

# Esquemas de watermarking semifrágil para la protección de imágenes de obtención remota

Por  
Jordi Serra Ruiz

Programa de Doctorado en  
Sociedad de la Información y el Conocimiento

Director:  
Dr. David Megías Jiménez

Junio 2011



Grupo de investigación KISON  
Internet Interdisciplinary Institute  
Universitat Oberta de Catalunya



©Copyright 2011 por

Jordi Serra Ruiz

Todos los derechos reservados





*“(Al camarero de un restaurante)*

*Hoy no tengo tiempo para almorzar.*

*Traiga la cuenta. ”*

Groucho Marx



*A mi esposa Eli,  
mis hijos Arnau y Lluç  
y a mis padres.*



# Agradecimientos

Vull agrair la dedicació del director de tesi. Durant molt temps ha supervisat el treball molt encertadament i li vull agrair especialments el temps dedicat a la correcció d'aquesta memòria, ha estat excel.lent.

Vull agrair també a tots els meus companys de la universitat els ànims que m'han donat en els darrers moments en els quals es fa molt cansat escriure i revisar la memòria.

Vull agrair als meus pares i germans el suport i ànims que m'han donat durant tots aquests anys.

I finalment vull agrair a la meva dona la dedicació que ha tingut en la cura de l'Arnau i en Lluç mentre jo estava realitzant els experiments o en els moments de la redacció d'aquesta memòria. Agrair també el suport que vaig tenir per part d'Eli mentre vaig estar més d'un any malalt enmig de la realització d'aquesta tesi.

Este trabajo ha sido subvencionado por el Ministerio de Ciencia e Innovación de España y los fondos FEDER con la subvención a los proyectos TSI2007-65406-C03-03 E-AEGIS y CONSOLIDER CSD2007-00004 ARES.



## Resumen

La flexibilidad de la transmisión de la información y los reducidos precios de los dispositivos digitales hacen posible que la gran mayoría de personas tengan acceso a reproducir, editar e intercambiar todo tipo de contenidos multimedia. La Red permite que los contenidos se puedan transmitir sin ningún problema y la redistribución se realiza sin pérdida de calidad alguna.

La facilidad con la que se puede modificar un contenido y la distribución sin permiso de éste han contribuido a que la protección del contenido y la prevención de las manipulaciones se hayan convertido en una importante área de investigación. Esto hace que las técnicas de protección de contenidos digitales y, entre ellas, las técnicas de watermarking sean de gran importancia para proteger los derechos de autor o para autenticar el contenido.

Dejando de lado la protección de los derechos de autor, la tesis se centra en el estudio de la autenticación de las imágenes de obtención remota usando técnicas de watermarking semifrágil. Las imágenes de obtención remota son costosas de obtener, dado que se necesita de un sensor especial que capte la reflexión de la luz para diferentes longitudes de onda. Estos sensores se instalan en aviones o en satélites, como pueden ser los LANDSAT o los enviados a otros planetas.

En la tesis se presenta una familia de métodos de watermarking para este tipo de imágenes que detectan las manipulaciones fraudulentas de las mismas. Los métodos tienen en cuenta la principal característica de las imágenes hiperespectrales: la representación de los materiales de la superficie mediante firmas espectrales específicas. La marca se incrusta teniendo en cuenta que

cada píxel de la imagen está compuesto por un conjunto de componentes (más de 100 en las imágenes hiperespectrales) que representan la reflexión de la luz para diferentes longitudes de onda.

Dos de los métodos presentados utilizan el valor de cada componente para incrustar la marca, mientras que los otros marcan sobre los coeficientes de la transformada discreta *wavelet* para obtener mayor robustez frente a ataques de compresión con pérdida.

Los métodos propuestos se han probado frente a ataques de copia y reemplazo de zonas de la imagen (sobrescribiendo una parte de la imagen sobre otra) obteniendo la localización de estas modificaciones. También se ha probado la robustez de los métodos frente a compresiones JPEG2000 y JPEG, obteniendo resultados positivos en cuanto al compromiso entre transparencia y robustez. La evolución de los diferentes métodos propuestos se ha realizado en base a las propiedades básicas de los esquemas de watermarking: la imperceptibilidad, la robustez frente a ataques de compresión y la detección de manipulaciones fraudulentas (ataques de copia y reemplazo). La capacidad no es una propiedad relevante para la aplicación que se presenta en esta tesis.

Una vez presentados los esquemas y sus resultados, se comparan los diferentes métodos presentados con otros sistemas semifrágiles de localización de manipulaciones de la literatura, mostrando que se mejoran los resultados obtenidos por éstos.



# Abstract

The flexibility of content transmission and the low prices of digital players make it possible for that vast majority of people to play, edit and exchange all kinds of multimedia contents. The Internet allows that all these contents are transmitted without effort and redistribution can be performed without any kind of quality loss.

The easiness to modify contents and the unauthorised distribution of them have contributed to the development of content protection and tampering prevention as relevant research topics. Because of this, digital content protection techniques and, among them, watermarking systems have become suitable strategies for copyright protection and content authentication.

Putting copyright protection aside, this thesis is focused on the authentication of remote sensing images using semi-fragile watermarking techniques. Remote sensing images are costly, since they require special sensors to capture light reflection for different wavelengths. These sensors are usually installed in aircraft or satellites, such as LANDSAT or those sent to explore other planets.

This thesis presents a family of watermarking schemes for tampering detection in remote sensing images. These methods consider the most relevant property of hyperspectral images: the representation of different surface materials by specific spectral signatures. The mark is embedded taking into account the fact that pixels are formed by a set of components (above 100 for hyperspectral images) which represent the reflection of light for different

wavelengths.

Two of the suggested methods use the component value to embed the mark, whereas the rest of them embed the mark in the coefficients of the discrete wavelet transform in order to attain better robustness against lossy compression attacks.

The proposed schemes have been tested against copy-and-replace attacks in specific areas of the image (by overwriting a region of the image over a different one) and they provide the localisation of the tampered area. They have been also tested for robustness against JPEG and JPEG2000 compression and the obtained results are very satisfying as the trade-off between transparency and robustness is concerned. The evolution of the different suggested methods has been undertaken taking into account the basic properties of watermarking schemes: imperceptibility, robustness against lossy compression and tampering detection (copy-and-replace attacks). Capacity is not a relevant property for the application which is proposed in this thesis.

Once all the schemes and their results are presented, they are also compared with several semi-fragile tampering detection systems of the literature, showing that the results of the suggested schemes overcome those of the other methods.





# Índice general

Índice de contenidos	XVI
Índice de figuras	XIX
Índice de tablas	XXIII
<b>1. Introducción</b>	<b>1</b>
1.1. Introducción a la protección de contenidos digitales . . . . .	1
1.2. Objetivos de la tesis . . . . .	4
1.3. Estructura de la tesis . . . . .	5
<b>2. Conceptos básicos y herramientas para imágenes de obtención remota</b>	<b>7</b>
2.1. Introducción a las imágenes . . . . .	7
2.2. Imágenes de obtención remota ( <i>Remote Sensing Images</i> ) . . .	8
2.3. Transformadas matemáticas . . . . .	13
2.3.1. Transformada de Fourier . . . . .	13
2.3.2. Transformada discreta del coseno (DCT) . . . . .	16
2.3.3. Transformada <i>wavelet</i> . . . . .	18
2.4. Compresión . . . . .	21
2.4.1. Compresión con pérdida de imágenes de obtención remota . . . . .	22
2.4.2. Cuantización vectorial y cuantización vectorial en estructura de árbol . . . . .	25

2.4.3.	<i>Joint Photographic Experts Group (JPEG)</i> . . . . .	29
2.4.4.	JPEG2000 . . . . .	31
<b>3.</b>	<b>Estado del arte de los esquemas de watermarking</b>	<b>35</b>
3.1.	Historia y situación actual . . . . .	35
3.2.	Aplicaciones de los esquemas de watermarking . . . . .	41
3.2.1.	Protección de los derechos de autor . . . . .	42
3.2.2.	Copia y control de acceso . . . . .	44
3.2.3.	Autenticación del contenido . . . . .	44
3.2.4.	Garantía de la integridad del contenido . . . . .	45
3.2.5.	Monitorización de las emisiones . . . . .	46
3.2.6.	Información oculta . . . . .	47
3.3.	Propiedades de los esquemas de watermarking . . . . .	47
3.3.1.	Capacidad . . . . .	48
3.3.2.	Robustez . . . . .	49
3.3.3.	Transparencia o imperceptibilidad . . . . .	50
3.3.4.	Seguridad . . . . .	52
3.3.5.	Detección y extracción ciega o informada . . . . .	53
3.4.	Métodos de watermarking para imágenes hiperespectrales . . . . .	53
3.5.	Otros métodos de watermarking frágil y semifrágil . . . . .	55
<b>4.</b>	<b>Sistemas de watermarking para imágenes hiperespectrales</b>	<b>59</b>
4.1.	Introducción . . . . .	59
4.2.	Proceso de incrustación de la marca . . . . .	60
4.2.1.	Preprocesado . . . . .	61
4.2.2.	Descripción general del método de marcado . . . . .	61
4.2.3.	Método basado en la extracción de los LSB . . . . .	77
4.2.4.	Marcado sobre los coeficientes de la DWT . . . . .	81

4.2.5. Marcado sobre DWT y aplicación de regiones de tamaño variable . . . . .	85
4.2.6. Marcado sobre DWT y automatización de la selección de las regiones a marcar . . . . .	89
4.3. Proceso de detección de modificaciones . . . . .	93
<b>5. Resultados experimentales</b>	<b>97</b>
5.1. Introducción . . . . .	97
5.2. Resultados experimentales . . . . .	97
5.2.1. Proceso de incrustación de la marca . . . . .	105
5.2.2. Método basado en la extracción de los LSB . . . . .	117
5.2.3. Método de marcado sobre los coeficientes de la DWT .	131
5.2.4. Método de marcado sobre DWT y aplicación de regiones de tamaño variable . . . . .	138
5.2.5. Método de marcado sobre DWT y automatización de la selección de las regiones a marcar . . . . .	147
5.3. Análisis comparativo con otros esquemas de watermarking . .	156
<b>6. Conclusiones</b>	<b>161</b>
6.1. Conclusiones destacadas . . . . .	161
6.2. Resultados de la tesis . . . . .	163
6.3. Trabajo futuro . . . . .	166
<b>Bibliografía</b>	<b>168</b>
<b>Publicaciones</b>	<b>178</b>





# Índice de figuras

2.1. Ejemplo de una curva de firma espectral para un píxel. . . . .	9
2.2. Algunas firmas espectrales típicas. . . . .	10
2.3. imágenes LANDSAT de Barcelona, 1984 (a) y 2004 (b). . . . .	12
2.4. Ejemplo de señal discreta. . . . .	16
2.5. Ejemplo de transformada discreta de Fourier. . . . .	17
2.6. Transformada <i>wavelet</i> de 3 niveles. . . . .	20
2.7. <i>Wavelet</i> de 4 niveles para la imagen lena. . . . .	20
2.8. Reconstrucción de la imagen lena usando sólo la banda de más baja frecuencia. . . . .	21
2.9. Diagrama de bloques del codificador para un método de com- presión con pérdida. . . . .	23
2.10. Función VQ óptima ( <i>convex hull</i> ) y aproximación con función sub-óptima. . . . .	26
2.11. Regiones de Voronoi. . . . .	27
2.12. Ejemplo de compresión del árbol TSVQ. . . . .	28
2.13. Imagen original (a), DWT 1 nivel (b) y DWT 2 niveles (c). . . . .	33
3.1. Carta completa (a) y ejemplo de rejilla de Cardano (b). . . . .	37
3.2. Carta original . . . . .	38
3.3. Texto obtenido con la rejilla de Cardano . . . . .	38
3.4. Parte inferior de la imagen de lena. . . . .	43
3.5. Esquema genérico del watermarking . . . . .	48

4.1. Construcción de los vectores TSVQ. . . . .	65
4.2. Selección de bandas, división por bloques y generación de los vectores de marcado. . . . .	67
4.3. Diagrama de bloques del proceso de marcado. . . . .	72
4.4. Esquema de restauración de los LSB. . . . .	78
4.5. Selección de bandas, división por bloques, extracción de los LSB y generación de los vectores de marcado. . . . .	80
4.6. Esquema de marcado: construcción y reconstrucción de los bloques con DWT. . . . .	83
4.7. Esquema iterativo de marcado con DWT. . . . .	84
4.8. Ejemplo de elección del tamaño variable de los bloques de la imagen. . . . .	86
4.9. Selección de bandas, división por bloques de tamaño variable, aplicación de la DWT y generación de los vectores de marcado. . . . .	87
4.10. Agrupación de áreas para automatizar la creación de bloques. . . . .	91
4.11. Inclusión del bloque $B_{3,2}$ dentro de $B'$ . . . . .	91
4.12. Bloques resultantes de la automatización de la selección de las regiones. . . . .	93
4.13. Diagrama de bloques del proceso de detección de modificaciones. . . . .	94
5.1. Ejemplo de imagen Cuprite. . . . .	98
5.2. Sección de la banda 151 de la imagen Indian Pines. . . . .	99
5.3. Sección de la banda 151 de la imagen WTC del 16 de septiem- bre de 2001. . . . .	100
5.4. Mapa geotermal del WTC del 16 de septiembre de 2001. . . . .	100
5.5. valores máximos, mínimos y medios de cada banda de la ima- gen Cuprite. . . . .	102
5.6. valores máximos, mínimos y medios de cada banda de la ima- gen Indian Pines. . . . .	103
5.7. valores máximos, mínimos y medios de cada banda de la ima- gen WTC. . . . .	103
5.8. Histograma de las diferencias entre valores originales y marcados. . . . .	111
5.9. Diferencia entre la firma espectral original y marcada (despla- zada) de un píxel marcado. . . . .	111

5.10. Imagen original (a) y marcada (b) de la banda 2. . . . .	113
5.11. Imagen modificada (a) y localización de la modificación (b). . .	114
5.12. Proceso de marcado de un componente con $n = 3$ . . . . .	118
5.13. Histograma de las modificaciones del proceso de marcado con $n = 2$ . . . . .	121
5.14. Diferencia entre la firma espectral original y marcada (despla- zada) de un píxel marcado. . . . .	122
5.15. Imagen Cuprite reducida a 8 bpp (banda 2). . . . .	124
5.16. Imagen Cuprite de 8 bpp (banda 2). . . . .	129
5.17. Histograma de las diferencias entre valores originales y mar- cados con la transformada <i>wavelet</i> . . . . .	136
5.18. Proceso de división de los bloques. . . . .	141
5.19. Histograma de las diferencias entre los valores originales y mar- cados para la imagen Cuprite aplicando la transformada DWT. . . . .	142
5.20. Diferencia entre las firma espectrales, original y marcada. . . .	143
5.21. Imagen modificada (a) y detectada (b) (para la banda 9). . . .	146
5.22. Bloques resultantes de la automatización de la selección de las regiones para la imagen Cuprite. . . . .	148
5.23. Diferencia entre las firma espectrales, original y marcada. . . .	149
5.24. Histograma de las diferencias entre valores reales y marcados para la imagen Cuprite. . . . .	151
5.25. Imagen modificada y detectada (banda 2). . . . .	153



# Índice de tablas

2.1. Ejemplo de <i>codebook</i> inicial. . . . .	29
2.2. Ejemplo de <i>codebook</i> final. . . . .	29
4.1. Ejemplo de valores obtenidos con el algoritmo BFOS. . . . .	71
5.1. Identificación de las bandas modificadas. . . . .	105
5.2. PSNR de la imagen Cuprite marcada para cada banda. . . . .	107
5.3. PSNR total de la imagen Cuprite marcada. . . . .	107
5.4. PSNR de la imagen Indian Pines marcada para cada banda. . . . .	108
5.5. PSNR total de la imagen Indian Pines marcada. . . . .	108
5.6. PSNR de la imagen del WTC marcada para cada banda. . . . .	108
5.7. PSNR total de la imagen del WTC marcada. . . . .	109
5.8. Resultados de ataques de compresión JPEG2000. . . . .	116
5.9. PSNR de la imagen Cuprite marcada para cada banda. . . . .	120
5.10. Valores de PSNR para diferentes imágenes y valores de $n$ . . . . .	123
5.11. Componentes de la imagen WTC reducida a 8 bpp. . . . .	125
5.12. Robustez del esquema contra ataques de compresión JPEG2000 para la imagen Cuprite de 14 bpp. . . . .	128
5.13. Robustez del esquema LSB contra ataques de compresión JPEG para la imagen Cuprite de 8 bpp. . . . .	132
5.14. PSNR de la imagen Cuprite marcada para cada banda. . . . .	133
5.15. PSNR de la imagen Cuprite marcada para DB1, DB3 y DB5. . . . .	134

5.16. Resultados de los ataques de compresión con JPEG2000 para la imagen Cuprite de 14 bpp. . . . .	137
5.17. Robustez del esquema DWT frente a ataques de compresión JPEG para la imagen Cuprite de 8 bpp. . . . .	139
5.18. Imperceptibilidad del esquema de marcado para las bandas marcadas (usando la transformada Daubechies 1 entera). . . .	143
5.19. PSNR de la imagen Cuprite marcada para DB1, DB3 y DB5. .	144
5.20. Resultados de los ataques de compresión con JPEG2000 para la imagen Cuprite. . . . .	147
5.21. PSNR de la imagen Cuprite marcada para DB1, DB3 y DB5. .	149
5.22. Resultados de los ataques de compresión con JPEG2000 para la imagen Cuprite de 14 bpp. . . . .	152
5.23. Robustez del esquema DWT con selección de areas automática frente a ataques de compresión JPEG para la imagen Cuprite de 8 bpp. . . . .	155
5.24. Comparación de los métodos presentados con otros sistemas de watermarking . . . . .	157







# Capítulo 1

## Introducción

### 1.1. Introducción a la protección de contenidos digitales

El gran auge de Internet en estos últimos años y la revolución de la tecnología digital, que permite reproducir todo tipo de contenidos a bajo coste, está provocando cambios continuos y muy significativos en la sociedad, que ve cómo la comunicación digital se ha convertido en uno de los principales medios de relacionarse. La prensa escrita, la radio, y la televisión ya difunden sus contenidos en Internet. Los cantantes lanzan sus nuevos temas musicales en la Red y ven cómo se comparten cada día más música y películas, ya sea de forma legal o ilegal. La comunicación entre las empresas también se realiza cada vez más en la Red. En consecuencia, las ventas de todo tipo de contenidos digitales están al alza.

La flexibilidad de la transmisión de los contenidos y los reducidos precios de los dispositivos digitales (cámaras, reproductores de sonido o vídeo, ordenadores portátiles, teléfonos inteligentes, etc.) hacen posible que la gran mayoría de gente tenga acceso a reproducir, editar e intercambiar todos estos contenidos multimedia. La Red permite que todos estos contenidos se puedan

transmitir sin ningún problema y la redistribución de los contenidos digitales se realiza sin pérdida de calidad alguna.

La facilidad con la que se puede modificar un contenido y la distribución sin permiso de éste han contribuido a que la protección del contenido y la prevención de las manipulaciones se hayan convertido en una importante área de investigación (Yu et al., 2001). Es imprescindible disponer de algún sistema de control para abordar estas problemáticas.

Los métodos iniciales basados en la inclusión de algún patrón distintivo en la cabecera del contenido digital se demostraron inútiles, ya que un simple cambio de formato o la supresión directa de estos bits permiten eliminar la protección. En el caso del cifrado del contenido, en el que se opta por aplicar criptografía, tanto el proveedor como el cliente necesitan disponer de la clave: el primero para cifrar el contenido y el segundo para poderlo reproducir descifrado. En este contexto, el cliente tiene acceso al contenido en claro (sin cifrar) y lo puede distribuir sin ningún obstáculo. Las herramientas disponibles actualmente posibilitan, en gran medida, la realización del descifrado y la distribución del contenido en claro por Internet. A partir del instante en que se usaron estos métodos, surgieron rápidamente métodos para romperlos. Los métodos iniciales de protección de contenido se volvieron insuficientes (Bloom et al., 1999).

Todo ello ha llevado a desarrollar nuevas estrategias en la protección del contenido y ahí es donde se introduce la esteganografía, la cual se ha convertido en poco tiempo en una muy buena herramienta de protección de la información.

La **esteganografía** –del griego “stego” (esconder o secreto) y “grapho” (escribir)– es una técnica que consiste en insertar un mensaje secreto oculto

dentro de una información aparentemente inocua (Katzenbeisser and Petitcolas, 1999). Así pues, una comunicación esteganográfica consiste en la transmisión de un mensaje secreto entre dos o más personas o entidades a través de un canal público. Estos tipos de técnicas se utilizan a menudo en comunicaciones militares o terroristas. A diferencia de la *criptografía* (Menezes et al., 1996), en la esteganografía el contenido no se cifra, si no que se transmite “en claro” entre el emisor y el receptor.

Un ejemplo de comunicación esteganográfica la podemos encontrar en el texto siguiente:

Aquí los días pasan muy lentamente. A veces nos encontramos en un bar para tomar unas cañas y ver un partido de fútbol por la televisión. Por ejemplo, la semana pasada vimos el partido de la selección. En una pancarta, pudimos leer el mensaje de un aficionado que declaraba su amor secreto por una vecina. Fue muy divertido.

Si se toma la primera palabra de cada línea encontramos el texto escondido siguiente:

“Aquí encontramos un ejemplo de mensaje secreto”.

Por tanto, la esteganografía abarca la ocultación de mensajes en contenido digital (*data hiding*), o la protección del contenido, conocida como marcas de agua o *watermarking*, usando a menudo la criptografía para hacer que los mensajes ocultos sean ilegibles en caso de recuperarlos.

La idea principal de esta tesis es aplicar técnicas de *watermarking* para proteger el contenido de un tipo especial de imágenes. En concreto, la tesis se focaliza en la protección de imágenes de obtención remota usando técnicas de compresión por cuantización vectorial y la aplicación de diferentes

herramientas de manipulación de señales, como la transformada *wavelet*.

## 1.2. Objetivos de la tesis

El objetivo principal de la tesis es el diseño e implementación de una familia de métodos que permitan marcar las imágenes de obtención remota para detectar manipulaciones fraudulentas de éstas. Los métodos desarrollados deben permitir una cierta compresión con pérdida pero no modificaciones significativas o cambios importantes en alguna de las partes de la imágenes.

Se diseñará un esquema general a partir del cual se obtendrán diferentes métodos de marcado usando diferentes técnicas o herramientas que permitirán llegar a los resultados deseados.

Los métodos a desarrollar han de permitir marcar la imagen completa, de manera que se obtenga una única versión de toda la imagen marcada. Además, deberá ser posible comprimir la imagen a más o menos ratio para poder tener diferentes versiones marcadas de la imagen con mayor o menor calidad, sin tener que volver a realizar el proceso de marcado. Esto permitirá que el proveedor de las imágenes de obtención remota pueda distribuir a diferentes clientes la misma imagen con diferentes calidades sin tener que volver a marcar la imagen otra vez. A partir de la imagen marcada sólo será necesario comprimirla para generar las diferentes versiones de la misma.

Además, se realizará un estudio, finalizándolo con una comparativa exhaustiva, de los diversos métodos de marcado sobre imágenes hiperespectrales mostrando los resultados obtenidos por los métodos propuestos y otros siste-

mas de la literatura, destacando las ventajas e inconvenientes de los métodos descritos en la tesis.

Los métodos señalarán la zona modificada por una manipulación de la imagen marcada. Si el proceso de detección identifica un área como modificada respecto a la imagen marcada, el método señalará en blanco la zona modificada. Además, esto se ha de realizar sin comparar las imágenes modificadas y las imágenes no marcadas. Los métodos desarrollados deberán de ser ciegos, y por tanto, no han de requerir de la imagen original para detectar las manipulaciones.

### 1.3. Estructura de la tesis

El resto de esta tesis está organizado de la siguiente manera.

El segundo capítulo describe los conceptos básicos de las imágenes hiperespectrales y algunas herramientas necesarias para la manipulación y el procesamiento de éstas, como la transformada de Fourier, la transformada del coseno, la transformada *wavelet* y la transformada de Karhunen-Loève, que se utilizan en algunos de los métodos desarrollados. También se describen las técnicas de compresión por cuantización vectorial, y los algoritmos de compresión *Joint Photographic Experts Group* (JPEG) y JPEG2000 que inspiran algunos de los métodos de la tesis.

El tercer capítulo presenta los conceptos básicos de los esquemas de watermarking, introduciendo los aspectos claves y ofreciendo una visión resumida de sus características más significativas. En el capítulo se describen breve-

mente las propiedades necesarias para poder comparar los diferentes métodos presentados.

El capítulo cuarto describe diferentes métodos de marcado desarrollados, explicando paso a paso su diseño y desarrollo.

El capítulo quinto muestra los resultados obtenidos con cada uno de los métodos y una comparativa detallada con otros métodos de la literatura.

Para finalizar, se presentan las conclusiones obtenidas así como las líneas de investigación futura que se pueden seguir a partir de este trabajo.

