



UNIVERSIDAD DE CHILE  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

ESTUDIO DE TÉCNICAS DE COMPRESIÓN DE IMÁGENES  
ASTRONÓMICAS

TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN CIENCIAS DE LA  
INGENIERÍA MENCIÓN ELÉCTRICA

CÉSAR FERNANDO VALENZUELA LEÓN

PROFESOR GUÍA:  
JORGE SILVA SÁNCHEZ

MIEMBROS DE LA COMISIÓN:  
MARCOS ORCHARD CONCHA  
RENÉ MÉNDEZ BUSSARD

SANTIAGO DE CHILE  
2013

RESUMEN DE LA TESIS  
PARA OPTAR AL GRADO DE  
MAGÍSTER EN CIENCIAS DE LA INGENIERÍA  
MENCIÓN ELÉCTRICA  
POR: CÉSAR VALENZUELA LEÓN  
FECHA: DICIEMBRE DE 2013  
PROF. GUÍA: SR. JORGE SILVA SÁNCHEZ

## “ESTUDIO DE TÉCNICAS DE COMPRESIÓN DE IMÁGENES ASTRONÓMICAS”

Esta tesis presenta un estudio acerca de la compresión, con pérdidas, de imágenes astronómicas mediante el estándar JPEG2000. La principal motivación de esta investigación surge al comparar los desempeños de compresión para diversas morfologías de imágenes, se observa que para una misma tasa de bits, el nivel de PSNR de las imágenes naturales es mayor que el de las imágenes astronómicas, es decir la compresión de dichas imágenes presenta menos pérdidas. Debido a esto, es necesario realizar ajustes sobre JPEG2000 de manera que ambos desempeños puedan equipararse.

El primer enfoque utilizado se basa en la implementación de la wavelet Haar, aquella de mayor compresibilidad sobre imágenes astronómicas, en JPEG2000. Sin embargo, al realizar esta modificación se aprecia que los resultados obtenidos no son mejores a los que proporciona la versión original del estándar, con la base CDF 9/7.

A partir del análisis de dependencias intraescala en el dominio Wavelet, se logra verificar que la característica que tiene mayor repercusión en el desempeño de la compresión es la presencia de estructura. Razón que explica el por qué la base CDF 9/7, a pesar de no presentar mejor compresibilidad que la base de Haar, permite alcanzar mejores niveles de compresión.

Finalmente, se propone dividir las imágenes astronómicas en dos partes: “Background” y “Cartoon”, esta última contiene los píxeles de mayor intensidad y al ser una señal *sparse*, es decir con muy pocos coeficientes distintos a cero, su alto nivel compresibilidad en base de Haar (reconstrucción perfecta con aproximadamente un 20 % de coeficientes) permite soslayar la carencia de estructura. La compresión del “Cartoon”, mediante JPEG2000 con la base de Haar, entrega resultados superiores a los obtenidos por la base CDF 9/7, por lo que se puede concluir que este enfoque puede ser aprovechado en aplicaciones que puedan prescindir de la totalidad de los píxeles y que hagan énfasis en aquellos valores más significativos.

La investigación presentada en esta memoria complementa los resultados encontrados en la literatura y abre una nueva línea que deberá seguir siendo explorada en el futuro.

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Antecedentes . . . . .	1
1.2. Descripción del Problema . . . . .	2
1.3. Aportes del Trabajo . . . . .	3
1.3.1. Aportes . . . . .	3
1.3.2. Alcances . . . . .	3
1.4. Estructura de la Tesis . . . . .	4
<b>2. Compresión de imágenes usando la transformada wavelet</b>	<b>5</b>
2.1. Transformada Wavelet . . . . .	5
2.2. Análisis Multiresolución . . . . .	6
2.3. Análisis en tiempo-frecuencia y el principio de incertidumbre de Heisenberg .	6
2.3.1. Aplicación de la Transformada Wavelet en imágenes . . . . .	8
2.4. Algoritmos de compresión de imágenes con la transformada wavelet . . . . .	10
2.4.1. EZW . . . . .	10
2.4.2. SPIHT . . . . .	12
2.4.3. JPEG2000 . . . . .	14
<b>3. Análisis del problema</b>	<b>15</b>
3.1. Aplicación de JPEG2000 en imágenes astronómicas . . . . .	15
3.1.1. Selección de base . . . . .	19
3.2. Implementación Wavelet de Haar en JPEG2000 . . . . .	21
3.2.1. Modificación JPEG2000 . . . . .	22
<b>4. Análisis de dependencias entre coeficientes en el dominio Wavelet</b>	<b>24</b>
4.1. Información Mutua . . . . .	25
4.2. Caracterización de los coeficientes en el dominio wavelet . . . . .	25
4.2.1. Construcción de $f$ . . . . .	27
4.3. Estimador de la Información Mutua . . . . .	27
4.4. Análisis de dependencias intraescala . . . . .	28
<b>5. División de Imágenes Astronómicas</b>	<b>31</b>
5.1. Umbral en el dominio de los píxeles . . . . .	31
5.2. Análisis del “Background” . . . . .	31
5.2.1. Compresibilidad . . . . .	33
5.2.2. Dependencias Intraescala . . . . .	34
5.2.3. JPEG2000 . . . . .	36

5.3. Análisis del “Cartoon” . . . . .	38
5.3.1. Compresibilidad . . . . .	38
5.3.2. Dependencias Intraescala . . . . .	40
5.3.3. JPEG2000 . . . . .	42
<b>6. Conclusiones y Trabajo Futuro</b>	<b>45</b>
6.1. Conclusiones . . . . .	45
6.2. Trabajo Futuro . . . . .	46
<b>Referencias</b>	<b>47</b>
<b>Anexo A</b>	<b>51</b>

# Capítulo 1

## Introducción

### 1.1. Antecedentes

En la actualidad, la cantidad de información disponible crece a pasos agigantados. La necesidad de guardarla sin sobrepasar la capacidad disponible de almacenamiento impulsa la generación de técnicas que permitan disponer de la información, sin pérdidas relevantes, pero con un tamaño significativamente menor al original.

El proceso anterior se conoce como *Compresión de Datos* y se origina cuando Shannon establece las bases teóricas, particularmente lo que se conoce como *Teoría de Tasa-Distorsión* [1].

A partir de entonces, los esfuerzos por lograr avances en esta área han sido incesantes, en especial con la masificación de los sistemas digitales. Por ello no es de extrañar el desarrollo de algoritmos de compresión de audio [2], [3], [4], video [5], [6], datos en general [7] [8] e imágenes. Cabe señalar que todos estos nichos son sometidos a esquemas de compresión *sin pérdidas* cuando la señal reconstruida es idéntica a la original, y *con pérdidas* en el caso opuesto.

Uno de los enfoques más empleados en la compresión de datos es el *Transform Coding*, [9] que consiste en reducir la cantidad de información a transmitir por medio de la aplicación de una transformación del espacio de la señal. Es decir, se toma la señal original y se lleva a un dominio donde se representa de una forma *sparse* (es decir, donde la mayoría de los coeficientes son iguales a cero). Tal como indica el esquema en la Figura 1.1, los pasos siguientes son la cuantización [10] y la codificación. En estas etapas (principalmente en la cuantización) se genera la distorsión respecto a la señal original, lo que provoca que la versión cuantizada no sea idéntica a la señal original [11].

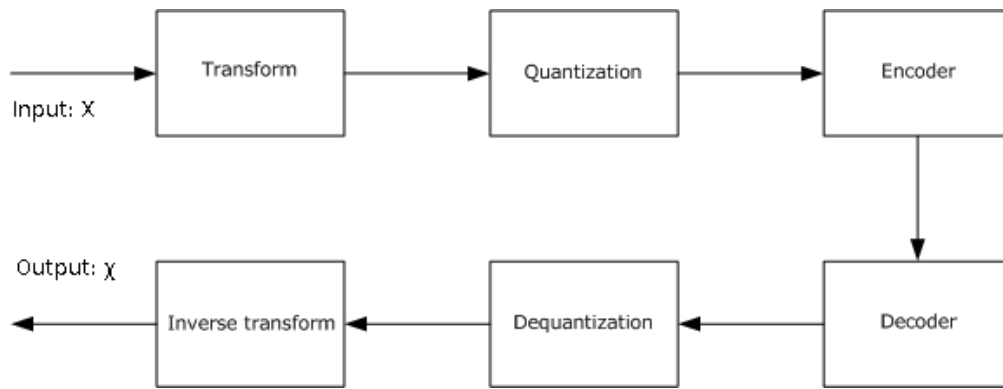


Figura 1.1: Diagrama de bloques del esquema *Transform Coding* para compresión de datos

*Transform Coding* ha sido utilizado en la compresión de imágenes, destacándose el estándar JPEG [12] el cual emplea la transformada DCT (*Discrete Cosine Transform*) [13]. Sin embargo, JPEG presenta dificultades técnicas (principalmente la creación de artefactos en el régimen de baja resolución [14]) que instaron a buscar una variante a la hora de efectuar la compresión. Por esta razón, la transformada wavelet [15] ha ganado importancia, posicionándose como una herramienta fundamental en lo que respecta a compresión de imágenes. De hecho, está presente en los algoritmos EZW [16], SPIHT [17] y en el estándar JPEG2000, entre otros.

En efecto, JPEG2000 [18] es el *estado del arte* en compresión con pérdidas y sus características son: transmisión progresiva, selección de área de interés, mayor compresión a bajas tasas, entre otras. Estas características lo transforman en el referente a la hora de investigar en la compresión de imágenes. En esta investigación se propone evaluar el desempeño de JPEG2000 en la compresión de imágenes astronómicas y al mismo tiempo, se proponen nuevas estrategias que permitan mejorar el desempeño del método, idealmente hasta lograr valores similares a los presentados por las imágenes naturales de prueba, presentes en la literatura.

## 1.2. Descripción del Problema

Las imágenes astronómicas se caracterizan por presentar una morfología muy particular: en líneas generales corresponde a una colección de objetos de alta intensidad, puntuales y difusos (o extendidos), que se distribuyen en la imagen de una forma “sparse” y sin estructura, ocupando un % muy menor de ella. Dentro de esta tipo de imágenes hay dos grupos dependiendo la cantidad de objetos que contengan [19]

1. **Low-photon:** Menos del 10 % de la imagen está comprendida por objetos de alta intensidad.
2. **High-photon:** Más del 10 % de la imagen está comprendida por objetos de alta intensidad.

Lo anterior, desde la óptica del análisis de señales, implica que esta morfología es una señal de alta frecuencia, todo lo contrario a las imágenes utilizadas en el procesamiento de imágenes

convencional. Por otro lado, las imágenes astronómicas han sido estudiadas profundamente, ya sea en problemas de reconstrucción [20] [21] [22], *denoising* [23] e incluso compresión [24] [25].

Además, tomando en cuenta que la cantidad de imágenes astronómicas disponibles aumentará en el corto plazo [26], la necesidad de almacenarlas de forma más eficiente, en un régimen *con pérdidas*, se hace cada vez más imperiosa. Por lo tanto, esto plantea la necesidad de diseñar métodos específicos para este tipo de imágenes, es decir, se debe modificar los esquemas de compresión para obtener mejores resultados.

Una primera aproximación es abocarse a mejorar de los algoritmos de compresión de imágenes astronómicas existentes, tales como FITS [27] y H-COMPRESS [28], entre otros. Sin embargo, desde el punto de vista de análisis de señales, estos algoritmos no son compatibles con esta investigación por los siguientes motivos:

- están diseñados para el caso de compresión sin pérdidas, el cual no será cubierto en este trabajo
- sus principales innovaciones están orientadas al almacenamiento de metadatos, tales como posición espacial (coordenadas), intensidad, color, composición, etc.

Alternativamente en este trabajo se propone modificar el algoritmo JPEG2000 de manera que para imágenes de tipo astronómicas los niveles de compresión y distorsión sean similares o mejores a los logrados por las imágenes tradicionales y/o naturales.

## 1.3. Aportes del Trabajo

### 1.3.1. Aportes

Los objetivos trazados son:

- Estudio de las características de las imágenes astronómicas y cuantificar sus diferencias con las imágenes tradicionales.
- Modificación y adaptación de JPEG2000 para las señales de estudio.
- Análisis cuantitativo de la solución propuesta.
- Comparación de los resultados con la literatura existente.

### 1.3.2. Alcances

Esta problemática se centra dentro de los esfuerzos del Grupo de Sistemas de Información y Decisión del Departamento de Ingeniería Eléctrica para generar un conjunto de herramientas que permitan comprimir señales de alta complejidad. El caso particular de esta tesis se enmarca dentro del planteamiento de una configuración de JPEG2000 para la compresión de imágenes provenientes y del interés de la Astronomía.

## 1.4. Estructura de la Tesis

Esta tesis se organiza de la siguiente forma: El Capítulo 2 trata la compresión de datos mediante la transformada wavelet. En el Capítulo 3 se analizará el comportamiento de las imágenes astronómicas al ser procesadas y sintetizadas por diversas bases y se procederá a modificar el algoritmo JPEG2000. La justificación del por qué las modificaciones desarrolladas funcionan de una forma determinada, se detallan en el Capítulo 4, por medio de la observación de los coeficientes en el dominio transformado y la relación que tienen entre ellos. Posteriormente, el Capítulo 5 presenta una formulación alternativa que explota la morfología de las señales astronómicas.



# Capítulo 2

## Compresión de imágenes usando la transformada wavelet

El término wavelet hace referencia a las funciones de la forma  $\psi_{ab}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right)$ , es decir, el conjunto de funciones formado por el escalamiento, controlado por el parámetro  $a \in \mathfrak{R}^+$ , y las traslaciones controladas por el parámetro  $b \in \mathfrak{R}$ , de una función  $\psi(x)$  llamada *wavelet madre*.

Una propiedad sobresaliente es que las wavelets forman una base *ortonormal* de las funciones  $\mathbb{L}^2$ . Esto es, para todo  $f(t) \in \mathbb{L}^2$  se tiene

$$f(t) = \sum_{j,k} a_{j,k} \psi_{j,k}(t), \quad (2.1)$$

donde  $a_{j,k}$  son los coeficientes originados como producto interno entre  $f(t)$  y la función wavelet.

### 2.1. Transformada Wavelet

Se define la *transformada wavelet continua* de la forma

$$W_f(j, k) = \int f(t) \psi_{j,k}(t) dt. \quad (2.2)$$

De forma análoga se presenta la transformada wavelet *discreta* (DWT), basada en la expresión (2.2), con la salvedad que los parámetros de escala y traslación se parametrizan de forma discreta, dando origen a los valores

$$a_j = 2^{-j} \quad \text{y} \quad b_{j,k} = 2^{-j}k, \quad \forall (j, k) \in \mathbb{Z}^2.$$

La función resultante se localiza en el espacio *tiempo-frecuencia*, es decir ambas dimensiones son relevantes en el análisis, lo que genera un *trade-off* entre la precisión que se puede obtener en dichos rangos. Por ejemplo, si se tiene una caracterización con alta resolución en el dominio de la frecuencia, la localización en el tiempo será pobre [30].

El razonamiento anterior es una extensión de lo postulado por Heisenberg en su *Principio de Incertidumbre* [29].

## 2.2. Análisis Multiresolución

Un aspecto fundamental de los wavelets y su teoría es la noción de aproximación por multi-resolución de  $L^2(\mathbb{R}^n)$ , acuñado por Mallat [30].

**Definición 1** Una aproximación por multiresolución de  $L^2(\mathbb{R}^n)$  es una secuencia creciente  $V_j$ ,  $j \in \mathbb{Z}$  de subespacios cerrados de  $L^2(\mathbb{R}^n)$  tales que:

1.  $\bigcap_{-\infty}^{\infty} V_j = \{0\}$ ;
2.  $\bigcup_{-\infty}^{\infty} V_j$  es denso in  $L^2(\mathbb{R}^n)$ ;
3.  $f(x) \in V_j \iff f(2x) \in V_{j+1}$ , para cualquier  $f \in L^2(\mathbb{R}^n)$  y cualquier  $j \in \mathbb{Z}$ ;
4.  $f(x) \in V_0 \iff f(x - k) \in V_0$ , para cualquier  $f \in L^2(\mathbb{R}^n)$  y cualquier  $j \in \mathbb{Z}$ ; y
5. Existe una función  $g(x)$ , conocida como la función de escala, tal que  $\{g(x - k) : k \in \mathbb{Z}^n\}$  es una base de Riesz de  $L^2(\mathbb{R}^n)$ .

Debido a la propiedad (2), toda función perteneciente  $L^2(\mathbb{R}^n)$  puede ser aproximada arbitrariamente bien por su proyección en el espacio  $V_j$ . Además, cuando se pasa desde un espacio  $V_j$  hacia su sucesor  $V_{j+1}$ , se introducen funciones del doble de la frecuencia, tal como dice la propiedad 3). En otras palabras, esto significa que una función  $f$  puede ser aproximada a la resolución  $2^{-j}$  por un elemento de  $V_j$ .

## 2.3. Análisis en tiempo-frecuencia y el principio de incertidumbre de Heisenberg

Sin importar el método que se utilice para descomponer una señal de manera simultánea en tiempo y frecuencia, siempre se presentará el problema que si se requiere una información localizada en el tiempo, se incrementará el nivel de incertidumbre en el dominio de la frecuencia y si se requiere información precisa sobre la frecuencia se perderá precisión en el tiempo. Este compromiso puede ser ilustrado con un plano en el cual el tiempo varia horizontalmente y la frecuencia verticalmente.

El principio de incertidumbre de Heisenberg establece que el área de cada celda, o caja, debe ser de al menos  $\frac{1}{4}\pi$ , pero dependiendo de la base utilizada, las celdas tendrán diferentes formas y posiciones. Por ejemplo, en la base del tiempo, la cual representa el instante cuando los eventos ocurren, el plano contiene las más altas y delgadas celdas permitidas por el intervalo

de muestreo. En la base de Fourier, las bases muestran las frecuencias que se presentan en eventos que ocurren en periodos de tiempo largos, por lo que las celdas de Heisenberg son cortas (buena resolución en frecuencia) pero son demasiado anchas (mala resolución en el dominio del tiempo).

Con la transformada de Fourier de tiempo corto, la forma de las celdas de Heisenberg dependen del tamaño de la ventana. Una ventana pequeña tiene intervalos cortos en el tiempo a costa de ser vaga acerca de la frecuencia. Una ventana grande es menos precisa en el tiempo, pero más precisa en frecuencia. En ambos casos, el tamaño de la ventana se mantiene fija para cada descomposición, es decir, es la misma para frecuencias altas y bajas.

Con Wavelets, una ventana grande es utilizada para bajas frecuencias y ventanas más pequeñas son empleadas para frecuencias altas. Pero, como postula el principio de Heisenberg, las celdas que corresponden a una wavelet corta son más altas, es decir, el tiempo se privilegia a expensas de la frecuencia, tal como se ilustra en la Figura 2.5.

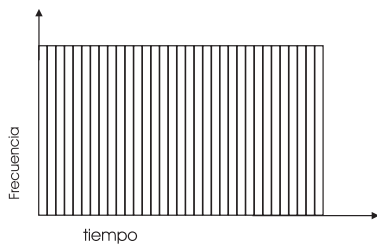


Figura 2.1: Análisis en tiempo

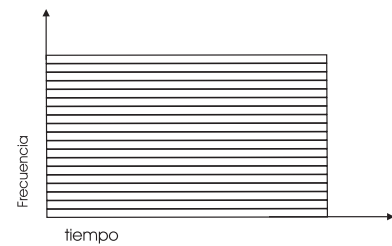


Figura 2.2: Análisis en frecuencia

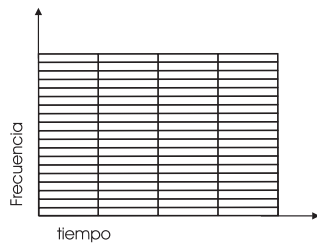


Figura 2.3: Análisis en tiempo-frecuencia utilizando la transformada corta de Fourier. Caso 1

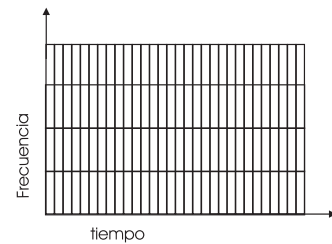


Figura 2.4: Análisis en tiempo-frecuencia utilizando la transformada corta de Fourier. Caso 2

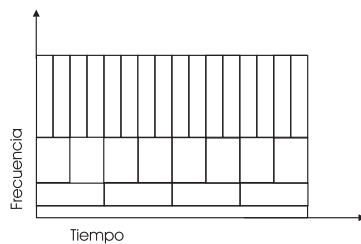


Figura 2.5: Análisis en tiempo-frecuencia utilizando la transformada Wavelets

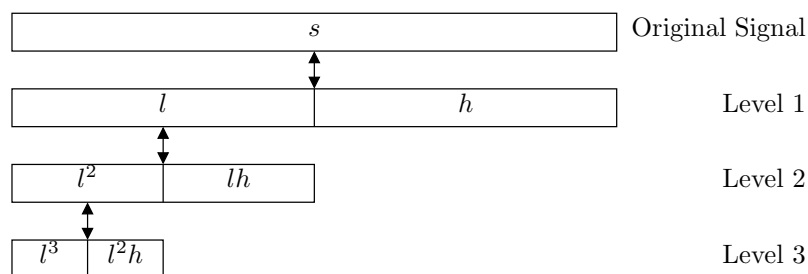


Figura 2.6: Una descomposición DWT diádica de 3 niveles.

### 2.3.1. Aplicación de la Transformada Wavelet en imágenes

La segunda etapa de compresión se basa en la transformada wavelet discreta o DWT (*Discrete Wavelet Transform*). Esta representa los puntos de la imagen a comprimir en un dominio espacio-frecuencial que posee dos ventajas fundamentales: (1) reducir la correlación espacial de cada componente de la imagen, lo que es esencial de cara a maximizar las tasas de compresión, y (2) encontrar una representación multiresolución para cada componente, propiedad que puede ser de ayuda a la hora de procesar imágenes muy grandes.

