufren cierto deterioro. Este efecto es mucho más notable con el algoritmo #1, que con el algoritmo #2.



Gráfica 3.11. Valores de detección de la imagen “Bárbara” (algoritmo #2).

Tanto en las gráficas 3.11 y 3.12 se puede ver que el valor R' casi tiene el mismo comportamiento de decrecimiento al igual que en la gráfica 3.3, la cual muestra el comportamiento en el primer experimento (imagen Lena). Sin embargo, en estos casos no existe una irregularidad en la curva, si no que el decrecimiento es monotonóico desde el principio de la gráfica, cosa que no ocurrió en el primer caso.

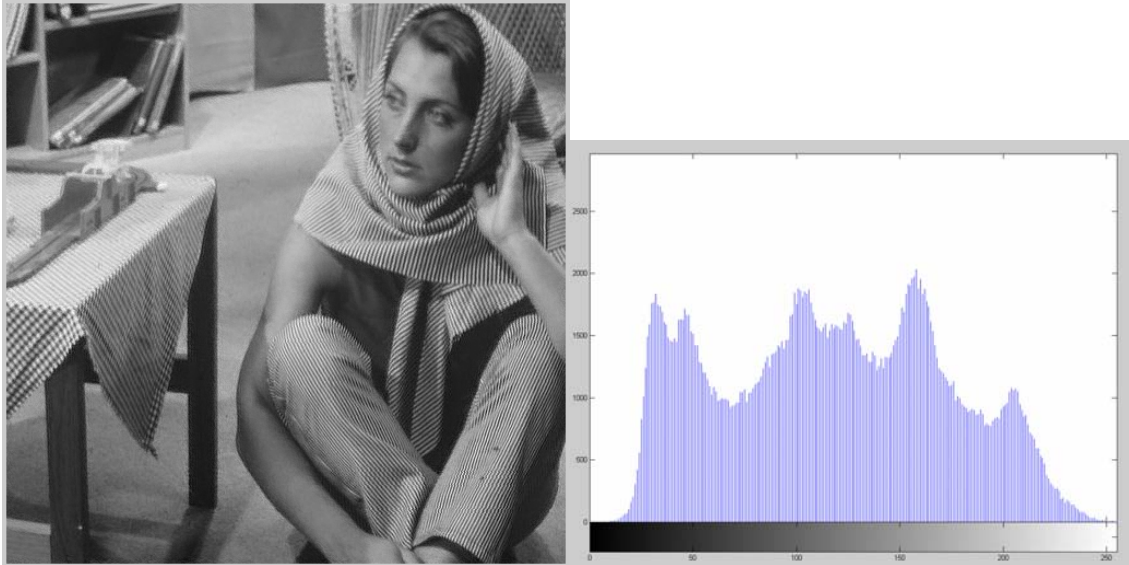


Figura 3.7 a) Imagen “Bárbara” marcada en 12 capas con algoritmo #2 b) Histograma de a).