# Capítulo 2

## Conceptos básicos y herramientas para imágenes de obtención remota

### 2.1. Introducción a las imágenes

Existen diversos tipos de imágenes que podemos clasificar, entre otros factores, por la cantidad de información que guardan por cada píxel. Así, por ejemplo, en el caso de las monocromáticas o en escala de grises, por cada uno de los píxeles, se almacena un solo valor, que puede ir comprendido entre 0 y 255 para imágenes de baja profundidad de color (en las que sólo se usa un byte por cada píxel) o entre 0 y 65 535 para las imágenes de alta profundidad de color, en las que se usan dos bytes de información para cada píxel. Estas últimas tienen mucha información lo que hace que sean costosas de editar y, por consiguiente, se usan muy poco (sólo en aquellos casos en los que la precisión que proporcionan justifica su uso).

Las imágenes se pueden clasificar, también, por el número de bandas de información que tienen asociadas. Así, las monocromáticas tienen una sola banda (un solo valor en cada píxel). Las imágenes en color pueden tener

entre 3 y 4 bandas de información por cada uno de los píxeles. Por ejemplo, en las imágenes en color representadas con el modelo de color RGB hay un valor para cada una de las tres bandas de colores básicos (*Red*, *Green*, *Blue*) que, dependiendo de la resolución de la imagen que se quiera tener, ocuparán desde 32 bits por píxel (8 bits por color RGB más 8 del factor alfa, que es el que indica la transparencia del píxel) o hasta 48 bits por píxel (16 por cada uno de los 3 canales RGB). Otro caso especial lo encontramos en las imágenes multibandas, en las que tenemos un número más elevado de bandas (de una decena hasta miles) y cada una de éstas se representa con uno o dos bytes de información por píxel. Así, tenemos imágenes con cientos o miles de bandas de información que representan la refracción de la luz para una determinada longitud de onda.

### 2.2. Imágenes de obtención remota (*Remote Sensing Images*)

Las imágenes de obtención remota (o *Remote Sensing Images*) están compuestas por una gran cantidad de bandas de información. Cada banda de la imagen almacena uno o dos bytes de datos, dependiendo de si la imagen está obtenida con baja o alta profundidad de color, y el conjunto de todas las bandas del mismo píxel se denomina firma espectral, como se muestra en la Figura 2.1. Como se describe más adelante, cada banda almacena la reflexión del píxel para una longitud de onda diferente.

A partir de esta definición, denominaremos “píxel” al conjunto de valores de todas las bandas para una misma posición y “componente” a cada uno de



los valores que tiene cada banda. Es decir, que un píxel estará compuesto por  $b$  componentes, siendo  $b$  el número de bandas que tiene la imagen multibanda. Por ejemplo, una imagen hiperespectral de dimensión  $512 \times 512$  con 224 bandas, tendremos  $512 \times 512$  píxeles con 224 componentes cada uno de ellos.

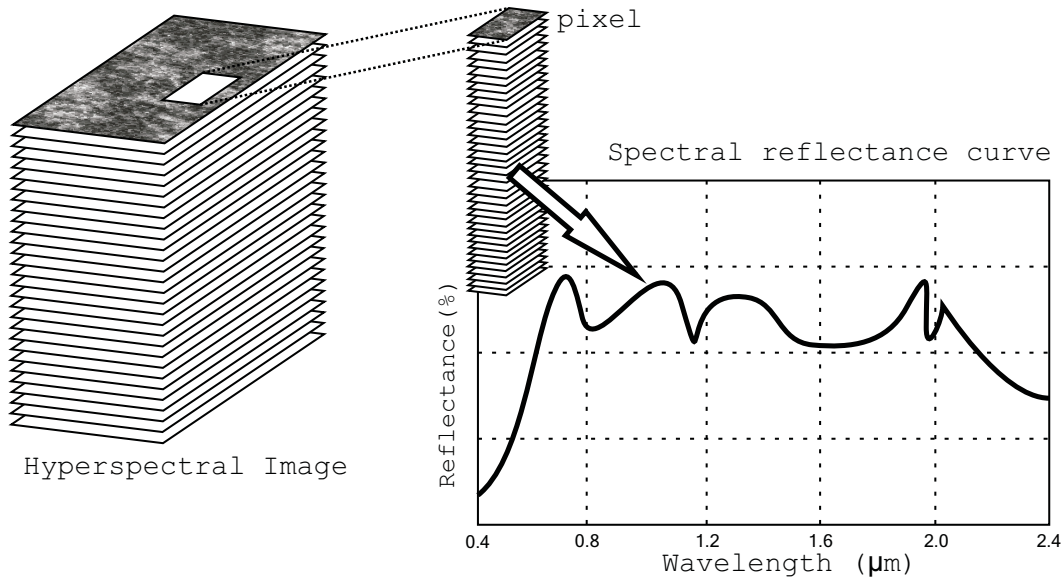


Figura 2.1: Ejemplo de una curva de firma espectral para un píxel.

Cada firma espectral de la imagen proporciona información sobre el material de la zona que se ha fotografiado. Los diferentes valores que hay para cada píxel forman una curva concreta denominada firma espectral. La Figura 2.2 muestra algunos ejemplos de firmas espectrales conocidas para agua clara de un lago, agua turbia de un río, vegetación, suelo árido y suelo húmedo.

En la captura de las imágenes remotas se emplean sensores especiales que se montan en aviones o en satélites, dependiendo de la zona de la cual se quiere obtener la imagen o de la resolución de la misma. Estos sensores recogen la reflexión de la luz en cada uno de los puntos de la imagen y son capaces de cuantificar en diferentes valores de la longitud de onda la reflexión de la luz solar y así obtener información sobre el tipo de material que predomina en esa región concreta. Diferentes materiales producen diferentes

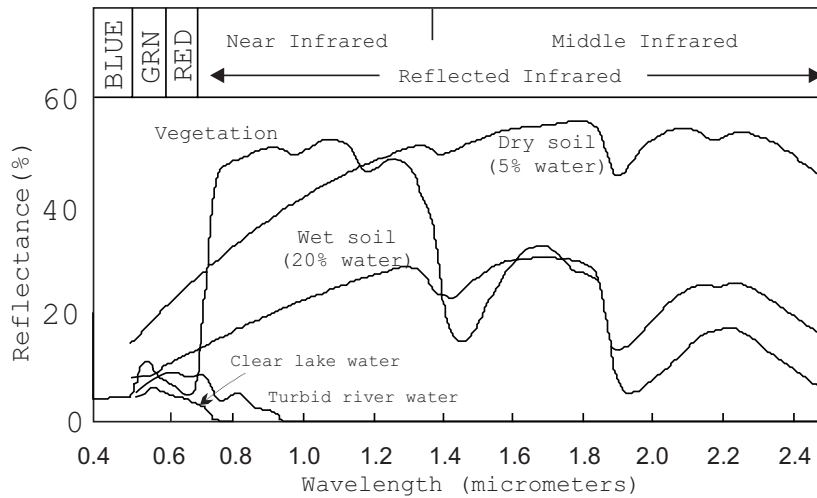


Figura 2.2: Algunas firmas espectrales típicas.

firmas espectrales en la luz reflejada.

Las firmas espectrales obtenidas mediante los sensores, llamadas *Endmembers* (Smith, 2006), se comparan con las curvas establecidas como correctas para cada uno de los materiales conocidos y, de esta manera, se puede realizar una “clasificación” del material más predominante en cada uno de los píxeles a partir de la firma espectral obtenida con el sensor. Hay que tener en cuenta que un sensor puede tener una resolución de entre unos centímetros a unos metros, por lo tanto, un único píxel puede representar una superficie de unos centímetros cuadrados o de unos pocos metros cuadrados. Por eso se dice que el *endmember* es una combinación de todos los materiales que hay en ese punto en concreto. La combinación de todas las firmas espectrales que representa un píxel de una región de  $1 \text{ m}^2$  es el *endmember* obtenido para ese píxel.

Las imágenes de obtención remota se pueden clasificar dependiendo del número de bandas de información que tengan y, por tanto, de la resolución de la firma espectral. De esta manera se pueden dividir en tres grandes grupos:

- Multiespectrales: son aquellas imágenes que sólo tienen unas decenas de bandas.
- Hiperespectrales: son aquellas imágenes que disponen de unos centenares de bandas de información.
- Ultraespectrales: son aquellas imágenes que disponen de miles de bandas.

El sensor que se emplee para capturar la imagen determinará el tipo de imagen que se puede obtener, ya que los sensores cuantizan el ancho de banda del espectro de la reflexión de la luz en diferentes cantidades. Los sensores más baratos y antiguos capturan imágenes multiespectrales, mientras que los más caros y modernos capturan imágenes ultraespectrales.

Como ejemplo, las imágenes Airbone Visible / Infrared Imaging Spectrometer (AVIRIS) (NASA, Jet Propulsion Laboratory, 2004) contienen 224 bandas del espectro y usualmente están almacenadas con alta profundidad de color, es decir, con 16 bits de precisión para valores de intensidad de cada banda. Por otro lado, las imágenes con menor resolución del espectro son las multiespectrales, como por ejemplo las LANDSAT, que únicamente disponen de 8 bandas de información para cada imagen y, además, usualmente, están almacenadas con baja profundidad de color (un solo byte). Las imágenes que tienen más de mil bandas de información son las ultraespectrales y son difíciles de manejar por la gran cantidad de datos que almacenan para una pequeña área de la superficie terrestre.

La obtención de estas imágenes puede ser muy costosa, ya sea en dinero, para poder iniciar el proyecto, como en el manejo de la gran cantidad de información asociada a ellas. El coste del proyecto puede ser muy

## 12 Conceptos básicos y herramientas para imágenes de obtención remota

---

variable, ya que dependiendo de lo que se quiera obtener se deberán montar los sensores en aviones o en satélites.

El proyecto más antiguo de la NASA de obtención de imágenes terrestres es el LANDSAT (NASA, U.S. Geological Survey, 1972). Éste proyecto se inició en 1972 con el satélite LANDSAT1 (se puso posteriormente el nombre del satélite a la misión) y ya se alcanzan los siete satélites lanzados desde entonces, dos de los cuales todavía se encuentran en funcionamiento, el LANDSAT 5 (1984) y LANDSAT 7 (1999).

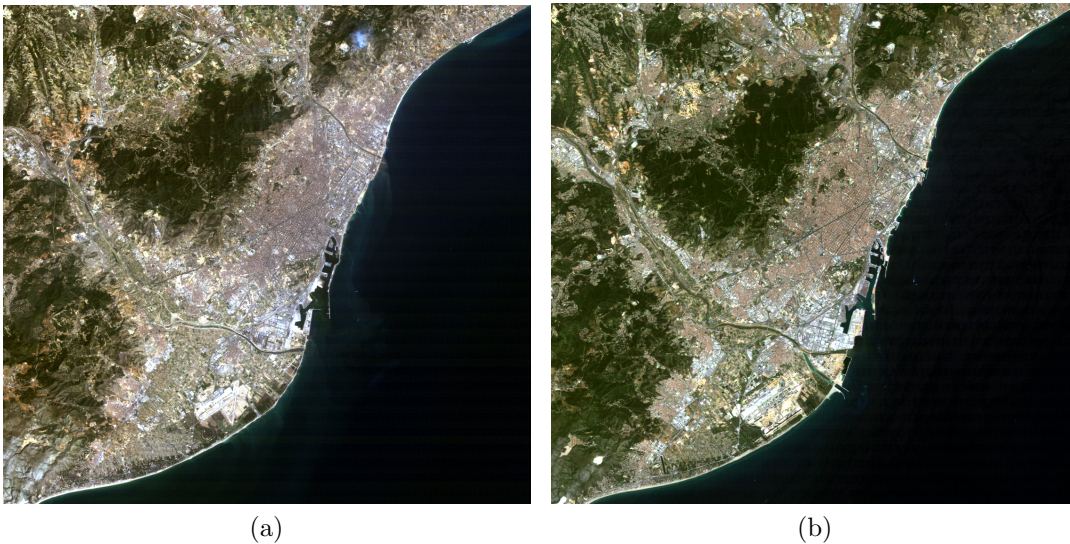


Figura 2.3: imágenes LANDSAT de Barcelona, 1984 (a) y 2004 (b).

El proyecto LANDSAT continúa obteniendo imágenes de la superficie terrestre para poder llevar a cabo estudios científicos, ya sea sobre la deforestación, el clima, el cambio climático, etc. La Figura 2.3 muestra dos imágenes tomadas en diferentes años (1984 y 2004) de la ciudad de Barcelona, en la que se pueden observar cambios en el puerto marítimo o en la desembocadura del río Llobregat. Estas imágenes, en realidad, son la composición de tres bandas hiperespectrales para convertirlas en una imagen a color. Simplemente tomando una banda en cada uno del rango de los colores básicos

visibles, se puede obtener la imagen en color aproximado a la realidad.

## 2.3. Transformadas matemáticas

En este apartado se describen, muy brevemente, las transformadas más utilizadas en el procesamiento de las señales digitales, ya sean imágenes, sonido o vídeo. Más concretamente, se resumen las transformadas más utilizadas: la transformada de Fourier, la transformada discreta del coseno y la transformada discreta *wavelet*. Además de éstas, existe la transformada *Karhunen-Loève* (KLT) más específica para imágenes multibanda, ya que aprovecha mucho mejor la redundancia de la información en las diferentes bandas o en la transmisión de contenido redundante. Sin embargo, en esta tesis no se ha usado esta transformada por centrarse este trabajo en el uso de cuantización vectorial y la aplicación de la transformada *wavelet* para obtener mejores resultados frente a ataques de compresión JPEG2000.

### 2.3.1. Transformada de Fourier

La teoría de Fourier (Balmer, 1996) explica que, mediante la suma de señales elementales de seno y coseno (o las funciones equivalentes de variable compleja) con diferentes amplitudes, frecuencias y fases, es posible construir casi cualquier función. Una de estas señales puede tener frecuencia cero, para representar el término constante de la función (desplazamientos en los valo-

## 14 Conceptos básicos y herramientas para imágenes de obtención remota

---

res), que es conocido como componente continua. La de Fourier es una de las transformadas más utilizadas para simplificar operaciones en el espacio frecuencial. Por ejemplo, la convolución de funciones en el dominio temporal se convierte en una simple multiplicación de funciones en el dominio frecuencial.

La representación gráfica de la transformada de Fourier es un diagrama, llamado espectro de Fourier, donde se representa la frecuencia y amplitud de cada una de las componentes frecuenciales y el desplazamiento respecto a la fase de las funciones trigonométricas. La ecuación 2.1 describe de forma matemática la transformada de Fourier para señales continuas en el tiempo.

$$g(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} f(t)e^{-i\omega t} dt, \quad (2.1)$$

donde  $t$  es el tiempo,  $\omega$  es la frecuencia en radianes,  $f(t)$  la función a transformar y  $g(\omega)$  es el espectro de la función  $f$ .

Por las características discretas de las señales con las que estamos trabajando (imagen, sonido y vídeo digital) se suele emplear la versión discreta de esta ecuación 2.1, que se describe como muestra la ecuación 2.2. Esta versión es la transformada discreta de Fourier (*Discrete Fourier Transform*, DFT).

$$X(k) = \sum_{n=0}^{N-1} x[n]e^{-\frac{2\pi i}{N}kn} \quad (2.2)$$

Para la serie bidimensional (imágenes) usaremos la siguiente ecuación 2.3.

$$X(k, l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n]e^{-\frac{2\pi i}{M}km}e^{-\frac{2\pi i}{N}ln} \quad (2.3)$$

Y para obtener la función a partir de los coeficientes de la transformada de

Fourier, conocida como la transformada inversa de Fourier (*Inverse Discrete Fourier Transform*, IDFT) se aplicará la ecuación 2.4.

$$f(m, n) = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} F[k, l] e^{\frac{2\pi i}{M} km} e^{\frac{2\pi i}{N} ln} \quad (2.4)$$

Usando la fórmula de Euler

$$e^{ix} = \cos x + i \sin x$$

sobre la Ecuación 2.2 o 2.4, se observa que, a partir de la suma de componentes trigonométricos, se puede obtener cualquier función  $f$  unidimensional, como por ejemplo un archivo de sonido, o bidimensional, como una imagen.

A modo de ejemplo muy simplificado se muestran las dos Figuras 2.4 y 2.5, en las que se puede ver una gráfica de un contenido digital que se ha de transmitir y la consiguiente gráfica del espectro de esta señal calculado con la transformada discreta de Fourier. De esta manera, la gráfica discreta de la señal se convierte a una suma de componentes de las funciones de senos y cosenos y resulta más fácil diseñar filtros para estas señales. Esto permite, por ejemplo, la eliminación de ruido en la señal o buscar los componentes de bajas frecuencias que son los que aportarán más información al contenido digital. En la figura del espectro se observa que la señal original se puede obtener mediante la suma de dos sinusoides de frecuencias 50 Hz y 200 Hz.

Evidentemente este caso es para una sola dimensión y con señales sin ruido, pero el concepto de transformada de Fourier queda bien ilustrado con este simple ejemplo.

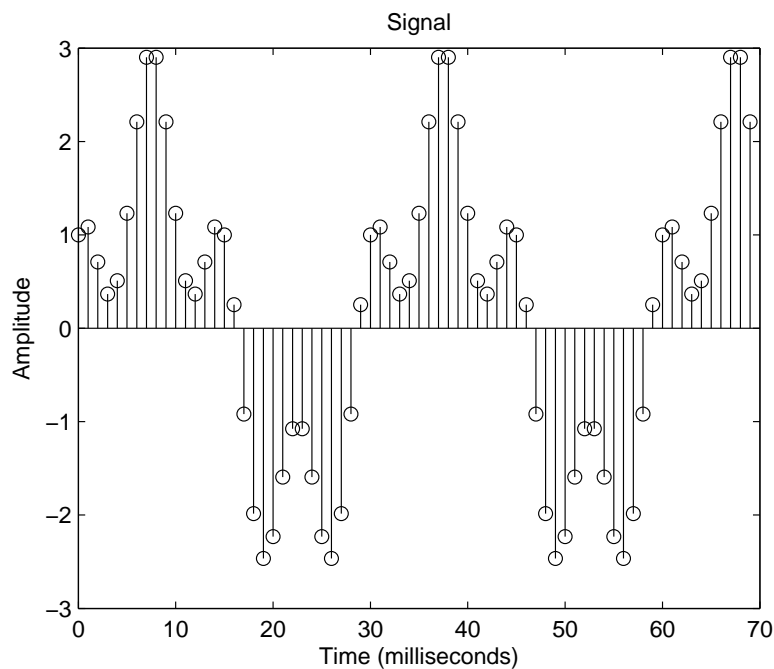


Figura 2.4: Ejemplo de señal discreta.

### 2.3.2. Transformada discreta del coseno (DCT)

La transformada discreta del coseno (*Discrete Cosine Transform*, DCT) (Miano, 1999; Jähne, 2005), es la que más se utiliza en aplicaciones de compresión de imágenes, ya que los coeficientes que produce no están correlacionados y, por tanto, se puede comprimir mejor la imagen sin perder datos por la correlación entre los coeficientes de la transformada que se aplica. Modificar un coeficiente de la transformada no implicará cambios en los demás coeficientes, y de esta forma, cada coeficiente se puede tratar independientemente. Esto también sucede para la transformada KLT, con la diferencia de que los vectores base de la DCT dependen del orden de la transformada y no de las propiedades estadísticas de los datos de entrada, como pasa en la KLT. Otro aspecto importante de la transformada del coseno es la capacidad de cuantificar los coeficientes mediante valores de cuantización.



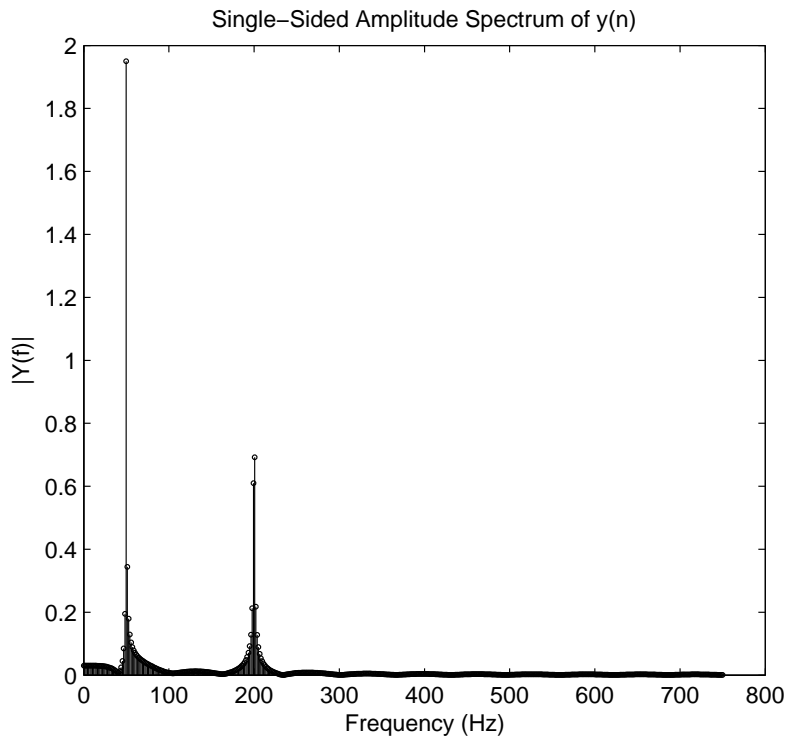


Figura 2.5: Ejemplo de transformada discreta de Fourier.

La DCT bidimensional se define de la manera siguiente:

$$C[k, l] = \alpha(k)\alpha(l) \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} F[m, n] \cos \left[ \frac{(2m+1)k\pi}{2N} \right] \cos \left[ \frac{(2n+1)l\pi}{2N} \right], \quad (2.5)$$

para  $0 \leq k, l \leq N-1$ , donde los coeficientes  $\alpha(k)$  y  $\alpha(l)$  toman los valores siguientes:

$$\alpha(k), \alpha(l) = \begin{cases} \sqrt{\frac{1}{N}}, & k, l = 0, \\ \sqrt{\frac{2}{N}}, & k, l = 1, \dots, N-1. \end{cases}$$

La transformada inversa se muestra en la ecuación 2.6.

$$F[m, n] = \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} \alpha(k)\alpha(l)C[k, l] \cos \left[ \frac{(2m+1)k\pi}{2N} \right] \cos \left[ \frac{(2n+1)k\pi}{2N} \right],$$

para  $0 \leq m, n \leq N-1$   
(2.6)

para  $0 \leq m, n \leq N-1$

Como en el caso de la transformada de Fourier, se puede obtener la función discreta  $F$  a partir de sumas y productos de funciones coseno.

### 2.3.3. Transformada *wavelet*

La transformada *wavelet* (Mallat, 1998) se basa en separar los valores de mayor resolución de los de menor resolución aplicando filtros de forma recursiva. Así, esta transformada crea una serie de bandas que representaran a la imagen en función de la resolución asociada a la que se quiera llegar. Al tratarse de una técnica recursiva, se puede aplicar la transformada hasta un cierto nivel  $D$  deseado. Este nivel  $D$  determinará el número de veces que se aplica la transformada a la banda de menor resolución, por lo que al final se obtendrán  $(2D-1)$  bandas.

Para las imágenes digitales se emplea transformada *wavelet* discreta. Las ecuaciones 2.7 a 2.9 muestran el desarrollo de esta transformada. Por simplicidad, se muestran las ecuaciones de la DWT para una sola dimensión.

El primer paso es aplicar a la señal  $x$  unos filtros pasa bajas (que suprimirán las altas frecuencias de la imagen) con la función  $g$  impulso, para obtener  $y[n]$  como la convolución de las dos señales (Ecuación 2.7).

$$y[n] = (x * g)[n] = \sum_{k=-\infty}^{\infty} x[n] g[n - k]. \quad (2.7)$$

A la vez, la señal se descompone también con filtros pasa altas  $h$  que proporcionarán los detalles de la señal, mientras que el filtro pasa bajas produce la aproximación de la señal. Estos dos filtros tienen que estar relacionados y cumplir con las características de los filtros espejo en cuadratura en los que se separa la señal de entrada en dos partes, alta y baja, y se reduce la dimensión de la señal en dos.

En este proceso, la mitad de las frecuencias se pierden (por el efecto de la aplicación de los dos filtros) por lo que la señal resultante se tiene que reducir según el criterio de Nyquist, lo que obliga a remuestrear la salida del filtro dividiendo las muestras por 2.

$$\begin{aligned} y_{\text{low}}[n] &= \sum_{k=-\infty}^{\infty} x[k] g[2n - k], \\ y_{\text{high}}[n] &= \sum_{k=-\infty}^{\infty} x[k] h[2n - k], \end{aligned} \quad (2.8)$$

resumiendo los sumatorios en la siguiente ecuación 2.9

$$\begin{aligned} y_{\text{low}} &= (x * g) \downarrow 2, \\ y_{\text{high}} &= (x * h) \downarrow 2. \end{aligned} \quad (2.9)$$

donde el operador  $\downarrow$  indica el remuestreo comentado anteriormente.

La convolución entre las funciones  $x$  y  $g$  o  $h$  implica una gran cantidad de tiempo y de recursos. Esto ha provocado que ya se haya desarrollado la segunda generación de transformadas *wavelet*, con los esquemas de “*lifting*” (Sweldens, 1998), donde se obtiene la transformada *wavelet* de forma mucho más sencilla y rápida.