A continuación se describirá la DWT desde el punto de vista de la Teoría de los Bancos de Filtros

Según dicha teoría, una secuencia unidimensional de  $N$  muestras  $x[n]$ ,  $n = 0, \dots, N-1$  puede ser representada mediante dos secuencias  $x_l[n]$ ,  $n = 0, \dots, \frac{N}{2}-1$  y  $x_h[n]$ ,  $n = 0, \dots, \frac{N}{2}-1$  de la mitad del tamaño de la señal original, donde  $x_l[n]$  es el resultado de filtrar  $x[n]$  usando un filtro pasa bajos  $L$  y  $x_h[n]$  es el resultado de filtrar  $x[n]$  mediante un filtro pasa altos  $H$ . Este banco de dos filtros se define de forma que desde el punto de vista de la frecuencia, lo que el filtro  $L$  deja pasar es lo que  $H$  no deja pasar y viceversa. Por este motivo, la información contenida en las bandas  $x_l[n]$  y  $x_h[n]$  es la misma que en la secuencia original  $x[n]$ , que puede ser regenerada usando los correspondientes filtros de síntesis.

Este proceso de descomponer una señal en dos señales, una de baja frecuencia y otra de alta puede aplicarse recursivamente a ambas subbandas. Cuando sólo se aplica a la banda de baja frecuencia, se habla de la DWT en su versión diádica (*dyadic*). Si se aplica a ambas, obtenemos la DWT en su versión paquetizada (*packet*). En el caso de la compresión de imágenes, la forma más utilizada en la diádica porque la energía se acumula principalmente en la zona de baja frecuencia.

En la Figura 2.6 se muestra gráficamente el proceso de descomposición aplicado en la DWT diádica unidimensional. El número máximo de etapas o niveles es igual a  $\log_2(N)$ . A los coeficientes de las bandas  $l^{\log_2(N)}$  y  $l^j h[n]$ ,  $j = 0, \dots, \log_2(N) - 1$ ;  $n = 0, \dots, \frac{N}{2^{j+1}} - 1$ , donde  $j$  es el nivel de la descomposición, se les llama coeficientes *wavelet*.

Usando  $l^n[0]$  y  $l^j h[n]$  es posible obtener  $\log_2(N)$  niveles de resolución a la hora de representar  $x[n]$ . Así, el primer nivel de resolución estaría formado por una señal constante e igual a  $l^{\log_2(N)}[0]$ , el segundo nivel utilizaría los coeficientes  $l^{\log_2(N)}[0]$  y  $l^{\log_2(N)-1}h[0]$ , y así sucesiva-

mente. Finalmente, nótese que se cumple que  $x[n] = l^0[n]$ .

La DWT se diferencia de otras transformadas clásicas como la DFT (Discrete Fourier Transform) o la DCT (Discrete Cosine Transform) en un aspecto fundamental: se trata de una representación espacio-frecuencial, no únicamente frecuencial. Gracias a esto es posible seleccionar un subconjunto de coeficientes *wavelet* para reconstruir sólo una parte de la señal original. Esto es muy útil cuando procesamos imágenes muy grandes y sólo podemos visualizar una parte de la imagen.

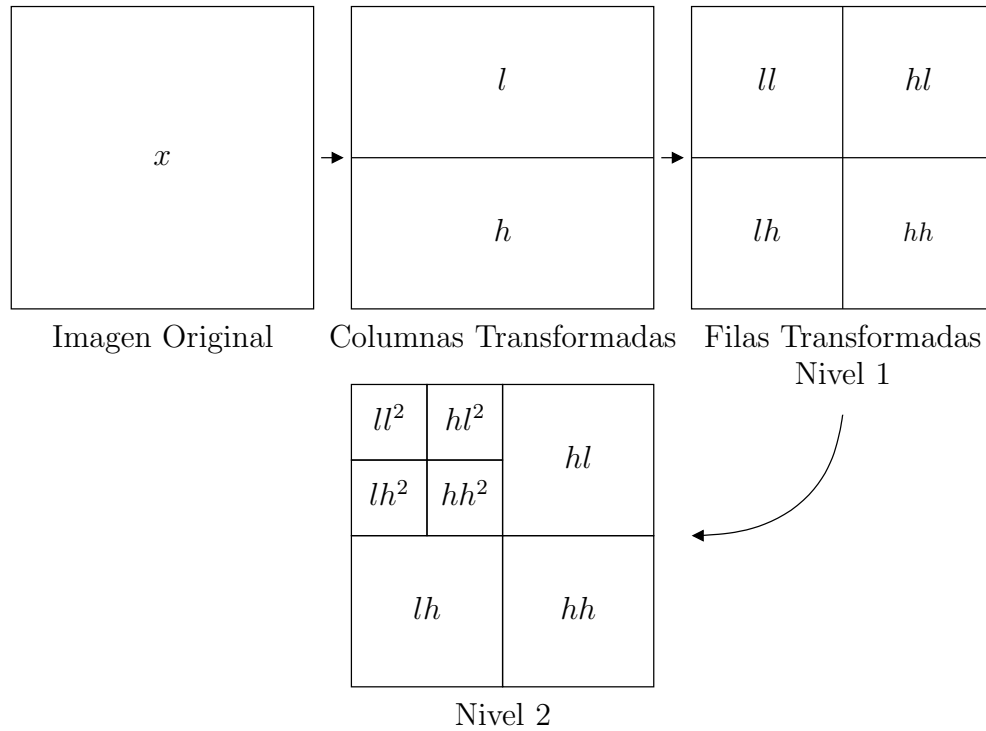


Figura 2.7: Una descomposición DWT diádica bidimensional de 2 niveles.

Ya que estamos trabajando con imágenes y éstas tienen dos dimensiones, es necesario extender este proceso al caso bidimensional. Para esto podemos calcular la DWT diádica bidimensional aplicando la DWT diádica unidimensional primero a las filas y luego a las columnas de la imagen (o viceversa). Este proceso se describe gráficamente en la Figura 2.7. De esta forma, tras cada fase de descomposición se obtienen cuatro bandas en lugar de dos.

En lo que resta de este trabajo, se hará referencia al concepto de *Niveles de Profundidad* (o profundidad), el cual no es más que el número de veces que se aplica la transformada Wavelet, en 2 dimensiones, sobre una imagen. Tomando como ejemplo la Figura 2.8, se puede señalar que el nivel de profundidad para ese caso es igual a 2.



Figura 2.8: Transformada diádica DWT a la imagen “lena”.

## 2.4. Algoritmos de compresión de imágenes con la transformada wavelet

### 2.4.1. EZW

El algoritmo EZW (Embedded Zerotree Wavelet, EZW) [16] es un método sencillo de compresión de imágenes, que tiene la propiedad de que los bits en el *bitstream*, o secuencia de bits codificados, se generan en orden de importancia. Esto permite que tanto el codificador como el decodificador puedan truncar su operación en cualquier parte de la cadena, generando imágenes comprimidas con una distorsión proporcional a la cantidad de bits omitidos. El algoritmo EZW puede llegar a producir resultados de compresión competitivos respecto a los otros algoritmos disponibles.

#### Funcionamiento del algoritmo

Un coeficiente wavelet  $x$  es no significativo respecto a un umbral dado  $T$ , si se tiene que  $|x| \leq T$ . Este método conocido como *zerotree* se basa en la hipótesis de que si un coeficiente wavelet, en cualquier escala, no es significativo respecto a un umbral  $T$ , entonces todos los coeficientes en la misma orientación y ubicación espacial, a escalas más finas, tienen alta probabilidad de ser insignificantes para el mismo umbral. Específicamente, en un sistema jerárquico de subbandas, con la excepción de las subbandas de más alta frecuencia, cada coeficiente a cierta escala está relacionado con un conjunto de coeficientes en la escala más fina, a similar orientación. El coeficiente de la escala más tosca se conoce como *padre* y todos los coeficientes en la próxima escala más fina, que tienen la misma orientación y ubicación espacial se denominan *hijos*. De la misma forma, se pueden definir los conceptos *ancestros* y *descendientes*. Dado un umbral  $T$  para determinar si un coeficiente es significativo o no, un coeficiente  $x$  es un *elemento del zerotree* si para un umbral  $T$ , él y todos sus descendientes son insignificantes respecto a  $T$ . Por lo tanto, dado un umbral, cada coeficiente wavelet puede pertenecer a uno de los siguientes cuatro tipos de datos: *raíz del zerotree* (ZRT), *cero aislado* (IZ) (el coeficiente es insignificante, pero su descendiente no lo es), *significante positivo* (POS) y *significante negativo* (NEG).

El algoritmo EZW crea árboles usando como raíz un pixel en las subbandas de resolución

más fina (exceptuando la subbanda de aproximación) y como hijos de este, emplea a coeficientes posicionados de forma similar en las otras subbandas, tal como ilustra la Figura 2.9. Hay dos tipos de “pasadas” en el algoritmo: una pasada dominante y otra subordinada. La pasada dominante encuentra los píxeles cuyo valor es mayor al umbral, mientras que la pasada subordinada cuantiza estos valores.

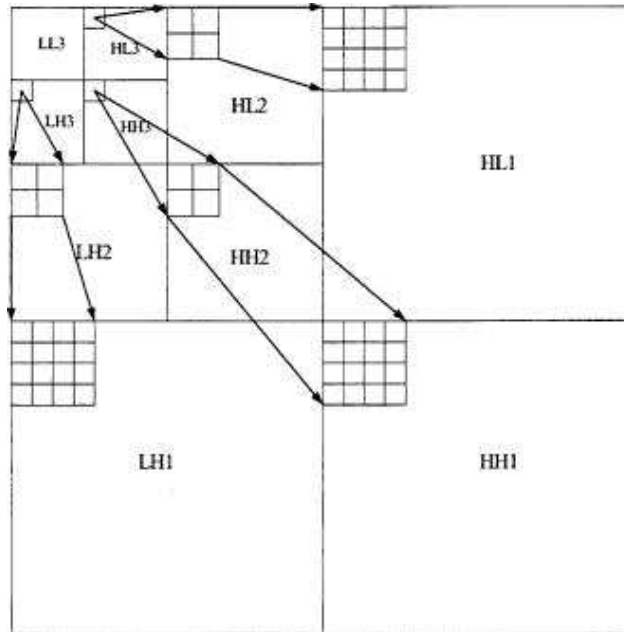


Figura 2.9: Representación gráfica de la estructura de árbol en la Transformada Wavelet, donde cada coeficiente tiene 4 hijos en la escala inferior.

### Pasada dominante

En la pasada dominante se revisan todos los árboles buscando valores significativos con respecto a un umbral determinado. El umbral inicial se escoge como la mitad de la mayor magnitud de los coeficientes. En las pasadas siguientes, el umbral es la mitad del umbral anterior. Cuando un coeficiente se considera insignificante y se revisa que sus hijos también lo son, estos son codificados en un símbolo, una raíz del *zerotree*, en lugar de un símbolo para el padre y un símbolo para cada descendiente, de esa forma se logra la compresión. Los coeficientes que son significativos se codifican con un símbolo positivo, para valores mayores que cero, o negativo, para valores menores que cero, luego éstos se añaden a la “lista subordinada” para luego ser cuantizados y el valor del coeficiente se deja en cero para la próxima pasada dominante. Los coeficientes no significativos, pero cuyos hijos no lo son, se codifican como “ceros aislados”. De esta forma, la pasada dominante mapea los píxeles en un alfabeto de cuatro símbolos, los cuales pueden ser codificados aritméticamente [31].

### Pasada subordinada

Luego de cada pasada dominante, se realiza una pasada subordinada sobre la lista que contiene a aquellos coeficientes etiquetados como significativos. La pasada subordinada cuantiza los

valores de dicho grupo, lo que permite compresión al decirle al decodificador cuál es el valor del coeficiente mediante un símbolo, en lugar de transmitir el valor en sí, lo que requeriría una precisión infinita. Ya que el umbral inicial es la mitad de la magnitud máxima entre todos los píxeles, en la primera pasada subordinada los valores se dividen en dos grupos: aquellos en la mitad superior del intervalo *umbral-píxel máximo* y aquellos pertenecientes a la mitad inferior. Un píxel en la mitad superior se codifica con el símbolo “*upper*”, mientras que uno en la mitad inferior se codifica con el símbolo “*lower*”. Cada punto se cuantiza, según el punto de vista del decodificador, en el punto medio de cada intervalo. En las siguientes pasadas subordinadas el umbral se ha reducido a la mitad, lo que genera el doble de intervalos de la pasada subordinada anterior más los dos nuevos correspondientes al nuevo umbral, con esto el decodificador es capaz de determinar el intervalo en el cual se encuentra cada coeficiente y reconstruye su valor en el punto medio de éste. Por lo tanto, la estimación del valor de las magnitudes se refina a medida que se efectúen más pasadas subordinadas

## Decodificación

Lo que se necesita para decodificar una imagen comprimida por el algoritmo EZW es el umbral inicial, el tamaño de la imagen, la descomposición en subbandas y el *bit-stream*. El decodificador descomprime el bit-stream en símbolos, crea la descomposición adecuada en subbandas, pues ya conoce los tamaños de ésta, luego deshace la compresión iterando sobre el umbral  $T$  que ya es conocido.

### 2.4.2. SPIHT

La idea principal tras el algoritmo SPIHT (Set Partitioning In Hierarchical Trees) [17] es que los coeficientes wavelets de mayor magnitud tienen una mayor influencia en la imagen reconstruida que aquellos de menor magnitud. Por lo mismo, el *bit budget*, o cantidad de bits disponibles para realizar la compresión, debería emplearse en la transmisión de aquellos coeficientes. Así, el algoritmo SPIHT transmite los coeficientes en el dominio wavelet en un orden parcial.

1. Pasada de ordenamiento: Se transmite el orden de los coeficientes a partir del plano de bits más significativo.
2. Pasada de refinamiento: Se transmiten los bits ubicados en planos más bajos que el actual.

El algoritmo examina cada nivel y establece la magnitud máxima de los coeficientes wavelet y por ende el máximo plano de bits. Una vez definido esto, se inspeccionan todos los coeficientes y todos aquellos que tengan el bit, correspondiente a ese plano, en 1 son llamados *significativos*, al contrario los que tengan el bit en 0 son *no significativos* para esta pasada. Esta información es transmitida al decodificador. El contador de planos de bits se decrementa en uno y se transmite la nueva información de significancias. También se transmite la información de refinamiento para aquellos valores establecidos como significativos en pasadas anteriores.

Una característica de algoritmo SPIHT es la forma es que las posiciones de los coeficientes no significativos son codificadas. Estas son transmitidas como sólo un símbolo: *zerotree*. Este



15	4	12
10	0	7
3	15	1

(a) Valores de los coeficientes transformados.

1111	0100	1100
1010	0000	0111
0011	1111	0001

(b) Valores de los coeficientes transformados en representación binaria.

1	0	1
1	0	0
0	1	0

(c) Primer plano de bits (MSB).

1	1	1
0	0	1
0	1	0

(d) Segundo plano de bits.

1	0	0
1	0	1
1	1	0

(e) Tercer plano de bits.

1	0	0
0	0	1
1	1	1

(f) Cuarto plano de bits (LSB).

Figura 2.10: Representación gráfica del concepto “Planos de bits”, para valores codificados a cuatro bits, donde el primer plano incluye a todos los bits más significativos (MSB), mientras que el último incluye a los de menor significación (LSB).

hecho se basa en la suposición que en las subbandas de bajo nivel (HH1, HL1, LH1, HH2, HL2 y LH2) hay una alta probabilidad de que se forme un árbol sólo con coeficientes no significativos, aprovechando las dependencias inter-escalas de la transformada wavelet. En este contexto, significancia quiere decir que todos los coeficientes de un árbol tienen una magnitud mayor a  $2^{n-1}$ , donde  $n$  es el contador de planos de bits. Claramente, mientras mayor sea  $n$ , mayor es la probabilidad de que se formen *zerotrees*.

Otra innovación que presenta SPIHT es la de tomar un árbol de coeficientes wavelets y, en el caso de que este no sea un *zerotree*, éste se particiona de forma que exista la posibilidad de que uno de esos sub-árboles sea un *zerotree* y de esa forma pueda ser transmitido de una manera más eficiente. El algoritmo realiza este proceso de tal forma que todas las particiones hechas por el codificador puedan ser rastreadas en el decodificador sin necesidad de ingresar más información que la significancia de los coeficientes.

### **2.4.3. JPEG2000**

Dada la relevancia de este algoritmo para este trabajo, en los Anexos se profundizará sobre las características de este estándar de compresión de imágenes.

# Capítulo 3

## Análisis del problema

Los algoritmos de compresión de imágenes, basados en Transform Coding (TC), tienen como principal característica el convertir la señal de entrada en una señal *sparse*, mediante la aplicación de una transformación o cambio de dominio, de manera que la energía de ésta quede concentrada en la menor cantidad de coeficientes. Ejemplos de lo anterior son JPEG y JPEG2000 con la Discrete Cosine Transform (DCT) y la Transformada Wavelet Cohen-Daubechies-Fauveau 9/7 (CDF 9/7) respectivamente.

Si se está en el escenario de compresión *con pérdidas* es necesario, de acuerdo al límite de bits por emplear, o *bit budget*, seleccionar los coeficientes más relevantes en términos de energía de la señal; es decir aquellos cuya magnitud, en el dominio transformado, sea mayor. En síntesis, el desempeño de los algoritmos depende de la eficacia de dicha selección y de la *sparsidad* de la entrada.

### 3.1. Aplicación de JPEG2000 en imágenes astronómicas

Tomando en cuenta el funcionamiento de JPEG2000, la naturaleza de las imágenes astronómicas y lo expuesto anteriormente, surge una pregunta ¿el estándar JPEG2000 mantendrá su desempeño - en términos de PSNR <sup>1</sup>- respecto a las imágenes naturales?.

Para efectuar los análisis en este trabajo, se utilizarán dos imágenes astronómicas, las cuales serán denominadas como “Imagen Astronómica 1” (Figura 3.1a) e “Imagen Astronómica 2” (Figura 3.1b).<sup>2</sup>

Una vez definidas las imágenes por analizar, se procede a comprimirlas mediante el estándar JPEG2000. En la Figura 3.2 se aprecia la curva de desempeño de la compresión para dichas imágenes y en la Figura 3.3 se muestran las imágenes comprimidas a una tasa de 0,16 bits por pixel. Para poder efectuar una comparación adecuada, se requiere contrastar estos resultados con los que JPEG2000 obtendría, bajo condiciones similares, al comprimir imágenes

---

<sup>1</sup>La relación Señal a Ruido de Pico o PSNR se define como,  $PSNR = 20 \log \left( \frac{255}{\sqrt{x - \hat{x}}} \right)$  [dB]

<sup>2</sup>Esta nomenclatura será empleada de forma permanente a lo largo de este trabajo.

naturales.

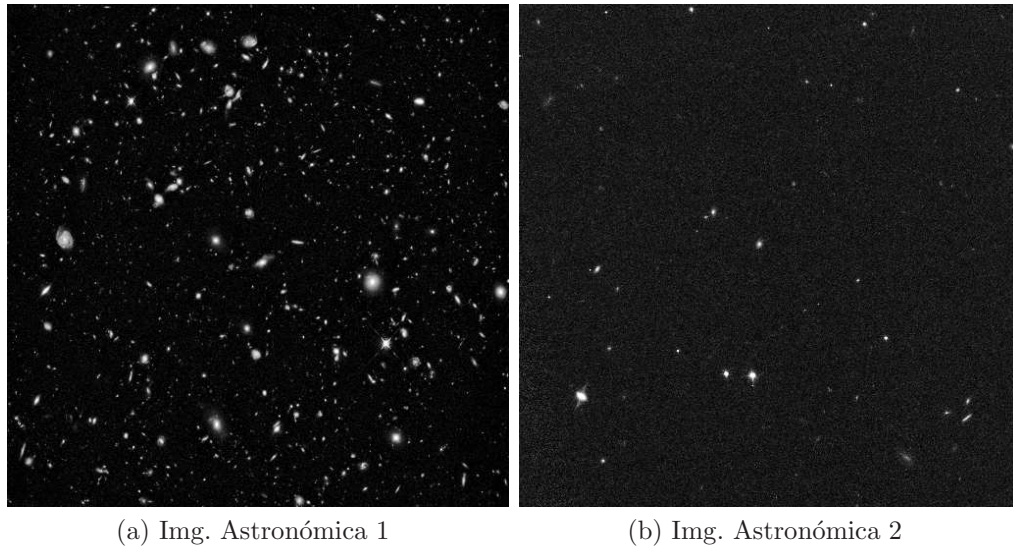


Figura 3.1: Imágenes astronómicas empleadas para realizar las pruebas.

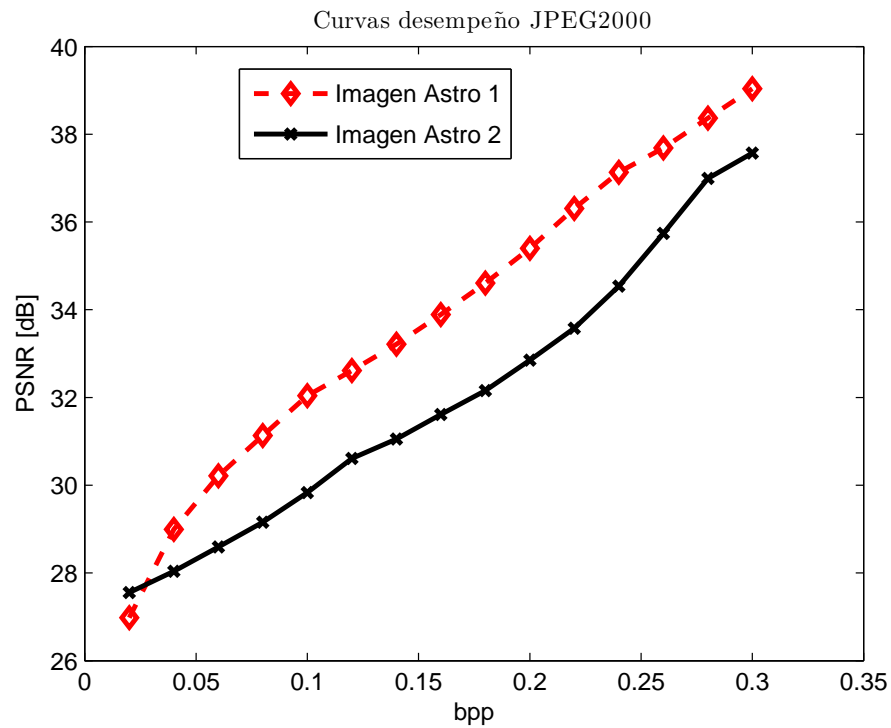
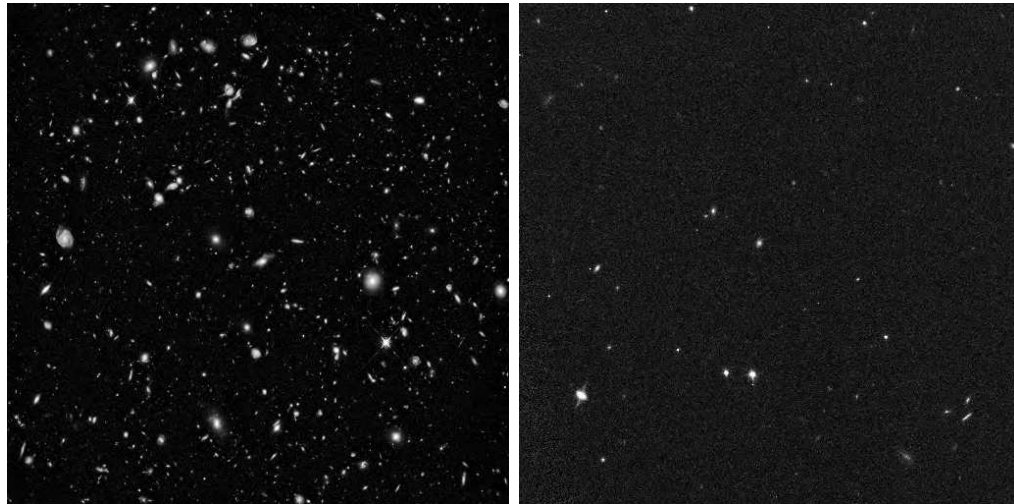


Figura 3.2: Curva PSNR vs Bits por pixel obtenida por la compresión de las dos imágenes astronómicas de prueba, mediante el estándar JPEG2000.