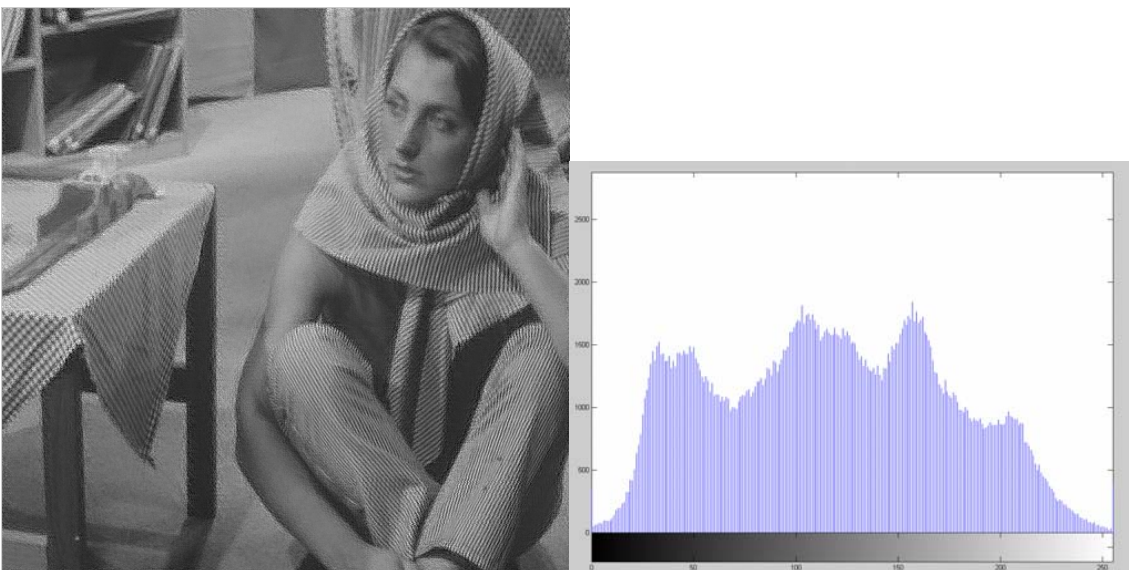
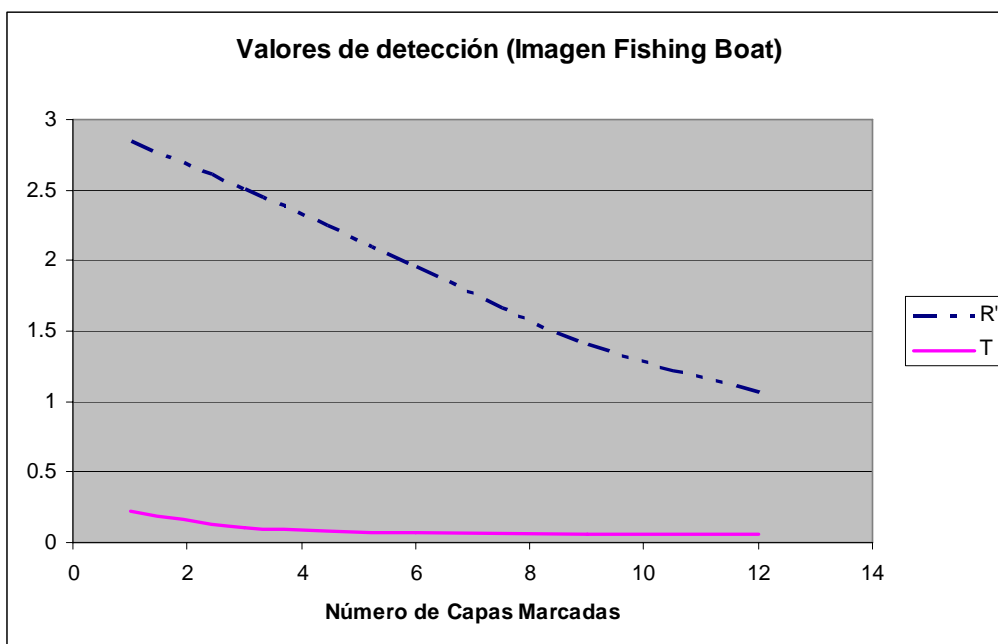
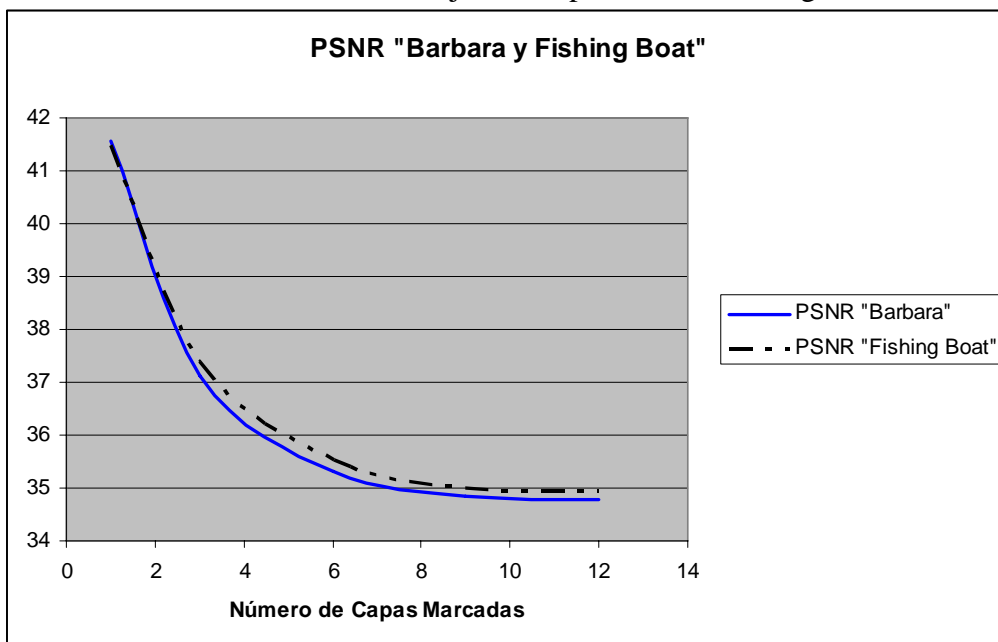