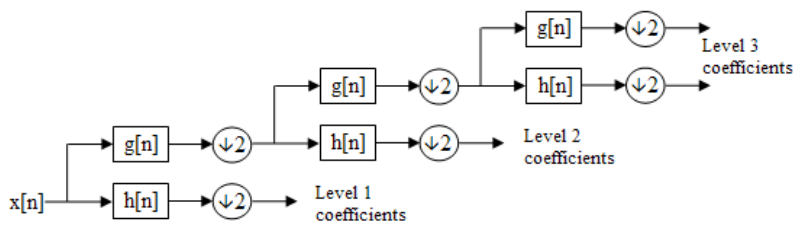


Figura 2.6: Transformada *wavelet* de 3 niveles.

La imagen 2.6 muestra de forma esquemática cómo se obtienen todas las bandas de la DWT para tres niveles. De forma recursiva se aplica la DWT sobre los valores de las bandas bajas, extrayendo así los detalles en cada uno de los pasos. A modo de ejemplo, la Figura 2.7 muestra el resultado de la aplicación de la DWT de tres niveles a la imagen lena.

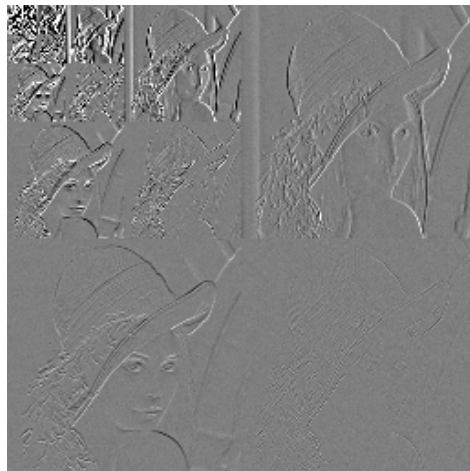


Figura 2.7: *Wavelet* de 4 niveles para la imagen lena.

La transformada *wavelet* se puede utilizar para temas relacionados con la compresión y la transmisión de imágenes y vídeos por Internet, ya que esta transformada permite ir enviando la imagen, sonido o vídeo con diferentes resoluciones por el canal de transmisión. Se empieza enviando la banda más baja para que, con poca información enviada, el receptor ya pueda mostrar la imagen a muy baja resolución. seguidamente se van enviando las siguientes

bandas, hasta llegar a enviarlas todas, con lo que el visualizador del receptor podrá mostrar la imagen o el contenido digital con todos los detalles contenidos en todas las bandas recibidas. La Figura 2.8 muestra cómo se mostraría la imagen una vez el visualizador aplicara la transformada *wavelet* inversa para mostrar el contenido con sólo la banda más baja y, por tanto, sin ningún detalle proporcionado por las bandas más altas.

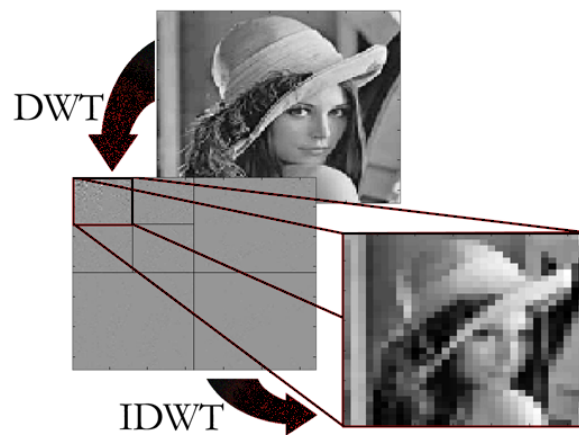


Figura 2.8: Reconstrucción de la imagen lena usando sólo la banda de más baja frecuencia.

## 2.4. Compresión

En este apartado se repasan diferentes herramientas y conceptos relacionados con la compresión de contenido digital, ya sean imágenes, sonido o vídeo.

### 2.4.1. Compresión con pérdida de imágenes de obtención remota

Una de las técnicas más usadas para reducir la gran cantidad de información que se almacena en las imágenes de obtención remota es la aplicación de compresión con pérdida (Aiazzi et al., 2006; Mielikainen and Toivanen, 2006). La compresión con pérdida es aquella en la que una vez se comprime la imagen original y se vuelve a descomprimir, la imagen resultante no es exactamente igual a la original, ya que alguna de las partes menos significativas de la imagen se ha eliminado para reducir el tamaño del archivo digital. Usualmente los algoritmos de compresión eliminan las altas frecuencias para almacenar y reconstruir las bajas frecuencias que son las que más información aportan al contenido digital. Aunque existen aplicaciones de las imágenes de *Remote sensing* que no admiten alteraciones en ella, afortunadamente la compresión hoy en día está muy desarrollada, con lo que se comprime eficazmente a costa de perder muy poca información de la imagen original.

Los métodos de compresión con pérdida eliminan la información del contenido digital que no es relevante para la reconstrucción posterior, es decir, en el momento en que se quiera recuperar el archivo. La información a descartar dependerá, en gran manera, de la aplicación que se quiera dar a ese contenido digital a posteriori. Existen muchos criterios con respecto a la calidad de la imagen, que pueden ser definidos, como se describe en (Cristophe et al., 2005), dependiendo del uso final que se quiera dar a la imagen. Diversos autores (Minguillón et al., 2000a,b) muestran que es posible alcanzar altos ratios de compresión sin eliminar información crítica de la imagen que se está manipulando.

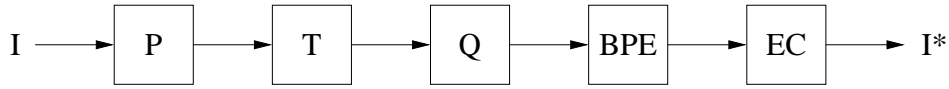


Figura 2.9: Diagrama de bloques del codificador para un método de compresión con pérdida.

La Figura 2.9 muestra un diagrama muy simplificado de los componentes del codificador típico de los esquemas de compresión con pérdida. Éste consiste en cuatro pasos, aunque de hecho, en la mayoría de compresores incluyen un paso previo (P) en el que, si es necesario, se realiza una conversión del modelo de color o una reducción de la dimensión a tratar, para simplificar un poco los siguientes pasos. El esquema se puede describir brevemente de la manera siguiente:

1. Se aplica una transformación (T) en los datos para obtener la decorrelación de los coeficientes de esta transformación y una mayor compactación de la energía de éstos en pocos coeficientes.
2. Se realiza una cuantización (Q) de los datos para eliminar toda la información que no se estima de importancia o útil.
3. Se realiza una codificación de planos de bits (BPE) para dar cuenta de la importancia de los coeficientes cuantizados.
4. Se aplica un esquema de codificación (EC) de alta entropía para reducir la cantidad de bits necesarios para ser enviados a través del canal de transmisión, es decir, en el archivo comprimido.

En el otro lado, en el caso de la descompresión del archivo, el proceso descompresor (descodificador) realiza los mismos pasos en sentido inverso. El objetivo, en este caso, es reconstruir una imagen lo más parecida posible

a la imagen original  $I$ , teniendo en cuenta el *bit rate* de la imagen recibida  $I^*$  que será lo más bajo posible para que la imagen comprimida ocupe lo menor posible maximizando la calidad. Como se comenta más adelante, el hecho de aplicar la cuantización hace que no sea posible obtener la imagen  $I$  exacta a partir de la imagen comprimida  $I^*$ . Es decir, la cuantización no es un paso invertible.

El esquema de codificación, o compresión, se tiene que adaptar a las características esenciales del tipo de imágenes al que van dirigidos, para maximizar la calidad y la ratio de compresión de la imagen. No es lo mismo tratar imágenes en tres dimensiones que una monocromática. El codificador ha de tener en cuenta esta tipología y se deberá realizar un paso previo de sintonización a las características de las imágenes.

En el caso que nos ocupa, la característica más importante de las imágenes de obtención remota es la componente de tres dimensiones intrínseca en ellas. Dos de las dimensiones que tienen estas imágenes son espaciales, mientras que la tercera es espectral y representa la longitud de onda de la luz reflejada en cada punto. Un proceso de compresión ideal para estas imágenes debería tener en cuenta esta característica y aprovechar la redundancia existente entre la parte espacial y la espectral. Si se codifica cada una de las bandas por separado no se podrá alcanzar el mismo ratio que si se aplica un método que tenga en cuenta las tres dimensiones de la imagen, aprovechando la redundancia que existen en cada una de las bandas que representan la misma región de terreno para cada una de las cuantizaciones realizadas del espectro de la luz solar. Existen muchos trabajos que proponen transformaciones 3D para descorrelacionar la parte espacial de la espectral. Por ejemplo (Motta et al., 2005) presenta una propuesta de compresión de imágenes hiperespectrales. En cambio, en esta tesis, se aplica la cuantización vectorial



para realizar la compresión con pérdida teniendo en cuenta las tres dimensiones de las imágenes a la vez. La compresión se realiza procesando todas las bandas de la imagen o un grupo suficientemente representativo de ellas de manera conjunta.

### 2.4.2. Cuantización vectorial y cuantización vectorial en estructura de árbol

La cuantización vectorial (*Vector Quantization*, VQ), como está descrita en (Gersho and Gray, 1992), hace posible la compresión óptima de una imagen desde el punto de vista de la teoría de *rate-distortion* de Shannon. Como se detalla en (Gersho and Gray, 1992; Gray and Neuhoff, 1998), (Shannon, 1948) mostró que, dada una ratio de codificación, la distorsión mínima alcanzable por cuantización vectorial de cualquier tipo es igual a una función, posteriormente llamada función *rate-distortion* de Shannon, la cual viene determinada por las propiedades estadísticas de la imagen original y la medida de distorsión.

La teoría de la función *rate-distortion* de Shannon establece la mínima cantidad de información que debe ser transmitida por un canal para que la fuente original pueda ser reconstruida aproximadamente por el receptor de la transmisión sin exceder de una distorsión dada. Así, aplicando sistemas de compresión VQ, la imagen resultante minimiza la distorsión para una ratio de compresión fijada. Sin embargo, la compresión VQ suele ser computacionalmente prohibitiva. Por lo tanto, se necesitan métodos alternativos que sean subóptimos pero factibles de utilizar en las aplicaciones prácticas reales.

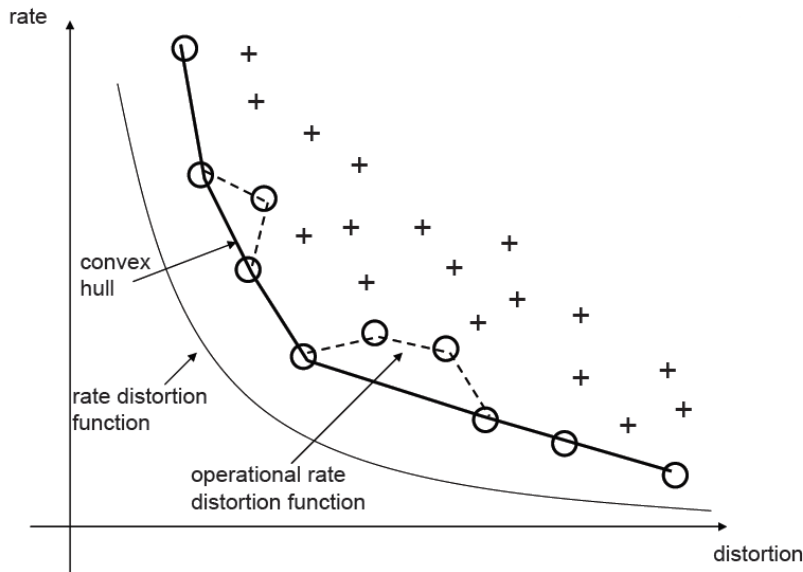


Figura 2.10: Función VQ óptima (*convex hull*) y aproximación con función sub-óptima.

La Figura 2.10 muestra la curva óptima de la cuantización vectorial, señalada como *convex-hull* y la aproximación a este límite que se puede conseguir con otras técnicas que requieren menos cálculos y que se puede implementar con menor coste computacional.

La cuantización vectorial en estructura de árbol (*Tree-Structured Vector Quantization*, TSVQ) es una aproximación subóptima del problema de la cuantización vectorial, la cual funciona empezando con un único valor como libro de códigos (*codebook*), llamado centroide, que forma un árbol con una sola hoja. Este método consiste en aplicar un criterio de calidad (en este caso por ejemplo el error cuadrático) y, si no se ha llegado a este criterio, la hoja con mayor distorsión se divide en dos centroides similares y se aplica el algoritmo Linde-Buzo-Gray, descrito en (Linde et al., 1980), para calcular los dos nuevos centroides resultantes de la división. Este proceso se repite para todos los centroides hasta que se alcance el criterio de calidad, o cuando

todas las hojas del árbol, los centroides, contengan los mismos valores que los reales, teniendo así el árbol  $T$  perfecto, en el cual no existirá compresión ni distorsión.

Para imágenes grandes, donde se dispone de muchos vectores para poder encontrar los centroides, el árbol resultante de este proceso suele ser bastante profundo y en general desequilibrado. Sin embargo, el número de árboles que se pueden generar con este sistema permite explorar muchas posibilidades a la hora de encontrar ciertas características que servirán para los propósitos de incrustar marcas de agua (*watermarking*) en las imágenes generadas a partir de esta compresión.

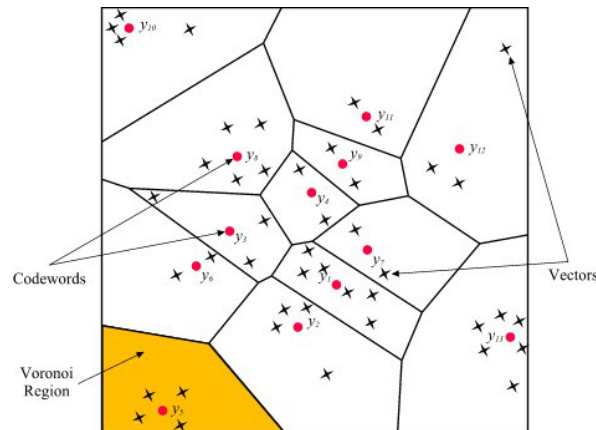


Figura 2.11: Regiones de Voronoi.

La Figura 2.11 muestra cómo queda, en un determinado estado, la división del espacio de los valores de los vectores. Todos los valores (marcados con "x") de la misma región serán sustituidos por los valores del centroide (marcado con un punto) correspondiente. Las regiones de Voronoi (Voronoi, 1907), se calculan a partir del método de división Linde-Buzo-Gray.

Finalmente, se usará la imagen original para codificarla con uno de los subárboles posibles que se obtiene con en el método de creación de todos los posibles subárboles TSVQ, reemplazando cada vector original por el cen-

troide más cercano posible, el que está dentro de su región de Voronoi. Esta operación es muy rápida y eficiente, lo que permite poder codificar la imagen muy rápidamente.

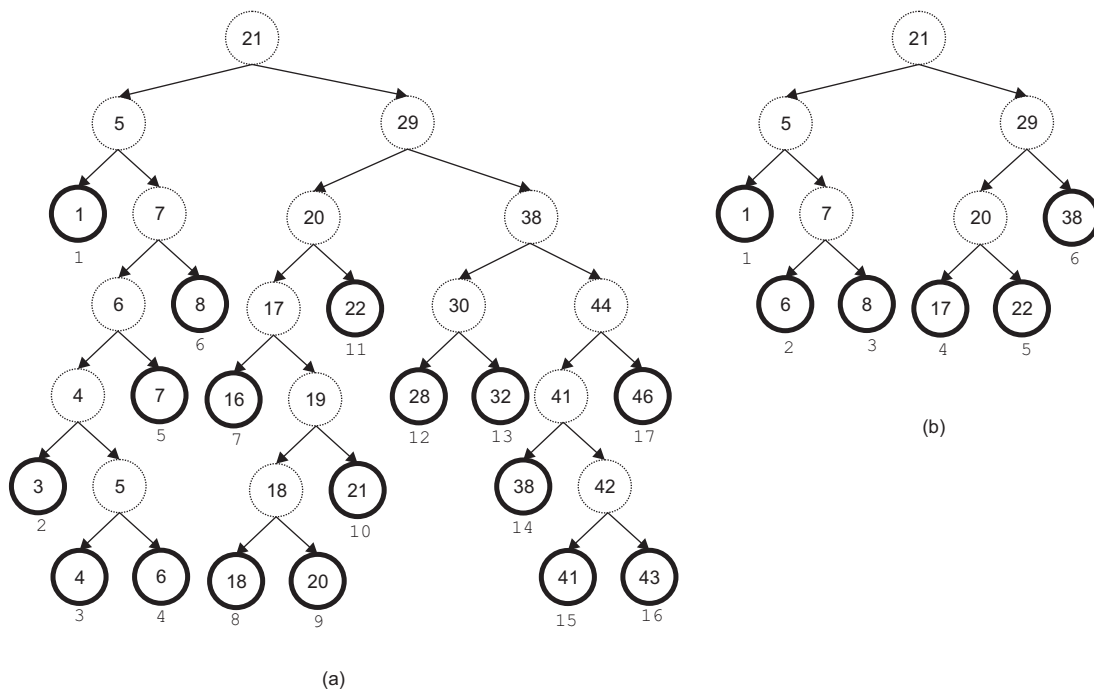


Figura 2.12: Ejemplo de compresión del árbol TSVQ.

La Figura 2.12 muestra un pequeño ejemplo del funcionamiento de los árboles de compresión. En el ejemplo se puede apreciar cómo ha quedado el árbol completo para todos los valores reales en las hojas, es decir, que tenemos un conjunto de 17 valores: 1, 3, 4, 6, 7, 8, 16, 18, 20, 21, 22, 28, 32, 38, 41, 43 y 46. Para representar estos números necesitamos un *codebook* de 5 bits (de 0 a  $2^5 - 1 = 31$ ) que convierta los valores guardados en los reales. En concreto tendremos las asociaciones descritas en la Tabla 2.1

En cambio, si se aplica la poda del árbol, es decir, si seleccionamos un

Tabla 2.1: Ejemplo de *codebook* inicial.

<i>Codebook</i>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Valor	1	3	4	6	7	8	16	18	20	21	22	28	32	38	41	43	46

subárbol de todos los posibles que se pueden obtener del árbol general mediante un criterio ya establecido anteriormente, podríamos encontrarnos con el subárbol (b) de la Figura 2.12, en la que se han cambiado algunos componentes por su centroide. En este caso el *codebook* necesita únicamente 3 bits para poder representar los seis valores diferentes de los centroides.

Tabla 2.2: Ejemplo de *codebook* final.

<i>Codebook</i>	1	2	3	4	5	6
Valor	1	6	8	17	22	38

Esto permite comprimir la información almacenada, al tener que guardar ahora sólo seis datos diferentes y un *codebook* de tres bits. Evidentemente cuanto más grande sea el archivo, más diferencia habrá entre los ficheros que representan a los nodos finales, los centroides.

El proceso de compresión acaba una vez se hacen coincidir los valores de los componentes finales con los centroides asignados en el subárbol (b) de la Figura 2.12.

### 2.4.3. *Joint Photographic Experts Group (JPEG)*

La compresión de imágenes que está más generalizada es la compresión JPEG, que se ha convertido en un estándar de facto, aunque actualmente

## 30 Conceptos básicos y herramientas para imágenes de obtención remota

---

no es la que mejores resultados obtiene, aunque sí sigue siendo de las más rápidas.

El estándar JPEG de compresión de imágenes está basado en la DCT. En uno de los pasos se aplica esta transformada para reducir el tamaño y aplicar cuantización sobre ellos. Básicamente, los pasos que sigue el algoritmo JPEG son los siguientes:

1. Convierte la imagen al espacio de color YCbCr.
2. Divide cada una de los planos de color en bloques de  $8 \times 8$  píxeles.
3. A todos los bloques se les aplica la DCT.
4. Se cuantizan los coeficientes al entero más próximo a partir de una tabla fija preestablecida y el valor de calidad que se asigna a la imagen comprimida.
5. Los coeficientes cuantizados de la DCT resultantes son comprimidos usando un codificador de alta entropía, como puede ser un código de Huffman (Huffman, 1952).

El paso inverso, el descompresor, únicamente ha de descuantizar los coeficientes que se han obtenido en la compresión, multiplicando los valores por una constante que aproxima el proceso inverso de la cuantización. Esto provoca que la imagen resultante no sea idéntica a la original, ya que la cuantización no es reversible, pero es tan parecida a la original que el ojo humano no lo puede detectar.

Existen muchos codificadores y decodificadores de JPEG ya que es una de las técnicas de compresión más utilizadas hasta la fecha.

#### 2.4.4. JPEG2000

El proyecto JPEG2000 (Taubman and Marcellin, 2002) nace en el año 1996 en la empresa RICOH con la necesidad de mejorar el rendimiento del ya conocido compresor de imágenes JPEG. En el año 1997, en la reunión global en Sydney, se evalúan las 24 propuestas para mejorar o crear un nuevo algoritmo y se adopta la propuesta *Wavelet/Trellis Coded Quantization* (WTCQ) como la que mejor se adapta a las especificaciones establecidas anteriormente. En marzo de 1998 en Ginebra, Suiza, se presentan la primera ronda de experimentos a partir de un núcleo de 40 imágenes seleccionadas en la reunión de Sydney y se crea un modelo de verificación (*Verification Model*, VM) para JPEG2000, que se actualizará cada reunión global a partir de los experimentos obtenidos con el algoritmo seleccionado y el modelo de verificación.

Básicamente el algoritmo WTCQ se compone de la transformada *wavelet*, el TCQ (Kasner et al., 1999; Taubman and Marcellin, 2002) usando tamaños de paso elegidos a través de la asignación de Lagrange y una codificación binaria de las subbandas generadas por los índices del TCQ (*trellis quantized wavelet coefficients*).

Se realizan dos reuniones más y en la tercera, en noviembre de 1998 en Los Angeles (EEUU), David Taubman presenta el algoritmo *Embedded Block Coding with Optimized Truncation*, (EBCOT). Este algoritmo incluye la idea de dividir cada subbanda de coeficientes en bloques rectangulares y realizar la codificación de planos de bits en cada uno de estos bloques y no sobre la subbanda entera como se realizaba en los anteriores modelos de verificación. Esta propuesta reduce considerablemente los requerimientos de memoria y, además, incluye una sintaxis eficiente para formar la codificación de los blo-

ques de los coeficientes, dejando una gran flexibilidad en la formación de estos bloques. A partir de esta tercera reunión se realizan seis más en la que se van incluyendo algunas mejoras al algoritmo EBCOT para que, en enero de 2001, se obtenga la norma ISO/IEC 15444-1 que designa la parte 1 (características básicas de la compresión JPEG2000). Posteriormente, a partir del 2002, se fueron obteniendo las siguientes partes designadas para desarrollar: implementación en el 2003, extensiones en el 2004, vídeo en el 2007, aspectos de seguridad en el 2008, herramientas de interacción en el 2008, etc.

Básicamente, y de forma muy resumida, el compresor JPEG2000 sigue los siguientes pasos:

- Transformación de color a YCbCr o YUV.
- Cálculo de la transformada *wavelet*.
- Creación de *codeblocks* (*tiling*) para ser codificados independientemente.
- Para cada *tiling*:
  - Cuantización de los coeficientes de la DWT.
  - Codificación con *MQ-coder*.
- Creación de la imagen comprimida completa.

### 2.4.4.1. Transformación de color

El primer paso se realiza para descorrelacionar los valores que después se obtengan en los siguientes pasos. En el caso de la transformación YCbCr se perderán datos y no se podrá realizar el proceso inverso y obtener exactamente la imagen original.



### 2.4.4.2. Cálculo de la transformada *wavelet*

Se aplica la transformada *wavelet* que, en función de la calidad que se requiera y la ratio de compresión, implica más o menos niveles a la transformada. El objetivo de este paso es compactar la información de la imagen en una parte muy pequeña, en la banda de más bajas frecuencias. Así, a partir de esta banda se puede ir generando la imagen transmitiendo el resto de bandas a posteriori. Es decir, que a la hora de transmitir una imagen comprimida se hará a partir de esta banda y de frecuencias más bajas a frecuencias más altas.

La Figura 2.13 muestra un ejemplo de cómo son las diferentes bandas de los coeficientes de la DWT. La banda con frecuencias más bajas se vuelve a subdividir en la siguiente iteración de la DWT. Las bandas con altas frecuencias apenas contienen información relevante para la imagen.

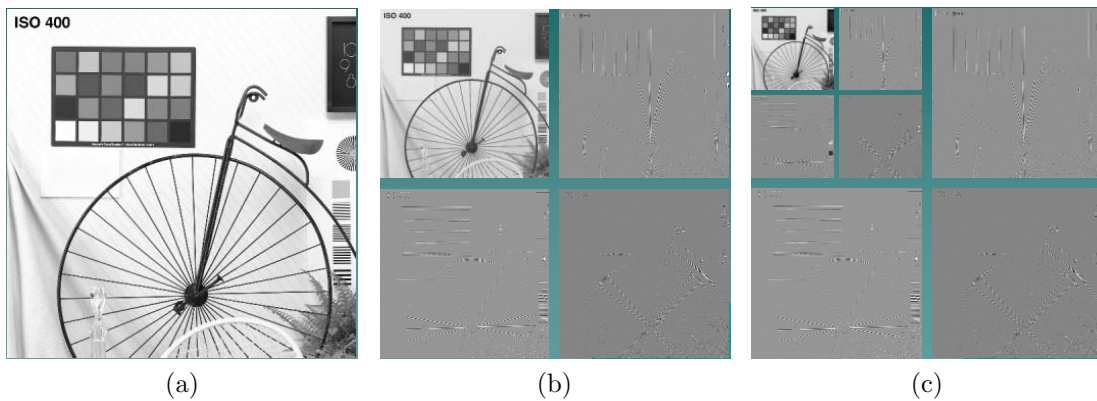


Figura 2.13: Imagen original (a), DWT 1 nivel (b) y DWT 2 niveles (c).