De manera análoga, se escogen dos imágenes naturales de prueba, Figuras 3.4a y 3.4b y se realiza compresión a la misma tasa (0,16 bpp). La curva de desempeño, Figura 3.5 muestra que los valores de PSNR, para esta morfología son muy superiores a los obtenidos por las



(a) Img. Astronómica 1 comprimida      (b) Img. Astro Astronómica 2 comprimida

Figura 3.3: Imágenes astronómicas comprimidas, codificadas a 0.16 bpp.

imágenes astronómicas.



(a) Img. Natural 1

(b) Img. Natural 2

Figura 3.4: Imágenes naturales empleadas para realizar las pruebas de compresión.

Se aprecia notoriamente que en la Figura 3.6 que existe una diferencia no despreciable, del orden de 5 [dB] a favor de las naturales, en el PSNR. de esto se plantea lo siguiente: es necesario modificar JPEG2000 para que la compresión de imágenes astronómicas mejore.

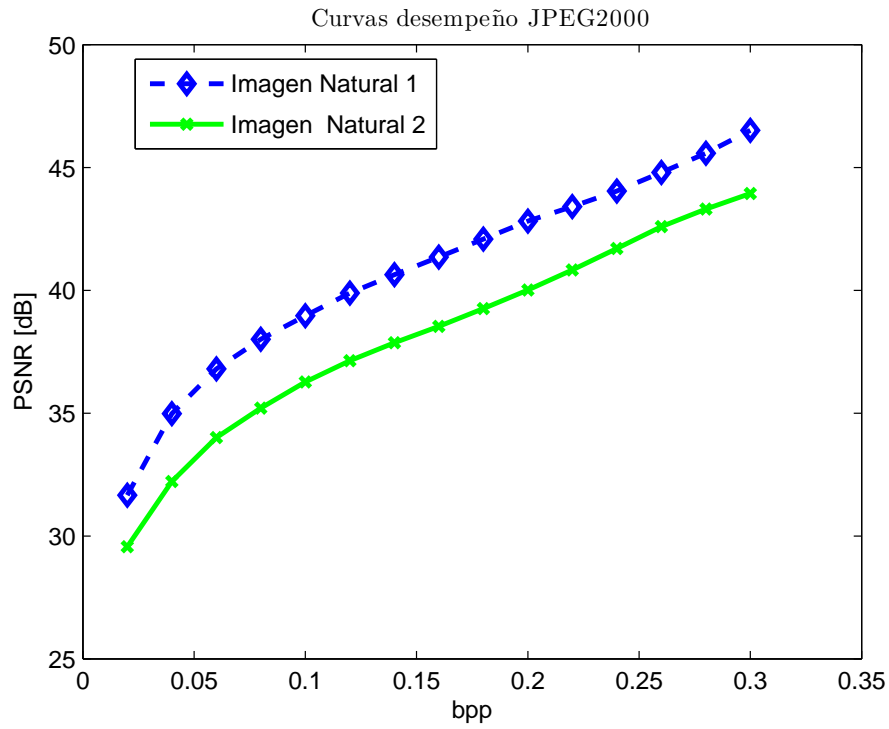


Figura 3.5: Curva PSNR vs Bits por pixel obtenida por la compresión de las dos imágenes naturales de prueba, mediante el estándar JPEG2000.

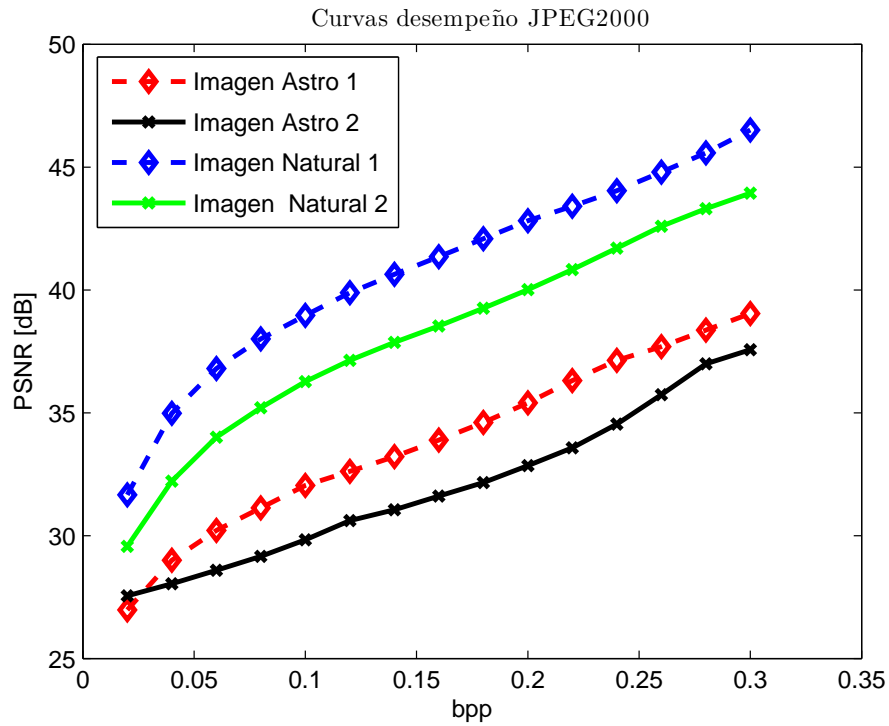


Figura 3.6: Curva PSNR vs Bits por pixel obtenida por la compresión de las imágenes astronómicas y naturales de prueba, mediante el estándar JPEG2000.

### 3.1.1. Selección de base

Una de las razones que puede justificar la merma en desempeño de JPEG2000 es que la base utilizada, CDF 9/7 [32], puede no ser la más idónea, en términos de compresibilidad de imágenes astronómicas. Es decir, puede existir una o varias bases ortonormales que ofrezcan mejor compresibilidad para la familia de imágenes astronómicas dada su naturaleza puntual. Para probar este hecho y, de paso, encontrar la base más adecuada para la morfología en estudio se efectuarán pruebas de selección de bases.

#### Procedimiento

A la imagen de referencia (Figura 3.1a) se le aplicará el esquema más simple de compresión: llevar la señal de entrada a un dominio transformado y hacer cero cierto porcentaje de los coeficientes. Una vez efectuado este proceso se implementa la transformación inversa, lo cual originará una imagen reconstruida. Finalmente se cuantifica el error de reconstrucción asociado a la omisión de los coeficientes menos significativos.

En este caso, se comparará la Wavelet de Haar no sólo con la CDF 9/7, sino que con todas las familias de wavelets presentes en el *toolbox WaveLab 850*<sup>3</sup>, de manera que sea posible discernir cuál es la más apropiada para señales de alta localización espacial.

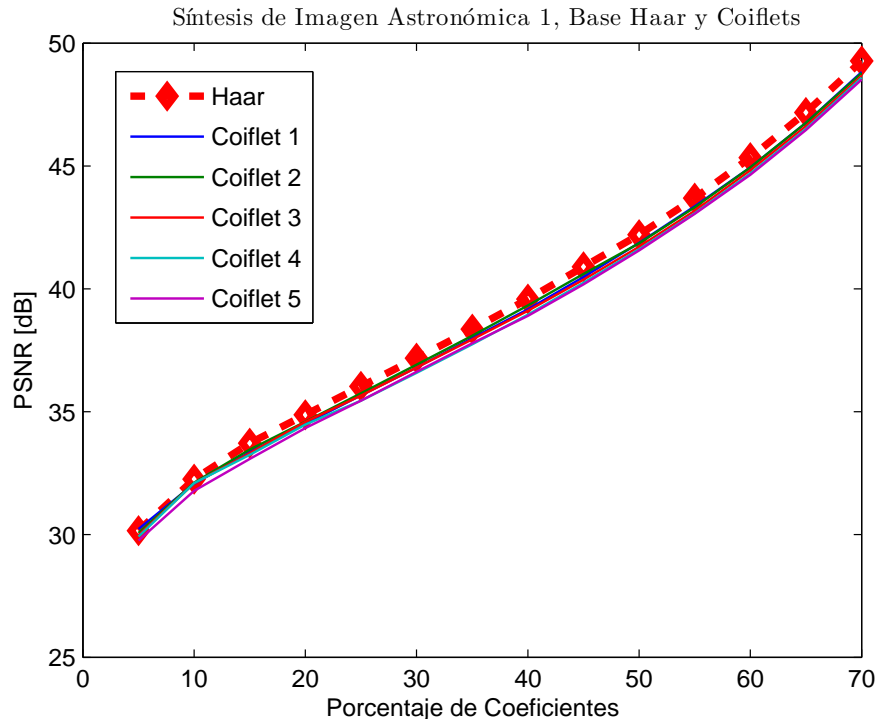


Figura 3.7: Prueba de compresibilidad, para una imagen astronómica, realizada para la base de Haar y las base pertenecientes a la familia Coiflet.

<sup>3</sup>WaveLab850: <http://www-stat.stanford.edu/~wavelab/>

Para realizar esta prueba se procedió a aplicar la transformada Wavelet, con distintas bases, sobre diversas imágenes astronómicas, luego un porcentaje variable de los coeficientes menos significativos fue llevado al valor cero y con la parte de la información de la señal no descartada se aplicó la Transformada Inversa, lo cual permitió obtener reconstrucciones de la imagen con distinta calidad.

En las Figuras 3.7, 3.8 y 3.9 se comprueba que la wavelet de Haar ofrece mejor compresibilidad para esta morfología de imágenes. La razón de dicho atributo radica en que la wavelet de Haar por su fisonomía captura de forma eficiente las discontinuidades de la señal de entrada, es decir, las componentes de alta frecuencia (cambios de negro a blanco y viceversa, en el dominio de los píxeles) son bien caracterizados por esta base.

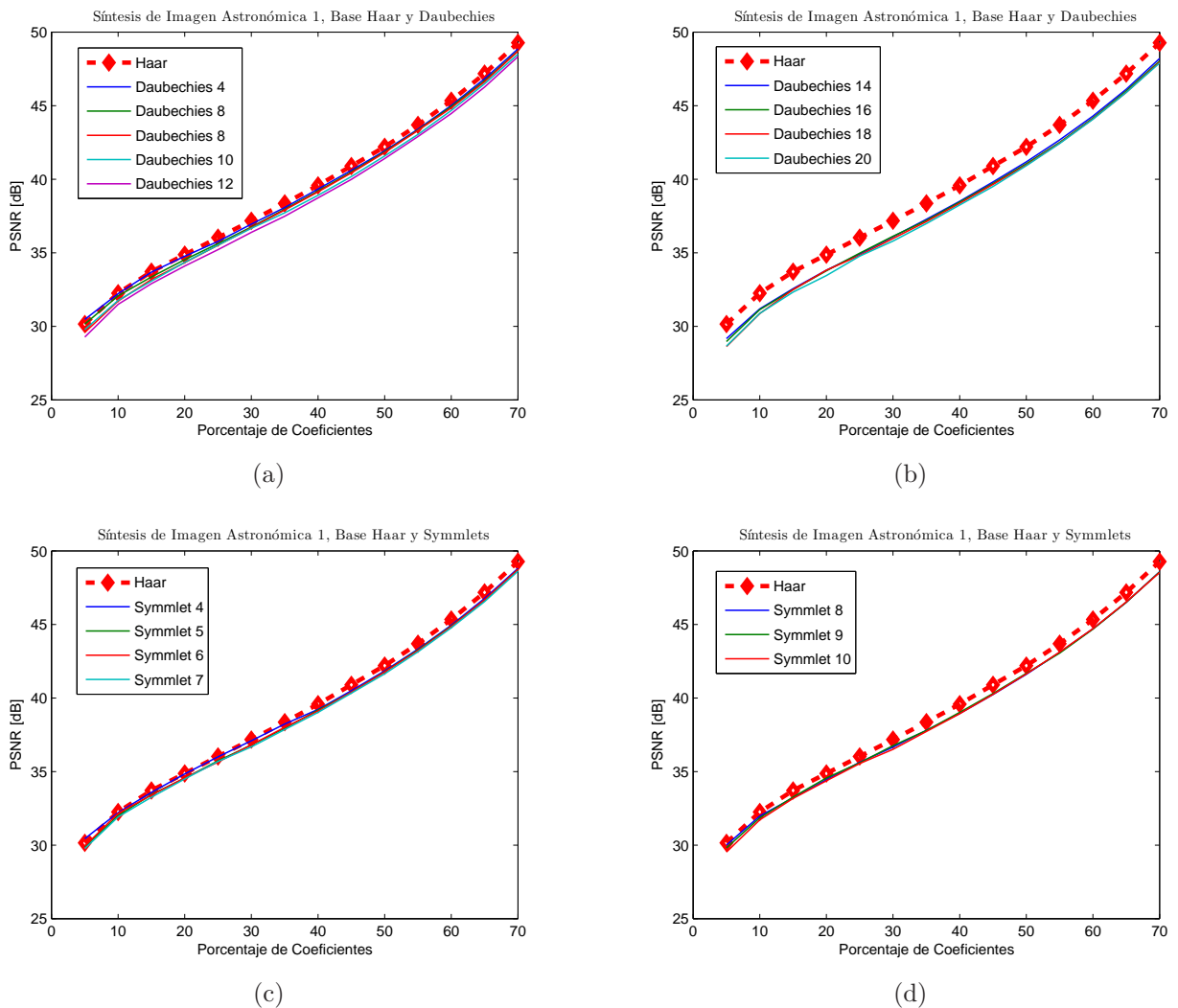


Figura 3.8: Prueba de compresibilidad, para una imagen astronómica, realizada para la base de Haar y las base pertenecientes a las familias Daubechies y Symmlets.



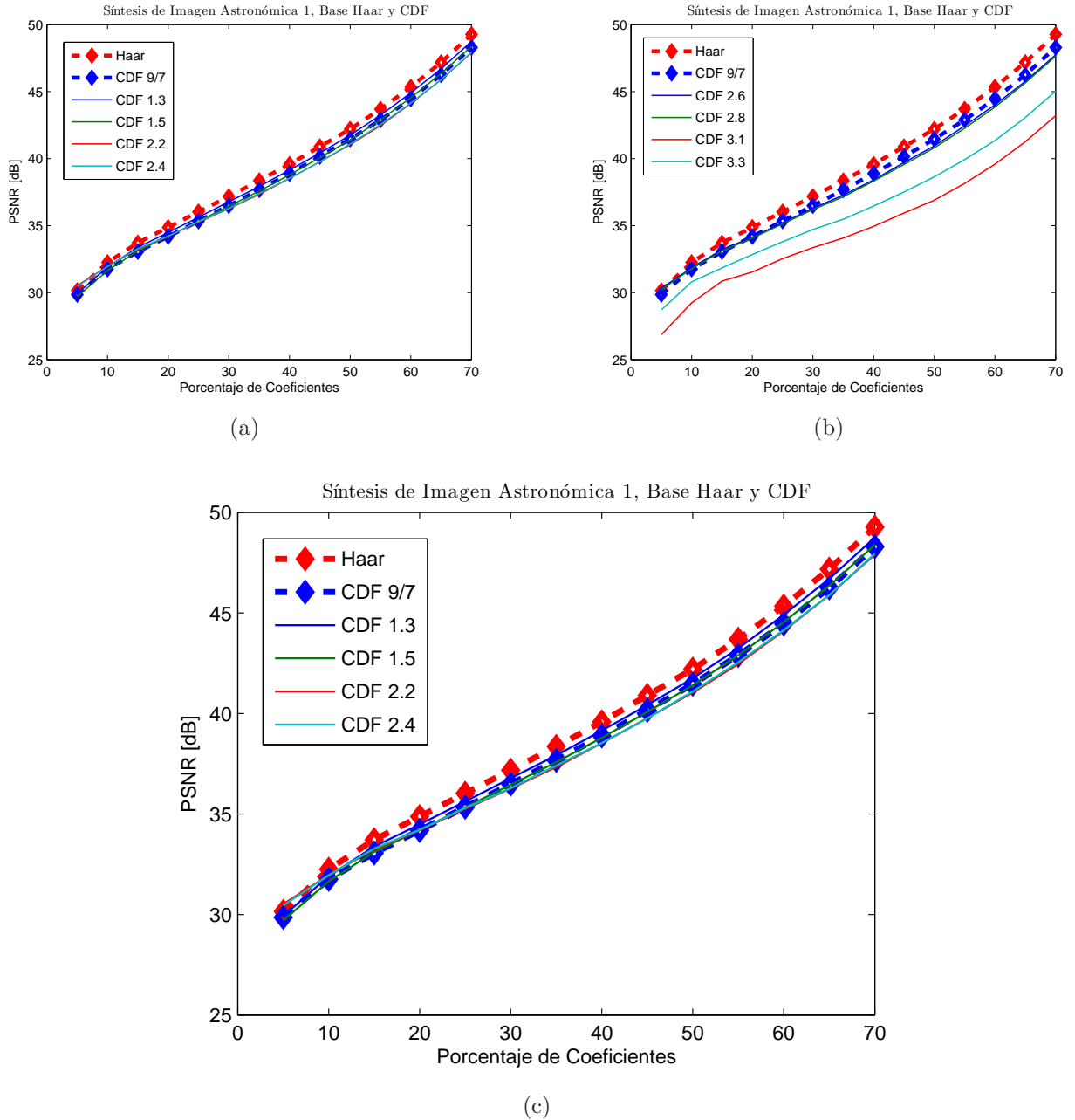


Figura 3.9: Prueba de compresibilidad, para una imagen astronómica, realizada para la base de Haar y las base pertenecientes a la familia bio-ortogonal CDF, la cual incluye a la base CDF 9/7, empleada por el estándar JPEG2000.

### 3.2. Implementación Wavelet de Haar en JPEG2000

Cuando ya se tiene conocimiento de la base más idónea para imágenes astronómicas, el paso lógico es intentar insertarla dentro de JPEG2000, de manera que las mejoras en compresibilidad se traduzcan en mejor calidad de la compresión para distintas tasas de bits.

### 3.2.1. Modificación JPEG2000

El algoritmo EBCOT [33] es el núcleo del funcionamiento del estándar JPEG2000, en él están integrados los módulos de selección de soporte, codificador aritmético y formación del *codestream*. Por consiguiente, las variaciones en el desempeño de la compresión efectuadas por este algoritmo se verán reflejadas en la aplicación de JPEG2000 [34].

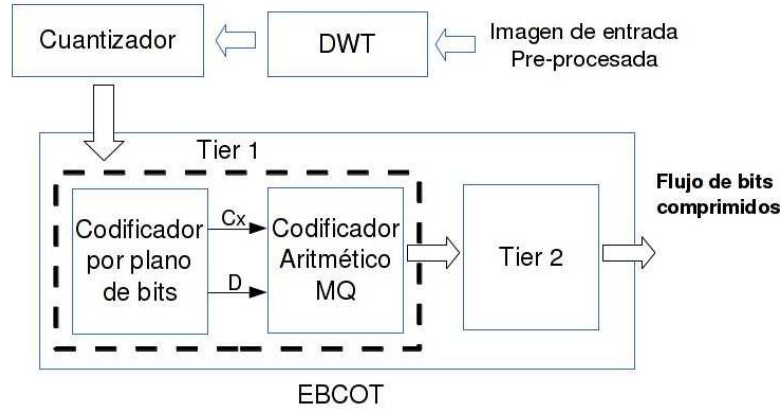


Figura 3.10: Diagrama de bloques del esquema de codificación de JPEG2000.

La modificación consiste en la implementación del filtro Haar en reemplazo del filtro -por defecto- CDF 9/7 en el módulo DWT. Debido a la naturaleza del algoritmo (ver Figura 3.10) y principalmente a su arquitectura modular, no es necesario efectuar una alteración más profunda, de manera que es posible obtener nuevas curvas de desempeño, las cuales serán comparadas con aquellas generadas con la configuración original.