Figura 3.8 a) Imagen “Bárbara” marcada en 12 capas con algoritmo #1 b) Histograma de a).



Gráfica 3.12. Valores de detección de la imagen “Fishing boat” (algoritmo #2).

En lo referente a la gráfica 3.13, se puede ver que el PSNR decrece sólo en aproximadamente 7db, lo cual se asemeja al comportamiento de la gráfica 3.1.



Gráfica 3.13. PSNR obtenidos al analizar las imágenes “Bárbara “ y “Fishing boat” (algoritmo #2).

La figura 3.9 muestra la imagen “Fishing boat” y su histograma, al compararlo con los histogramas de la figura 3.10 se corrobora como con el algoritmo #1 la curva se suaviza mas, respecto al obtenido con el algoritmo #2.

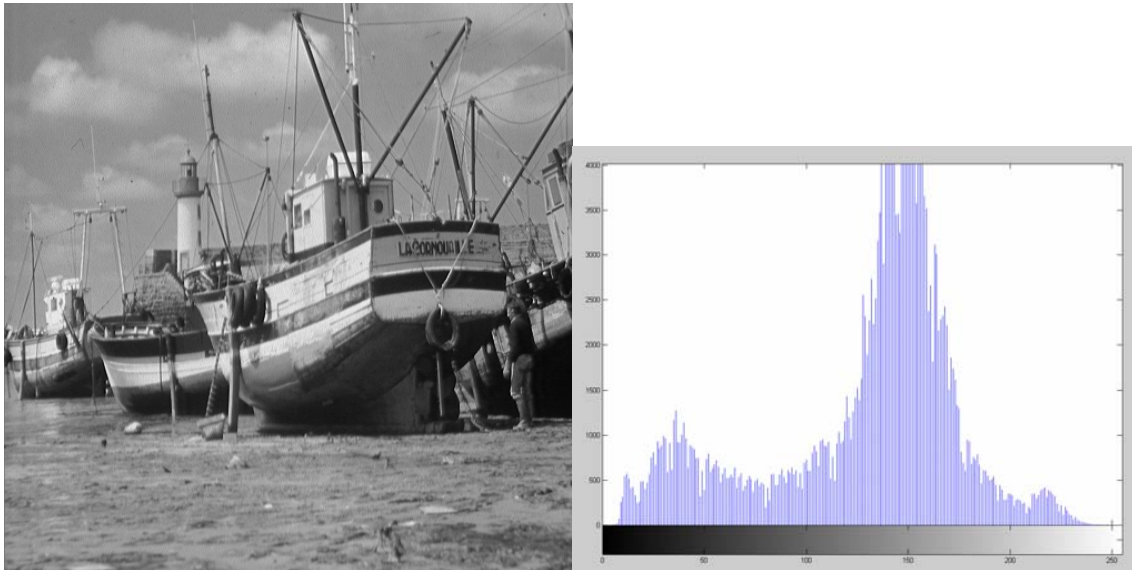


Figura 3.9 a) Imagen “Fishing boat” b) Histograma de “Fishing boat”.