### 2.4.4.3. *Tiling*

Se permite que las imágenes puedan ser divididas en partes rectangulares iguales, a excepción de los bordes que pueden ser diferentes, para simplificar

## **34 Conceptos básicos y herramientas para imágenes de obtención remota**

---

los cálculos. No es necesario dividir las imágenes y es posible crear una sola imagen para tratar que contenga todos los píxeles.

### **2.4.4.4. Cuantización**

Los coeficientes de la DWT se cuantizan teniendo en cuenta los criterios de calidad y ratio de compresión. Existe, al igual que en el caso de JPEG, una tabla de cuantización en el caso de que se quiera utilizar.

### **2.4.4.5. Codificación**

Los valores cuantificados en el paso anterior se transforman con el codificador aritmético binario MQ, que mejora en rendimiento del codificador más conocido QM.

### **2.4.4.6. Ventajas**

Una de las ventajas más importantes de JPEG2000 es que el proceso de compresión es único para las diferentes aplicaciones que se quiera dar a la imagen. A partir del fichero comprimido se puede extraer la imagen con diferentes ratios de compresión, se puede transmitir por un canal enviando a partir de las bandas de más bajas frecuencias para acabar enviando las de más altas frecuencias, o extraer o enviar una parte la de imagen. Se puede resumir esto con la frase “se comprime una vez para descomprimir de muchas maneras” (*One codestream, many applications*).

## Capítulo 3

# Estado del arte de los esquemas de watermarking

### 3.1. Historia y situación actual

Hace miles de años ya se usaban técnicas de ocultación de información para poder pasar mensajes de forma secreta y segura. Las Historias de Herodotus de Halicarnassus (484 AC - 425 AC) (Katzenbeisser and Petitcolas, 1999) explican lo siguiente:

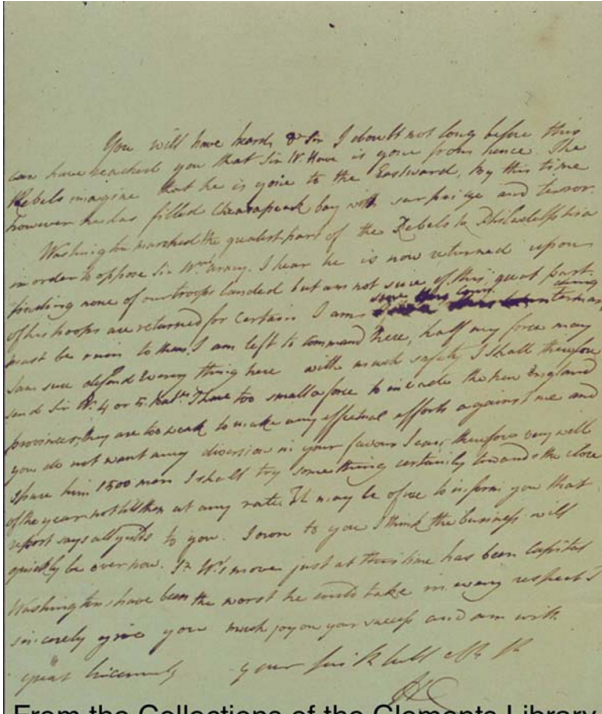
*A Histiaeus no le gustaba vivir en Susa (Susa es una ciudad de la provincia de Khuzestan en Irán) e hizo planes para restaurar su poder a la ciudad de Miletus instigando una revuelta a Ionia, las dos ciudades de la antigua Turquía. En el año 499 antes de Cristo afeitó la cabeza de su esclavo más seguro y leal, le tatuó un mensaje en la cabeza y esperó hasta que el pelo volviera a crecer y tapar toda la cabeza. El esclavo fue enviado a su yerno Aristagoras en Miletus, viajando el mensaje absolutamente desapercibido. Aristagoras tenía instrucciones específicas de afeitar la cabeza del esclavo otra vez y leer*

*el mensaje, en que le decía que iniciara la revuelta contra los persas desde dentro de la misma ciudad.*

Otra técnica pionera en la ocultación de información fue en la antigua China (Katzenbeisser and Petitcolas, 1999). Se tiene constancia de transferencia de mensajes esteganográficos en que, para ocultar los mensajes dentro de otras informaciones, tanto el emisor como el receptor disponían de una copia idéntica de un papel con agujeros que servía de máscara. Colocando la máscara sobre un papel en blanco, escribían en los agujeros los caracteres del mensaje que necesitaban transmitir de forma oculta. Después, se sacaba la máscara y se llenaba el papel con mensajes arbitrarios, de forma que los caracteres del mensaje quedaran bien ocultos dentro el texto. Esta técnica fue reinventada el año 1550 por el matemático italiano Gerolamo Cardano y es conocida como rejilla de Cardano (*Cardan grille*).

A continuación se comenta un caso famoso por el uso de esta técnica.

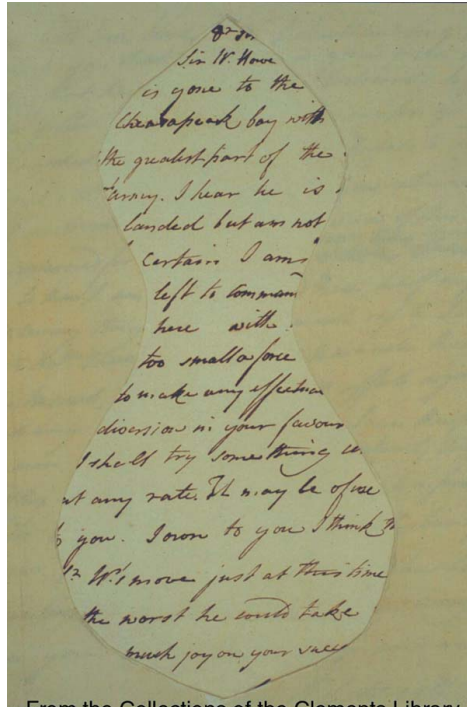
Durante las tres semanas anteriores al 10 de agosto de 1777, en la guerra de independencia de EEUU, en el estado de New York, el general William Howe del ejército inglés se acercó a Filadelfia con las tropas. El general Henry Clinton trató de convencerlo de cooperar y cumplir con las órdenes del comandante Burgoyne de quedarse en el estado de Nueva York. Howe, sin embargo, no se apartaría de su plan y dejó órdenes a Clinton para cubrir Manhattan, Staten Island, Long Island y las Highlands. En ese momento, Henry Clinton envió una carta al general John Burgoyne en la que alababa y exageraba los éxitos militares de los soldados británicos frente al ejército de los EE.UU. Temiendo que la carta pudiera ser leída por los comandantes del ejército de los EE.UU., escondió en ella otro mensaje, el que realmente quería que leyese el comandante Burgoyne.



You will have heard &c. I don't not long before this  
 can have heard you that Sir W. Howe is gone from hence. The  
 He tells imagine that he is gone to the Westward, by this time  
 however he has filled Chesapeake bay with his ships and troops  
 Washington marched the greater part of the Rebels to Philadelphia  
 in order to oppose Sir W. Howe. I hear he is now advanced upon  
 spending some of our troops landed but am not sure of this part  
 of the troops are ordered for capture. I am ~~sure~~ ~~that~~ ~~the~~ ~~troops~~  
 must be ruin to them. I am left to command here, half my force may  
 can see defend every thing here with much safety. I shall therefore  
 send Sir W. G. or G. H. I have too small a force to invade the New England  
 provinces, they are too weak to make any effectual efforts against me and  
 you do not want any divisions in your favour. I am therefore very well  
 of the year, whether at any rate. It may be of use to inform you that  
 support myself with you. I own to you I think the business will  
 quickly be over now. Sir W. Howe just at this time has been captured  
 Washington have been the worst he could take in every respect  
 in early of you much joy on your success and am with  
 your sincerely your friend &c. &c. &c.

From the Collections of the Clements Library

(a)



Sir W. Howe  
 is gone to the  
 Chesapeake bay with  
 the greater part of the  
 Army. I hear he is  
 landed but am not  
 certain I am  
 left to command  
 here with  
 too small a force  
 to make any effectual  
 divisions in your favour  
 I shall try something  
 at any rate. It may be of use  
 to you. I own to you I think  
 Sir W. Howe just at this time  
 the worst he could take  
 much joy on your success

From the Collections of the Clements Library

(b)

Figura 3.1: Carta completa (a) y ejemplo de rejilla de Cardano (b).

En dicha carta, en realidad, manifestaba su frustración con las acciones del general William Howe, dejándolo sin soldados para poder ayudarlo con efectividad y comentaba la que era la peor decisión que se podía tomar en ese momento 3.1b. Clinton recibió el permiso para mover sus escasas tropas para ayudar a Burgoyne, pero no fue suficiente y se perdió la batalla de Saratoga frente a los secesionistas de EE.UU. Al año siguiente el general Clinton sustituyó al general Howe al mando de las tropas británicas.

You will have heard, Dr Sir I doubt not long before this can have reached you that Sir W. Howe is gone from hence. The Rebels imagine that he is gone to the Eastward. By this time however he has filled Chesapeak bay with surprize and terror. Washington marched the greater part of the Rebels to Philadelphia in order to oppose Sir Wm's. army. I hear he is now returned upon finding none of our troops landed but am not sure of this, great part of his troops are returned for certain. I am sure this countermarching must be ruin to them. I am left to command here, half of my force may I am sure defend everything here with much safety. I shall therefore send Sir W. 4 or 5 Bat. I have too small a force to invade the New England provinces; they are too weak to make any effectual efforts against me and you do not want any diversion in your favour. I can, therefore very well spare him 1500 men. I shall try some thing certainly towards the close of the year, not till then at any rate. It may be of use to inform you that report says all yields to you. I own to you that I think the business will quickly be over now. Sr. W's move just at this time has been capital. Wahingtons have been the worst he could take in every respect. sincerely give you much joy on your success and am with great Sincerity your. HC

Figura 3.2: Carta original

Sir. W. Howe  
is gone to the  
Chesapeak bay with  
the greatest part of the  
army. I hear he is  
landed but am not  
certain. I am  
left to command  
here with  
too small a force  
to make any effectual  
diversion in your favour.  
I shall try something  
at any rate. It may be of use  
to you. I own to you I think  
Sr W's move just at this time  
the worst he could take.  
Much joy on your success.

Figura 3.3: Texto obtenido con la rejilla de Cardano

Todo y que el papel se inventó aproximadamente sobre el año 200 antes de cristo en China, durante la dinastía Han Occidental, no es hasta el año 1282 en Bolonia, Italia, en que aparecen las primera muestras de papel marcado de forma invisible o, en este caso, prácticamente invisible. En estos primeros momentos, se incrustaba una marca de agua en el papel mediante la inclusión de un fino alambre con la forma deseada en los moldes de fabricación del papel, que recibe el nombre de filigrana. De esta manera, el papel era más fino en ese punto y, por lo tanto, más traslúcido, dejando ver la marca insertada en el papel si se miraba a contraluz. Actualmente el papel de buena calidad se sigue marcando mediante la presión de un gran rodillo que contiene las marcas cuando el papel todavía está húmedo.

Las primeras muestras de aplicaciones (Moulin and Koetter, 2005) comerciales en que se camuflaba información para proteger la propiedad intelectual fueron en los siglos XVII y XVIII, cuando en tablas logarítmicas se introducían errores concretos en algunos dígitos menos significativos de algunas entradas. De este modo, en caso de ser copiadas, se podía demostrar que las tablas plagiadas eran copias de las originales. También es conocido que los editores de diccionarios antiguos introducían palabras inexistentes (entradas del diccionario erróneas) para detectar copias de sus libros (Alford, 2005). Hay referencias del uso de esta técnica de protección de diccionarios que se remontan al siglo XIX.

Uno de los principales usos que han tenido estas técnicas ha sido la protección del papel moneda. Ya por los inicios de la edición del papel moneda en Europa se usaron técnicas de marcado para que no fuera fácil falsificarlo. Como anécdota, en el año 1779 la revista *Gentleman's magazine* (Mathison, 1779) publicó la historia de John Mathison, que encontró la manera de falsificar la marca del papel moneda. A cambio del perdón, se ofreció a contar

y enseñar cómo lo había hecho para poder mejorar el proceso de marcado. Dicha confesión y ayuda no fue convincente para el banco y John Mathison fue ejecutado.

Actualmente, con la gran cantidad de información digital almacenada en sistemas informáticos, las posibilidades de esta técnica son enormes. Es posible incrustar información secreta en muchos tipos de contenidos diferentes, como por ejemplo, fotografías digitales, archivos de sonido, películas, etc. En el caso de contenidos digitales, esconder un mensaje es mucho más sencillo, debido a la facilidad con que se puede conseguir que el mensaje permanezca prácticamente imperceptible.

Hoy en día, se sabe que los miembros de algunas organizaciones terroristas usan técnicas de esteganografía para comunicarse entre sí. El periódico americano *New York Times*, en octubre de 2001, ya anunció que la organización Al-Qaeda había usado técnicas esteganográficas de ocultación de mensajes en imágenes que se enviaron por e-mail para preparar el atentado del 11 de septiembre de 2001 en Estados Unidos.

El watermarking digital es una técnica consistente en insertar una cierta información, denominada “marca”, en un objeto digital, denominado “objeto portador”. Por lo tanto, el watermarking es un caso especial de técnica esteganográfica. Ahora bien, hace falta tener presente que, en general, en la esteganografía lo más importante es el mensaje oculto, mientras que el objeto portador tiene un interés secundario o nulo. En el caso del watermarking, en cambio, es muy importante proteger el objeto portador que contiene la marca y ésta es simplemente una herramienta que sirve para la protección del contenido en algún aspecto. El término inglés “watermarking” se traduce a menudo como “marca de agua”, porque en el supuesto de que el contenido



sea una imagen la marca es un patrón (casi invisible) que se superpone con la imagen portadora para producir la imagen marcada.

Los esquemas de protección del copyright electrónicos basados en el principio de protección contra la copia digital se han visto insuficientes en los últimos años, como se describe por ejemplo en (Petitcolas et al., 1998; Petitcolas and Anderson, 1999). Uno de los claros ejemplos de aplicaciones contra la copia han sido los sistemas de gestión de derechos digitales (*Digital Rights Management*, DRM) que incorporan algunos DVD (Bell, 1999), y su fracaso ha sido evidente.

Intentar realizar un sistema en el que se impida la copia es muy difícil, y aquí es donde entran en juego las técnicas de watermarking, en las que no se impide la copia digital de los medios pero, por contra, se detecta quién ha realizado la distribución del contenido. Aún más allá, las aplicaciones de *fingerprinting*, o huella digital, consisten en incrustar una marca diferente para cada comprador del contenido, lo que permite identificar a un cliente concreto si éste decide distribuir el contenido de manera ilegal.

## 3.2. Aplicaciones de los esquemas de watermarking

En este apartado se describen las principales aplicaciones (Moulin and Koetter, 2005) en los que se aplican técnicas de marcado imperceptible al contenido digital.

### 3.2.1. Protección de los derechos de autor

La gran expansión de las tecnologías de la información y las comunicaciones ha generalizado el uso de la información en formato digital, cosa que ha facilitado la distribución de réplicas idénticas de los contenidos a través de la red Internet. Este hecho tiene dos repercusiones muy negativas en el ámbito socio-económico. Por un lado, se frena el desarrollo tecnológico de la sociedad. Los fabricantes y creadores del contenido no se atreven a lanzar sus productos en el nuevo formato por miedo de la distribución masiva de copias fuera del mercado. Por otro lado, las pérdidas económicas derivadas de fenómenos como Emule u otros sistemas de igual a igual (*peer-to-peer*, P2P) actuales son muy elevadas. Es evidente, por lo tanto, la necesidad de mecanismos tecnológicos que protejan la propiedad intelectual de los contenidos digitales. En los sistemas de protección de los derechos de autor a veces interesa identificar al productor del contenido y, en este caso, la marca incrustada identifica de manera única el productor, cosa que permite que éste demuestre la propiedad intelectual de sus documentos digitales (Katzenbeisser and Petitcolas, 1999). En otras ocasiones, interesa que la marca identifique el usuario o comprador del contenido para detectar posibles usos fraudulentos de esta información (como por ejemplo la distribución ilegal por parte del comprador de un CD). En esta situación los esquemas de marcado se denominan *fingerprinting* (Liu et al., 2005) y consisten en insertar un identificador diferente para cada comprador que pueda ser detectado cuando se identifica una distribución no permitida del contenido. En cualquiera de los casos, la marca incrustada no se debería poder borrar cuando se modifica el objeto marcado a no ser que éste quede inservible.

En el caso de las imágenes en papel, el propietario de los derechos incrusta

una marca visible en una de las esquinas de la imagen. A simple vista se puede comprobar quién tiene la propiedad intelectual de esa imagen. ¿Pero qué pasa si esa imagen se recorta o modifica?



Figura 3.4: Parte inferior de la imagen de lena.

Dependiendo de cómo se recorte la imagen, la marca visible incrustada en la imagen se perderá completamente. Por ejemplo, una de las imágenes más utilizada en trabajos de investigación, ya sea de watermarking, compresión o tratamiento de imágenes, es la fotografía de Lena Söderberg, modelo sueca que fue la *playmate* de la página central de la revista Playboy de noviembre de 1972, bajo el seudónimo de Lenna Sjööblom. La imagen real tomada por Dwight Hooker contiene la marca de la propiedad intelectual de esta imagen. Como se puede ver en la Figura 3.4, sí existe una marca en la imagen original, aunque no en la que se usa habitualmente para la investigación en compresión o tratamiento de imágenes. En este caso la revista Playboy ha renunciado a perseguir a las personas que usan esta imagen por la cantidad de investigadores involucrados (Brown, 1997).

Actualmente, este problema no se produce si se incrusta la marca en el objeto digital. En este caso la marca también es digital y, además, debe ser robusta a los recortes en el contenido. De poco serviría incrustar una marca que desapareciera simplemente recortando el objeto.

### **3.2.2. Copia y control de acceso**

En aplicaciones de control de copia o control de acceso (Craver and Stern, 2001), la marca incrustada representa la validez del contenido. Después de detectar la marca correctamente se descifra la copia o se permite el acceso, que se controla mediante hardware o software especial. Estas aplicaciones necesitan de algoritmos de watermarking robustos contra ataques ya que sirven para validar el acceso o no al contenido.

### **3.2.3. Autenticación del contenido**

En este caso (Arnold et al., 2003), las marcas se incrustan en el contenido no para proteger la propiedad intelectual, sino para que el usuario pueda comprobar la procedencia y autenticidad de los datos. Esta clase de aplicaciones se encuentra, por ejemplo, en la distribución de imágenes de captura remota (obtenidas por satélite o aviones). Estas imágenes acostumbran a tener un coste muy elevado y los usuarios han de estar protegidos frente posibles falsificaciones.

### 3.2.4. Garantía de la integridad del contenido

A veces no sólo interesa garantizar la procedencia de un contenido, sino que el usuario también puede necesitar comprobar que nadie ha realizado ninguna manipulación fraudulenta. En este caso, la marca debería ser frágil de forma que un pequeño cambio en el objeto marcado provoque la pérdida de la marca y, por lo tanto, permitir la detección de una manipulación (*tampering detection*). En esta misma línea, encontramos aplicaciones que no sólo permiten detectar si se ha producido un cambio fraudulento, sino que también hacen posible identificar qué parte del contenido se ha modificado. Esta clase de aplicaciones se conocen como “técnicas forenses” (Arnold et al., 2003).

Estos esquemas tratan los datos del contenido digital para que tengan una determinada propiedad diferenciándolos de los datos originales. Así, si una persona, ya sea de forma intencionada o no, modifica el contenido de alguna región del objeto, se detectará esa modificación y se podrá concluir que el objeto marcado ha sido modificado en alguna parte. En el caso especial de que se quiera, además, localizar la posible modificación del contenido digital, los objetos se dividen en regiones más o menos grandes, que permitirán a posteriori afirmar si se ha modificado esa área o si se ha mantenido intacta. Existen algunos métodos (Caldelli et al., 2006) en los que la región es muy pequeña, pero esto obliga a forzar propiedades o valores diferentes en todos los datos, por lo que el error introducido será mucho más visible o audible, provocando que el resultado final del proceso de marcado sea de inferior calidad. Por el contrario, con regiones un poco más grandes, las modificaciones son menores, ya que no hace falta tener tantos valores diferentes de la propiedad con la que se marca de manera que se reduce la diferencia entre el

valor original y el valor marcado. Por lo tanto, el error que se introduce en el contenido es mucho menor y la calidad perceptual del objeto es mayor.

En la actualidad, la mayoría de métodos que se implementan tienen detección de modificaciones. De poco sirve a día de hoy determinar que un contenido se ha modificado si no se detecta dónde. Es más, ya se está trabajando en restaurar el contenido del objeto a los datos originales sin marcar, con lo que en el último proceso tendríamos la imagen original sin ninguna variación. Estos métodos se conocen como watermarking reversible (Lee et al., 2007; Tian, 2002).

### **3.2.5. Monitorización de las emisiones**

Otra posible aplicación del watermarking es la monitorización automática de las emisiones de medios de comunicación (Megías et al., 2010), como la radio o la televisión. Esta aplicación permite detectar cuántas veces se emite un determinado contenido durante un cierto período de tiempo. En las actuales emisiones digitales, tanto de televisión como de radio, estas técnicas permiten detectar, en tiempo real y de forma automática, qué se está emitiendo en todo momento. Así, esto permite tener un control automático de las canciones que se emiten en las radios o televisiones y, lo que hace más útil estas técnicas, el control de las emisiones de los anuncios publicitarios. Éstos tienen diferentes precios según la franja horaria en la que se contratan. Con estas técnicas, las agencias de publicidad pueden controlar de forma automática la emisión de sus anuncios con sólo recuperar las marcas de los anuncios emitidos en tiempo real.

### 3.2.6. Información oculta

En algunos casos interesa poder ocultar información dentro del propio contenido de forma imperceptible. Datos como el propietario, la licencia de uso o metadatos específicos de cada contenido pueden ser incrustados dentro del contenido digital. En estos casos es importante que el esquema sea robusto ya que un ataque podría eliminar toda esta información. Por ejemplo, una simple compresión MPEG podría eliminar los datos del propietario legal del contenido. Se podría incrustar un identificador a una base de datos externa en la que se almacenasen todos los datos deseados.

## 3.3. Propiedades de los esquemas de watermarking

La Figura 3.5 muestra en un esquema muy simple las propiedades y elementos que intervienen en los esquemas de watermarking.

A partir del objeto original, y mediante la incrustación de la marca en él, se obtiene la versión del mismo objeto marcada. La diferencia entre el objeto original y el marcado determina la transparencia o imperceptibilidad de la marca. Este objeto marcado puede ser atacado, intencionalmente o no, para eliminar la marca o para falsificarla. Se emplean diferentes técnicas de ataques dependiendo del tipo de contenido. En el caso de las imágenes se pueden realizar conversiones de tamaño, compresión, rotaciones, etc. y para el caso de objetos de audio pueden ser ataques tan simples como la compresión, la inclusión de ruido blanco, el eco, o transformadas frecuenciales. La robustez

determinará cuáles de estos ataques es capaz de soportar el algoritmo sin que se elimine la marca incrustada y, por tanto, el algoritmo de detección podrá determinar si está presente la marca o si el método puede, o no, extraer la marca incrustada.

A continuación se definen muy brevemente las propiedades básicas de los esquemas de marcado (Fridrich, 1998; Dittmann et al., 2006).

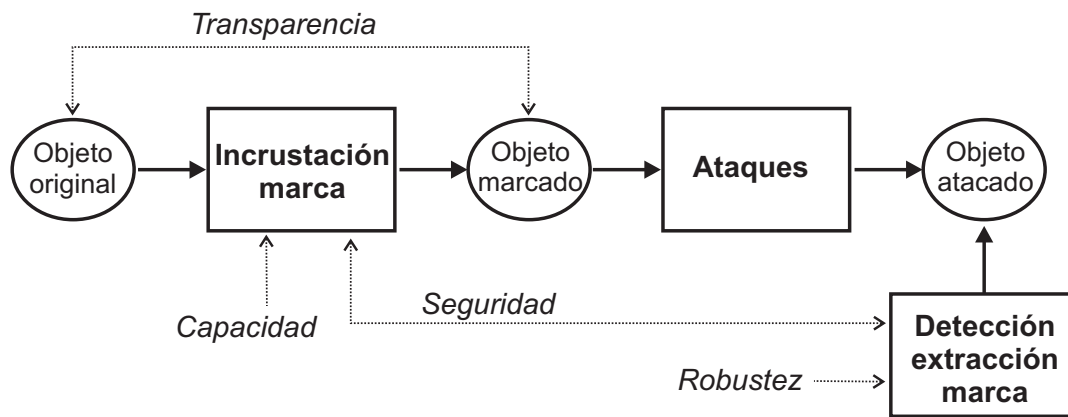


Figura 3.5: Esquema genérico del watermarking

### 3.3.1. Capacidad

La capacidad de un esquema de marcado mide la cantidad de bits de información que se pueden incrustar por unidad de tiempo para contenidos de audio, o espacio en el caso de las imágenes, y se medirán en bits por segundo (bps) y bits por píxel (bpp) respectivamente. Usualmente, en los archivos de audio se desea incrustar un número de serie o el identificador del



propietario de los derechos de autor, por lo que con unos 0.5 bps suele ser suficiente. En el caso de la monitorización de difusión, se necesitan más bits de información para poder controlar todo el contenido y no sólo incrustar la marca en el inicio. En este caso se podrían necesitar cerca de 15 bps. En el caso de las imágenes y aplicando la incrustación de información oculta, se dispone de mucha más información, alcanzando de 2 a 40 kbit para una imagen de  $512 \times 512$  píxeles en escala de grises.