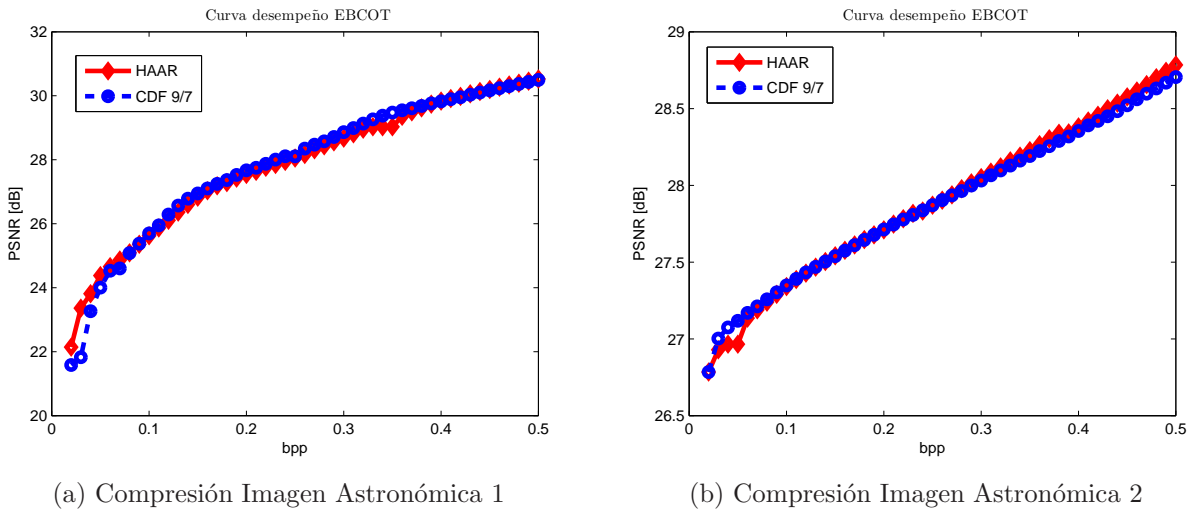
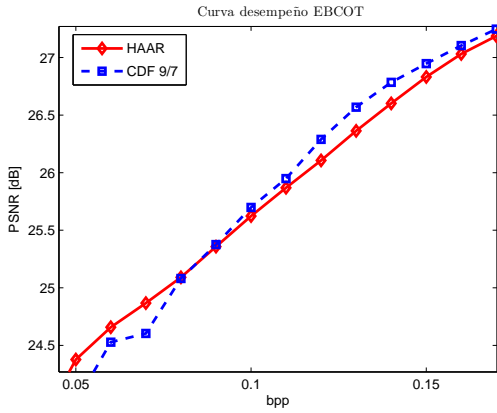
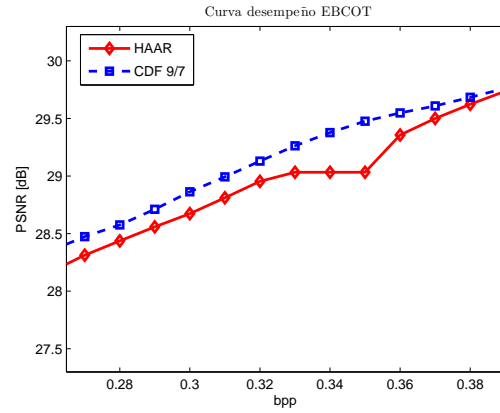


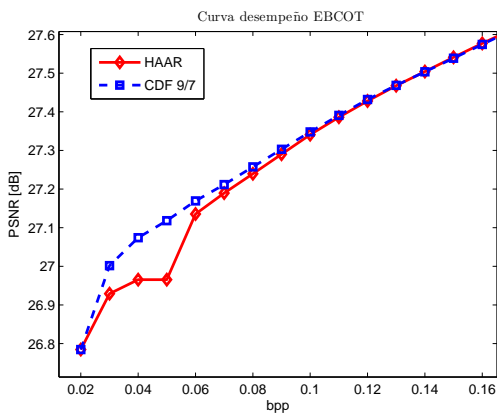
Figura 3.11: Curvas PSNR vs Bits por pixel obtenidas por la compresión de “Imagen Astronómica 1” e “Imagen Astronómica 2” mediante el algoritmo EBCOT modificado, contrastando la implementación de la wavelet de Haar con la versión original que ocupa la wavelet CDF 9/7.



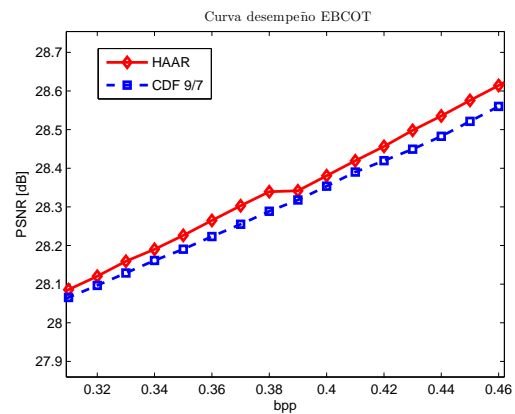
(a) Imagen Astronómica 1.



(b) Imagen Astronómica 1.



(c) Imagen Astronomica 2.



(d) Imagen Astronomica 2..

Figura 3.12: Acercamiento para distintos regímenes de las curvas PSNR vs Bits por pixel obtenidas por la compresión de “Imagen Astronómica 1” e “Imagen Astronómica 2” mediante el algoritmo EBCOT modificado, contrastando la implementación de la wavelet de Haar con la versión original que ocupa la wavelet CDF 9/7.

Al observar los resultados obtenidos, Figura 3.11 y Figura 3.12, se aprecia que, a pesar de la existencia de regímenes en los cuales el rendimiento de JPEG2000 con filtro de Haar es superior a la versión original (entre los 0,01 y 0,07 [bpp] y entre los 0,4 y 0,5 [bpp]), no se muestra de forma categórica que la compresibilidad sea el factor dominante en el desempeño del estándar JPEG2000. Es más, al contrastarlas con la Figura 3.9, asociada netamente a compresibilidad, las tendencias de las curvas no coinciden, pues en compresibilidad, se nota el dominio de Haar para cualquier régimen de porcentaje de coeficientes. Por lo tanto, del análisis de esta evidencia podemos concluir que una base con buenas propiedades de compactación de energía (respecto a las de la base CDF 9/7) no garantiza una mejor compresión en JPEG2000.

En efecto, se debe hacer un estudio más detallado del problema de compresión que no sólo considere como factor único la compresibilidad. En la siguiente sección haremos un análisis más completo del problema.

## Capítulo 4

# Análisis de dependencias entre coeficientes en el dominio Wavelet

De acuerdo a lo obtenido en el capítulo anterior, se concluye que la compresibilidad no es la característica más influyente en el desempeño de JPEG2000. Por ende, se debe profundizar el análisis de las imágenes astronómicas de entrada, de manera que sea posible inferir el por qué se producen las mermas de rendimiento en comparación a sus símiles naturales.

Para determinar las posibles causas de dicho comportamiento hay que remitirse al proceso de selección de soporte que realiza JPEG2000, explicado en los Anexos 6.2, consistente en la localización de coeficientes significativos (por plano de bits) y la posterior predicción efectuada sobre los coeficientes vecinos del mismo. Dicho de otra forma, el codificador asume si se está en la presencia de una magnitud relevante, es muy probable que en su vecindad exista un valor con propiedades similares.

En términos prácticos, los bits empleados en un esquema de compresión de imágenes  $\mathcal{B}$  se dividen en dos cantidades: los bits utilizados para codificar las magnitudes de los coeficientes significativos  $b_1$  y los bits empleados para codificar sus posiciones espaciales  $b_2$ , tal como se aprecia en la Ecuación 4.1

$$\mathcal{B} = b_1 + b_2. \quad (4.1)$$

Si la señal tiene  $N$  coeficientes, de los cuales sólo  $K$  son distintos a cero y cada magnitud se codifica con  $c$  bits, la expresión anterior puede reescribirse de la forma

$$\mathcal{B} = K \cdot c + K \log_2(N). \quad (4.2)$$

La Ecuación 4.2 muestra el caso más pesimista, es decir cuando la señal carece de estructura, de esa forma la cantidad de bits asignados para la codificación de las posiciones asciende a  $K \log_2(N)$ . Lo cual, ante un escenario de *bit-budget* limitado (como ocurre en la compresión con pérdidas) se deberá prescindir forzosamente de algunos coeficientes significativos con tal de no sobrepasar los recursos disponibles.

Supongamos que existe una estructura en el dominio transformado, esto quiere decir que la posición de los coeficientes significativos puede inferirse en base a ciertas propiedades de la señal. En el caso de JPEG2000, este proceso se realiza asumiendo que ante la presencia de

un coeficiente significativo, sus vecinos tienen altas probabilidades de serlo. Esto se conoce como *codificación predictiva* [35].

De esta forma, el análisis que se realizará cobra suma importancia, pues si se comprueba la existencia de dicha estructura, la cantidad de bits empleados para codificar las posiciones de los coeficientes será menor, lo que implica que cierto nivel de distorsión (o de PSNR) se alcanzará a una tasa menor de bits.

Para realizar el análisis de existencia de estructura en el dominio transformado se utilizará el enfoque desarrollado por Liu [36], quien emplea la *Información Mutua* [37] para determinar la dependencia entre los coeficientes. Se analizará la dependencia *intraescala*, coeficientes dentro de la misma subbanda en posiciones que rodean al valor estudiado.

## 4.1. Información Mutua

Esta medida de información permite conocer el grado de dependencia entre dos variables aleatorias. Sean  $X \in \mathbf{X}$  e  $Y \in \mathbf{Y}$  variables aleatorias, con una función densidad de probabilidad conjunta  $p(x, y)$ . La información mutua entre  $X$  e  $Y$  se define como

$$I(X; Y) = \int_{\mathbf{X}} \int_{\mathbf{Y}} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} dx dy, \quad (4.3)$$

o bien

$$I(X; Y) = D(p(x, y) \| p(x)p(y)), \quad (4.4)$$

donde  $D(\|)$  es la divergencia de *Kullback-Liebler* [38]. La información mutua es simétrica en  $X$  y en  $Y$ , no negativa y es igual a cero si y sólo si  $X$  e  $Y$  son independientes.

La información mutua  $I(X; Y)$  indica cuánta información contiene  $Y$  respecto a  $X$ . Visto desde la óptica de compresión de datos: si se codifica  $X$  a una precisión  $\Delta_X$  se necesitarán  $h(X) - \log(\Delta_X)$ <sup>1</sup>, pero si  $Y$  es conocido, la codificación de  $X$  se realizará con sólo  $h(X) - \log(\Delta_X) - I(X; Y)$  bits.

## 4.2. Caracterización de los coeficientes en el dominio wavelet

Para estudiar las dependencias de los coeficientes dentro de una misma subbanda, es decir la *dependencia intraescala*, es necesario definir previamente algunos elementos.

- $\mathbf{NX}$ , vecino del coeficiente  $X$ , tal como muestra la Figura 4.1.

---

<sup>1</sup>Donde la *entropía diferencial*  $h(x) = E_X [-\log(P(x))]$ .

- $I(X; \mathbf{NX})$ , información mutua, entre un coeficiente  $X$  y sus vecinos.

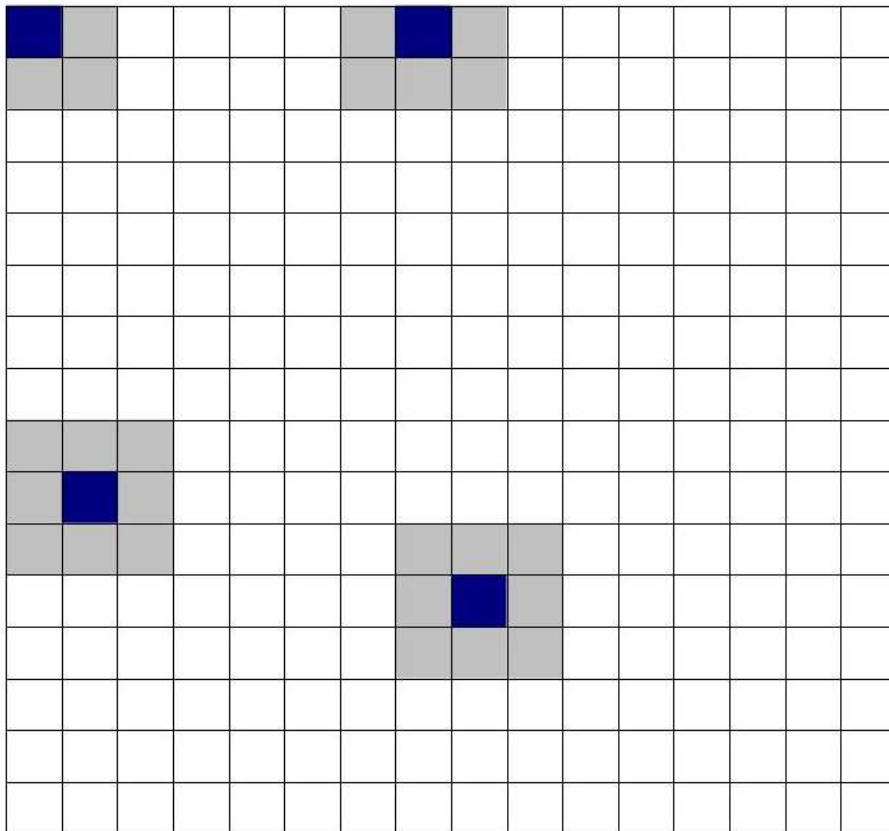


Figura 4.1: Dependencias intraescalas dentro de una subbanda: Se muestra la relación espacial entre los coeficientes  $X_i$  (en azul) y sus vecinos  $\mathbf{NX}_i$  (en gris). De acuerdo a la ubicación de los coeficientes, se determina el número de vecinos que éste tiene, siendo aquellos coeficientes ubicados en los bordes quienes cuentan con la menor cantidad de vecinos.

Un problema que surge es la estimación de  $I(X; \mathbf{NX})$ . Esto puede generar un problema de estimación de alta dimensión. Por ejemplo, si se consideran los ocho coeficientes vecinos de  $X$  en una subbanda, la función de densidad de probabilidad  $p(x, \mathbf{nx})$ , de nueve dimensiones, sería muy difícil de calcular. Para evitar este problema, se asumirá que la función  $T = f(\mathbf{NX})$  contiene la información del vecindario  $\mathbf{NX} = \{\mathbf{NX}_1, \dots, \mathbf{NX}_k\}$  de forma escalar, donde se asume que,

$$I(X; \mathbf{NX}) \approx I(X; T). \quad (4.5)$$

De esta forma, el problema se reduce a estimar la información mutua para una densidad de dos dimensiones. Sin embargo, se debe tener en cuenta que la expresión (4.5) no se cumple necesariamente. Por otro lado, sin importar la función  $f$  escogida, se tiene que

$$X \longrightarrow \mathbf{NX} \longrightarrow T = f(\mathbf{NX}), \quad (4.6)$$

es una cadena de Markov [39] y por ende la desigualdad de procesamiento de datos sostiene que

$$I(X; \mathbf{NX}) \geq I(X; T), \quad (4.7)$$

donde la igualdad sólo se obtiene si el estadístico  $T$  es suficiente con respecto a  $X$ .

### 4.2.1. Construcción de $f$

Para la formulación de  $f$  se tomará como referencia el modelo sugerido por Simoncelli [40], el cual indica que

$$T = f(NX) = \sum_i W_i |\mathbf{NX}_i|^2, \quad (4.8)$$

donde  $W_i$  es el peso de cada coeficiente y además

$$\frac{X}{\sqrt{T + \sigma_e^2}} \sim N(0, 1), \quad (4.9)$$

es independiente de  $T$ , con  $\sigma_e^2$  la varianza en el error de predicción en la subbanda. De acuerdo a lo anterior, las dependencias quedan caracterizadas principalmente por la función  $f$ , lo que gatilla que la Ecuación 4.5 se cumpla de forma aproximada.

Sólo resta definir el valor de cada  $W_i$ , los cuales por simplicidad serán considerados iguales a  $\frac{1}{8}$ .

## 4.3. Estimador de la Información Mutua

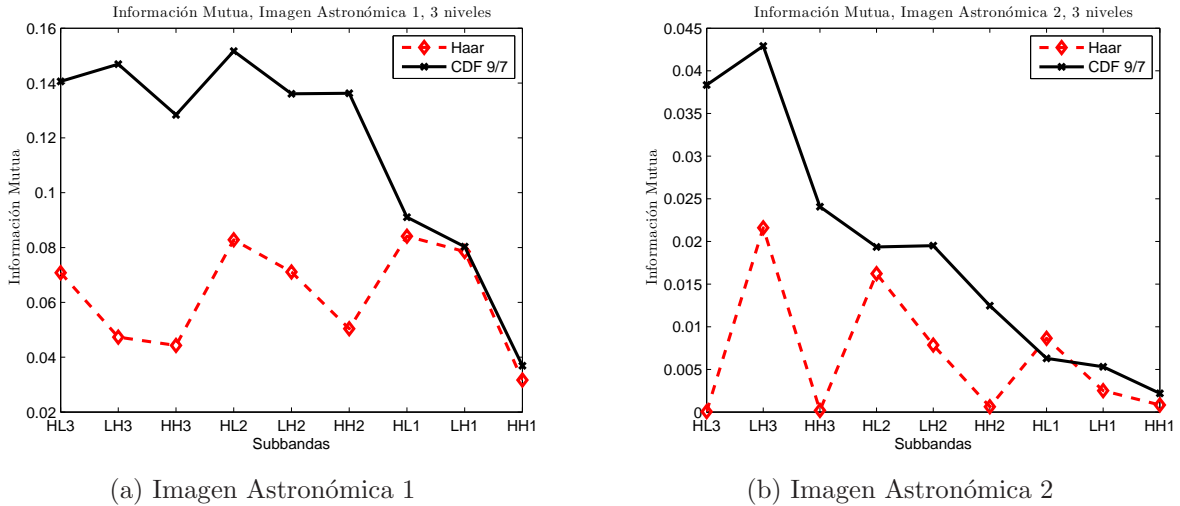
Para estimar la Información Mutua se empleará un método iterativo que particiona, de forma recursiva, los datos en celdas no uniformes. Este enfoque es dependiente de los datos y adaptable. Cada operación de particionado divide una celda en cuatro cuadrantes. Se toma la probabilidad conjunta  $(X, Y)$ , previamente normalizada, en cada celda. La partición se realiza en cada celda, a la “altura” de la *mediana* de  $X$  y de  $Y$ .

Si la contribución de un cuadrante al estimador  $\hat{I}(X; Y)$  está por debajo de un umbral  $\delta$ , previamente definido, se considera que  $X$  e  $Y$  son *aproximadamente independientes* dentro de ese cuadrante, por lo tanto ya no es posible seguir particionándolo. Por otro lado, si la contribución es superior a  $\delta$ , el cuadrante se vuelve a dividir. Los cuatro cuadrantes son sometidos al mismo proceso de manera independiente. El algoritmo se detiene cuando ya no quedan más celdas por particionar.

Los detalles del funcionamiento de este método pueden consultarse en el trabajo de Darbellay y Vajda [41].

## 4.4. Análisis de dependencias intraescala

Habiendo especificado las funciones de densidad de probabilidad para los coeficientes de cada subbanda (omitiendo la de aproximación), se estima la información mutua de las imágenes astronómicas estudiadas. Se considerarán las bases Haar y CDF 9/7.



(a) Imagen Astronómica 1

(b) Imagen Astronómica 2

Figura 4.2: Estimación de la Información Mutua, para las 9 subbandas (se excluye la banda de aproximación), entre un coeficiente en el dominio wavelet y sus vecinos. Se consideran las Imágenes Astronómicas 1 y 2, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad.

Observando los resultados obtenidos, Figura 4.2, la base de Haar presenta valores muy por debajo de su “competidora”. Por lo tanto, sus características impiden que se manifieste una persistencia en una vecindad en torno a un coeficiente significativo. Por esta razón, a pesar de ser la base idónea en términos de compresibilidad, sacrifica las dependencias intraescala, lo que repercute en una selección de soporte deficiente en el algoritmo JPEG2000 y como resultado de eso, un peor desempeño compresivo.

### Comparación con imágenes naturales

Si bien se constató que la base de Haar carece de las mismas propiedades de persistencia que la CDF 9/7, no se tiene conocimiento de qué magnitudes alcanza la información mutua en condiciones normales: es decir, sobre imágenes naturales en las base CDF 9/7. Por ende, se repetirá el ejercicio anterior para este tipo de señales. Los resultados son mostrados en la Figura 4.3.

Los valores de esta medida de información en imágenes naturales están muy por encima de su contraparte astronómica: por lo menos dos órdenes de magnitud en las subbandas de nivel 3. Esto puede otorgar una respuesta sobre la brecha en los desempeños entre los distintos tipos de morfología.



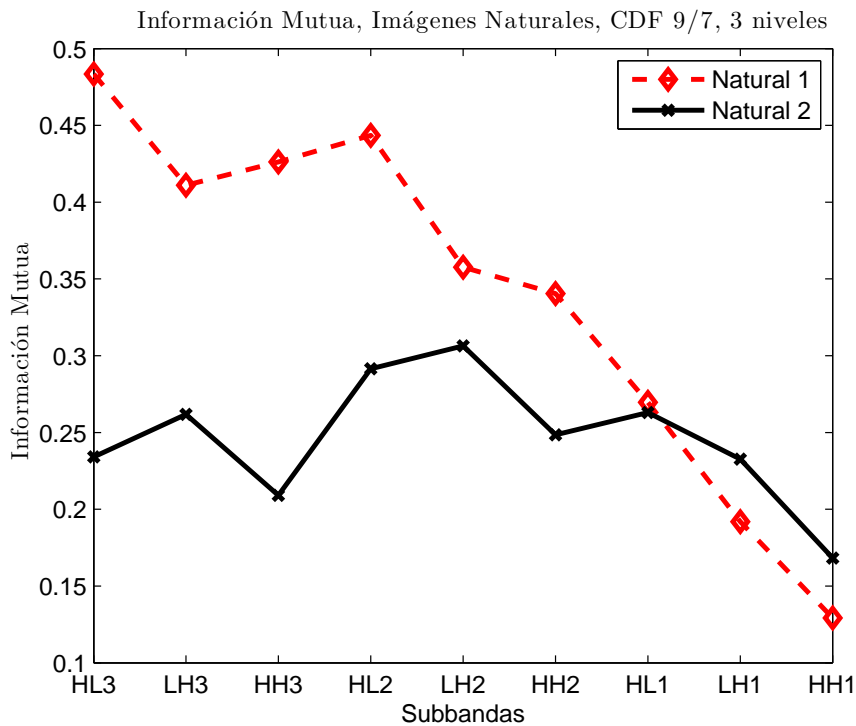


Figura 4.3: Estimación de la Información Mutua, para las 9 subbandas (se excluye la banda de aproximación), entre un coeficiente en el dominio wavelet y sus vecinos. Se consideran las Imágenes naturales de prueba, utilizando la base CDF 9/7, a 3 niveles de profundidad.

Un hecho llamativo que ayuda a confirmar la relevancia del problema de selección de soporte es que aquellas señales con mayor información mutua (en dominio wavelet CDF 9/7) también tienen un mejor desempeño al ser comprimidas vía JPEG2000, es decir, de acuerdo a lo visto anteriormente (Figura 3.6) se aprecia que el nivel de PSNR sigue el siguiente orden (decreciente):

1. Imagen Natural 1
2. Imagen Natural 2
3. Imagen Astronómica 1
4. Imagen Astronómica 2

A modo de resumen, la Figura 4.5 dan cuenta de esa situación.