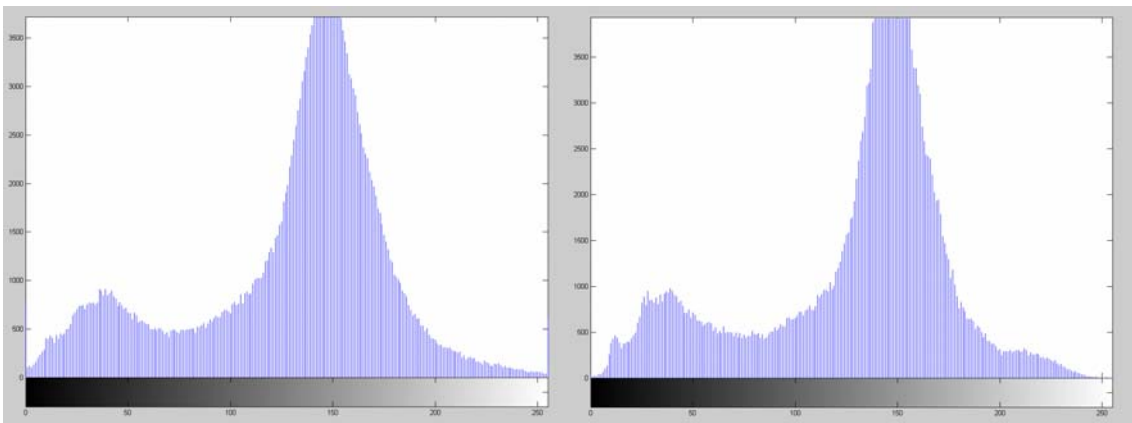


Figura 3.10 a) Histograma de “Fishing boat” con marcado.en 12 capas con algoritmo #1
b) Histograma de “Fishing boat” ” con marcado.en 12 capas con algoritmo #2.

La figura 3.11 muestra las figuras “Fishing boat” y “Lena” marcadas con los algoritmos #1 y #2 respectivamente, con un factor de compresión 100 en las cuales no se pudo detectar la marca de agua.



Figura 3.11 a) Imagen “Fishing boat” marcada en 1 capa con algoritmo #1 y con factor de compresión JPEG 100 b) Imagen “Lena” marcada en 1 capa con algoritmo #1 y con factor de compresión JPEG 100.

CONCLUSIONES

Al presentar las principales técnicas de marcado de agua, escogidas en esta tesis, se ha considerado que las características de la transformada *contourlet* son las más eficientes en cuestión de agrupar mejor la estructura geométrica de la información visual y con ello situarse como la mejor transformada para lograr una inserción de una marca de agua invisible en una imagen digital.

Se pudo examinar detalladamente el comportamiento del algoritmo #1 (ref [1]), el cual por medio de la transformada *contourlet* redundante y de las características del sistema visual humano (HVS) realiza un enmascaramiento que se apoya en las principales características de la imagen, el cual permite introducir la marca de agua en el dominio frecuencial de dicha transformada y, permite así, tener una mejor invisibilidad de la marca de agua. Al aplicar el proceso de detección de la marca de agua se pudo corroborar que el método es bastante eficaz, primeramente siempre detectó la marca genuina y nunca detecto marcas falsas. Por otra parte, al aplicar ciertos ataques, por lo menos entre los más importantes como lo es la compresión JPEG, también tubo excelente eficiencia. (Nota: antes se debe explicar qué fue lo que motivó a hacer el siguiente tipo de variación)

Al utilizar la transformada *contourlet* redundante se generan 12 capas *contourlet*, la marca de agua se puede insertar en cada una de ellas, o bien, dependiendo de la aplicación en el número que se quiera, pudiendo así utilizar hasta un total de 12 capas y en cualquier orden. Se realizaron algunas pruebas, haciendo variar el número de capas de la transformada *contourlet* que intervienen en el marcado. Debido a que cada banda marcada incrementa la información insertada en la imagen digital, se pudo constatar que al utilizar las 12 capas *contourlet*, el valor de la razón señal pico a ruido PSNR disminuye significativamente, en aproximadamente 10 dB, lo cual es altamente visible en la imagen marcada.