Para realizar la detección de manipulación de los objetos, la capacidad pasa a en un segundo plano, ya que no es importante que el algoritmo sea capaz de incrustar una gran cantidad de información.

### 3.3.2. Robustez

La robustez es la capacidad que tiene el esquema de marcado de extraer la marca insertada en el contenido después de que el objeto marcado sea manipulado intencionalmente o no. Aunque depende de cada caso y de la aplicación a que esté diseñado, los sistemas de marcado requieren de esta propiedad para poder ser más seguros contra modificaciones. Ataques típicos de compresión con JPEG para imágenes o MP3 (*MPEG-1 Audio Layer 3*) para audio, o de filtrado han de ser soportados por el esquema de marcado, ya que son ampliamente utilizados en la manipulación de los contenidos y pueden suponer la eliminación de la marca incrustada (Johnson et al., 2000; Lang et al., 2005; Swanson et al., 1998).

Por contra, se dice que un método es semifrágil cuando se permite alguna modificación leve del contenido del archivo sin que la marca o información

sean borradas. Por otro lado, al contrario que para los métodos robustos, si se realiza una alteración importante del objeto marcado sí que se eliminará la marca o la información incrustada. Diversos métodos que detectan modificaciones de las imágenes en general utilizan esquemas semifrágiles, por ejemplo (Gantasala and Prasad, 2009) y (Zhu et al., 2007).

Los métodos frágiles son aquellos en que la más mínima alteración del contenido del objeto marcado, aunque sea un ataque muy leve, elimina la marca o la información incrustada. También pueden usarse en técnicas forenses pero sólo en aquellos casos en que no sea admisible ninguna modificación, ni siquiera una ligera compresión con pérdida.

### **3.3.3. Transparencia o imperceptibilidad**

La transparencia de un esquema de marcado mide cómo afecta a la calidad final del contenido la inserción de la marca desde el punto de vista perceptual.

La transparencia indica hasta qué punto las personas pueden detectar que en el contenido existe alguna modificación perceptible. Esta modificación sería perceptible si en las imágenes aparecen zonas con tonalidades completamente diferentes del resto, colores diferentes o borrosos, líneas, etc. En el caso de los contenidos de audio, la modificación sería perceptible con señales audibles que no deberían estar, o con un excesivo ruido añadido.

Una marca muy robusta implica, en la mayoría de los casos, poca transparencia, ya que se necesita modificar considerablemente el contenido original.

La transparencia se puede medir de diferentes maneras, a partir de méto-

dos perceptuales, o mediante el error cuadrático medio (*Mean Squared Error*, MSE) y el *Peak Signal-to-Noise Ratio* (PSNR).

Formalmente denotaremos  $D(x, \hat{x})$  la distorsión entre el objeto original  $\mathbf{x} \equiv x[M, N]$  y el marcado  $\hat{\mathbf{x}} \equiv \hat{x}[M, N]$ , siendo  $M$  y  $N$  las dimensiones del objeto.

### 3.3.3.1. Mean Squared Error (MSE)

Una de las medidas de distorsión más utilizadas es el error cuadrático medio, *Mean Squared Error* (MSE), que se define como la media de las diferencias al cuadrado de las distancias entre los valores originales y modificados de los píxeles. La ecuación 3.1 muestra su definición.

$$\text{MSE} = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (x[i, j] - \hat{x}[i, j])^2 \quad (3.1)$$

### 3.3.3.2. Peak Signal-to-Noise Ratio (PSNR)

Otra manera de expresar la transparencia es usar una medida derivada del MSE que se expresa en decibelios y que se conoce como *Peak Signal-to-Noise Ratio* o PSNR, que se calcula como se muestra en la ecuación 3.2.

$$\text{PSNR} = 10 \log_{10} \frac{(2^B - 1)^2}{\text{MSE}}, \quad (3.2)$$

donde  $2^B - 1$  es el máximo valor que puede tener el píxel.

Existen ataques o modificaciones en los que no se aprecia a simple vista ninguna diferencia entre las dos imágenes, la original y la atacada, aunque

el PSNR es realmente muy bajo, por ejemplo ataques de desplazamiento o recorte de un número de píxeles. Por esto ya existen técnicas (Ramasubramanian et al., 1999) que van más allá de la comparación de los píxeles y aplican técnicas perceptuales, por lo que se obtienen medidas de calidad mejores que el PSNR.

### 3.3.4. Seguridad

Los algoritmos de marcado han de ser seguros en el sentido de que un atacante o adversario no debe ser capaz de eliminar la marca o falsificarla.

Por lo general, los métodos de watermarking son conocidos y basan la seguridad en una clave secreta, no difundida, para ser seguros. Es decir, que la seguridad no se basa en la ocultación del método sino en el uso de esta clave.

En el caso de la ocultación de información dentro de un contenido digital, ya sea audio, imagen o vídeo, usualmente ésta se incrusta cifrada para que no sea posible obtener la marca en claro incluso en el caso de que el ataque consiga obtener la marca o información incrustada.

### 3.3.5. Detección y extracción ciega o informada

El proceso de detección de la marca puede ser ciego (Eggers and Girod, 2001; Silvestre et al., 2001), es decir, que en el momento de determinar si un contenido tiene o no la marca o de extraerla no se necesita el contenido original. En caso contrario el proceso de detección o extracción de la marca se denomina informado (El-Taweel et al., 2007; Arnold and Huang, 2001).

Durante los primeros años de investigación en técnicas de watermarking los métodos eran informados, es decir, necesitaban el contenido original para poder extraer la marca. Posteriormente los investigadores se dieron cuenta que no siempre es factible comparar los dos contenidos (original y marcado) porque no siempre se dispondrá del original. De este modo empezaron a surgir los métodos de marcado en los que no es necesario el contenido original para saber si el objeto contiene o no la marca o para extraer la información secreta incrustada en él.

## 3.4. Métodos de watermarking para imágenes hiperespectrales

En la literatura existen muchos esquemas de marcado de imágenes que usan diferentes técnicas para incrustar la marca, pero existen muy pocos que se apliquen específicamente a las imágenes hiperespectrales.

Entre los métodos que se aplican a las imágenes hiperespectrales, se encuentra el trabajo (Sal and Graña, 2008), que se aplica a cada banda por

separado, lo que significa que la firma espectral no se modifica de manera uniforme y por tanto la curva de la firma puede ser significativamente alterada. El sistema descrito por (Sal and Graña, 2008) se puede utilizar para detectar alteraciones, pero no dispone de localización de las firmas espectrales manipuladas. El esquema de marcado se basa en la búsqueda de los mejores coeficientes de la transformada del coseno (DCT) teniendo en cuenta una optimización de dos funciones a la vez, la distorsión y la robustez.

El esquema de (Tamhankar et al., 2003) trabaja con la firma espectral de un conjunto de píxeles, pero no proporciona información sobre la localización de las manipulaciones de la imagen. Este método utiliza la transformada discreta *wavelet* redundante (RDWT) para calcular los coeficientes que después modifica a partir de una simple ecuación adaptativa para cada coeficiente.

En cuanto al tamaño de la localización se refiere, (Qin et al., 2004; Ho et al., 2005; Caldelli et al., 2006) detectan pequeñas áreas alteradas en comparación con los esquemas propuestos en esta tesis, pero los métodos se aplican a cada banda por separado y no ofrecen los resultados adecuados en cuanto a la calidad de la imagen marcada.

El método descrito en (Caldelli et al., 2006) aprovecha la compresión JPEG-LS y el algoritmo presentado en (Fridrich, 2002) para realizar una cuantización sobre la predicción del error de la compresión, en el que se inserta la marca, para finalmente realizar una codificación de los errores reales, ya marcados, y obtener la imagen marcada. Este proceso se realiza en bloques de  $8 \times 16$  píxeles para imágenes de una sola banda.

El artículo presentado en (Ho et al., 2005) describe el proceso de marcado semifrágil para imágenes de satélite. La imagen se divide en dos partes, mediante la transformada *Pinned Sine* (PST), el contorno de las formas y

el contenido, o cubierta, de éstas. Este esquema inserta la marca mediante la modificación de los coeficientes de la PST que denotan al contenido, cambiándolos a un cierto valor  $\alpha_1$  o  $\alpha_2$  dependiendo de si se quiere incrustar un 0 o un 1.

Finalmente, el algoritmo descrito en (Qin et al., 2004) utiliza la transformada *wavelet* para realizar el marcado de la imagen. Se aplica la transformada *wavelet* diádica (a medio camino entre la transformada discreta y la continua) para detectar los contornos de las formas y poder codificarlos.

### 3.5. Otros métodos de watermarking frágil y semifrágil

Los métodos de marcado frágiles o semifrágiles se utilizan en gran medida para realizar localización de manipulaciones de las imágenes, ya que las manipulaciones eliminan la marca que se ha incrustado. Con una simple extracción de la marca se puede saber si todavía existe ésta y no se ha modificado la imagen o, por el contrario, si la marca incrustada en la imagen ya no existe, o hay alguna zona donde no exista, de manera que la imagen habrá sido manipulada.

Los métodos frágiles no permiten ningún tipo de modificación del objeto. Cualquier simple modificación del objeto elimina la marca. Por el contrario, los algoritmos semifrágiles permiten una cierta manipulación del contenido por debajo de un umbral. Usualmente estos algoritmos permiten la comprensión con pérdida dentro de un cierto rango.

A continuación se detallan algunos ejemplos de métodos de estas características.

El método presentado en (Lin et al., 2000) incrusta la marca deseada en el dominio de la DCT, en bloques de  $8 \times 8$  en una imagen gris o, en el caso de una imagen a color, en la capa de luminancia. Se marca cambiando los valores de los coeficientes de baja frecuencia de la DCT de la imagen para cada bloque generado por la transformada, para así soportar compresiones JPEG. El artículo no proporciona valores de calidad de la imagen marcada.

El método presentado en (Li, 2004) realiza una transformada DCT sobre toda la imagen y cuantiza sus coeficientes. Este sistema genera una secuencia binaria con la marca secreta. Se crea un mapa binario con el valor uno para las posiciones de los coeficientes de la transformada DCT no nulos y cero para los nulos. Posteriormente se realiza una suma exclusiva entre la marca y el mapa binario de los coeficientes. Para cada bloque de la DCT, se seleccionan cuatro coeficientes no nulos con las frecuencias mínimas y se calcula el valor final a cambiar a partir de una ecuación que tiene en cuenta la suma exclusiva entre la marca y el mapa binario de los coeficientes y los valores vecinos al coeficiente que se marca.

En (Yang et al., 2010) se aplica la transformada DCT sobre el plano de la luminancia de la representación del color YCbCr y se cuantiza a partir de una tabla dada. Los coeficientes de baja frecuencia y la localización de cada uno de los bloques de la transformada DCT ( $8 \times 8$  píxeles) se usan como entrada de un modelo de autenticación que genera una secuencia de bits que identifica a cada uno de los bloques. Esta secuencia se incrusta en las frecuencias medias de la DCT para cada bloque, obteniendo unos resultados de alrededor de 47 dB en PSNR para imágenes en color. El método detecta



modificaciones y permite compresión JPEG de calidad hasta 30, pero con rango de errores de 4 a 13.6 % de no detección.

Otros ejemplos de métodos semifrágiles se describen en (Gantasala and Prasad, 2009), donde se presenta un esquema de marcado semifrágil basado en la cuantización de los coeficientes de la transformada discreta *wavelet* para imágenes de una sola banda y en (Zhu et al., 2007), donde se describe otro método semifrágil utilizando regiones irregulares, marcando sobre regiones de dimensión  $n \times n$ , pero teniendo en cuenta los valores de la transformada DCT de regiones  $m \times m$  más grandes.

Ninguno de estos métodos semifrágiles o frágiles se adapta a las características específicas de las imágenes de obtención remota, ya que se aplican sobre una sola banda o bien sobre la luminancia.



# Capítulo 4

## Sistemas de watermarking para imágenes hiperespectrales

### 4.1. Introducción

En esta tesis se proponen nuevos esquemas de watermarking para imágenes hiperespectrales, teniendo en cuenta su principal característica, las tres dimensiones de este tipo de imágenes: dos dimensiones espaciales y una dimensión espectral.

Este capítulo muestra la evolución de los diferentes métodos propuestos, desde una primera aproximación en la que simplemente se marca el contenido de la imagen por regiones sin tener en cuenta ataques específicos (Serra-Ruiz et al., 2006), para posteriormente introducir mejoras en el proceso de marcado que proporcionaran más robustez contra ataques de compresión. En este sentido, se realiza una primera modificación del método inicial teniendo en cuenta la extracción de los bits menos significativos (*Least Significant Bits*, LSB) (Serra-Ruiz and Megías, 2011), y posteriormente se propone un nuevo método en el que se introduce la transformada *wavelet* en el proceso de marcado sobre los coeficientes de esta transformada (Serra-Ruiz and Megías,

2010a). Además se realizó una nueva propuesta en la que la selección de las zonas a marcar no son todas del mismo tamaño (Serra-Ruiz and Megías, 2010b). En un último resultado se automatiza la selección de las zonas a marcar con un proceso para agruparlas en función de la diferencia o similitud entre sus píxeles.

Por lo tanto los métodos desarrollados se exponen de manera evolutiva:

1. Marcado sobre valores de los píxeles por regiones.
2. Marcado sobre los valores de los píxeles extrayendo los LSB.
3. Marcado sobre los coeficientes de la DWT.
4. Marcado sobre DWT y aplicación de regiones de diferentes tamaños.
5. Marcado sobre DWT y automatización de la selección de las regiones a marcar.

## 4.2. Proceso de incrustación de la marca

Para simplificar llamaremos  $I$  a la imagen original hiperespectral, de tres dimensiones, de medidas  $M \times N \times b$ , donde  $b$  es el número de bandas y  $M$  y  $N$  las dimensiones de la imagen. En general, tendremos en cuenta que se ha hecho el estudio con imágenes AVIRIS (NASA, Jet Propulsion Laboratory, 2004) de 224 bandas de una región de Indiana (EE.UU.), de una cantera en Nevada (EE.UU.) y del World Trade Center de Nueva York el 16 de septiembre del 2001.

### 4.2.1. Preprocesado

Se puede aplicar un proceso previo al marcado de la imagen con el fin de eliminar o reducir el posible ruido del sensor. Este preprocesado podría consistir en comprimir y descomprimir la imagen original con JPEG2000 (Taubman and Marcellin, 2002), por ejemplo usando el compresor KaKadu (Taubman, 2007) o Jasper (Image Power Inc., 2003), con 14 bits por píxel<sup>1</sup> (bpp) como parámetro de compresión para cada banda. Si se aplica este paso previo se incrementará la robustez del método, aunque el *Peak Signal-to-Noise Ratio* (PSNR) de las imágenes marcadas se decrementará, haciendo que la imagen final resultante tenga mayor distorsión que la que se puede conseguir sin este paso previo. Por otro lado la realización de este paso de procesamiento ayuda a reducir el ruido del sensor y elimina los valores anormales de la imagen. Tras aplicar este paso, los valores de los componentes se *alisan* un poco pero sin cambiar la forma de la firma espectral.

### 4.2.2. Descripción general del método de marcado

De forma general los métodos de marcado de las imágenes se componen de los siguientes pasos.

1. Selección de las bandas a marcar.
2. División por bloques.
3. Construcción de los vectores para la cuantización vectorial.
4. Incrustación de la marca: construcción de los árboles TSVQ.

---

<sup>1</sup>Se escoge el valor de 14 bpp porque las imágenes seleccionadas tienen los dos bits más significativos de los componentes a 0.

5. Reconstrucción de los vectores.
6. Reagrupación de los bloques.
7. Reagrupación de las bandas.

Los diferentes métodos de marcado que se han desarrollado afectan a uno o más de los pasos anteriores, pero la estructura de los diferentes esquemas es siempre la misma.

#### **4.2.2.1. Selección de las bandas**

Como se describe en la Sección 2.2, las imágenes con las que trabajamos tienen la característica principal de que, para cada píxel, disponen de múltiples bandas de información, dependiendo del tipo de imagen que sea. Por eso, en el caso en que se trabaje con imágenes hiperespectrales o ultraspectrales, el primer paso que se realiza en el método es la selección de un grupo de estas bandas para facilitar el trabajo. Únicamente por eficiencia y para simplificar los cálculos y el tiempo empleado en el marcado, se seleccionan 16 bandas del total de la imagen. Hay que recordar, en este punto, que las imágenes AVIRIS disponen, generalmente, de 224 bandas, mientras que las ultraspectrales disponen de miles de bandas. En el caso de las imágenes multiespectrales, las LANDSAT disponen de 8-10 bandas, por lo que este primer paso no es necesario y se pueden marcar todas las bandas de la imagen.

Para seleccionar las bandas podemos escoger diferentes criterios, como realizar una selección aleatoria o siguiendo alguna regla que tenga en cuenta las características de las imágenes.

Un paso previo que habrá que realizar será la eliminación de aquellas ban-

das que no dispongan de información correcta o sean erróneas. Por ejemplo, se eliminarán aquellas bandas en las que el sensor hiperespectral produzca errores y todos o una parte importante de los valores de la banda queden fijados al valor mínimo (0) o máximo (255 ó 65 535). En este sentido, no podemos marcar valores de los componentes que no dispongan de suficiente información como para poder modificar el valor original.

En el caso de no escoger aleatoriamente las bandas (ya que poca información se puede extraer de este proceso), éstas se pueden escoger en función de diferentes criterios como por ejemplo los siguientes:

- **Equidistantes:** si las bandas escogidas son equidistantes a lo largo de toda la firma espectral aseguramos que la forma de la firma espectral del píxel está protegida, de manera que una alteración en una parte de ésta se detecte.
- **Clasificación:** se pueden tener en cuenta métodos de clasificación de materiales para escoger las bandas a marcar. En la literatura existen diversos métodos de clasificación (Plaza et al., 2005; Chang et al., 1999; Camps-Valls et al., 2007) en los que se determina qué material es predominante en cada punto, o qué bandas tienen mejor energía para un determinado material.
- **Reflexión:** los valores de cada componente proporcionan la reflexión de la luz solar para cada una de las divisiones del espectro que captura el sensor. Así, se puede tener en cuenta qué valores son los mejores para recuperar y los valores que puede captar el sensor teniendo en cuenta las condiciones climáticas del momento en que se realizó la captura.
- **Forma:** la firma espectral en cada punto tiene una forma diferente para cada material. Si lo que se quiere es proteger un tipo de material en

concreto (uno sólo), la selección de las bandas se puede basar en escoger aquellas para las que el material es característico. Por ejemplo, si se quiere proteger vegetación, la parte del espectro de color verde será característica y se podrá escoger un número de bandas más elevado en esa parte del espectro y dejar muchas menos para la otra parte. Esto permitiría proteger mucho mejor la forma del espectro de la vegetación.

Para la realización de este método, siguiendo la recomendación de (Gersho and Gray, 1992), que determina la proporción mínima entre vectores y componentes para realizar cuantización vectorial, se han escogido 16 bandas con una distribución equidistante entre ellas y así asegurar la forma de las firmas espectrales tras el proceso de marcado. Se ha optado por esta selección por simplicidad y por no estar dentro de los objetivos del trabajo realizado el estudio de qué bandas son mejores para realizar la clasificación de los materiales.

#### 4.2.2.2. División por bloques

El siguiente paso a realizar es la división de las bandas en diferentes áreas, o bloques tridimensionales, con el objetivo de marcar el bloque y, a posteriori, determinar si éste se ha modificado o no.

La imagen  $I$  se segmenta en bloques de medida  $W \times H \times b'$ , donde  $W \leq M$  y  $H \leq N$  son dimensiones en píxeles de cada uno de los bloques. De esta manera, se crean  $W \times H$  vectores con la agrupación de todos los componentes y los valores de las bandas y donde  $b' \leq b$  es el número de bandas seleccionadas. En este caso se usan 16 bandas del global de 224.



Esta división de la imagen hiperespectral en bloques tridimensionales permitirá determinar si se ha modificado o no cada uno de los bloques por separado, ya sea porque se ha modificado una parte de la firma espectral, porque se ha copiado otro bloque de la imagen en la posición del bloque modificado o por algún otro tipo de ataque que modifique de manera significativa la imagen.

#### 4.2.2.3. Construcción de los vectores

Una vez se han construido los bloques de  $W \times H \times b'$  componentes de la imagen se crea una matriz con los vectores de las firmas espectrales que después se utilizarán para ser marcadas.

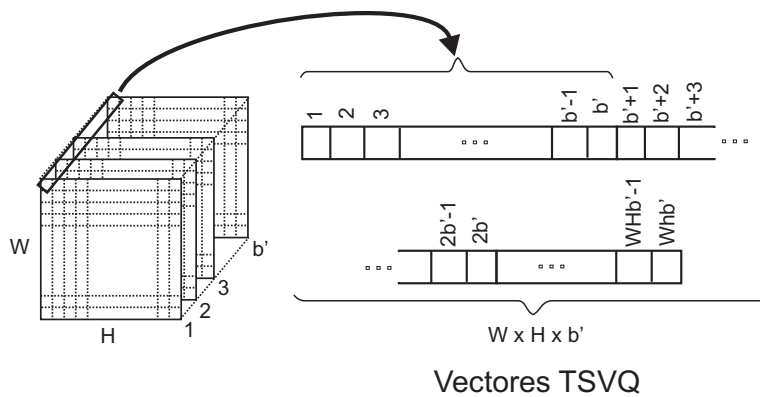


Figura 4.1: Construcción de los vectores TSVQ.

La Figura 4.1 muestra la manera en que se agrupan los vectores para poder realizar el marcado de la imagen. Cada vector se construye con cada uno de los  $b'$  componentes seleccionados de las firmas espectrales del bloque.

La división de la imagen original en bloques se realiza siguiendo las re-

comendaciones de (Raudys, 1997), donde se muestra que la relación entre el número de valores disponibles y la dimensión del vector ha de ser por lo menos de 30 con el fin de minimizar el sesgo estadístico causado por un número insuficiente de muestras del vector. Por lo tanto, como en nuestro caso la dimensión de las firmas espectrales es 16 (que son el número de bandas seleccionadas), necesitamos por lo menos  $16 \times 30$  muestras diferentes para poder construir después los árboles TSVQ. Por esta razón, las regiones han de ser por lo menos de 480 píxeles. Si tenemos en cuenta regiones cuadradas cuya dimensión sea potencia de dos, la mínima región deberá de ser de  $32 \times 32$  píxeles (la cual tiene 1 024 vectores). Valores más pequeños de esta dimensión no satisfacen las recomendaciones de (Raudys, 1997) y provocarían problemas en el momento de la construcción del árbol.

En los primeros métodos presentados en esta tesis se usan regiones de  $64 \times 64$  píxeles para reducir el número de bloques resultantes de la imagen. Así, la imagen original de  $512 \times 512$ , se ha dividido en  $8 \times 8$  regiones resultando un total de 64 bloques a ser marcados separadamente. No obstante, en los tres últimos métodos se ha optado por usar bloques de diferentes tamaños, para, como se explicará en cada método, dificultar algunos ataques.

La Figura 4.2 muestra el resumen todos los pasos realizados hasta este punto: la selección de las bandas, la división en regiones y la construcción de los vectores para realizar el marcado de éstos.