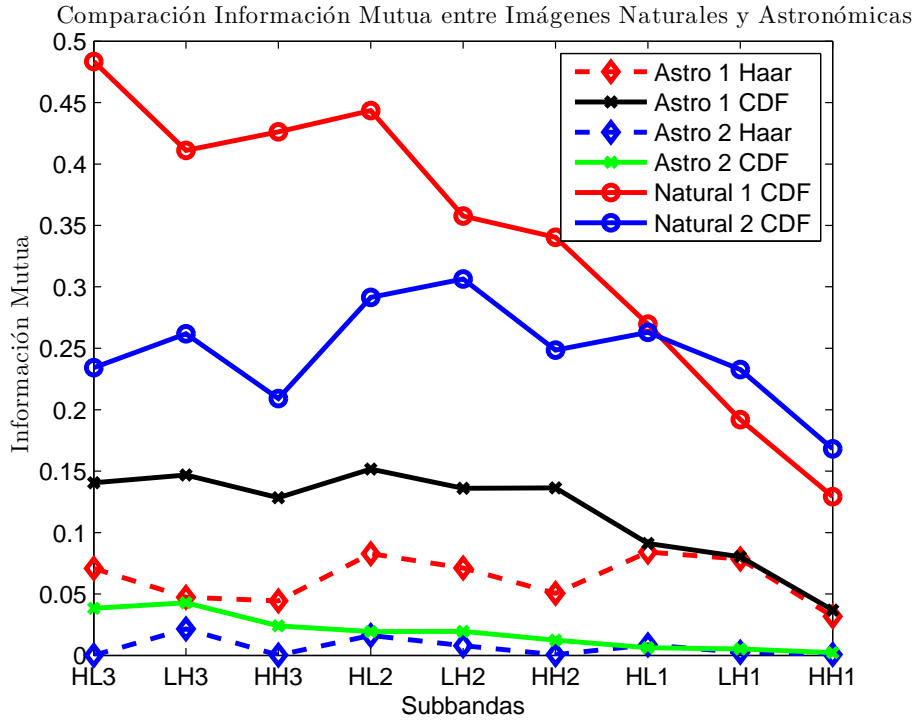
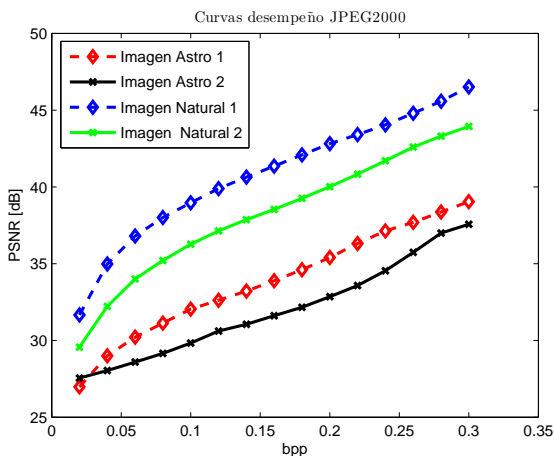
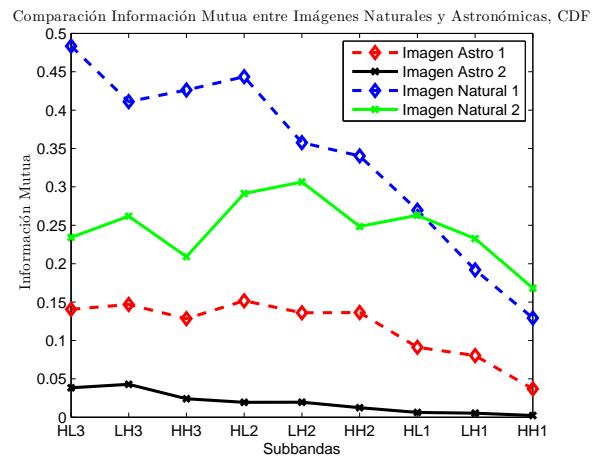


Figura 4.4: Estimación de la Información Mutua, para las 9 subbandas (se excluye la banda de aproximación), entre un coeficiente en el dominio wavelet y sus vecinos. Se consideran las imágenes naturales y astronómicas de prueba, utilizando la bases CDF 9/7 y Haar, a 3 niveles de profundidad.



(a) Desempeño JPEG2000



(b) Información Mutua

Figura 4.5: Paralelo entre el valor de la estimación de la Información Mutua y el desempeño en JPEG2000, medido en decibeles (PSNR).

# Capítulo 5

## División de Imágenes Astronómicas

### 5.1. Umbral en el dominio de los píxeles

Luego de comprobar experimentalmente que la implementación de la base de Haar en JPEG200 no garantiza una mejor compresión, es necesario idear un escenario que permita explotar de mejor manera las propiedades que entrega dicha base. Tomando como referencia el trabajo de Dong et al. [42] que propone dividir el contenido de las imágenes astronómicas y en base a eso efectuar la compresión, se aplicará una separación de las imágenes de entrada en dos componentes:

1. **Background**: Como el nombre lo indica, contiene los sectores de menor intensidad, los cuales pueden ser modelados como “ruido” [43].
2. **“Cartoon”**: Agrupa a los píxeles de mayor intensidad lo que, puesto en contexto astronómico, equivale a los sectores de interés del observador.

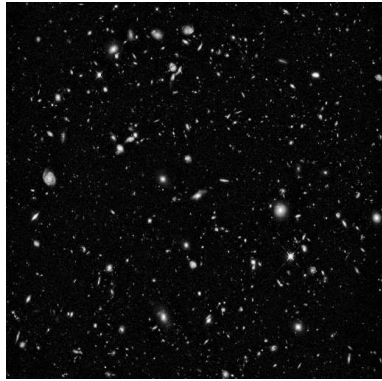
El proceso de separación de componentes se hará vía “thresholding” [44] en el dominio de los píxeles, para ello se establecerá un umbral y las zonas menores a él pertenecerán al *Background* mientras que las mayores serán etiquetadas como *Cartoon*. En la Figura 5.1 se muestra la imagen original y ambos componentes, para los umbrales 10, 30 y 60 respectivamente.<sup>1</sup>

### 5.2. Análisis del “Background”

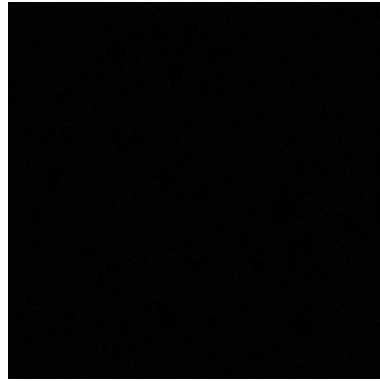
A continuación se repetirán los análisis hechos en los Capítulos 3 y 4: compresibilidad, información mutua e implementación en JPEG2000, con las bases CDF 9/7 y Haar, para las imágenes catalogadas como *Background*. Es necesario acotar que, al tener una morfología similar a la del ruido estacionario y sin memoria en imágenes, pueden presentar un desempeño muy pobre en compresión [25].

---

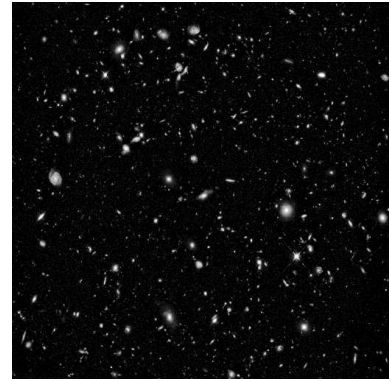
<sup>1</sup>Por simplicidad se usarán como muestra estos tres umbrales, sin embargo los experimentos fueron realizados para un total de 11 umbrales que van desde el 5 hasta el 60



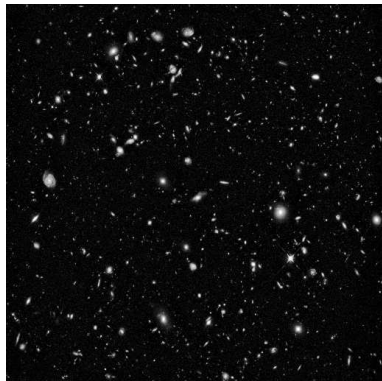
(a) Imagen Original.



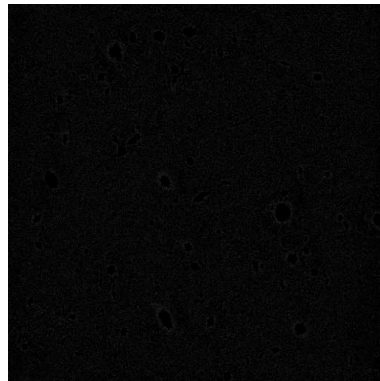
(b) Background, umbral 10.



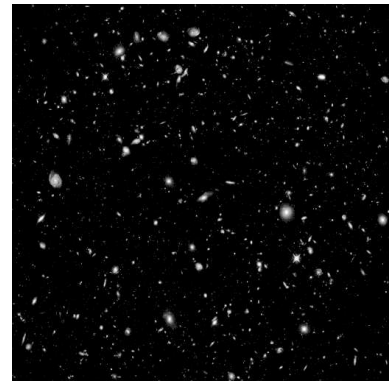
(c) Cartoon, umbral 10.



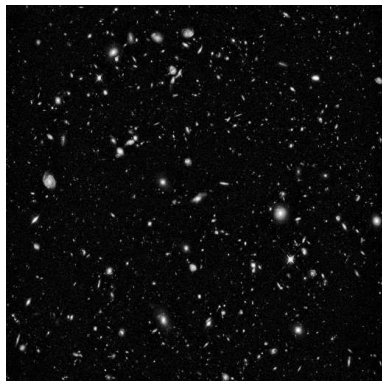
(d) Imagen Original.



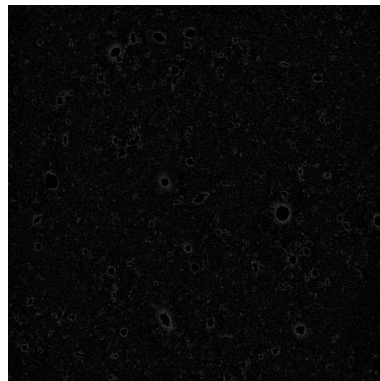
(e) Background, umbral 30.



(f) Cartoon, umbral 30.



(g) Imagen Original.



(h) Background, umbral 60.



(i) Cartoon, umbral 60.

Figura 5.1: División de Imagen Astronómica en Background y Cartoon para umbrales 10, 30 y 60.

## 5.2.1. Compresibilidad

### Umbral 10

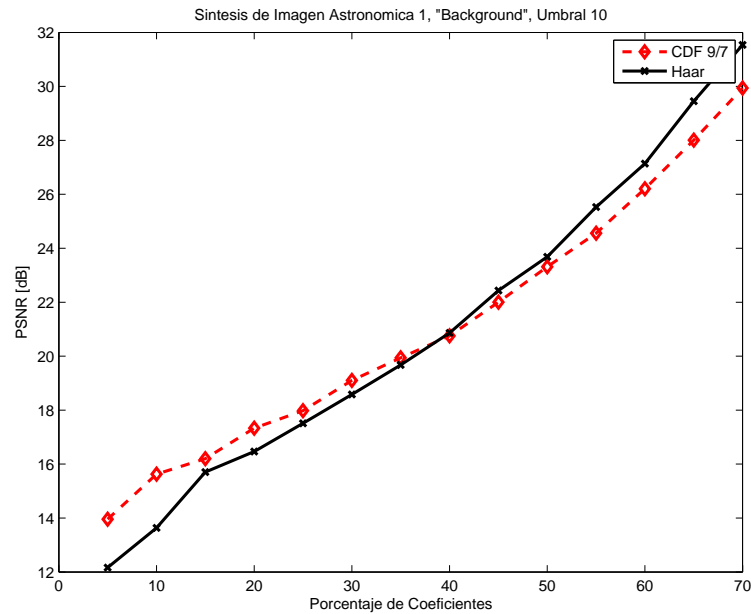


Figura 5.2: Análisis de compresibilidad para el "Background" de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 10.

### Umbral 30

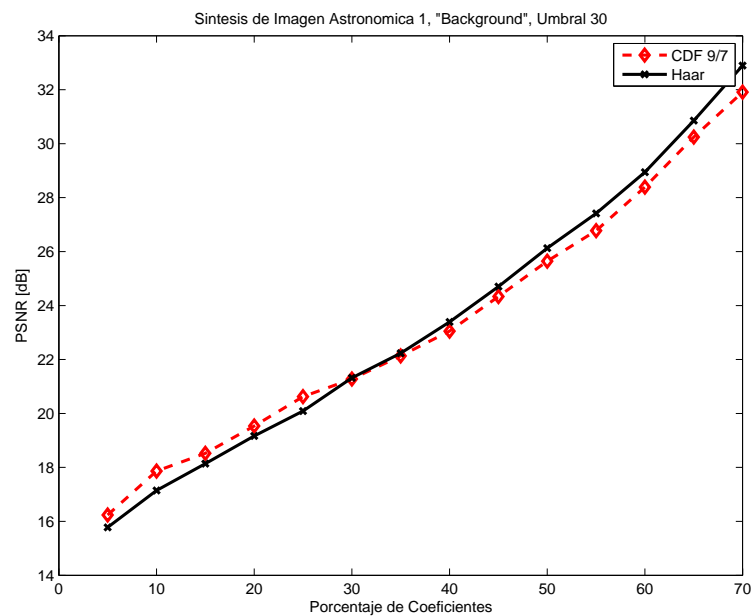


Figura 5.3: Análisis de compresibilidad para el "Background" de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 30.

## Umbral 60

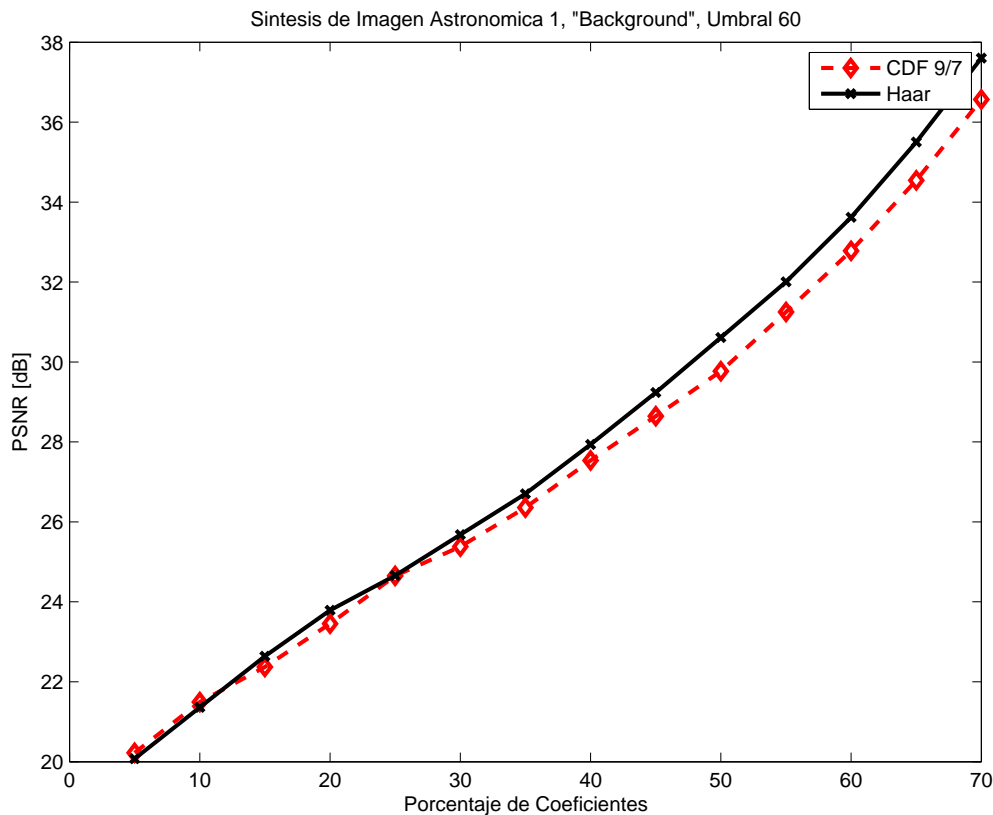


Figura 5.4: Análisis de compresibilidad para el “Background” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 60.

Se aprecia en los resultados expuestos que, a diferencia de lo ocurrido en la imagen original, el background no presenta una base dominante y, más importante, el error de reconstrucción es muy alto, incluso al efectuar la síntesis con un porcentaje elevado de coeficientes (70%). Si bien el argumento anterior no es el más determinante respecto a desempeño en un entorno JPEG2000, sí es decidor respecto a que estas bases no logran representar de forma compacta al background en el dominio transformado.

### 5.2.2. Dependencias Intraescala

A sabiendas de la importancia de la información mutua en la compresión de imágenes, se debe efectuar este análisis para cuantificar las relaciones ente coeficientes, para esta nueva dualidad: *Background* y *Cartoon*. Comenzando con el primero, para diferentes umbrales.

## Umbral 10

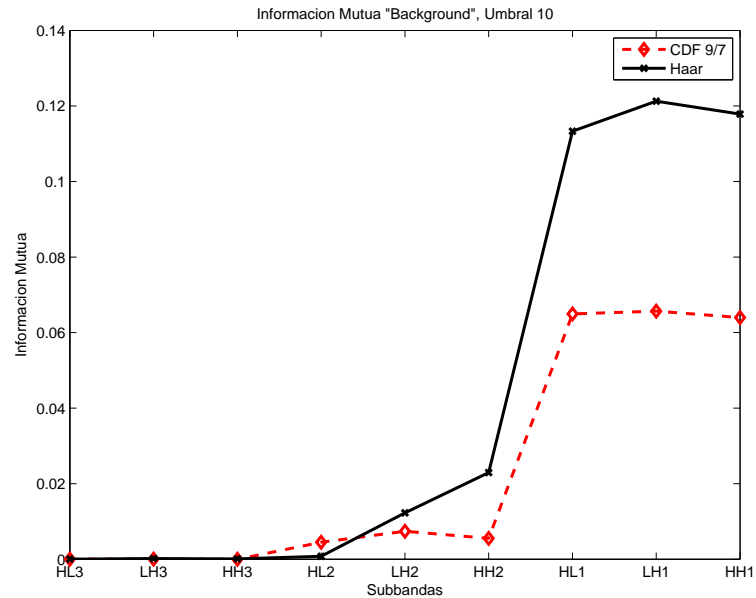


Figura 5.5: Estimación de la Información Mutua, para las 9 subbandas (se excluye la de aproximación), entre un coeficiente en el dominio wavelet y sus vecinos. Para el “Background” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 10.

## Umbral 30

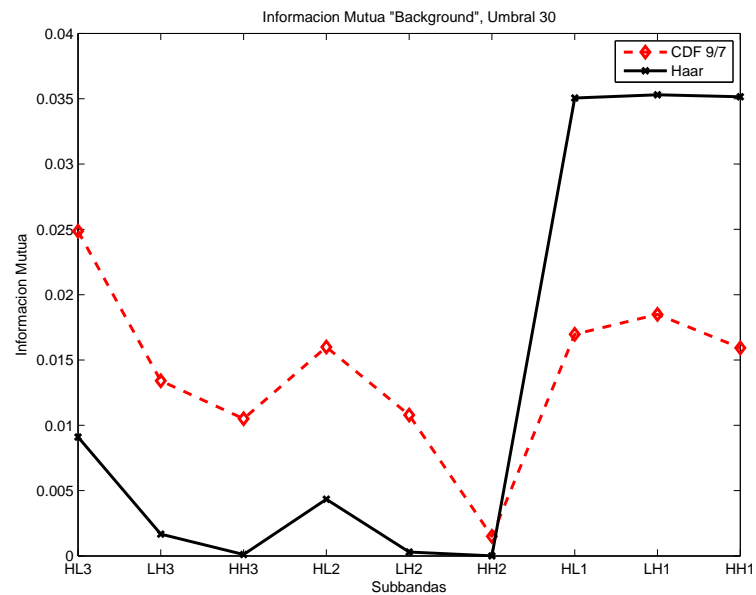


Figura 5.6: Estimación de la Información Mutua, para las 9 subbandas (se excluye la de aproximación), entre un coeficiente en el dominio wavelet y sus vecinos. Para el “Background” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 30.

## Umbral 60

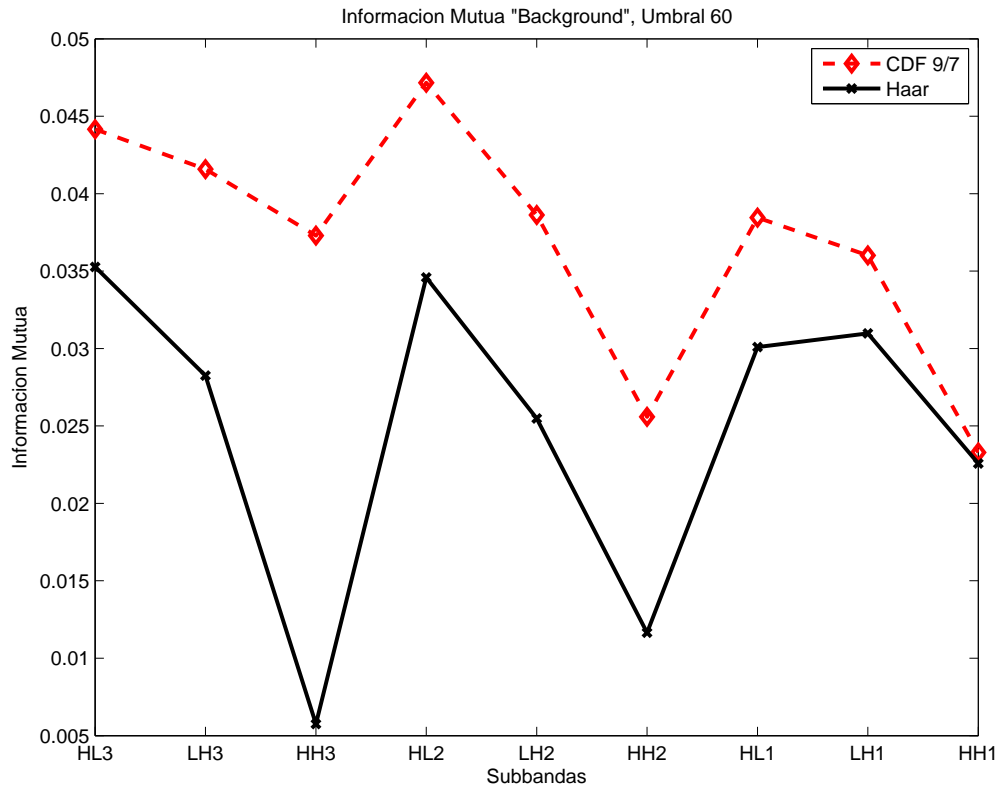


Figura 5.7: Estimación de la Información Mutua, para las 9 subbandas (se excluye la de aproximación), entre un coeficiente en el dominio wavelet y sus vecinos. Para el “Background” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 60.