Al considerar que en varias aplicaciones uno de los objetivos más importantes del marcado de agua es la invisibilidad de la marca, después de analizar el algoritmo #1 y corroborar la disminución del valor PSNR al utilizar varias capas *contourlet* en el marcado. Se propuso como parte de los objetivos de este trabajo realizar algunas modificaciones a dicho algoritmo con el propósito de mejorar el valor PSNR, que decrece al usar varias capas *contourlet* marcadas. Dichas modificaciones consistieron en la agrupación de la información de la marca de agua por medio de la propia transformada *contourlet* redundante, para así conservar los detalles de la marca de agua en unos cuantos coeficientes, y tener el mismo tamaño de marca que el de la imagen, para así facilitar la manipulación de la marca con respecto a las capas de la imagen digital a marcar.

Al aplicar la transformada *contourlet* a la marca de agua, se crearon 12 marcas de agua diferentes que corresponden a las 12 capas *contourlet*, y aunque dependan de una sola marca en el dominio espacial, son diferentes en el dominio *contourlet*. En otra aplicación se podrían insertar diferentes capas provenientes de diferentes marcas de agua, las cuales al igual que en el caso del algoritmo #1 serían insertadas en el dominio *contourlet*.

Al evaluar el sistema propuesto en esta tesis (algoritmo #2), el cual como ya se mencionó es una modificación del algoritmo #1 y por lo tanto se apoya en los mismos conceptos básicos de funcionamiento, se pudo corroborar que la hipótesis planteada fue cierta, ya que sí se mejoró el nivel PSNR, aún marcando la totalidad de capas.

La evaluación del algoritmo fue determinada principalmente por los resultados y pruebas utilizadas en la referencia [1] y de esta manera poder comparar los datos entre ambos algoritmos.

Para hacer un estudio sistemático, primeramente se insertó la marca de agua en 1 banda, después se incrementó progresivamente el número de capas hasta llegar al total de 12, obteniendo en cada caso los valores de PSNR, y de los parámetros R' y T usados en la detección de la marca. Con ellos se realizó una comparación de los resultados obtenidos, respecto a los del algoritmo #1. Después se realizaron las pruebas de robustez para ver si a pesar de que la marca de agua perdió intensidad, los valores de detección (R' y T) eran todavía capaces de proporcionar el resultado correcto en cuanto a la presencia de la marca de agua, según fuese el caso.

Mediante dicha evaluación, para el caso en el que se insertó la marca de agua en sólo una banda, la diferencia entre el algoritmo #1 y #2 fue mínima. Sin embargo, al incrementar el número de capas *contourlet* utilizadas en el marcado con nuestro algoritmo, el valor del nivel de PSNR no decrece en forma drástica, como en el resultado obtenido con el algoritmo #1, lo cual fue un cambio significativo. Los resultados obtenidos con nuestro algoritmo son buenos a pesar de que con la modificación realizada disminuye la información global de la marca. Esto por otro lado, causa que los valores de detección disminuyan el intervalo de eficiencia. Sin embargo, tanto en las pruebas de funcionamiento del algoritmo, como en las pruebas de robustez, la detección siempre fue positiva, en los mismos casos que lo fue con el algoritmo #1.

Considerando más a fondo las pruebas de robustez se pudo ver que ambos algoritmos son solo eficientes en circunstancias donde existe una compresión JPEG de la imagen, ya que cuando existe ruido aditivo, no es posible detectar la marca de agua, así como cuando existe una aplicación de filtro mediana donde solo en ciertos pequeños intervalos es posible detectar la marca de agua.

Trabajo a futuro

En base al trabajo desarrollado, se puede considerar como trabajo a futuro, el analizar e investigar la forma de generar una marca de agua detectable a pesar de ataques geométricos, como por ejemplo: ataque de rotación, para el cual consideramos se debe adaptar algún método que realice transformaciones geométricas por medio de matrices de rotación. Ataques de recorte, o escalados, donde se deberán de tener algoritmos de sincronización. Y finalmente, ataques de ruido aditivo.