#### 4.2.2.4. Marcado: construcción de los árboles TSVQ

Si disponemos de una imagen hiperespectral de  $512 \times 512$  píxeles, existen entonces 262 144 firmas espectrales, o vectores con 16 componentes, uno

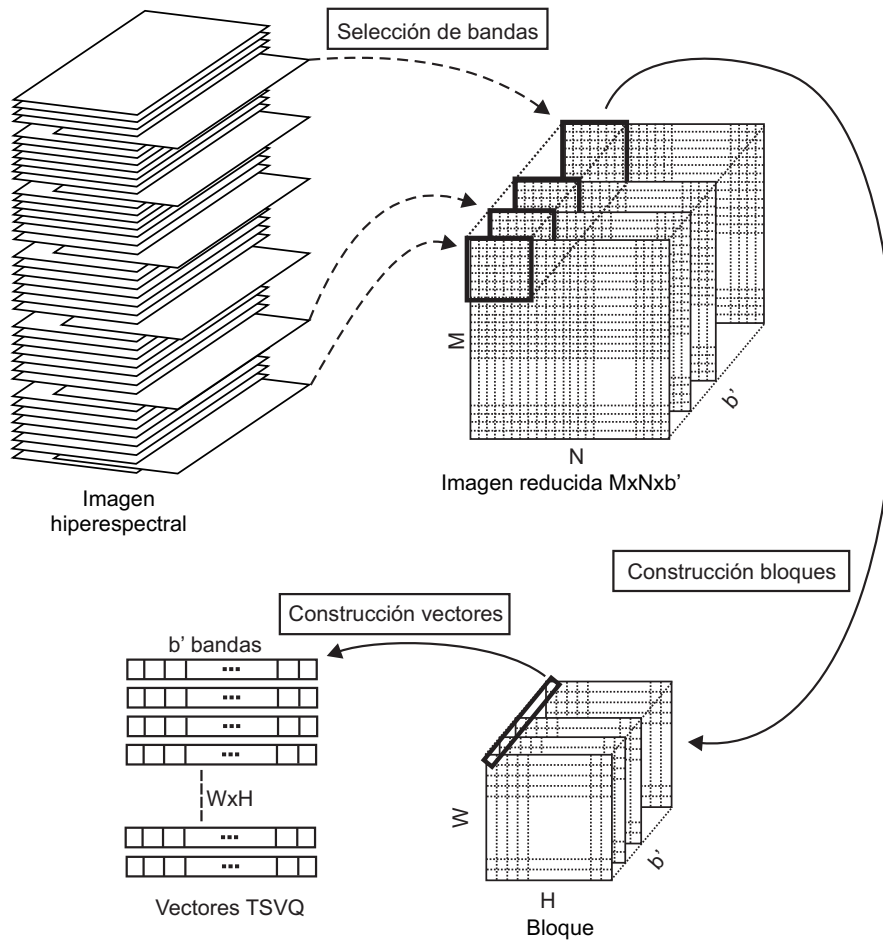


Figura 4.2: Selección de bandas, división por bloques y generación de los vectores de marcado.

para cada banda seleccionada. Por tanto, tenemos suficiente información para realizar la cuantización vectorial sobre ellos.

Tal y como se explica en la Sección 2.4.2, la cuantización vectorial que se usa en los diferentes métodos presentados en esta tesis se basa en la estructura de árbol TSVQ. Los vectores de cada uno de los bloques  $B$  (en la Figura 2.9 se denota como imagen  $I$ ) se reemplazan por un valor muy similar, que se determina con el centroide de la región de Voronoi a que pertenece cada vector, produciendo un bloque nuevo  $B^*$ . Estos centroides serán los valores

que se encuentran en las hojas del árbol de cuantización, como se explica en la Sección 2.4.2, que se ha escogido de toda la familia de árboles posibles. Por lo tanto, se aplica la compresión con pérdida TSVQ, descrita en (Gersho and Gray, 1992), para generar un bloque similar  $B^*$ . Los pasos del proceso TSVQ (Figura 2.9) que se aplican en los diferentes métodos son el preprocesado (P), la transformación (T) y la cuantización (Q), tal y como se describe en la Sección 2.4.1.

El algoritmo TSVQ construye el árbol completo de todos los centroides para el bloque con el que se ha generado. Este árbol contiene todos los posibles subárboles para la compresión de este bloque, desde el árbol raíz, con un solo nodo que representa el centroide de todos los valores de los píxeles, hasta el árbol completo que representa exactamente a todos los píxeles, en el que no existe compresión ni pérdida de datos. Dependiendo del criterio que se adopte, se escogerá cualquier subárbol de este árbol general.

El algoritmo BFOS (Breiman, Friedman, Olshen, and Stone) (Breiman et al., 1984) poda el árbol general para obtener todos los subárboles de la curva Compresión-Distorsión llamada *convex-hull*, como se indica en la Figura 2.10, minimizando la distorsión para una ratio de compresión dada. Por norma general, y teniendo en cuenta objetivos de compresión, se utilizan la ratio de compresión y el MSE (*Mean Squared Error* - Error Cuadrático Medio) para generar la curva *convex-hull*, pero el algoritmo BFOS puede usar otros criterios de poda que sean más acordes a los objetivos deseados. Uno de los criterios podría ser el watermarking, como se detalla en (Caldelli et al., 2004). Se pueden escoger diferentes parámetros para realizar la poda, como la entropía del subárbol resultante, el número de nodos, o la distorsión, entre otros. Este parámetro determinará qué árbol hay que generar con el algoritmo BFOS y éste será el que se utilizará para generar el bloque marcado

$B^*$ .

Con el objetivo de aprovechar la redundancia (3D) espacial y espectral, algunos autores proponen usar la transformada discreta *wavelet* (*Discrete Wavelet Transform*, DWT) para cada una de las bandas por separado y aplicar cuantización vectorial para cada firma espectral nueva usando una aproximación de multiresolución (Markas and Reif, 1993). El primer método desarrollado se basa en la misma idea de aplicar cuantización vectorial sobre los píxeles originales (vectores), sin aplicar ninguna transformación sobre ellos, como en el caso de (Markas and Reif, 1993), ya que no necesitamos maximizar la relación de compresión, sino obtener imágenes lo más parecidas posible a la original con el propósito de incrustar la información de marcado. Para los siguientes métodos, a partir de la Sección 4.2.4, se muestra cómo se aplica la transformada *wavelet* a los componentes para mejorar la robustez contra la compresión, pero no para mejorar la relación de compresión de las imágenes.

En este punto hay que hacer notar que la marca que se incrusta en cada uno de los bloques se determina por un valor del criterio escogido diferente en cada bloque, por lo que la relación de compresión no tiene el mismo valor en cada una de las bandas, sino que lo tienen las firmas espectrales en global de cada bloque. Los diferentes métodos propuestos en esta tesis trabajan con los vectores de las firmas espectrales y no con las bandas separadamente. Esta propuesta es diferente a la de otros métodos de watermarking semifrágil de la literatura (Rey and Dugelay, 2002; Ekici et al., 2004), ya que utilizan las bandas separadamente y, por lo tanto, la cuantización se realiza por bandas y no por firmas espectrales.

La Tabla 4.1 muestra un ejemplo de los diferentes subárboles que se ge-

neran en el proceso de poda del árbol general. Las columnas de esta tabla representan la relación entre la distorsión y la compresión ( $\lambda$ ), la ratio de compresión, la distorsión, las profundidades máxima y mínima del subárbol, el número de nodos del subárbol y la entropía del subárbol. Todos estos valores se muestran para cada uno de los posibles subárboles que se pueden generar a partir del árbol original. Éste, identificado con el índice 0, tiene distorsión 0, ya que todos los vectores están representados en una hoja final y, por lo tanto, no implica ninguna compresión. Una de estas medidas del árbol puede ser la escogida para realizar la selección del subárbol que se quiere podar y generar el bloque marcado a partir de éste. Un criterio simple pero eficiente es la máxima profundidad del subárbol, podando el árbol general a la altura deseada. Incluso con criterios simples como éste, es posible generar subárboles lo suficientemente complejos que son difíciles de reproducir mediante la manipulación de un bloque de la imagen. La Tabla 4.1 muestra un ejemplo muy reducido del proceso de selección de los subárboles, teniendo en cuenta la entropía del subárbol como criterio de poda.

En este caso se parte de la lista global de todos los posibles subárboles del árbol general que representa el bloque y se determina que este bloque en concreto ha de tener una entropía de 8.64, por lo que se busca en la lista el subárbol correspondiente (el 805). Se poda, por tanto, el árbol general para tener el subárbol 805 y con éste se generará el bloque  $B^*$ .

Una vez se obtiene el subárbol TSVQ generado a partir del criterio elegido, se usa éste para obtener un nuevo bloque  $B^*$  sustituyendo los vectores por los centroides. El nuevo bloque  $B^*$  tendrá menos vectores diferentes. Algunos vectores se habrán sustituido por otros que representan al centroide de la región de Voronoi, descrita en la Sección 2.4.2. Cabe destacar que el bloque  $B^*$  no generará un árbol TSVQ con esa misma entropía, ya que al volver a

Tabla 4.1: Ejemplo de valores obtenidos con el algoritmo BFOS.

Subárbol	$\lambda$	Bits por píxel	Distorsión	Num. nodos	Prof. mín.	Prof. máx.	<b>Entropía</b>
0	0.222	13.303	0.0000	6 813	6	24	11.651
1	0.666	13.302	0.0002	6 811	6	24	11.649
2	0.666	13.301	0.0006	6 809	6	24	11.649
3	0.833	13.301	0.0011	6 807	6	24	11.648
4	0.889	13.300	0.0019	6 805	6	24	11.647
5	1.000	13.299	0.0026	6 803	6	24	11.647
6	1.000	13.298	0.0035	6 801	6	24	11.645
7	1.111	13.296	0.0055	6 791	6	24	11.644
8	1.250	13.295	0.0063	6 789	6	24	11.643
...	...	...	...	...	...	...	...
<b>805</b>	10.000	9.5067	19.6232	2 867	6	17	<b>8.640</b>
...	...	...	...	...	...	...	...

construir el árbol TSVQ con el bloque  $B^*$ , los vectores se pueden agrupar de otra manera y obtenerse una entropía algo diferente.

Por lo tanto, el bloque  $B^*$  que se ha generado en el paso anterior, se vuelve a convertir en el bloque  $B$  para construir otra vez el árbol TSVQ y obtener todos los subárboles posibles de compresión. Se volverá a buscar el subárbol con entropía 8.64 y se podará para tener al final un nuevo bloque  $B^*$  ligeramente diferente al anterior. El proceso es iterativo hasta que se encuentre el bloque para el cual el árbol TSVQ que genere tenga el criterio establecido en el primer subárbol, es decir, que no se tiene que modificar para obtener la propiedad (por ejemplo la entropía) que se le quiere forzar en ese bloque.

Como se puede observar en la Figura 4.3, el proceso es iterativo hasta que el árbol TSVQ del bloque resultante tiene la propiedad deseada. En ese momento ya no se realizan más pasos iterativos y se genera el bloque final  $B^*$  con los centroides que están en las hojas de ese subárbol. Este bloque es

con el que después se construye la imagen marcada. Para conseguir esto se usa una clave secreta que generará diferentes valores del criterio para cada uno de los bloques de  $W \times H$  píxeles.

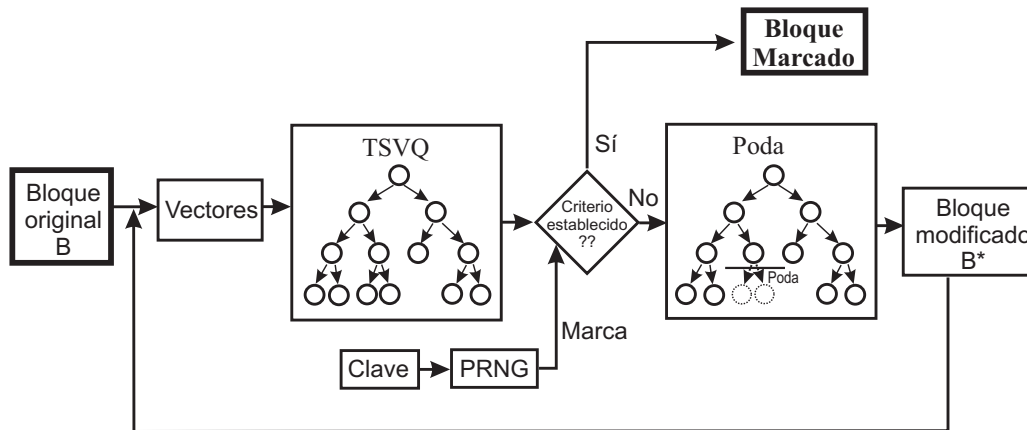


Figura 4.3: Diagrama de bloques del proceso de marcado.

Como se menciona en secciones anteriores, cada bloque de la imagen original se comprime ligeramente con una ratio de compresión diferente, de acuerdo al criterio seleccionado para cada uno de los bloques. Para detectar un ataque de manipulación (*tampering*) en la imagen, como una copia-reemplazo de una parte de la imagen, o la copia de una parte de otra imagen sobre la imagen marcada, se genera una secuencia pseudoaleatoria, que será la clave secreta, para determinar los valores que se usan como criterio para seleccionar el subárbol TSVQ en el algoritmo de poda. De esta manera se hace más difícil encontrar un patrón que revele las propiedades del esquema de watermarking, reduciendo así las posibilidades de manipular una imagen o modificarla usando una parte de la misma imagen. Esta secuencia, que asegura que los bloques sean localizables dentro de la imagen, se genera a partir de una clave secreta que dependerá de cada imagen. La clave puede ser idéntica para todas las imágenes, pero eso podría comprometer la seguridad en el



método (Cayre et al., 2004). Si se utiliza de una clave diferente para cada imagen, será más difícil poder generar la misma propiedad para el bloque que se quisiera modificar. En definitiva, la clave es un número que se puede asociar a cada una de las imágenes que se marcan y que incluso se podría incrustar, cifrada a partir de una clave común, a posteriori con técnicas de watermarking reversible (Fallahpour et al., 2009; Arjun and Rao, 2008; Ren et al., 2009; Lin et al., 2008) dentro de la propia imagen. Esto permitiría extraer la clave de la misma imagen y restaurar ésta a los mismos valores que tenía anteriormente a ocultar la clave en ella y poder así comprobar si se ha modificado o no la imagen. Este proceso reduciría la seguridad del método, ya que la clave estaría incrustada dentro del mismo contenido, pero se describe a modo de mejora en el caso en que se emplearan sistemas de ocultación reversibles lo suficientemente seguros como para justificar su uso. Esta posibilidad queda fuera de los objetivos de esta tesis.

La clave secreta es la semilla de un generador de números pseudoaleatorios (*Pseudo-random number generator*, PRNG) y será también requerida para el proceso de detección de la marca. Al margen de la clave, el método de marcado no necesita la imagen original para poder detectar dónde se ha modificado la imagen, por lo que se considera que el método es ciego (*blind*).

Para cada bloque la secuencia pseudoaleatoria determinará el valor del parámetro escogido que ha de tener el árbol TSVQ, pero como no todos los valores son posibles, ya que en el árbol sólo se disponen de unos valores discretos, se tendrá que cuantizar el conjunto de valores del parámetro para poder escoger el subárbol que determine el valor de la secuencia pseudoaleatoria.

El proceso se inicia con el valor máximo de la entropía para cada una de las imágenes que se quiere marcar. En la primera iteración del proceso de

marcado se calcula la entropía del árbol completo con todos las hojas y se guarda ésta para poder aplicar el método iterativo. En este caso en que se divide la imagen en bloques de  $64 \times 64$  píxeles la entropía tendrá como valor máximo el siguiente:

$$\log_2(64 \times 64) = 12$$

Aunque no llegará nunca a este valor, ya que siempre habrá algunos píxeles repetidos en la imagen.

El siguiente paso es cuantizar la entropía para poder marcar cada uno de los bloques con diferente información. Así se reduce el valor máximo encontrado en el árbol completo a 11, y a partir de ahí se cuantiza a la baja la entropía de los diferentes bloques a marcar a partir de la distancia escogida entre los bloques y el valor de la secuencia pseudoaleatoria escogida. La distancia entre los valores de la entropía es un parámetro de los métodos. A mayor distancia entre ellos menor será el número de errores en el proceso de detección (menos falsos positivos o negativos), pero peor calidad de imagen (PSNR) se obtendrá, ya que habrá bloques con muy poca entropía.

Para seleccionar el valor de la entropía para cada bloque, se obtiene un valor  $v$  de 6 bits, entre 0 y 63, con el PRNG. Con este valor, la entropía se escoge de la manera siguiente:

$$\text{Entropía del bloque} = 11 - v \times h,$$

donde  $h$  es el paso de cuantización.

Por ejemplo, teniendo en cuenta que la imagen se divide en 64 bloques, el paso de cuantización  $h$ , se puede escoger como  $\frac{1}{64}$  y, a partir de una secuencia pseudoaleatoria  $g = (5, 46, 24, 62, 0, \dots, 30)$  se obtienen los valores de

entropía para cada bloque siguientes:

$$11 - \frac{5}{64}, \quad 11 - \frac{46}{64}, \quad 11 - \frac{24}{64}, \quad 11 - \frac{62}{64}, \quad 11, \quad \dots, \quad 11 - \frac{30}{64}$$

obteniendo así un conjunto de valores de entropía comprendidos entre 11 y 10.

#### 4.2.2.5. Reconstrucción de los vectores

Una vez ya se tienen todos los bloques marcados, cada uno con un valor diferente del criterio escogido para cada bloque, se inicia el proceso contrario: se tienen que volver a generar los bloques a partir de los vectores que se han sustituido por centroides del árbol TSVQ. Para eso, este paso consiste en generar la estructura tridimensional de los bloques, ya que del árbol TSVQ obtenemos un conjunto de vectores (firmas espectrales) sin la forma deseada. Por lo tanto, habrá que pasar los  $W \times H$  vectores de  $b'$  componentes, a la forma rectangular  $W \times H \times b'$  que teníamos inicialmente. Esto se consigue con una simple transformación de almacenamiento en matrices.

#### 4.2.2.6. Reagrupación de los bloques

En este proceso se trata de formar la imagen tridimensional completa, una vez ya se disponen de todos los bloques marcados, colocándolos en la posición inicial para formar la imagen final marcada.

Es decir, se tienen que combinar los bloques de  $W \times H$  píxeles para formar la imagen final marcada de dimensiones  $M \times N \times b'$ , por ejemplo de  $512 \times 512 \times 16$ .

#### 4.2.2.7. Reagrupación de las bandas

El último paso es reagrupar todas las bandas, tanto las que se escogieron en el primer paso como las que no, generando otra vez la imagen hiperespectral completa de  $M \times N \times b$  componentes, es decir, la imagen marcada con las  $b$  bandas. Esta imagen es la que se podrá distribuir. Este último paso concluye el sistema general de marcado.

#### 4.2.2.8. Resumen

Los siguientes pasos resumen el proceso general de marcado:

0. Preprocesado: compresión y descompresión de la imagen original con JPEG2000 (por ejemplo usando el software KaKaDu (Taubman, 2007)).
1. Selección de bandas y construcción de los bloques de la imagen (como se muestra en la Figura 4.2).
2. Elección de la semilla del generador de números pseudoaleatorios y la inicialización de éste.
3. Elección de la propiedad a modificar (por ejemplo la entropía).
4. Para todos los bloques
  - 4.1 Generar un número con el PRNG y comprobar que no esté repetido en algún bloque ya procesado.
  - 4.2 Escoger el valor de la propiedad según el número generado en el paso 4.1.
  - 4.3 Seleccionar el árbol con la propiedad escogida en el paso 4.2 (ver Tabla 4.1).
  - 4.4 Generar el bloque usando el árbol obtenido en el paso 4.3.
  - 4.5 Comprobar la propiedad seleccionada en 4.2. Si no se cumple ir a 4.3 con el bloque generado en 4.4.
5. Reagrupar todos los bloques (y las bandas no seleccionadas en el paso 1) para construir la imagen marcada final.

En este método la clave secreta necesaria para el proceso de incrustación y en el de detección estará compuesto de la semilla del PRNG y el criterio de poda. La forma de la división por bloques, el valor inicial de la entropía y el paso de cuantización  $h$  también forman parte de la clave secreta, si bien son idénticos para todas las imágenes de esta tesis.

### 4.2.3. Método basado en la extracción de los LSB

La primera variante del método que se propone (Serra-Ruiz and Megías, 2011) para incrementar la robustez contra los ataques de compresión es la extracción de los bits menos significativos (*Least significant bits*, LSB) de cada uno de los valores de los componentes de las bandas en que se marca.

Después de aplicar el primer paso de la selección de las bandas, los  $n$  bits menos significativos se eliminan (y posteriormente se restauraran). Cuanto mayor sea el número  $n$  de LBS extraídos, mayor será la robustez del sistema frente a compresión con pérdida. Incrementar el número de LSB incrementa la robustez, pero en contrapartida decrementará la calidad de la imagen marcada. Por otro lado, el número de bits eliminados no se puede incrementar demasiado, ya que la información resultante ha de ser suficiente como para poder generar los árboles TSVQ. Si se dispone de poca información no se podrán generar correctamente los árboles y no se podrá marcar la imagen. Los valores de los  $n$  LSB extraídos en este paso del método se guardan para poder ser restaurados al final y generar la imagen marcada final. El proceso de inserción de los LSB extraídos se detalla a continuación.

El número ( $n$ ) de bits a extraer es un parámetro del esquema que deter-

minará la robustez del mismo, ya que los ataques de compresión y descompresión de la imagen afectarán generalmente a los bits menos significativos. De esta manera, una pequeña pérdida de datos por un ataque de compresión con pérdida se puede tolerar, ya que las ratios de compresión pequeñas modifican sólo los bits menos significativos de los componentes. Así, si un ataque modifica únicamente los bits menos significativos, el esquema seguirá funcionando y la imagen conservará la marca, ya que la propiedad que se fija en cada uno de los bloques ahora está en los valores de los componentes sin tener en cuenta los  $n$  últimos bits.

En este caso, los vectores que se utilizarán para construir el árbol TSVQ tendrán las firmas espectrales con componentes de  $16 - n$  o  $8 - n$  bits, dependiendo de si las imágenes hiperespectrales tienen una profundidad de color de 2 ó 1 bytes. Las firmas espectrales marcadas con este método no disponen de todos los bits al extraerles  $n$  LSB, por lo que se deberán restaurar los bits que se guardaron en el primer paso. Para minimizar la distancia entre el valor del componente original y el marcado se seguirá el proceso descrito en la Figura 4.4, que garantiza este objetivo.

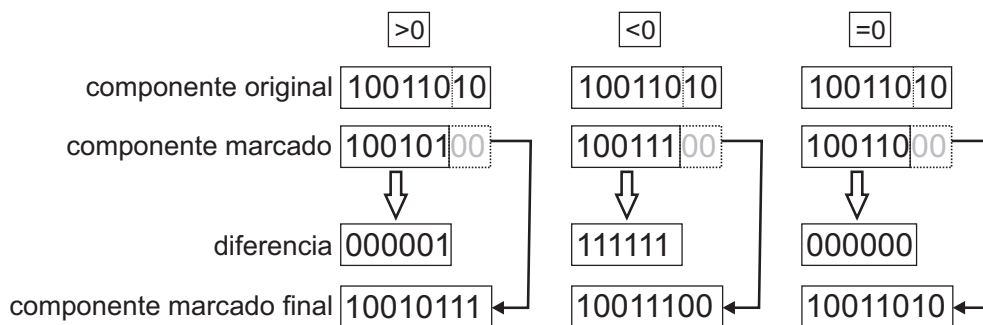


Figura 4.4: Esquema de restauración de los LSB.

1. Si el valor del componente marcado es menor que el valor original, los

- bits menos significativos se rellenan todos con 1.
2. Si el valor del componente marcado es mayor que el valor original, los bits menos significativos se rellenan todos con 0.
  3. Si el valor del componente marcado es idéntico al del valor original, los bits menos significativos se restauran idénticos al componente original.

Este proceso se formaliza en la ecuación 4.1. Consideremos  $LSB'$  como los nuevos valores de los LSB, donde  $i$  y  $j$  son las posiciones del componente y  $k$  la banda del componente que se está restaurando:

$$LSB' = \begin{cases} 1s : B_{i,j,k}^* < B_{i,j,k}, \\ 0s : B_{i,j,k}^* > B_{i,j,k}, \\ LSB : B_{i,j,k}^* = B_{i,j,k}. \end{cases} \quad (4.1)$$

Al final de este proceso se dispondrá de los componentes con 16 ó 8 bits marcados, con lo que se continúa con el mismo esquema de marcado general, es decir, que se sigue con la reconstrucción de los vectores descrita en la Sección 4.2.2.5.

En el caso de este nuevo método únicamente se intercalan dos nuevos pasos, uno antes de marcar y otro justo después. En el primero se extraen  $n$  LSB y en el segundo se restauran los  $LSB'$  minimizando la distancia entre el componente marcado y el original.

#### 4.2.3.1. Resumen

El esquema general se muestra en la Figura 4.5 y se resume en los siguientes pasos.

0. Preprocesado: compresión y descompresión de la imagen original con JPEG2000 (por ejemplo usando el software KaKaDu (Taubman, 2007)).