De acuerdo a los datos expuestos por las Figuras 5.5, 5.6 y 5.7, los valores de la información mutua, en especial los medidos en la base de Haar, son de menor magnitud a los de la imagen completa. Lo cual no es sorprendente, pues la forma en la que se formuló la división genera discontinuidades significativas en el dominio original.

Tomando en cuenta los magros resultados obtenidos, tanto en compresibilidad como en las relaciones entre coeficientes, es posible pronosticar que la compresión de este componente con JPEG2000 no alcanzará niveles altos de PSNR.

### 5.2.3. JPEG2000

El paso final es someter al *Background* a compresión real, mediante el algoritmo EBCOT. De manera que las suposiciones puedan contrastarse con los resultados empíricos. Los cuales se aprecian en las Figuras 5.8, 5.9 y 5.10



## Umbral 10

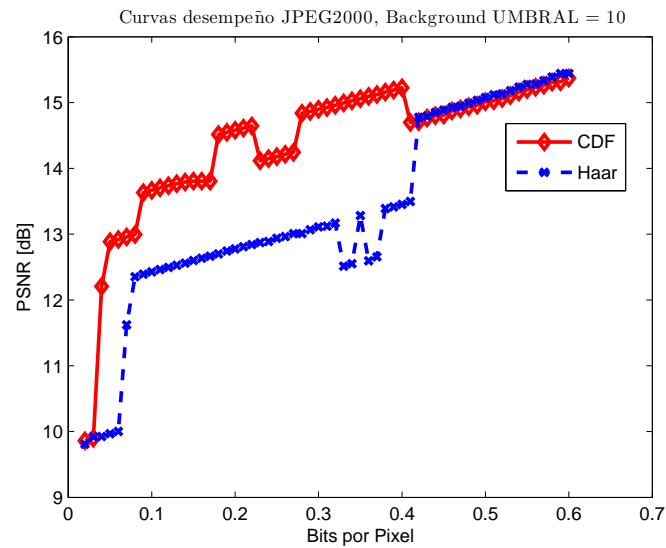


Figura 5.8: Curvas PSNR vs Bits por píxel obtenidas por la compresión del “Background” de la Imagen Astronómica 1, mediante el algoritmo EBCOT con bases CDF 9/7 y Haar, a 3 niveles de profundidad, para un umbral de valor 10.

## Umbral 30

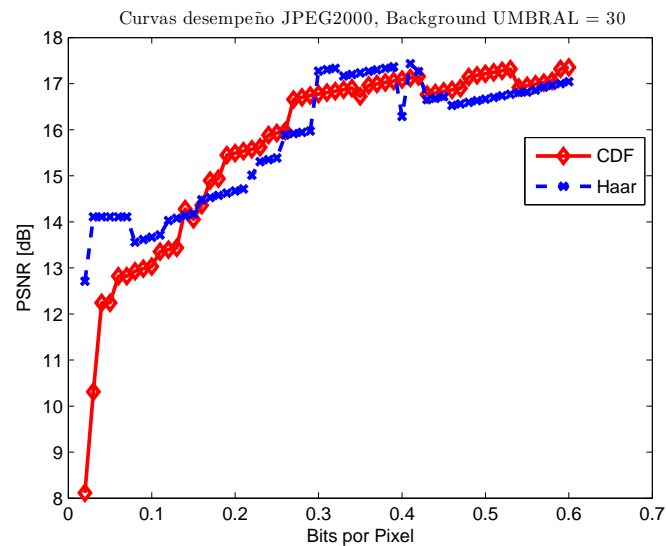


Figura 5.9: Curvas PSNR vs Bits por píxel obtenidas por la compresión del “Background” de la Imagen Astronómica 1, mediante el algoritmo EBCOT con bases CDF 9/7 y Haar, a 3 niveles de profundidad, para un umbral de valor 30.

## Umbral 60

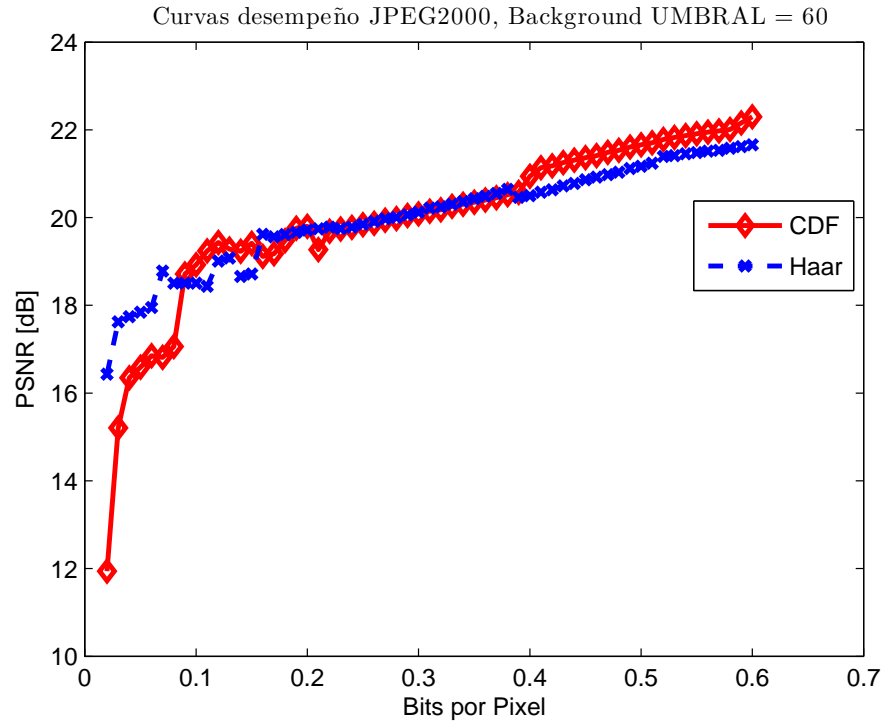


Figura 5.10: Curvas PSNR vs Bits por pixel obtenidas por la compresión del “Background” de la Imagen Astronómica 1, mediante el algoritmo EBCOT con bases CDF 9/7 y Haar, a 3 niveles de profundidad, para un umbral de valor 60.

La conclusión que se puede establecer en este punto, es la no existencia de una base dominante, ya que ambas poseen distintos regímenes en los cuales se alternan el liderazgo.

### 5.3. Análisis del “Cartoon”

De manera análoga se repetirán las rutinas de análisis para los detalles o *cartoon* de las imágenes astronómicas. A partir de lo obtenido será posible concluir si existe un enfoque conveniente para señales de alta frecuencia en lo concerniente a compresión vía JPEG2000 y de existir, qué rangos de operación permiten obtener ventajas en desempeño.

#### 5.3.1. Compresibilidad

Como ya se ha señalado en secciones anteriores, se efectuará el análisis de compresibilidad para el *Cartoon*, sin variar las condiciones ya empleadas, tanto de bases como de porcentaje de coeficientes a utilizar.

## Umbral 10

Para un umbral de 10, los resultados obtenidos son los siguientes

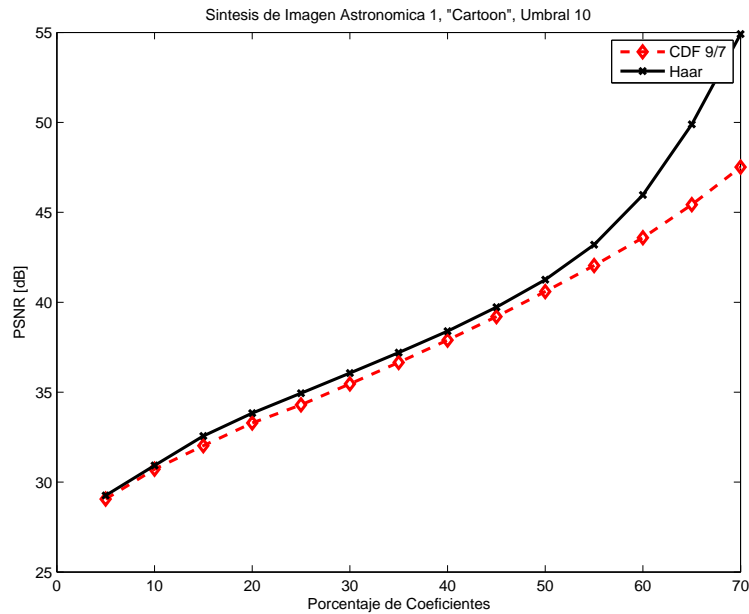


Figura 5.11: Análisis de compresibilidad para el “Cartoon” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 10.

De acuerdo a la Figura 5.11 no se logra apreciar una gran diferencia con los resultados de la imagen original, lo que es esperable, pues para un umbral tan bajo, la imagen a comprimir no difiere de forma cualitativa a la original.

## Umbral 30

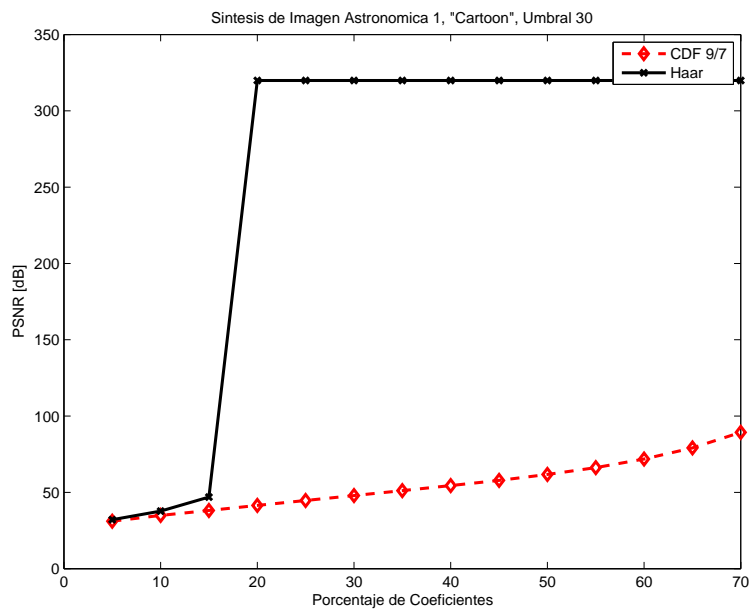


Figura 5.12: Análisis de compresibilidad para el “Cartoon” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 30.

De acuerdo a lo mostrado en la Figura 5.12, se produce un hecho digno de mención, para un porcentaje de coeficientes igual o mayor al 20% los niveles de reconstrucción son muy elevados, llegando a un valor máximo invariable. Esto sugiere que se ha alcanzado reconstrucción con un error mínimo para la base de Haar.

## Umbral 60

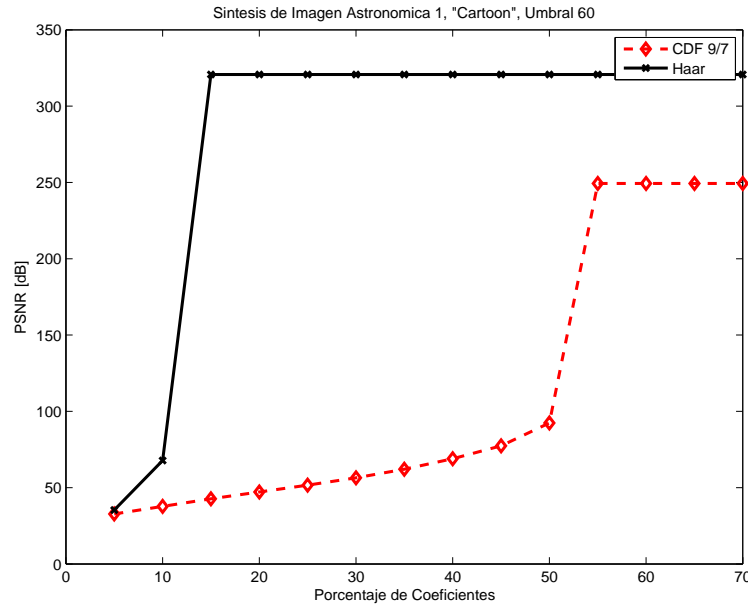


Figura 5.13: Análisis de compresibilidad para el “Cartoon” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 60.

De acuerdo a lo mostrado en la Figura 5.13, al igual que para un umbral de 30, se alcanza reconstrucción perfecta con la wavelet de Haar, con la diferencia de que ésta se presenta a partir de un 15% de coeficientes significativos.

### 5.3.2. Dependencias Intraescala

De la misma forma en que se analizó la dependencia entre coeficientes en el dominio wavelet, por medio de la Información Mutua, para el *Cartoon* se efectuará el mismo análisis que para el *Background*.

Los resultados obtenidos, para un umbral igual a 10, 30 y 60, se presentan a continuación.

## Umbral 10

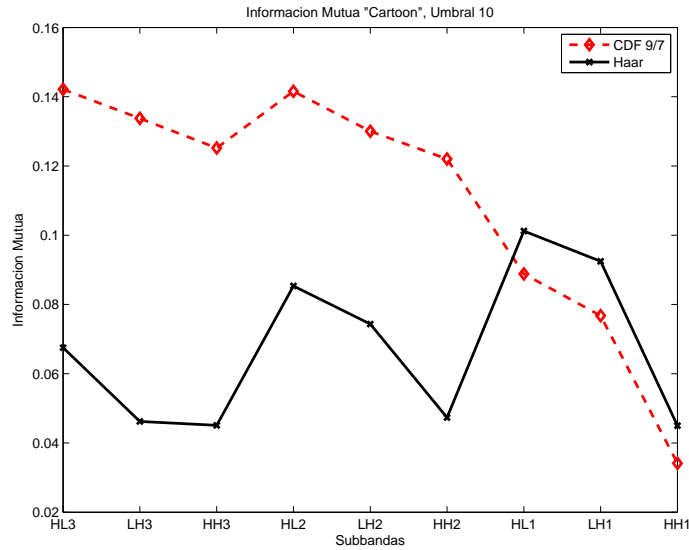


Figura 5.14: Estimación de la Información Mutua, para las 9 subbandas (se excluye la de aproximación), entre un coeficiente en el dominio wavelet y sus vecinos. Para el “Cartoon” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 10.

## Umbral 30

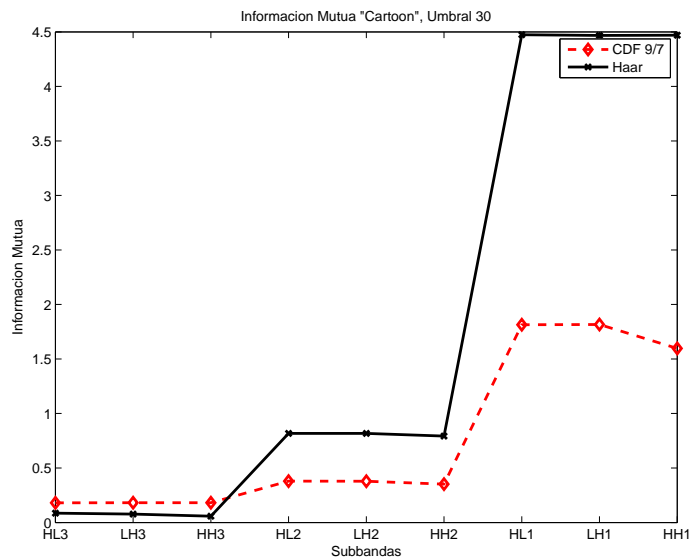


Figura 5.15: Estimación de la Información Mutua, para las 9 subbandas (se excluye la de aproximación), entre un coeficiente en el dominio wavelet y sus vecinos. Para el “Cartoon” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 30.

## Umbral 60

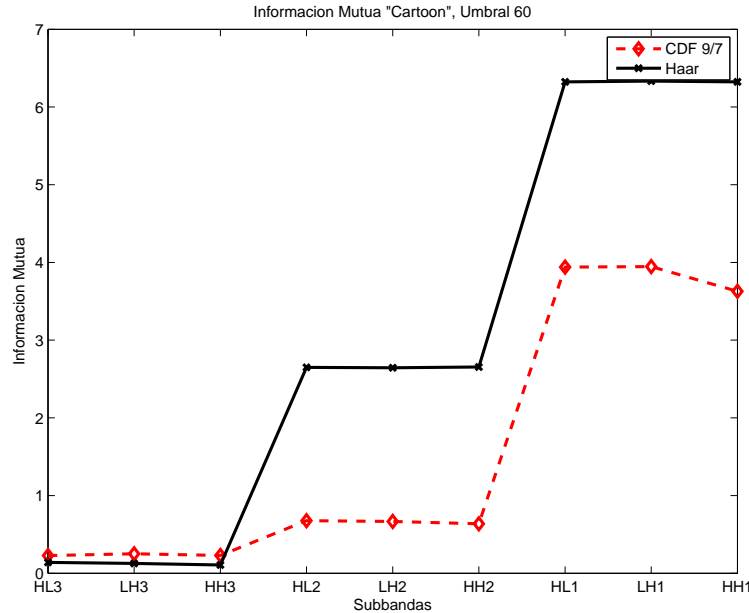


Figura 5.16: Estimación de la Información Mutua, para las 9 subbandas (se excluye la de aproximación), entre un coeficiente en el dominio wavelet y sus vecinos. Para el “Cartoon” de la Imagen Astronómica 1, utilizando las bases CDF 9/7 y Haar a 3 niveles de profundidad y un umbral de valor 60.

De acuerdo a los datos expuestos por las Figuras 5.14, 5.15 y 5.16, los valores de la información mutua varían en función del umbral escogido. Por ejemplo, para un umbral igual a 10 la dependencia intraescala se mantiene dominada por la base CDF 9/7.

En el caso de los umbrales 30 y 60, Figuras 5.15 y 5.16, el panorama es diametralmente distinto, pues se aprecia una dominancia de la base de Haar en el segundo nivel de profundidad (subbandas HH2, HL2 y LH2), donde el indicador de la información mutua alcanza valores muy altos, lo que sugiere una total dependencia en las distribuciones de probabilidad de los coeficientes centrales y la vecindad que los rodea.

Tomando en cuenta los resultados obtenidos, se podría esperar que la base de Haar alcance desempeños superiores a la CDF 9/7 en JPEG2000. Esto será analizado en la sección siguiente.

### 5.3.3. JPEG2000

Sólo resta someter al *Cartoon* a compresión real, mediante el algoritmo EBCOT. De manera que los análisis previos puedan contrastarse con los resultados empíricos. Los resultados se observan en las Figuras 5.17, 5.18 y 5.19

## Umbral 10

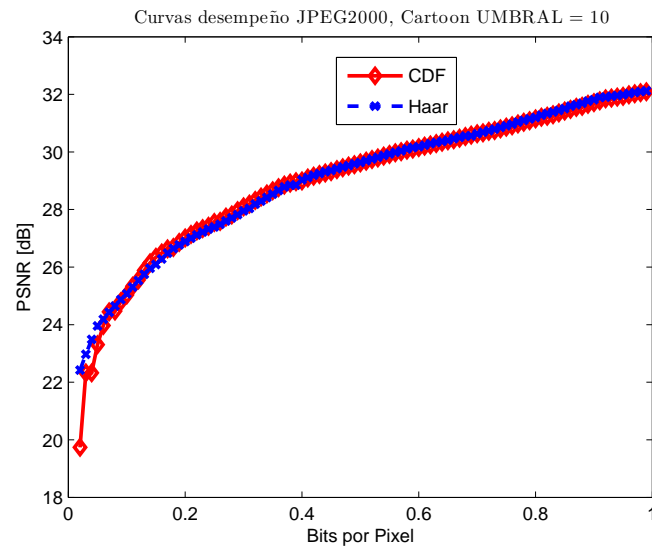


Figura 5.17: Curvas PSNR vs Bits por pixel obtenidas por la compresión del “Cartoon” de la Imagen Astronómica 1, mediante el algoritmo EBCOT con bases CDF 9/7 y Haar, a 3 niveles de profundidad, para un umbral de valor 10.

## Umbral 30

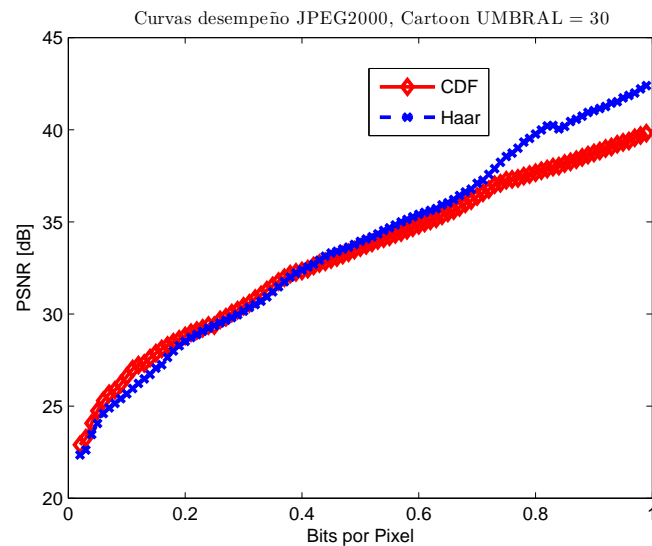


Figura 5.18: Curvas PSNR vs Bits por pixel obtenidas por la compresión del “Cartoon” de la Imagen Astronómica 1, mediante el algoritmo EBCOT con bases CDF 9/7 y Haar, a 3 niveles de profundidad, para un umbral de valor 30.

## Umbral 60

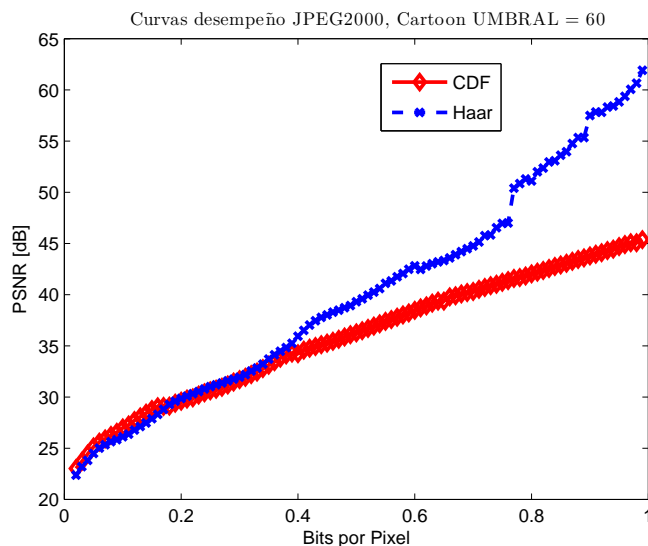


Figura 5.19: Curvas PSNR vs Bits por pixel obtenidas por la compresión del “Cartoon” de la Imagen Astronómica 1, mediante el algoritmo EBCOT con bases CDF 9/7 y Haar, a 3 niveles de profundidad, para un umbral de valor 60.