Estas ideas tienen sustento en el hecho de que en el caso de los dos algoritmos tratados en esta tesis, para el caso de ataques geométricos no se obtuvieron buenos resultados, es decir, no se detectó la marca de agua. Y al aplicar ataques de ruido aditivo Gaussiano, sólo con una baja varianza del ruido, fue posible detectar la marca de agua.

Por lo tanto para los dos algoritmos estudiados, algoritmo #1 (ref [1]) y algoritmo #2 propuesto, se deberán de buscar mejores resultados para las pruebas de robustez anteriormente mencionadas.

REFERENCIAS

- [1] N. Baaziz, "Adaptive watermarking schemes based on a redundant contourlet transform," *Proc. ICIP*, Genoa, Italy, September 2005.
- [2] Minh N. Do and Martin Vetterli, "The Contourlet Transform: An Efficient Directional Multiresolution Image Representation." *Proc. of IEEE International Conference on Image Processing (ICIP)*, Rochester, September 2002.
- [3] F. García-Ugalde, "Análisis de imágenes por transformada wavelet y su extensión a la segmentación", seminario de "Informática Fundamental" del LaBRI, U. Bordeaux 1, Francia, 19 de Enero 2006.
- [4] C. Valens, Embedded Zerotree Wavelet Encoding. http://tonelli.sns.it/pub/wavelets/C_Valens/ezwe.pdf, 1999.
- [5] E. J. Candès and D. L. Donoho, "Curvelets - a suprisingly effective nonadaptive representation for objects with edges," in *Curve and Surface Fzting*, A. Cohen, C. Rabut, and L. L. Schumaker, Eds., Saint-Malo, 1999, Vanderbilt University Press.
- [6] E. J. Candès and D. L. Donoho. Ridgelets: a key to higher-dimensional intermittency? *Phil. Trans. R. Soc. Lond. A.*, pages 2495-2509, 1999.
- [7] J.L. Starck, E.J. Candès, and D.L. Donoho. The curvelet transform for image denoising. *IEEE Trans. Image Proc.*, 11:670-684, Jun. 2002.
- [8] P. J. Burt and E. H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Trans. Commun.*, 31(4):532-540, April 1983.
- [9] M. N. Do and M. Vetterli, "Framing pyramids," *IEEE Trans. Signal Proc.*, pp. 2329-2342, Sep. 2003.
- [10] S. Park. New Directional Filter Banks and Their Applications in Image Processing. PhD thesis, Georgia Institute of Technology, 1999.
- [11] S. Park, M. J. T. Smith, and R. M. Mersereau. A new directional filterbank for image analysis and classification. In *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc.*, pages 1417-1420, 1999.
- [12] R. R. Coifman, Y. Meyer, and M. V. Wickerhauser. Wavelet analysis and signal processing. In M. B. Ruskai et al, editor, *Wavelets and their Applications*, pages 153-178. Jones and Barlett, Boston, 1992.
- [13] S. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 2nd edition, 1999.
- [14] S. Mallat. Multiresolution approximations and wavelet orthonormal bases of $L_2(\mathbb{R})$. *Trans. Amer. Math. Soc.*, 315:69-87, September 1989.
- [15] Y. Meyer. *Wavelets: Algorithms and Applications*. SIAM, 1992.

[16] M. Barni, F. Bartolini, and A. Piva, "Improved wavelet based watermarking through pixel-wise masking," *IEEE Trans. Image Proc.*, vol. 10, no. 5, pp. 783-791, May 2001.

[17] F. A. P. Petitcolas, M. Steinbach, J. Dittmann, C. Fontaine, F. Raynal, and N. Fatès, "A public automated Web-based evaluation service for watermarking schemes: Stirmark benchmark," *Proc. SPIE: Security and Watermarking of Multimedia Contents*, vol. 4314, San Jose, CA, USA, pp. 575-584, January 2001.