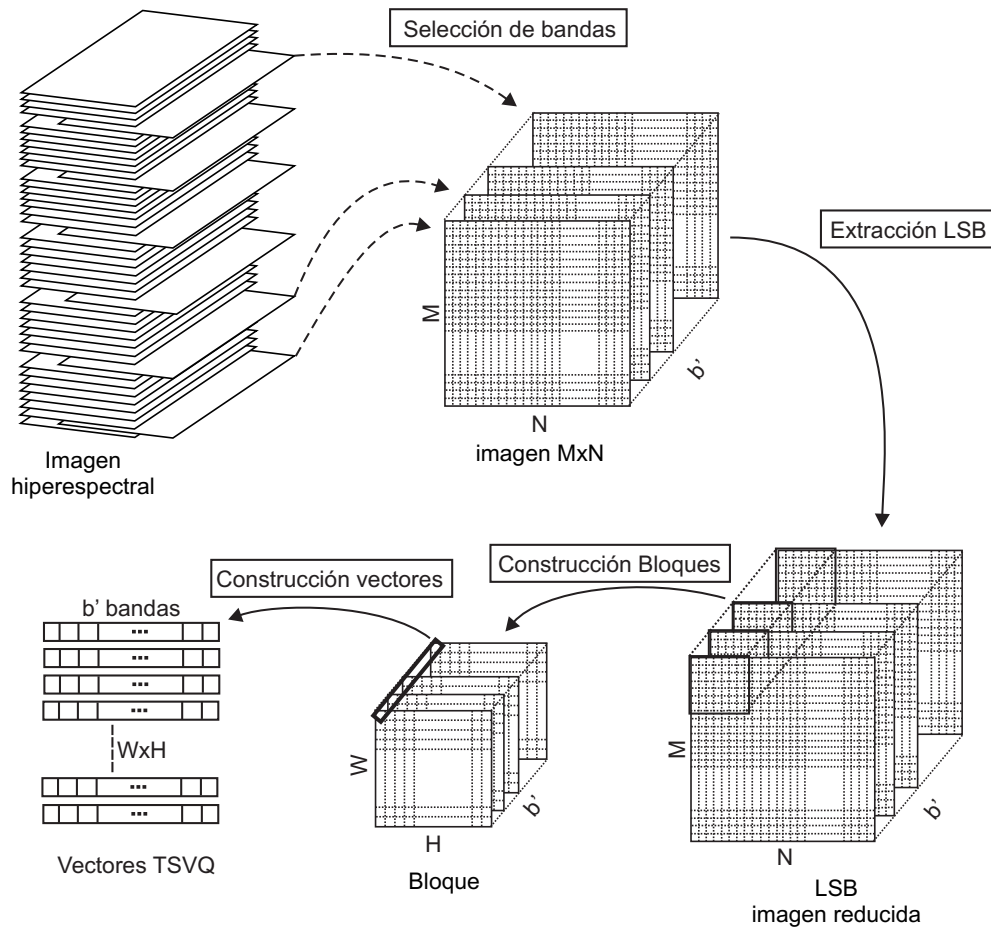


Figura 4.5: Selección de bandas, división por bloques, extracción de los LSB y generación de los vectores de marcado.

1. Selección de bandas y construcción de los bloques de la imagen (como se muestra en la Figura 4.5).
2. Extracción de los  $n$  LSB: se guardan estos valores para poderlos restaurar a posteriori.
3. Elección de la semilla del generador de números pseudoaleatorios y la inicialización de éste.
4. Elección de la propiedad a modificar (por ejemplo la entropía).
5. Para todos los bloques:
  - 5.1 Generar un número con el PRNG y comprobar que no esté repetido en algún bloque ya procesado.



- 5.2 Escoger el valor de la propiedad acorde al número generado en el paso 5.1.
  - 5.3 Seleccionar el árbol con la propiedad escogida en el paso 5.2 (ver Tabla 4.1).
  - 5.4 Generar el bloque usando el árbol obtenido en el paso 5.3.
  - 5.5 Comprobar la propiedad seleccionada en 5.2. Si no se cumple ir a 5.3 con el bloque generado en 5.4.
6. Restaurar los  $n$  bits menos significativos minimizando la distancia entre los componentes marcados y los originales.
  7. Reagrupar todos los bloques (y las bandas no seleccionadas en el paso 1) para construir la imagen marcada final.

En este método la clave secreta necesaria para el proceso de incrustación y para el de detección estará compuesto de la semilla del PRNG, el criterio de poda, el número  $n$  de LSB utilizado, además de la división de los bloques, el valor inicial de la propiedad (entropía) y el paso de cuantización de ésta.

#### 4.2.4. Marcado sobre los coeficientes de la DWT

Por ser una de las transformadas más utilizadas en los procesos de compresión, se propone un nuevo método que usa la transformada *wavelet*. Aprovechando que la transformada *wavelet* discreta (*Discrete Wavelet Transform*, DWT) genera bandas espectrales con la información más relevante y bandas con los detalles, tal y como se resume en la Sección 2.3.3, se trabaja con la banda de menos detalle (bajas frecuencias) para marcar esta vez sobre los coeficientes generados por la DWT. Así, en lugar de marcar sobre los valores de los píxeles (sobre la firma espectral), el método marca sobre los coeficientes de la banda LL (la banda de menor frecuencia) de la transformada *wavelet* de un nivel.

Los pasos a seguir son los mismos que en el proceso general. Únicamente se cambia el proceso de extracción de los bits menos significativos por la aplicación de la transformada *wavelet* discreta a cada una de las bandas. La idea es que una vez se hayan seleccionado las bandas para marcar, se aplique la DWT a cada bloque de  $W \times H$  píxeles y se escojan los coeficientes de la banda más baja para ser marcados con el proceso de construcción de los árboles TSVQ. Estos árboles representaran la cuantización vectorial de los coeficientes de la banda LL y no de los píxeles reales, como se propone en los métodos descritos anteriormente.

Las ratios de compresión que admite esta técnica, con la utilización de la transformada *wavelet*, son más altos que en las anteriores aproximaciones, ya que, de hecho, la compresión JPEG2000 (Taubman and Marcellin, 2002) utiliza esta transformada para realizar la compresión de las imágenes.

La Figura 4.6 muestra este proceso de construcción de los vectores de los coeficientes de la transformada *wavelet* discreta y la reconstrucción completa de la imagen marcada a partir de los coeficientes marcados. En este caso, por el hecho de tener bloques iniciales de  $W \times H$  píxeles y aplicar el proceso TSVQ sobre una sola de las bandas generadas por la DWT (recordemos que se trata únicamente de la banda LL) el conjunto de vectores se divide por 4, y se trabajará con bloques de  $W/2 \times H/2 \times b'$  componentes.

Para regenerar la imagen marcada a partir de los coeficientes de la banda LL marcados, sólo hay que utilizar el resto de las bandas de la DWT que no se han marcado (LH, HL, HH) para realizar la transformada *wavelet* discreta inversa (*Inverse Discrete Wavelet Transform*, IDWT). Esta IDWT generará el bloque marcado a partir de todas las bandas de coeficientes que se generan en el primer paso. La Figura 4.7 muestra este proceso en detalle.

Como se puede ver en la figura, únicamente se utilizan los valores de la banda LL para generar el árbol TSVQ e, iterativamente, se va creando una nueva banda LL' con los valores de los centroides del árbol que satisfacen el criterio seleccionado. Por lo tanto, al final tendremos una banda LL' que sí lo satisfaga y ésta se combinará con las otras bandas de la DWT para aplicar la IDWT sobre el bloque que se está marcando.

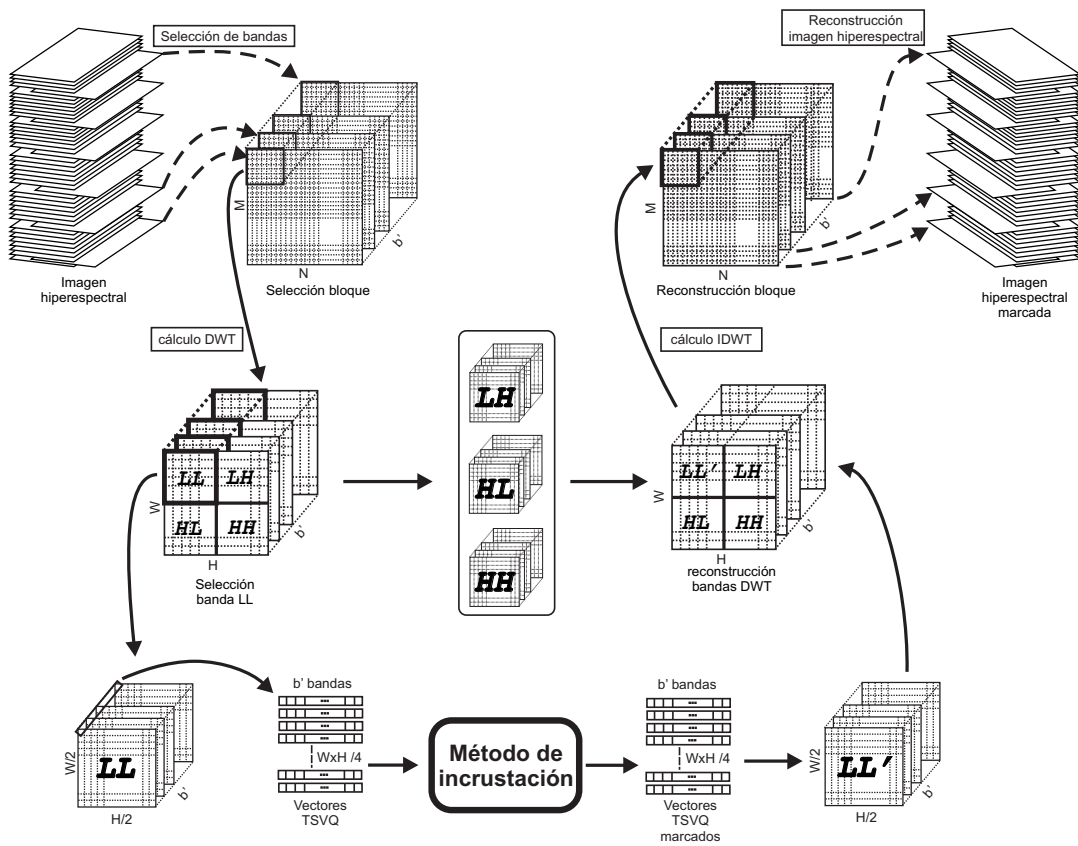


Figura 4.6: Esquema de marcado: construcción y reconstrucción de los bloques con DWT.

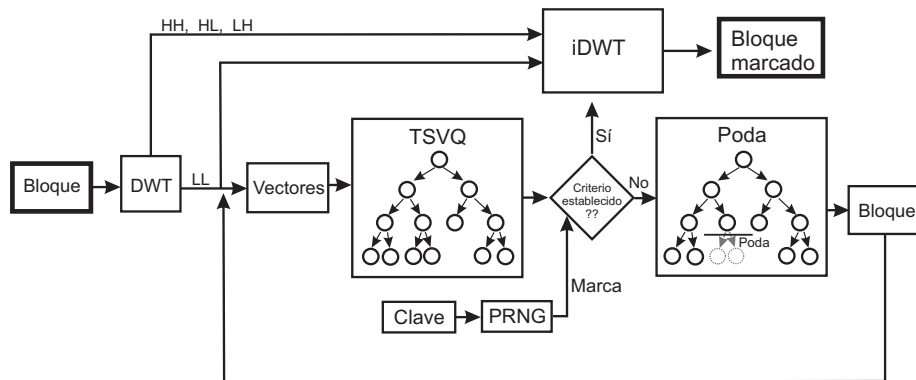


Figura 4.7: Esquema iterativo de marcado con DWT.

#### 4.2.4.1. Resumen

Los siguientes pasos resumen el proceso general de marcado:

0. Preprocesado: compresión y descompresión de la imagen original con JPEG2000 (por ejemplo usando el software KaKaDu (Taubman, 2007)).
1. Selección de bandas y construcción de los bloques de la imagen (como se muestra en la Figura 4.6).
2. Aplicación de la DWT (se guardan las bandas LH, HL y HH).
3. Elección de la semilla del generador de números pseudoaleatorios y la inicialización de éste.
4. Elección de la propiedad a modificar (por ejemplo la entropía).
5. Para todos los bloques LL generados en 2:
  - 5.1 Generar un número con el PRNG y comprobar que no esté repetido en algún bloque ya procesado.
  - 5.2 Escoger el valor de la propiedad según el número generado en el paso 5.1.
  - 5.3 Seleccionar el árbol con la propiedad escogida en el paso 5.2 (ver Tabla 4.1).
  - 5.4 Generar la nueva banda LL' usando el árbol obtenido en el paso 5.3.

- 5.5 Comprobar la propiedad seleccionada en 5.2. Si no se cumple ir a 5.3 con la banda generada en 5.4.
6. Aplicar la IDWT al bloque  $LL'$  marcado junto con las bandas guardadas en 2.
7. Reagrupar todos los bloques (y las bandas no seleccionadas en el paso 1) para construir la imagen marcada final.

En este método la clave secreta necesaria para el proceso de incrustación y en el de detección estará compuesto de la semilla del PRNG y el criterio de poda, además de la división de los bloques, el valor inicial de la propiedad (entropía) y el paso de cuantización de ésta.

#### 4.2.5. Marcado sobre DWT y aplicación de regiones de tamaño variable

El siguiente método selecciona regiones de tamaño variable en la creación de los bloques con los que se marca la imagen. Con los métodos anteriores, si se realiza un estudio completo de los valores de la imagen, se puede intentar atacar al método cambiando regiones muy similares entre sí o calculando el criterio (por ejemplo la entropía) de cada región y sustituyéndolo por otra región forzando un valor similar del criterio. Para minimizar este problema se opta por escoger regiones de la imagen de tamaño variable.

Hasta ahora los bloques seleccionados eran de  $W \times H$  píxeles pero con esta nueva aproximación existirán bloques comprendidos entre  $32 \times 32$  y  $128 \times 128$  dependiendo de cómo de uniforme sea la región que se quiere marcar. En una primera aproximación, en la que se basa este método, la elección de los bloques se realiza de forma manual, comprobando los valores

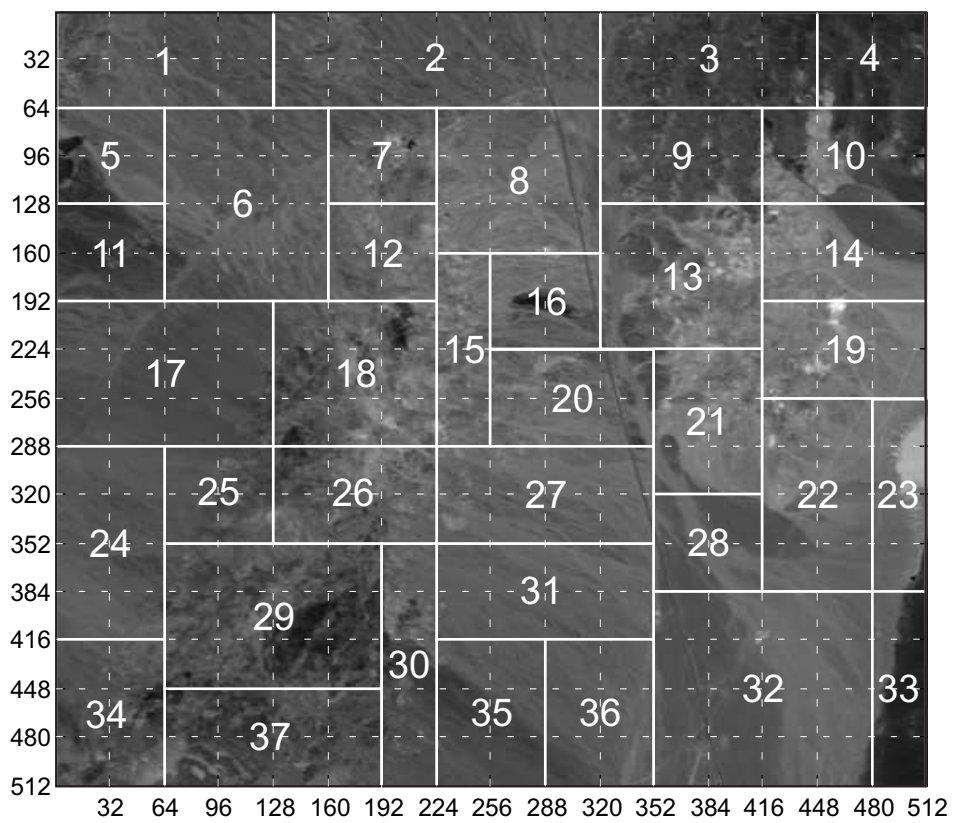


Figura 4.8: Ejemplo de elección del tamaño variable de los bloques de la imagen.

de los componentes que hay en las bandas a marcar. La Figura 4.8 muestra en detalle la elección de la dimensión de cada uno de los bloques en los que se dividirá la imagen hiperespectral para ser marcados.

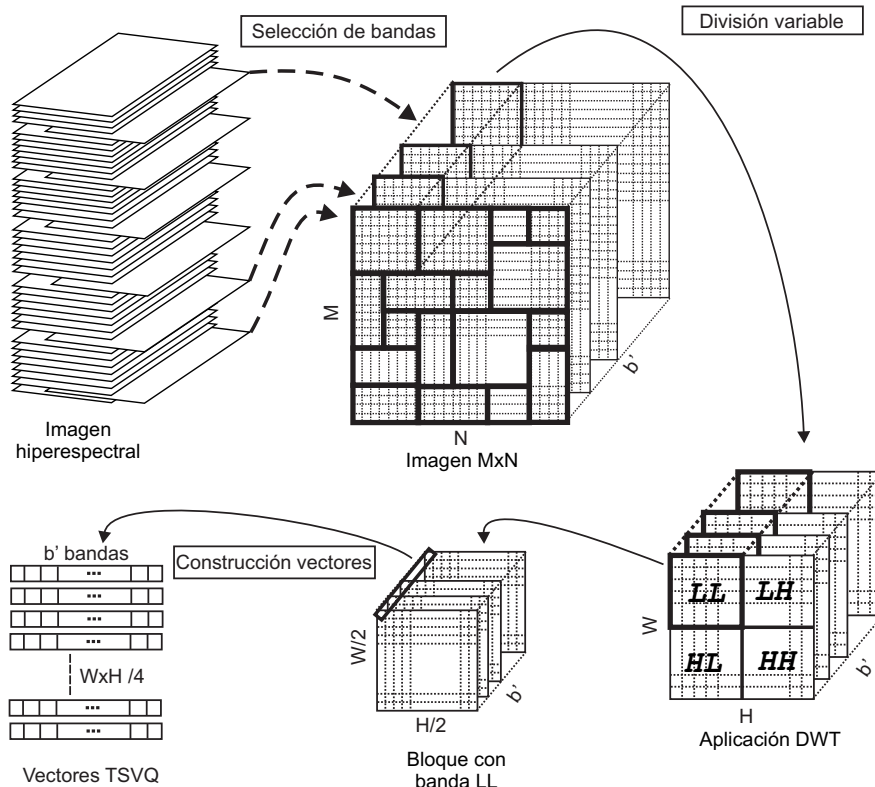


Figura 4.9: Selección de bandas, división por bloques de tamaño variable, aplicación de la DWT y generación de los vectores de marcado.

Al igual que en el método anterior, dividimos la imagen en zonas, que en este caso serán de tamaño variable: no todas las áreas tendrán el mismo número de píxeles como hasta ahora. La imagen entera se parte y para cada uno de los bloques se aplica la DWT del mismo modo que se explica en el método anterior, pero cambiando el paso de cuantización  $h$ , ya que con este método se genera un número diferente de bloques, por lo tanto el paso de

cuantización será en este caso:

$$h = \frac{1}{\text{número de bloques}}.$$

La Figura 4.9 muestra el proceso de generación de los vectores para este método. Una vez seleccionado el bloque, se aplica la transformada discreta *wavelet* y se crean los vectores a partir de la banda LL. A estos vectores se les aplica el algoritmo TSVQ y se comprueba, de forma iterativa, si tienen o no la característica requerida. En caso afirmativo se aplica la transformada inversa con los valores almacenados de las bandas HH, HL y LH, juntamente a la banda LL' marcada y, en caso negativo, se reconstruye el bloque con los parámetros LL de la DWT y se vuelve a crear el árbol hasta que éste satisfaga la característica requerida.

#### 4.2.5.1. Resumen

Los siguientes pasos resumen el proceso general de marcado:

0. Pre-procesado: compresión y descompresión de la imagen original con JPEG2000 (por ejemplo usando el software KaKaDu (Taubman, 2007)).
1. Selección de bandas.
2. Selección de la dimensión de cada bloque. Es necesario guardar la forma de estos bloques.
3. Construcción de los bloques de la imagen (como se muestra en la Figura 4.9).
4. Aplicación de la DWT (se guardan las bandas LH, HL y HH).
5. Elección de la semilla del generador de números pseudoaleatorios y la inicialización de éste.
6. Elección de la propiedad a modificar (por ejemplo la entropía).



7. Para todos los bloques LL generados en 4.
  - 7.1 Generar un número con el PRNG y comprobar que no esté repetido en algún bloque ya procesado.
  - 7.2 Escoger el valor de la propiedad acorde al número generado en el paso 7.1.
  - 7.3 Seleccionar el árbol con la propiedad escogida en el paso 7.2 (ver Tabla 4.1).
  - 7.4 Generar la banda LL' usando el árbol obtenido en el paso 7.3.
  - 7.5 Comprobar la propiedad seleccionada en 7.2. Si no se cumple ir a 7.3 con la banda generada en 7.4.
8. Aplicar la IDWT al bloque LL' marcado reagrupándolo con las bandas LH, HL y HH guardadas en 4.
9. Reagrupar todos los bloques, teniendo en cuenta las dimensiones guardadas (y las bandas no seleccionadas en el paso 1) para construir la imagen marcada final.

En este método la clave secreta necesaria para el proceso de incrustación y en el de detección estará compuesto de la semilla del PRNG, la propiedad de poda y la forma con que se divide la imagen. El paso de cuantización  $h$  se puede deducir a partir de la forma de la división por bloques. También forma parte de la clave secreta el valor inicial de la propiedad (entropía).

#### 4.2.6. Marcado sobre DWT y automatización de la selección de las regiones a marcar

La gran ventaja del método descrito en la Sección 4.2.5, es que minimiza un posible ataque basado en el estudio de las zonas marcadas. Como última mejora se propone un sistema para la automatización del proceso de selección de las regiones de la imagen.

Se realiza un estudio para cada uno de los componentes que hay en cada banda  $b_1, b_2, \dots, b_{16}$ . Para cada  $b_i$  se inicia el proceso dividiendo toda la banda en porciones pequeñas de  $32 \times 32$  componentes y se aplica un estudio para encontrar los valores medios de cada uno de estas porciones. Así pues, se dispone de  $16 \times 16 = 256$  bloques de componentes. Para cada una de las bandas de la imagen por separado, el siguiente paso es la agrupación de los bloques adyacentes teniendo en cuenta si los valores de los componentes son o no parecidos. Para cada uno de los 256 bloques que no esté ya agrupado en un bloque vecino, se comprueba si los bloques adyacentes pueden o no formar un bloque más grande. Es decir, que se irán agrupando bloques de  $32 \times 32$  de la misma banda de la imagen que tengan un valor parecido, dentro de un umbral que es uno de los parámetros del método. Se van formando así bloques más grandes para zonas en las que varía muy poco el contenido (las firmas espectrales son parecidas) y bloques más pequeños para las regiones en las que sí hay una variación por encima de un cierto umbral, fijado como parámetro del método. Al final se comparan las divisiones obtenidas para todas las bandas, y se escoge la distribución del tamaño de los bloques con un número mayor de divisiones, de manera que se obtienen los bloques más pequeños para todas las bandas.

Para cada banda  $b_k$  se realizan los siguientes cálculos:

Se agrupan las divisiones iniciales en grupos de  $128 \times 128$  componentes para que no existan zonas más grandes con la misma marca, como se muestra en la Figura 4.10.

Se obtienen los valores siguientes:

- $M_1$ , la media de los componentes del bloque de  $32 \times 32$  componentes.

	1	2	3	4
1	32x32	32x64	32x96	32x128
2	64x32	64x64	64x96	64x128
3	96x32	96x64	96x96	96x128
4	128x32	128x64	128x96	128x128

Figura 4.10: Agrupación de áreas para automatizar la creación de bloques.

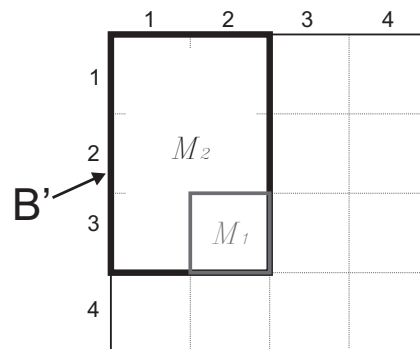


Figura 4.11: Inclusión del bloque  $B_{3,2}$  dentro de  $B'$ .

- $M_2$  la media de los componentes del bloque total en el que está incluido.

La Figura 4.11 muestra el ejemplo del cálculo de las medias para el bloque  $i = 3$  y  $j = 2$ .  $M_1$  será la media del bloque  $B_{3,2}$  situado en la posición (3,2) del bloque  $B'$  y  $M_2$  representará el valor medio de todos los componentes del bloque  $B'$ .

Una vez se disponen de los 16 valores medios para todos los bloques  $B_{i,j}$  indicados en la Figura 4.11 y de las medias de cada uno de los bloques  $B'$  de posibles agrupaciones de éstos, se escoge el  $B'$  que sea más grande y cumpla que la diferencia entre las medias esté dentro del umbral establecido, como

muestra la Ecuación 4.2.

$$\begin{aligned}
 M_1 &= \frac{\sum_{n=1}^{32} \sum_{m=1}^{32} B(n, m)}{32 \times 32}, \\
 M_2 &= \frac{\sum_{n=1}^{32i} \sum_{m=1}^{32j} B'(n, m)}{32i \times 32j}, \\
 |M_2 - M_1| &\leq l.
 \end{aligned} \tag{4.2}$$

donde  $i$  y  $j$  son los índices de los bloques  $B'$  que se están calculando. Por ejemplo, 3 y 2 para el caso descrito en la Figura 4.11

Este proceso se repite para cada una de las bandas  $b_k$  finalizando con la selección de la división en un mayor número de bloques, teniendo en cuenta todas las bandas.

$$Tiling = \text{máx}(\#\text{bloques}) \quad \forall b_k$$

La Figura 4.10 muestra un ejemplo de división por bloques a partir de este proceso. Como se puede observar, existen bloques más grandes para las zonas más homogéneas y bloques más pequeños en las partes en las que el contenido cambia bastante, intentando crear los bloques más grandes posibles para reducir cálculos y marcar de forma más uniforme.