En esta ocasión, para un umbral de 30 y 60 se confirma que, para un tasa superior a 0,4 bits por pixel, la base de Haar entrega mejores resultados en compresión. Tomando en cuenta las secciones previas, y contrastando los análisis de compresibilidad e información mutua, se desprende que el mejor desempeño es generado principalmente por la reconstrucción perfecta que se logra con un porcentaje de coeficientes reducido. Esto se fundamenta en el hecho de que para las subbandas que aportan la mayor cantidad de coeficientes significativos, es decir en la tercera profundidad: HL3, LH3 y HH3, la información mutua de la base de Haar no logra superar a la de CDF 9/7. Es decir, para dichos umbrales se logra que la imagen sea perfectamente *sparse* en el dominio transformado, lo que facilita la selección de soporte por parte del codificador, pasando a un segundo plano la ubicación especial de dichos valores.

Habiendo encontrado un escenario donde la compresión de imágenes astronómicas, con la base de Haar en JPEG2000, alcanza un desempeño mayor al de la configuración original del algoritmo, se hace necesario establecer los usos dónde éste pueda tener aplicaciones prácticas.

Considerando la construcción de esta dualidad *Background-Cartoon*, en la cual se prescinde de los pixeles de menor intensidad (caso *Cartoon*) de una forma drástica, la imagen resultante es una versión truncada de la imagen astronómica original, lo que gatilla en una compresión más efectiva, pero transando la exactitud y detalles de aquellos elementos menos perceptibles. Es decir, en aplicaciones en que sólo la ubicación de los elementos más visibles sea necesaria hay chances de emplear la solución propuesta en este trabajo.



# Capítulo 6

## Conclusiones y Trabajo Futuro

### 6.1. Conclusiones

En esta tesis se estudió la compresión de imágenes astronómicas, en particular el desempeño que éstas presentan al ser comprimidas bajo el estándar JPEG2000. La observación de las curvas de tasa distorsión muestran una merma, no despreciable en términos de PSNR, en la compresión a igual tasa de bits por parte de estas señales en comparación a las imágenes naturales. Tomando en cuenta lo anterior, se buscó mitigar este efecto realizando modificaciones en el algoritmo JPEG2000, de manera que éste último se adapte a las características de las señales de estudio.

Se focalizó el estudio en el bloque de Transformación del Tier-1 de JPEG2000, lo que en la práctica se tradujo en la búsqueda de un filtro más idóneo, de acuerdo a criterios establecidos, para la naturaleza del problema.

El criterio de evaluación de las bases empleado fue el de mayor compresibilidad, es decir, aquel filtro que permita reconstruir la imagen, con el menor error posible, con un porcentaje determinado de coeficientes. Es decir, una imagen sintetizada al 30 % sólo utiliza dicha cantidad de valores en el dominio transformado, haciendo los restantes iguales a cero. De acuerdo a este análisis, la base de mejor desempeño en imágenes astronómicas es la wavelet de Haar.

Habiendo determinado la mejor base, se implementó dentro de JPEG2000 con la noción de que una mayor compresibilidad se traduciría en un desempeño superior al obtenido mediante la base CDF 9/7. Sin embargo, los resultados refutaron esta hipótesis lo que llevó a buscar explicaciones sobre el por qué la base de Haar no logra traducir su ganancia en compresibilidad en una ganancia en la compresión.

Lo anterior determina que el criterio elegido no es el más decisivo al momento de optar por una base, porque omite un aspecto clave del estándar: la selección de coeficientes significativos y su posterior codificación. Entonces se debe recurrir a alguna herramienta que permita cuantificar las relaciones entre los coeficientes de cada subbanda, la cual corresponde a la información mutua. El análisis posterior mostró que hay una correspondencia entre los niveles de información mutua por subbanda y el desempeño en JPEG2000, es decir, aquella imagen que presente una mayor dependencia intraescala entre sus coeficientes tendrá un nivel más

alto de PSNR a la misma tasa de bits.

No obstante, se buscó un escenario en el cual la wavelet de Haar pueda proporcionar un mejor desempeño, para ello se dividió cada imagen astronómica en dos partes: el fondo o *background* y los detalles o *cartoon*. La separación fue realizada vía *thresholding* en el dominio de los píxeles. Esto quiere decir que aquellos objetos de mayor intensidad pertenecerán a los detalles, mientras que las tonalidades tenues serán asociadas al fondo.

Al efectuar los análisis de compresibilidad y dependencias sobre cada división de las imágenes, se puede apreciar que a partir de umbrales superiores a 30, es posible sintetizar casi de manera perfecta la imagen con un porcentaje reducido de coeficientes, en muchos casos el error de reconstrucción es igual a cero (PSNR infinito). Ante ese nuevo escenario, se volvieron a comprimir las imágenes en JPEG2000 implementado con filtros Haar y esta vez se obtuvieron desempeños mejores, en particular en rangos superiores a los 0,4 bits por pixel.

Ante la evidencia obtenida se puede concluir que la hipótesis desarrollada en esta tesis funciona en aplicaciones donde se pueda prescindir de un porcentaje de los elementos de la imagen, en especial de aquellos sectores menos visibles, o bien donde se priorice detectar la ubicación de los cuerpos más relevantes de la imagen estudiada, como el caso del problema de astronomía.

## 6.2. Trabajo Futuro

Las mejoras o líneas de trabajo que se pueden desprender del trabajo realizado son

- Modificación del resto de los bloques que componen JPEG2000.
- Estudio de la compresión y/o caracterización del ruido de fondo de una imagen astronómica.
- Diseño de filtros óptimos, para la morfología de señal estudiada, que conjuguen los criterios de compresibilidad y de dependencias entre coeficientes, de manera que puedan ser implementados en aplicaciones de compresión de imágenes.
- Análisis del caso de compresión sin pérdidas y evaluar la validez de los criterios de selección discutidos en esta tesis.

# Bibliografía

- [1] C. E. Shannon, “A mathematical theory of communication,” *Bell system technical journal*, vol. 27, 1948.
- [2] D. Pan, “A tutorial on mpeg/audio compression,” *IEEE MultiMedia*, vol. 2, pp. 60–74, 1995.
- [3] A. Gersho, “Advances in speech and audio compression,” *Proceedings of the IEEE*, vol. 82, no. 6, pp. 900–918, 1994.
- [4] M. Hans and R. Schafer, “Lossless compression of digital audio,” *Signal Processing Magazine, IEEE*, vol. 18, no. 4, pp. 21–32, 2001.
- [5] D. Le Gall, “Mpeg: a video compression standard for multimedia applications,” *Commun. ACM*, vol. 34, no. 4, pp. 46–58, Apr. 1991. [Online]. Available: <http://doi.acm.org/10.1145/103085.103090>
- [6] I. Richardson, *H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia*, ser. H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia. Wiley, 2004. [Online]. Available: <http://books.google.cl/books?id=jLL1xqOry2QC>
- [7] J. Ziv and A. Lempel, “A universal algorithm for sequential data compression,” *Information Theory, IEEE Transactions on*, vol. 23, no. 3, pp. 337–343, 1977.
- [8] —, “Compression of individual sequences via variable-rate coding,” *Information Theory, IEEE Transactions on*, vol. 24, no. 5, pp. 530–536, 1978.
- [9] R. A. DeVore, B. Jawerth, and B. J. Lucier, “Image compression through wavelet transform coding,” *Information Theory, IEEE Transactions on*, vol. 38, no. 2, pp. 719–746, 1992.
- [10] A. Gersho and R. M. Gray, *Vector quantization and signal compression*. Springer, 1992.
- [11] T. Berger, *Rate-Distortion Theory*. Wiley Online Library, 1971.
- [12] G. K. Wallace, “The jpeg still picture compression standard,” *Communications of the ACM*, vol. 34, no. 4, pp. 30–44, 1991.
- [13] N. Ahmed, T. Natarajan, and K. R. Rao, “Discrete cosine transform,” *Computers, IEEE Transactions on*, vol. 100, no. 1, pp. 90–93, 1974.

- [14] P. Schelkens, A. Skodras, and T. Ebrahimi, *The JPEG 2000 suite*. Wiley. com, 2009, vol. 15.
- [15] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, “Image coding using wavelet transform,” *Image Processing, IEEE Transactions on*, vol. 1, no. 2, pp. 205–220, 1992.
- [16] J. M. Shapiro, “Embedded image coding using zerotrees of wavelet coefficients,” *Signal Processing, IEEE Transactions on*, vol. 41, no. 12, pp. 3445–3462, 1993.
- [17] A. Said and W. A. Pearlman, “A new, fast, and efficient image codec based on set partitioning in hierarchical trees,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 6, no. 3, pp. 243–250, 1996.
- [18] M. Rabbani, “Jpeg2000: Image compression fundamentals, standards and practice,” 2002.
- [19] J.-L. Starck and F. Murtagh, “Astronomical image and signal processing: looking at noise, information and scale,” *Signal Processing Magazine, IEEE*, vol. 18, no. 2, pp. 30–40, 2001.
- [20] A. Fruchter and R. Hook, “Drizzle: a method for the linear reconstruction of undersampled images,” *Publications of the Astronomical Society of the Pacific*, vol. 114, no. 792, pp. 144–152, 2002.
- [21] S. M. Jefferies and J. C. Christou, “Restoration of astronomical images by iterative blind deconvolution,” *The Astrophysical Journal*, vol. 415, p. 862, 1993.
- [22] A. Fruchter and R. N. Hook, “Novel image-reconstruction method applied to deep hubble space telescope images,” in *Optical Science, Engineering and Instrumentation’97*. International Society for Optics and Photonics, 1997, pp. 120–125.
- [23] R. Puetter, T. Gosnell, and A. Yahil, “Digital image reconstruction: deblurring and denoising,” *Annu. Rev. Astron. Astrophys.*, vol. 43, pp. 139–194, 2005.
- [24] M. Louys, J. Starck, S. Mei, F. Bonnarel, and F. Murtagh, “Astronomical image compression,” in *Astronomy and Astrophysics, Suppl. Ser.* Citeseer, 1999.
- [25] J.-L. Starck, F. Murtagh, B. Pirenne, and M. Albrecht, “Astronomical image compression based on noise suppression,” *Publications of the Astronomical Society of the Pacific*, pp. 446–455, 1996.
- [26] J.-L. Starck and F. Murtagh, “Handbook of astronomical data analysis,” 2002.
- [27] D. Wells, E. Greisen, and R. Harten, “Fits-a flexible image transport system,” *Astronomy and Astrophysics Supplement Series*, vol. 44, p. 363, 1981.
- [28] R. White, M. Postman, and M. Lattanzi, “Compression of the guide star digitised schmidt plates,” in *Digitised Optical Sky Surveys*. Springer, 1992, pp. 167–175.
- [29] M. Vetterli and J. Kovačević, *Wavelets and subband coding*. Prentice Hall PTR Englewood Cliffs, New Jersey, 1995, vol. 87.

- [30] S. Mallat, *A Wavelet Tour of Signal Processing*, 3rd ed. Academic Press, 2009.
- [31] I. H. Witten, R. M. Neal, and J. G. Cleary, “Arithmetic coding for data compression,” *Communications of the ACM*, vol. 30, no. 6, pp. 520–540, 1987.
- [32] A. Cohen, I. Daubechies, and J.-C. Feauveau, “Biorthogonal bases of compactly supported wavelets,” *Communications on pure and applied mathematics*, vol. 45, no. 5, pp. 485–560, 1992.
- [33] D. Taubman, “High performance scalable image compression with ebcot,” *Image Processing, IEEE transactions on*, vol. 9, no. 7, pp. 1158–1170, 2000.
- [34] A. Skodras, C. Christopoulos, and T. Ebrahimi, “The jpeg 2000 still image compression standard,” *Signal Processing Magazine, IEEE*, vol. 18, no. 5, pp. 36–58, 2001.
- [35] A. Said and W. A. Pearlman, “An image multiresolution representation for lossless and lossy compression,” *Image Processing, IEEE Transactions on*, vol. 5, no. 9, pp. 1303–1310, 1996.
- [36] J. Liu and P. Moulin, “Information-theoretic analysis of interscale and intrascale dependencies between image wavelet coefficients,” *Image Processing, IEEE Transactions on*, vol. 10, no. 11, pp. 1647–1658, 2001.
- [37] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.
- [38] S. Kullback and R. A. Leibler, “On information and sufficiency,” *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.
- [39] S. M. Ross, *Introduction to probability models*. Access Online via Elsevier, 2006.
- [40] E. P. Simoncelli, “Modeling the joint statistics of images in the wavelet domain,” in *SPIE’s International Symposium on Optical Science, Engineering, and Instrumentation*. International Society for Optics and Photonics, 1999, pp. 188–195.
- [41] G. A. Darbellay and I. Vajda, “Estimation of the information by an adaptive partitioning of the observation space,” *Information Theory, IEEE Transactions on*, vol. 45, no. 4, pp. 1315–1321, 1999.
- [42] J. Dong, H. Boussalis, C. Liu, and K. Rad, “Content-based compression and transmission of astronomical images,” in *Information Technology: Research and Education, 2003. Proceedings. ITRE2003. International Conference on*. IEEE, 2003, pp. 17–21.
- [43] J.-L. Starck and F. Murtagh, *Astronomical image and data analysis*. Springer, 2006.
- [44] R. C. Gonzales and R. Woods, “Digital image processing, 1993.”
- [45] M. Boliek, “Information technology—the jpeg2000 image coding system: Part 1,” *ISO/IEC IS*, pp. 15 444–1, 2000.
- [46] D. Taubman and A. Zakhor, “A common framework for rate and distortion based scaling of highly scalable compressed video,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 6, no. 4, pp. 329–354, 1996.

- [47] D. Taubman and M. Marcellin, “Jpeg2000: Image compression fundamentals, practice and standards,” *Massachusetts: Kluwer Academic Publishers*, pp. 255–258, 2002.
- [48] I. Ueno, F. Ono, T. Yanagiya, T. Kimura, and M. Yoshida, “Proposal of the arithmetic coder for jpeg2000,iso,” IEC JTC1/SC29/WG1, Tech. Rep.
- [49] W. B. Pennebaker, J. L. Mitchell, G. Langdon, and R. B. Arps, “An overview of the basic principles of the q-coder adaptive binary arithmetic coder,” *IBM Journal of research and development*, vol. 32, no. 6, pp. 717–726, 1988.
- [50] P. G. Howard, F. Kossentini, B. Martins, S. Forchhammer, and W. J. Rucklidge, “The emerging jbig2 standard,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 8, no. 7, pp. 838–848, 1998.

# Anexo A

## El estándar JPEG2000

El estándar JPEG2000 es un sistema de compresión y descompresión de imágenes (de tonos de gris y de color) digitales, propuesto por el JPEG (*Joint Photographic Experts Group*). Este estándar está publicado como un ISO/IEC estándar [45] y consta de varias partes, empezando por la Parte 1, que define los requerimientos mínimos necesarios para la compresión/descompresión de las imágenes (codificación, sintaxis del *codestream*, etcétera). La primera parte define lo mínimo para cualquier implementación funcional. Las partes sucesivas añaden nuevas características.

Las partes existentes más importantes son presentadas en la Tabla 1.

### Características

Algunas de las características del estándar JPEG2000 son las siguientes:

- **Compresión de imágenes *continuous-tone y bi-levels*:** El estándar es capaz de comprimir imágenes de tono continuo (*continuous-tone*) y binarias (*bi-levels*).
- **Transmisión progresiva de las imágenes:** Ofrece una progresión en calidad, en resolución, en posición espacial o en componente de imagen, lo cual es muy apropiado para adaptarse a las condiciones de comunicación.
- **Posibilidad de compresión con o sin pérdida:** Gracias a la inclusión de una transformada wavelet reversible, el estándar JPEG2000 ofrece esta posibilidad (aunque en el presente trabajo, el estudio será dedicado a la compresión con pérdidas).
- **Acceso aleatorio al *codestream* de la imagen así como posibilidad de procesamiento del mismo sin necesidad de descompresión:** Los *codestreams* ofrecen diversos mecanismos para soportar el acceso aleatorio en función de una región de interés.
- **Robustez ante errores en los datos:** JPEG2000 define una serie de mecanismos (marcadores de sincronización, etc.) para permitir la detección de errores de los datos.
- **Arquitectura abierta:** Se le pueden añadir funcionalidades a la base (*core*) del estándar.

Parte	Descripción
1	Sistema base
2	Extensión. Añade funcionalidad al sistema base
3	JPEG 2000 para secuencias de vídeo
4	<i>Conformance</i>
5	Software de referencia (implementaciones en C y Java)
6	Formato de archivo compuesto
8	JPSEC, para habilitar opciones de seguridad
9	JPIP, protocolo para la transmisión progresiva de imágenes
10	JP3D, para imágenes tridimensionales
11	JPWL, para aplicaciones <i>wireless</i>

Tabla 1: Partes principales del estándar JPEG2000.

- **Mejor rendimiento a bajos *bit-rates*:** JPEG2000 ofrece un rendimiento superior a bajos *bit-rates* con respecto al resto de los estándares. Beneficioso para la transmisión de imágenes a través de la red.
- **Posibilidad de definir regiones de interés (*ROI*):** A menudo existen partes de una imagen que son más importantes que otras. El estándar permite definir ciertas ROIs en la imagen con el fin de que sean codificadas y transmitidas con mayor prioridad, mejor calidad y menor distorsión que el resto.

## Arquitectura

La Parte 1 del estándar especifica una arquitectura básica, formada por un conjunto de etapas de procesamiento, que se puede ver en forma de diagrama de bloques en la Figura 1. Dichas etapas son aplicadas a cada componente de cada *tile* de la imagen a comprimir.

Las etapas de la arquitectura del estándar JPEG2000 son las siguientes:

- **Offset:** Las muestras de la imagen,  $x[\mathbf{n}]$ , donde  $\mathbf{n}$  representa el punto de coordenadas  $(n_1, n_2)$  de la imagen, con una resolución de  $B$  bits, por tanto:

$$-2^{B-1} \leq x[\mathbf{n}] \leq 2^{B-1}$$

Por ello, si las muestras originales de la imagen son sin signo, como suele pasar en la mayoría de los casos, es necesario añadirles un *offset* de  $-2^{B-1}$ .

- **Transformada de color:** La transformada de color es opcional, ya que sólo puede ser aplicada cuando se poseen al menos tres componentes de color y las tres primeras son del mismo tamaño y la misma profundidad de bits. Se asume que estas primeras tres componentes son RGB, el rojo (*Red*), el verde (*Green*) y el azul (*Blue*).

El objetivo de la transformada de color es convertir las componentes RGB de la imagen en otras componentes con un modelo de color diferente, con el fin de reducir la



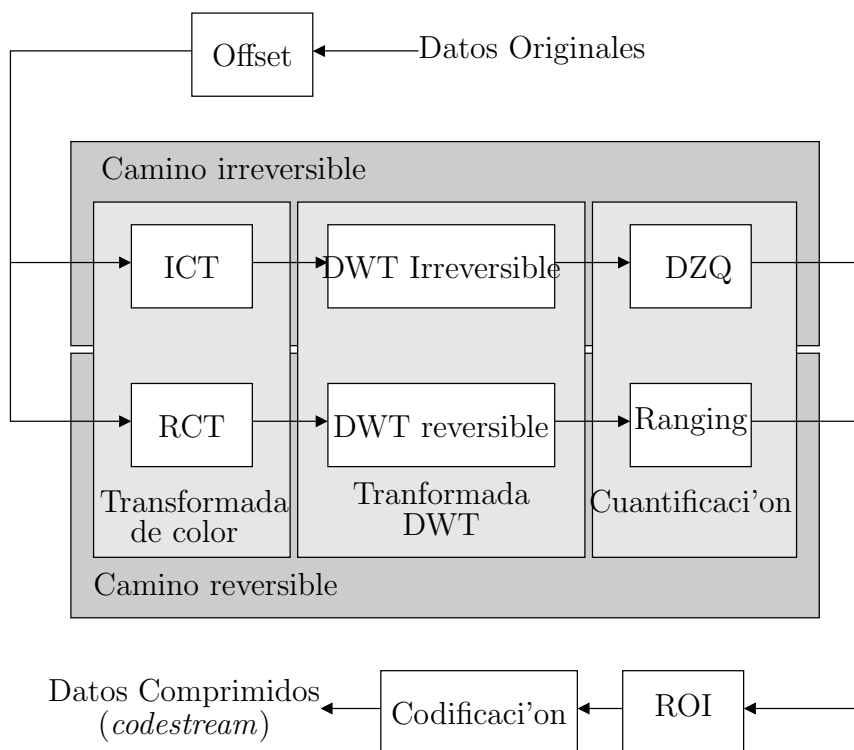


Figura 1: Arquitectura del estándar JPEG2000.

redundancia que existe entre los canales R, G y B, y poder así aumentar las tasas de compresión.

Como se puede apreciar en la Figura 1, existen dos tipos de transformadas de color, una para la compresión con pérdidas (irreversible), la ICT, y otra para la compresión sin pérdidas (reversible), la RCT.

La transformada ICT convierte a la imagen de RGB a YCbCr. Por otro lado, la transformada RCT convierte de RGB a Y'DbDr.

Como hemos comentado, asumimos que las tres componentes de la imagen,  $x_0[\mathbf{n}]$ ,  $x_1[\mathbf{n}]$  y  $x_2[\mathbf{n}]$ , son asociadas al rojo,  $x_R[\mathbf{n}]$ , al verde,  $x_G[\mathbf{n}]$ , y al azul,  $x_B[\mathbf{n}]$  respectivamente. La transformada ICT es:

$$\begin{aligned}
 x_Y[\mathbf{n}] &\triangleq \alpha_R x_R[\mathbf{n}] + \alpha_G x_G[\mathbf{n}] + \alpha_B x_B[\mathbf{n}] \\
 x_{Cb}[\mathbf{n}] &\triangleq \frac{0,5}{1 - \alpha_B} (x_B[\mathbf{n}] - x_Y[\mathbf{n}]) \\
 x_{Cr}[\mathbf{n}] &\triangleq \frac{0,5}{1 - \alpha_R} (x_R[\mathbf{n}] - x_Y[\mathbf{n}])
 \end{aligned}$$

Donde  $\alpha_R \triangleq 0,2999$ ,  $\alpha_G \triangleq 0,587$  y  $\alpha_B \triangleq 0,114$ .

Con las mismas condiciones que para la transformada ICT, se define la transformada RCT como sigue:

$$x_{Y'}[\mathbf{n}] \triangleq \left\lfloor \frac{x_R[\mathbf{n}] + 2x_G[\mathbf{n}] + x_B[\mathbf{n}]}{4} \right\rfloor$$

$$x_{Db} \triangleq x_B[\mathbf{n}] - x_G[\mathbf{n}]$$

$$x_{Dr} \triangleq x_R[\mathbf{n}] - x_G[\mathbf{n}]$$

- **Transformada wavelet:**

Todo lo concerniente a la Transformada Wavelet fue explicado en la Sección 2.3.1

- **Cuantización:** Cada coeficiente *wavelet*  $c$  se cuantiza mediante la siguiente expresión:

$$q_b[n] = \text{sign}(c) \left\lfloor \frac{|c|}{\Delta_b} \right\rfloor$$

El estándar JPEG2000 permite definir un *step-size* de cuantización,  $\Delta_b$ , específico para cada subbanda. Para el camino reversible, el *step-size* de cuantificación debe ser necesariamente igual a uno. Por el camino irreversible la cuantización recibe el nombre de *DZQ* [46], y por el reversible, el de *Ranging* [47].

- **Selección de coeficientes significativos:** Se detallará en la sección siguiente.
- **Ajuste de la ROI (*Region Of Interest*):** Como se comentó anteriormente, JPEG2000 ofrece mecanismos mediante los cuales el compresor puede asignar una prioridad más alta a ciertas regiones de la imagen. En esta etapa es donde se realiza la codificación de la ROI a tener en cuenta en la codificación. El método empleado en el estándar para la codificación de la ROI es el llamado método *MAXSHIFT*, que consiste en desplazar a la izquierda los coeficientes *wavelet* que conforman la ROI.
- **Codificación:** El codificador empleado por el estándar JPEG2000 es el codificador *MQ* [48], similar al codificador *QM* [49] empleado en el estándar JPEG. El codificador *MQ* es empleado también en el estándar JBIG-2 [50]. Se trata de un codificador aritmético binario que permite codificar sin redundancia estadística los coeficientes *wavelets*.

## Particiones de los datos

El estándar JPEG2000 define una gran variedad de particiones dentro de los datos asociados a una imagen, con el fin de permitir la manipulación eficiente de la imagen, o de un trozo de ella, a diferentes resoluciones.

### El concepto de *canvas*

El *canvas* es una grilla bidimensional donde se mapearán todas las particiones que forman la imagen, es decir, cada coeficiente wavelet de cada componente (en una imagen RGB existen 3 componentes) tendrá una posición específica en el canvas, de manera que el procesamiento y análisis posterior tendrá como referencia a las coordenadas espaciales en el canvas y no en la imagen transformada.

## Componentes

Una imagen está compuesta por una o más componentes. Por ejemplo, las imágenes a color de 24 bits, son imágenes con tres componentes: componente rojo, componente verde y componente azul (RGB).

En el estándar JPEG2000, las componentes tienen asociados además unos factores de espaciado, o *sub-sampling factors*. Sea una componente cualquiera  $c$ , definida por un conjunto bidimensional de muestras,  $x_c[n_1, n_2]$ , se definen dos factores de espaciado,  $S_1^c$  y  $S_2^c$ , uno para las filas y otro para las columnas respectivamente. Cada muestra de la componente,  $x_c[n_1, n_2]$  tiene la posición  $[n_1 S_1^c, n_2 S_2^c]$  dentro del *canvas*.

Por ejemplo, si para el componente  $c = 1$  se tiene que  $S_1^1 = 2$  y  $S_2^1 = 3$  significa que la posición de su primer coeficiente se encuentra en la posición  $(2, 3)$  del *canvas*. Además, cada coeficiente de ese componente se encontrará a una separación de  $(2, 3)$  del coeficiente más cercano de su misma clase.

## Tiles

El estándar JPEG2000 permite dividir una imagen en regiones rectangulares más pequeñas llamadas *tiles* (que en español sería algo así como baldosas). Cada *tile* es comprimido de forma independiente en relación al resto, por lo que los parámetros de compresión pueden ser distintos para cada *tile* si se desea.

Una de las posibles aplicaciones de estas particiones son las imágenes que contienen elementos variados y bien diferenciados, como texto, gráficos o material fotográfico. El uso de los *tiles* también reduce la memoria necesaria para la descompresión de una imagen, al tener que aplicar el proceso de descompresión *tile* a *tile* en vez de a la imagen completa.

La forma de crear los *tiles* está definida por cuatro parámetros enteros positivos: los dos primeros,  $T_1, T_2$ , definen el alto y el ancho de los *tiles*, y los dos restantes  $\Omega_1^T$  y  $\Omega_2^T$ , definen el punto de anclaje del conjunto de *tiles*. Esto puede verse gráficamente en la Figura 2.

Los *tiles* le dan al estándar JPEG2000 una gran versatilidad, pero en la mayoría de los casos, su uso no merecerá la pena e incluso, la calidad de la imagen puede verse afectada, como se puede apreciar en la Figura 3.

## Tile-componente

La transformada DWT y todos los pasos siguientes de cuantificación y codificación son aplicados independientemente a cada *tile*-componente. Un *tile*-componente, del *tile*  $t$  y la componente  $c$ , está formando por la zona bidimensional que define  $t$  sobre la cual se mapea la componente  $c$  en función de sus factores de espaciado. Esto quiere decir que si tenemos una imagen con un único *tile* que ocupe el total de la imagen, con tres componentes de color, tendremos tres *tile*-componentes, a los que a cada uno se le aplica el proceso de compresión

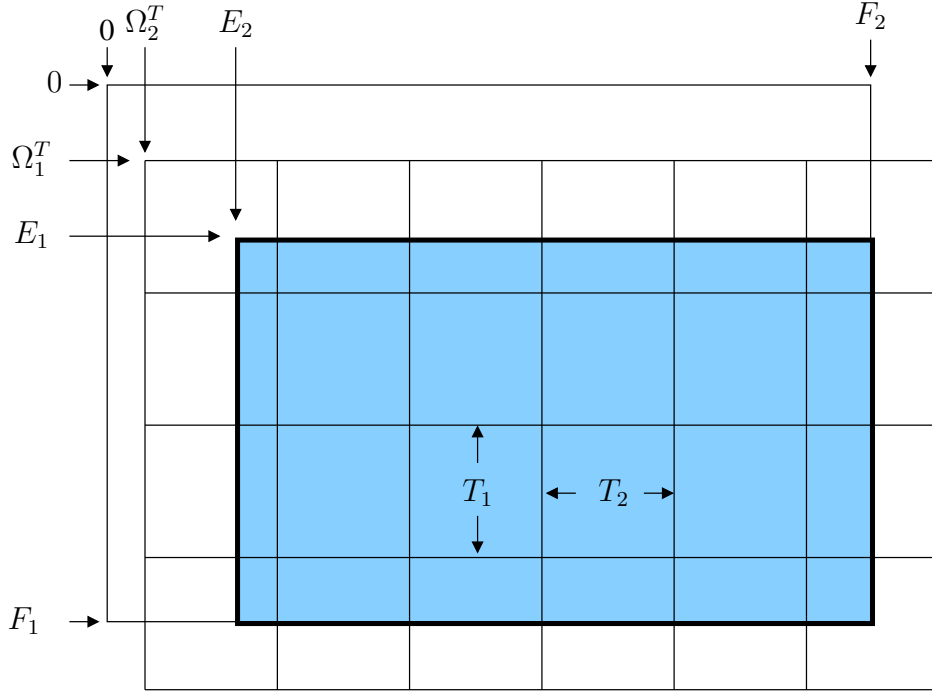


Figura 2: La partición en *tiles* sobre el *canvas*. La región en celeste identifica a la imagen, la región en la que está incluida al *canvas*, y los rectángulos grises los diferentes *tiles*.

de forma independiente.

## Resoluciones

Para cada *tile*-componente, identificado por el *tile*  $t$  y la componente  $c$ , hay un total de  $D_{t,c} + 1$  resoluciones, donde  $D_{t,c}$  es el número de etapas implementadas en la transformada. El  $r$ -ésimo nivel de resolución de la imagen comprimida se obtiene tras aplicar  $r$  veces el filtro de dos canales (DWT) o  $D_{t,c} - r$  etapas de la transformada inversa (IDWT), y se corresponde con la banda de frecuencia  $ll^r$ . El valor de  $r$  toma un valor en el rango  $0 \leq r \leq D_{t,c}$ . La menor resolución,  $r = 0$ , corresponde a la última subbanda (subbanda de aproximación) obtenida tras aplicar todos los pasos de la transformada DWT al *tile*-componente. La mayor resolución se corresponde al *tile*-componente original.

En la Figura 2.8 se pueden apreciar las distintas resoluciones para una imagen dada. En ese caso, se ha aplicado la DWT dos veces,  $D_{t,c} = 2$ , con lo que hay un total de 3 resoluciones. La resolución mayor sería la imagen original. Las restantes resoluciones se corresponden con la subbanda de aproximación de las restantes imágenes (LL1 Y LL2, respectivamente).

Habiendo dicho lo anterior, conociendo la región del canvas que ocupa el *tile*  $t$ , denominada como  $[E_1^t, F_1^t) \times [E_2^t, F_2^t)$ , entonces es posible determinar que porción de dicha región se utiliza para una componente  $c$  determinada a un nivel de resolución  $r$ , de la forma:

$$E_i^{t,c,r} = \left\lceil \frac{E_i^t}{2^{(D_{t,c}-r)S_i^c}} \right\rceil, F_i^{t,c,r} = \left\lceil \frac{F_i^t}{2^{(D_{t,c}-r)S_i^c}} \right\rceil, \text{ para } i \in 1, 2$$



Figura 3: La imagen “ski” de  $720 \times 576$ : (a) imagen original, (b)-(d) imágenes reconstruidas después de una compresión JPEG2000 a 0.25 bpp: (b) sin *tiling*, (c) con *tiles* de  $128 \times 128$ , y (d) con *tiles* de  $64 \times 64$ .

### *Code-blocks* y precintos

Cada *tile*-componente es dividido en *code-blocks*, que son codificados independientemente. Esta partición en *code-blocks* es definida mediante el punto de anclaje,  $[\Omega_1^C, \Omega_2^C]$ , y el alto y el ancho máximo del *code-block*,  $J_1^{t,c}$  y  $J_2^{t,c}$ . En la Parte 1 del estándar, el punto de anclaje debe ser obligatoriamente  $[0, 0]$  pero, en cambio, en la Parte 2, cada  $\Omega_i^{t,c}$  puede tomar el valor de 0 o de 1.

La división en *code-blocks* es fundamental para el paradigma EBCOT, empleado en el estándar JPEG2000.

En cada resolución  $r$  de cada *tile*-componente  $t, c$ , los *code-blocks* son agrupados en precintos (*precincts*). Esta agrupación viene definida por el alto, en número de *code-blocks*,  $P_1^{t,c,r}$ , y el ancho, también en número de *code-blocks*,  $P_2^{t,c,r}$  de cada precinto.

En la Figura 4 se puede observar la división de una resolución de un *tile*-componente en *code-blocks* y precintos.

Cada precinto  $P$  es identificado por un par de índices  $(p_1, p_2)$ , los cuales están comprendidos

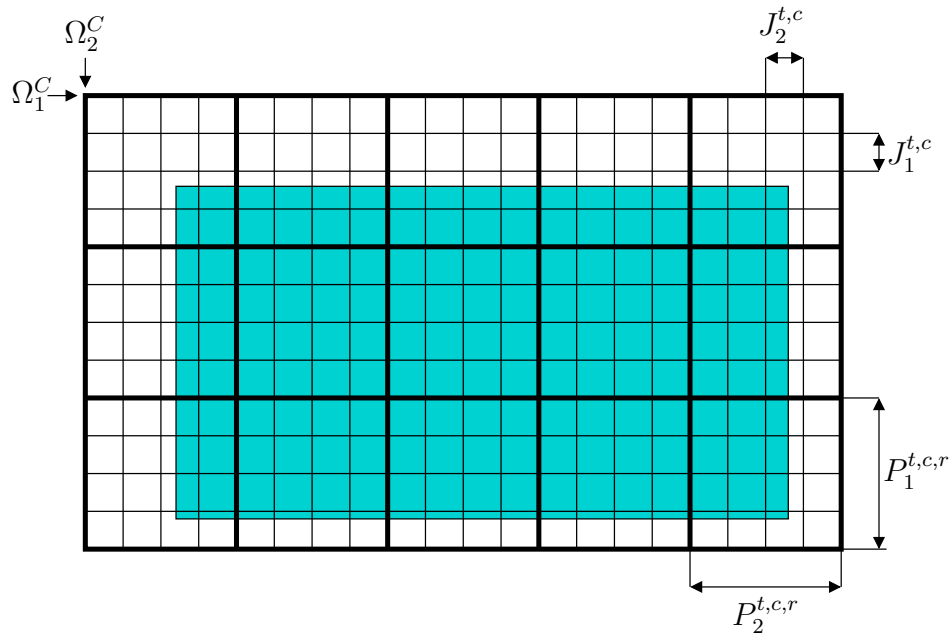


Figura 4: Partición en *code-blocks* y precintos de un *tile-componente* a una determinada resolución. La región en celeste identifica a la resolución del *tile-componente*.

en los siguientes rangos:

$$0 \leq p_1 < N_1^{P,t,c,r} \text{ y } 0 \leq p_2 < N_2^{P,t,c,r}$$

$N_1^{P,t,c,r}$  y  $N_2^{P,t,c,r}$  identifican el número de precintos vertical y horizontalmente, respectivamente. Estos valores son obtenidos mediante la siguiente ecuación:

$$N_i^{P,t,c,r} = \begin{cases} \left\lceil \frac{F_i^{t,c,r} - \Omega_i^C}{P_i^{t,c,r}} \right\rceil - \left\lfloor \frac{E_i^{t,c,r} - \Omega_i^C}{P_i^{t,c,r}} \right\rfloor & \text{si } F_i^{t,c,r} > E_i^{t,c,r} \\ 0 & \text{si } F_i^{t,c,r} = E_i^{t,c,r} \end{cases}$$

Los precintos juegan un papel importante en la organización de los datos comprimidos dentro del *codestream*.

### Capas y paquetes

El paquete es la unidad fundamental en la organización del *codestream*. Cada precinto contribuye al *codestream* con tantos paquetes como capas de calidad (*quality layer*) haya. Como hemos mencionado anteriormente, cada *code-block* es codificado independientemente. Los datos codificados de cada *code-block* son divididos en diferentes secciones llamadas capas de calidad. Todos los *code-blocks* de los precintos de un mismo *tile* son divididos en el mismo número de capas, aunque el tamaño de las capas entre *code-blocks* puede ser diferente (el tamaño puede ser incluso cero). Para una capa  $l$  determinada, el conjunto de las capas  $l$  de todos los *code-blocks* que forman un precinto forman un paquete. La Figura 5 muestra un ejemplo gráfico.

El número de capas de calidad para un *tile*  $t$  es identificado como  $\Lambda_t$ . Aunque el número de capas de calidad puede variar de *tile* a *tile*, los compresores harían bien en usar el mismo

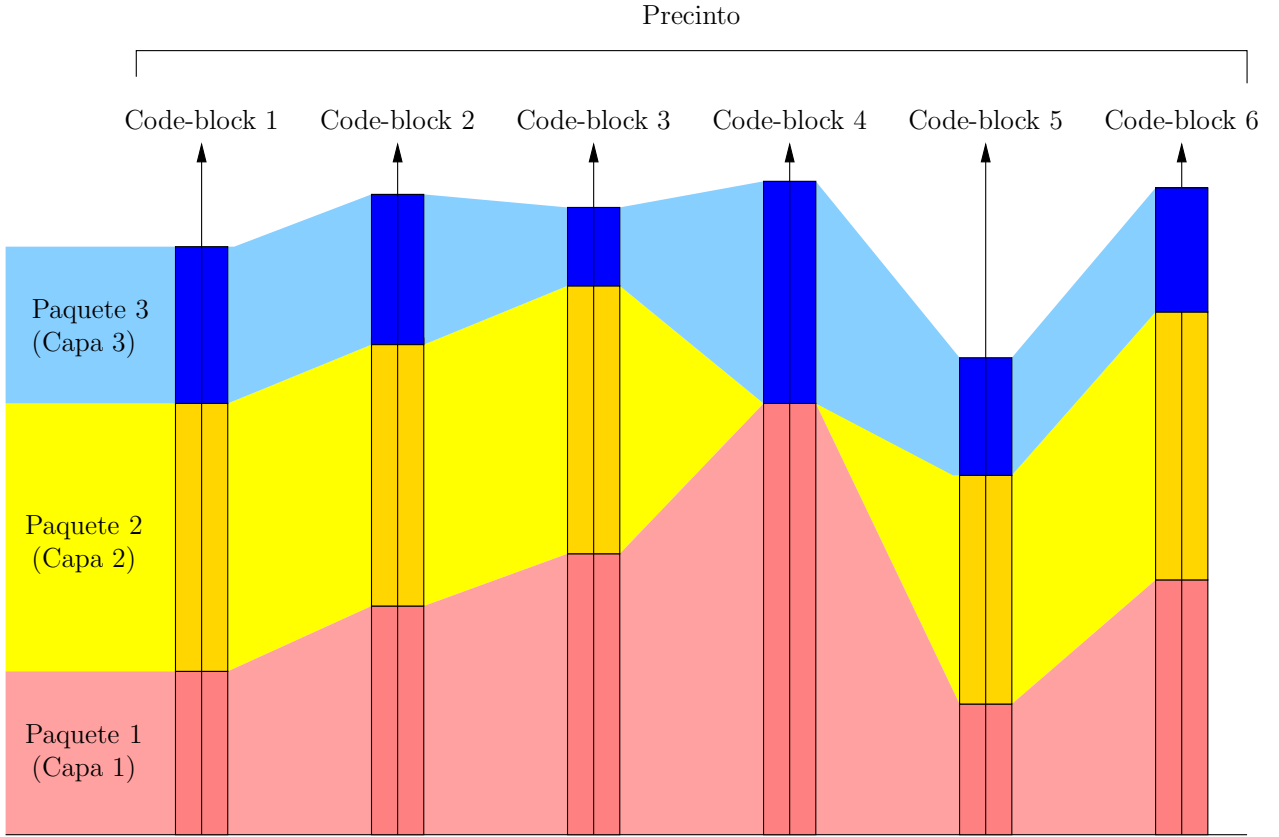


Figura 5: Ejemplo de capas de calidad y paquetes.

número de capas para todos los *tiles*, para evitar ambigüedades cuando se pretende descartar un determinado número de capas de calidad en una imagen. El número total de paquetes en un *codestream* sería:

$$\text{número de paquetes} = \sum_{t_1=0}^{N_1^T-1} \sum_{t_2=0}^{N_2^T-1} \sum_{c=0}^{C-1} \sum_{r=0}^{D_{t,c}} \Lambda_t N_1^{P,t,c,r} N_2^{P,t,c,r}$$

Finalmente, un paquete  $\zeta_{t,c,r,p,l}$  es identificado por el *tile*  $t$ , la componente  $c$ , la resolución  $r$ , el precinto  $p$  ( $p \equiv [p_1, p_2]$ ) y la capa de calidad  $l$ .

### Codificación: Selección de soporte

El proceso de codificación busca preparar la información obtenida de los anteriores procesos, eliminando la redundancia estadística que pudiera haber a fin de minimizar la cantidad de bits sin perder datos. La codificación es un proceso reversible, lo que hace que no haya que diferenciar esta vez si estamos comprimiendo con pérdida o sin pérdida.

Los coeficientes wavelet de un bloque de código se codifican individualmente por *plano de bits*, tal como muestra la Figura 2.10. Se va haciendo desde el plano de los bits más significativos hasta el plano de los bits menos significativos.

JPEG 2000 codifica los bits individualmente y usa contextos simétricos. El contexto de un bit se calcula a partir de los 8 bits vecinos. El problema es que el decodificador no va a poder disponer de todos ellos, con lo que no se podrá disponer de sus valores.

Por ello, hay que utilizar la significancia de los bits vecinos. Cada coeficiente wavelet generado tiene un bit que indica la significancia. Tras conocer el contexto, se estima la probabilidad del bit.

Por último, se transfiere el bit y probabilidad al codificador aritmético. Así es como se codifica un bit.

A la hora de codificar planos de bits completos, se siguen 3 pasos:

1. Propagación de la significancia
2. Refinamiento de magnitud
3. Limpieza (Cleanup)

El primer plano de bits más significativos no nulo sólo tendrá que aplicar el 3er paso. El resto, seguirá los 3 pasos.

Un plano de bits se recorre de la siguiente manera (Figura ?? a la hora de aplicar el proceso, por franjas verticales. Hay cuatro filas de franjas.

### **Propagación de la significancia**

Este paso codifica todos los bits pertenecientes a coeficientes wavelets que satisfagan:

1. El coeficiente wavelet asociado es insignificante
2. Al menos uno de los 8 bits vecinos son significantes.

Si un bit es codificado y vale 1, su coeficiente wavelet pasa a tener 1 como valor de significancia.

Subsecuentemente, los bits codificados en este paso (y los siguientes dos pasos) consideraran el nuevo valor de la significancia del coeficiente wavelet asociado. Este paso muestra que al menos es necesario que algunos coeficientes wavelet sean declarados significativos para que haya codificación. Es por ello por lo que el primer plano de bit más significativo sólo hace el 3er paso.

### **Refinamiento de magnitud**

Este segundo paso codifica todos los bits de los coeficientes wavelets que se convirtieron en significativos en el plano de bits anterior. Así, una vez que un coeficiente se convierte en significativo, todos sus bits menos significativos serán codificados uno por uno, cada uno en el segundo paso de un plano de bits diferente.



## **Limpieza**

Por último, en este paso se codifican los bits que no se han codificado en los pasos 1 y 2. Si el bit a codificado vale 1, el coeficiente wavelet correspondiente pasa a tener el coeficiente de significancia a 1.