Este método es idéntico al expuesto en la Sección 4.2.5, con la única diferencia que la selección del tamaño de los bloques no se realiza de forma manual, sino que se ha automatizado el proceso.

La Figura 4.12 muestra cómo queda la imagen finalmente con este proceso. Los bloques son ligeramente diferentes a los que se han escogido de forma

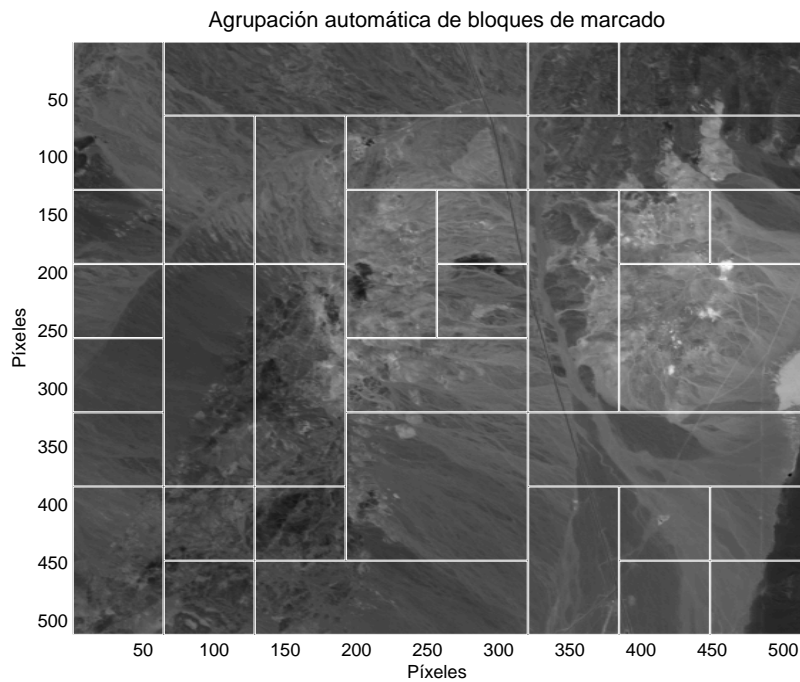


Figura 4.12: Bloques resultantes de la automatización de la selección de las regiones.

manual (Figura 4.8).

### 4.3. Proceso de detección de modificaciones

El proceso de detección para detectar posibles modificaciones fraudulentas en la imagen es muy sencillo y se muestra en la Figura 4.13. Como se puede observar, el proceso es análogo al proceso de marcado que se utiliza en cada uno de los métodos propuestos. La figura muestra el caso más general, en el que no se ha aplicado ninguna transformación a las bandas (el primer método presentado en la Sección 4.2.2). Para los demás métodos se han de construir los vectores teniendo en cuenta el proceso de marcado de la imagen que se

está analizando.

A modo de resumen habrá que:

1. Seleccionar las mismas bandas que se marcaron del total de las  $b$  bandas.
2. Seleccionar los bloques en las mismas posiciones que para el proceso de marcado, ya sea con tamaño constante o variable.
3. Aplicar el método que se ha usado (extracción de los LSB o aplicación de la DWT).
4. Generar la secuencia de números pseudoaleatorios con la misma semilla que se utilizó en el proceso de marcado y calcular la propiedad que debe satisfacer cada bloque a partir de ésta, del valor inicial de la propiedad y del paso de cuantización.

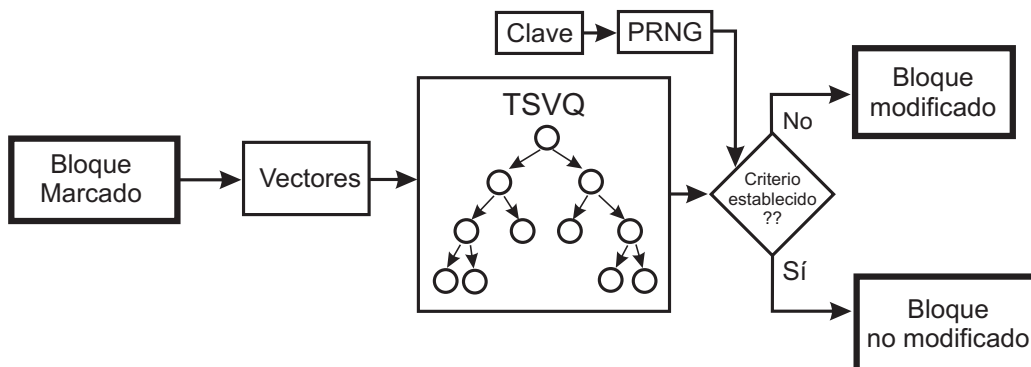


Figura 4.13: Diagrama de bloques del proceso de detección de modificaciones.

La detección de las modificaciones de la imagen se puede realizar comprobando simplemente si el árbol TSVQ que se genera con cada bloque de la imagen tiene o no la propiedad que se introdujo en ese bloque en concreto. Si el bloque verifica el criterio que debería tener, ese bloque se considera como no modificado, en caso contrario, en el que el árbol TSVQ que se ha generado no tiene la propiedad que debería tener, entonces se señala el bloque como modificado y se continúa hasta procesar el resto de la imagen.

Al final se dispondrá de una imagen con los bloques señalados para las zonas en las que se ha detectado una modificación fraudulenta de la imagen marcada.

El método de detección es muy simple y rápido de ejecutar, lo que supone una ventaja del método desarrollado. A diferencia del esquema de incrustación de la marca, la construcción de los árboles TSVQ no es un proceso iterativo, sino que sólo es necesario ejecutarlo una vez para comprobar si el bloque posee o no la propiedad requerida.





# Capítulo 5

## Resultados experimentales

### 5.1. Introducción

En este capítulo se presentan los resultados experimentales para los métodos descritos en el Capítulo 4. Para cada uno de los métodos se presenta la imperceptibilidad obtenida (en términos de PSNR), los resultados de los ataques de compresión que se han realizado y la detección de manipulaciones fraudulentas (ataques de copia y reemplazo). Además, la sección 5.3 muestra una comparativa de los métodos desarrollados en esta tesis con otras propuestas de watermarking para imágenes de obtención remota de la literatura.

### 5.2. Resultados experimentales

Como se describe en la Sección 4.2.2.4, se ha usado un conjunto de imágenes hiperespectrales para evaluar el sistema de detección de modificaciones

propuesto. Para evaluar los métodos, se mide la influencia del proceso de incrustación de la marca en la calidad de la imagen y se muestra un ejemplo de la detección de un ataque de copia y reemplazo y un estudio de ataques de compresión mediante JPEG y JPEG2000.

Se han escogido tres imágenes representativas de este tipo, una parte del área de una mina en Nevada (EE.UU.) (NASA, Jet Propulsion Laboratory, 2004), una área pequeña de unos campos de cultivo en el noroeste del estado de Indiana (EE.UU.) (Landgrebe, 1992) y las imágenes del World Trade Centre en Nueva York (EE.UU.) del 16 de septiembre de 2001 (Clark, 2001).

La Figura 5.1 muestra tres bandas de la imagen de la mina en Nevada, más conocida como Cuprite. La Figura 5.2 muestra la banda 151 de una sección de la imagen AVIRIS Indian Pines y finalmente la Figura 5.3 muestra la zona escogida para la imagen hiperespectral del WTC, concretamente la banda 151, donde aún se puede observar el humo que sale de los escombros.

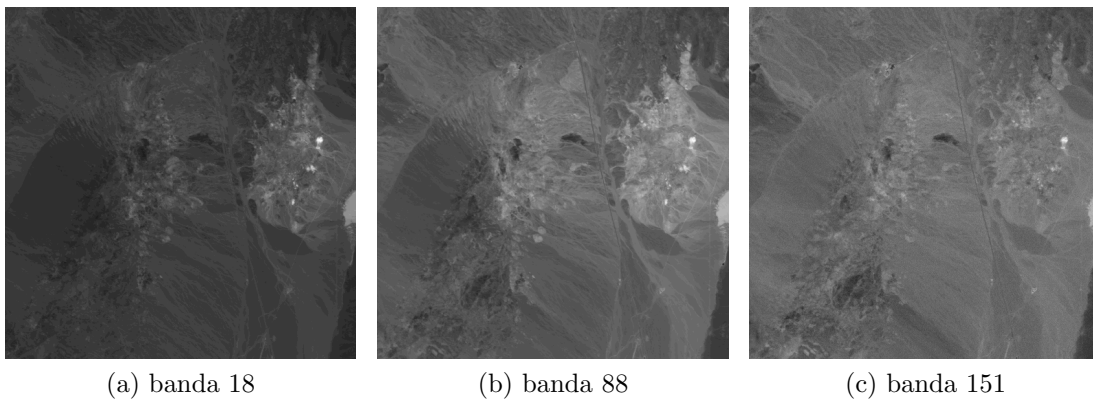


Figura 5.1: Ejemplo de imagen Cuprite.

Existe una gran variedad de imágenes hiperespectrales, pero para demostrar la eficacia de los métodos desarrollados se ha trabajado con estas tres imágenes. La imagen de la mina de Nevada contiene gran cantidad de información acerca de los minerales que hay en la superficie, lo que la hace idónea

para realizar ataques de modificaciones sobre ella. Los diferentes minerales que se pueden encontrar en la imagen darán como resultado una reflectancia diferente y, por lo tanto, como se ve en la Sección 2.1, tendrán firmas espectrales completamente diferentes, lo que nos ayudará a la hora de realizar el marcado de la imagen con árboles TSVQ con más niveles y valores más diferenciados.

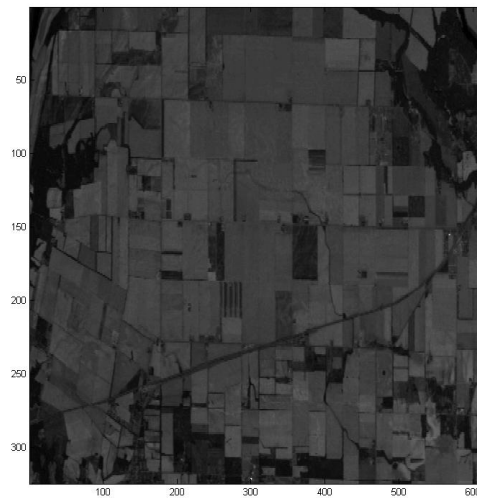


Figura 5.2: Sección de la banda 151 de la imagen Indian Pines.

La imagen Indian Pines refleja más vegetación, ya que corresponde a los campos de cultivo del noroeste del estado de Indiana. Como se puede observar en la Figura 5.2, se trata de una imagen muy regular en la que existe poca diferencia entre las diferentes zonas de la imagen, cosa que dificulta significativamente el proceso de división por zonas semejantes de la imagen.

La tercera imagen con la que se han realizado experimentos de los métodos es una imagen obtenida en día 16 de septiembre de 2001, justo 5 días después del ataque a las torres del WTC de Nueva York. Se tomaron diferentes fotografías durante esos días, tanto de imágenes AVIRIS como de otros formatos. El objetivo principal en el caso de estas imágenes fue la realización

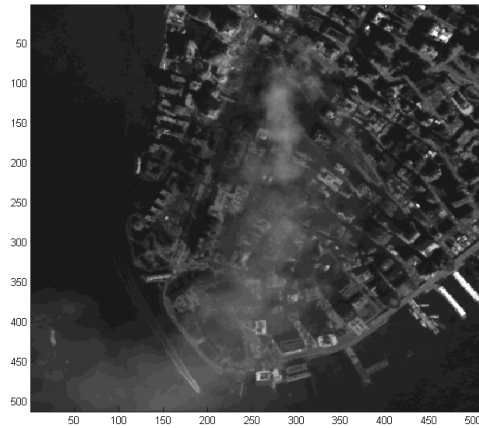


Figura 5.3: Sección de la banda 151 de la imagen WTC del 16 de septiembre de 2001.

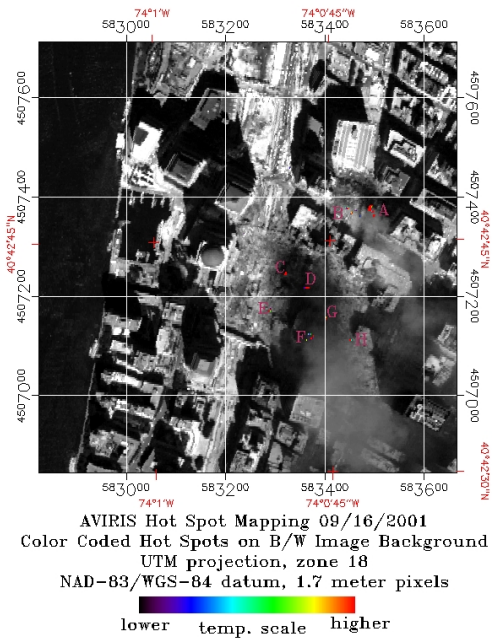


Figura 5.4: Mapa geotermal del WTC del 16 de septiembre de 2001.

de un estudio de la temperatura existente en el WTC los días posteriores al atentado (Clark, 2001). La Figura 5.4 muestra cómo, a partir de los datos obtenidos por estas imágenes, se puede determinar la temperatura a la que se encontraban los restos de las torres.

El objetivo principal en los métodos desarrollados en esta tesis es la protección del contenido para que un comprador potencial de estas imágenes (sólo son públicas una mínima parte de las imágenes hiperespectrales existentes) tenga la certeza de que lo que se está mostrando en la imagen es cierto, por ejemplo, que los minerales que se muestran las firmas espectrales de la mina, la vegetación de los campos de Indiana, o las zonas de altas temperaturas sean realmente ciertas y, en el caso de que se pueda sacar provecho del terreno que representan, sea realmente lo que se está mostrando.

Las tres imágenes representan una gran área de terreno, pero para la realización del método se ha escogido una parte de estas imágenes correspondiente a  $512 \times 512$  píxeles. Esta área se divide en bloques dependiendo del método de marcado empleado. Por ejemplo, en el primer método, con regiones de  $64 \times 64$  píxeles se obtendrán  $8 \times 8$  bloques diferentes.

Cada componente de la imagen está representado por un número entero de dos bytes (16 bits), aunque en algunos casos los dos bits más significativos (MSB) son siempre cero. Así se puede considerar que la imagen se puede representar con 14 bits reales, en lugar de los 16 teóricos.

Como se puede observar en la Figura 5.5, en la que se pueden ver los valores máximos, mínimos y medios para cada una de las bandas, la firma espectral de los píxeles de la imagen Cuprite no contiene ningún componente que supere el valor 10 000, por lo que con 14 bits ya se podrá representar la imagen. La Figura 5.6, que representa los valores máximos mínimos y

medios de los componentes de la imagen Indian Pines, alcanza valores más altos, llegando casi a los 12 000 en las primeras bandas, para decrecer considerablemente este valor máximo en las bandas finales. Como igualmente no se superan los  $2^{14} - 1 = 16\,383$  se puede seguir utilizando 14 bits para representar los valores. Por último, la imagen WTC sí que se tiene valores más elevados en las bandas iniciales, como se puede apreciar en la Figura 5.7, llegando a alcanzar los 21 000, lo que hace que sean necesarios 15 bits en lugar de 14 para representar los componentes, por lo que a la hora de calcular el PSNR de estas imágenes será necesario usar  $2^{15} - 1 = 32\,767$  como valor máximo del componente.

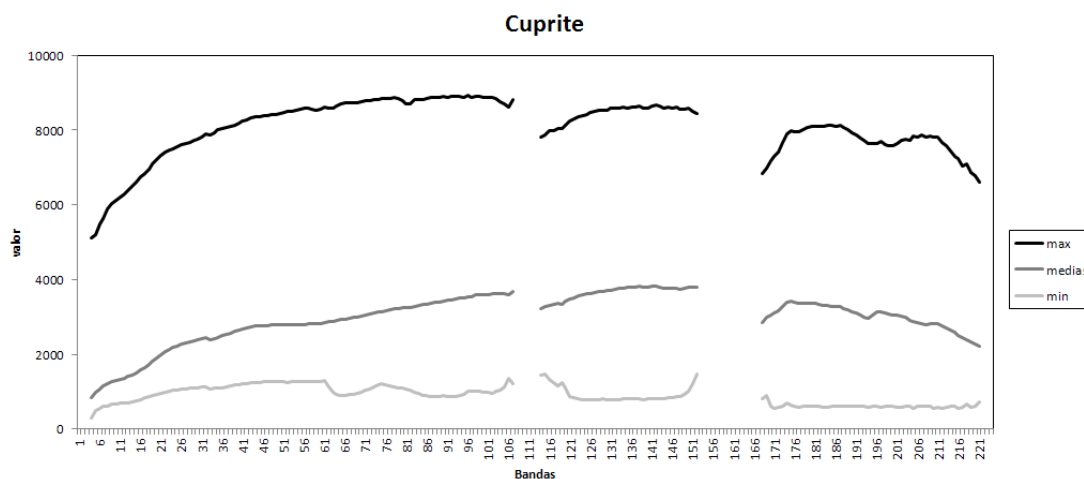


Figura 5.5: valores máximos, mínimos y medios de cada banda de la imagen Cuprite.

Las Figuras 5.5, 5.6 y 5.7 muestran también los valores mínimos y medios de todos los componentes de un píxel, (224 bandas), a parte del valor máximo, que ya se ha comentado. Estas curvas proporcionan una idea de cómo están distribuidos los valores de los componentes en toda esa banda. La imagen de la mina es la que más variedad de valores tiene a lo largo de todas las bandas, ya que hay una gran diferencia entre el valor mínimo y máximo de la banda, cosa que no ocurre para las otras dos. Por ejemplo, en la Figura 5.6 de la

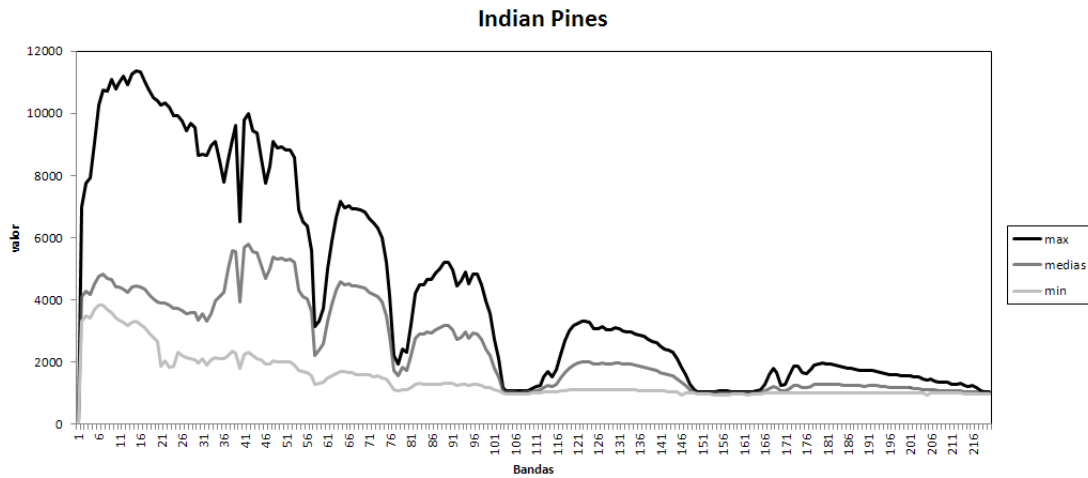


Figura 5.6: valores máximos, mínimos y medios de cada banda de la imagen Indian Pines.

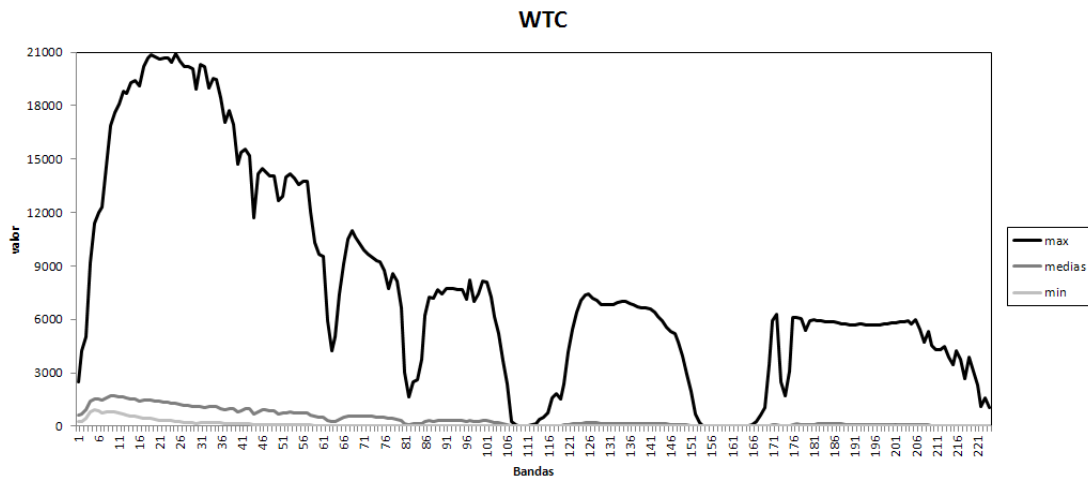


Figura 5.7: valores máximos, mínimos y medios de cada banda de la imagen WTC.

imagen Indian Pines, se puede observar que hay un grupo de bandas (150-166) en las que la diferencia entre todos los valores que todas esas bandas es prácticamente nula, el valor mínimo y máximo de toda la banda es idéntico. Lo mismo pasa en la Figura 5.7 de la imagen WTC, en la que se puede apreciar que, en las bandas iniciales (las bandas que representan al espectro visible), hay una gran variedad de datos, con valores mínimos cercanos a 1 000

y valores máximos cercanos al 21 000, mientras que para las demás bandas los valores máximos se reducen considerablemente. Hay que notar una cosa importante en esta imagen y es que los valores medios de los componentes de cada banda están muy cercanos al valor mínimo. Además, excepto para las primeras 80 bandas, la media es casi cero.

Tanto la imagen de la mina como la obtenida en el WTC de Nueva York disponen de 224 bandas. En cambio la imagen Indian Pines dispone de 220. Esto puede ser debido al tipo de sensor empleado o a que se ha realizado un filtrado previo de todas aquellas bandas de información en las que se produjo un error en el momento de obtener la imagen. Para minimizar los cálculos que se deben realizar en el método de incrustación, sólo se han seleccionado 16 bandas del total disponible. Con estas 16 bandas seleccionadas correctamente ya se puede proteger el contenido de la imagen, por lo tanto no hace falta cargar en exceso el proceso de cálculo del método para obtener los mismos resultados finales. Para escoger las bandas a marcar se pueden establecer diferentes criterios, como seleccionarlas de manera aleatoria, equidistante, en función de parámetros atmosféricos (la reflectancia es diferente para diferentes frecuencias dependiendo del clima), o dependiendo del material que se quiera proteger (algunas frecuencias se reflejarán mejor que otras para materiales concretos). En este caso, se han escogido las bandas equidistantes entre sí, por simplicidad y para proteger toda la forma de la firma espectral por igual. De este modo, se han escogido las siguientes bandas: 4, 18, 32, 46, 60, 74, 88, 102, 116, 130, 144, 151, 172, 186, 200 y 216. Siguiendo el criterio escogido de equidistancia las bandas 158 y 214 deberían haber sido escogidas en lugar de las 151 y 216, pero éstas presentan errores del sensor, ya que los componentes de estas bandas son cero o 65 535. Por lo tanto, se ha optado por escoger otras bandas cercanas. La Figura 5.1 muestra tres de las ban-



das escogidas, que se corresponden con las bandas 18, 88 y 151 teniendo en cuenta la numeración de las 224 bandas.

En el caso de las otras tres imágenes, y para poder comparar los resultados obtenidos de manera más homogénea, se ha escogido exactamente el mismo grupo de bandas.

Por simplicidad, las bandas escogidas para ser marcadas con los métodos se renombran con los índices 1 a 16. A partir de este punto, por ejemplo, la banda 4 pasa a nombrarse banda 1, la banda 18 se nombrará como banda 2, etc., de manera que los índices 1 a 16 se usan para identificar las bandas modificadas, tal y como se muestra en la Tabla 5.1.

Tabla 5.1: Identificación de las bandas modificadas.

Índice	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Posición	4	18	32	46	60	74	88	102	116	130	144	151	172	186	200	216

A partir de aquí se describen los diferentes métodos que se han diseñado y se presentan los resultados obtenidos para cada uno de ellos.

### 5.2.1. Proceso de incrustación de la marca

En este primer método (Serra-Ruiz et al., 2006) la marca se incrusta teniendo en cuenta todos los bits de cada uno de los componentes. Así pues, es de esperar que éste sea el peor método en cuanto a robustez contra ataques de compresión.

Hay que recordar que el PSNR (ecuación 5.1) se calcula usando el valor

máximo que puede tener cada componente de la imagen. Para estas imágenes es aproximadamente 12 000, de esta manera el PSNR se ha calculado con el valor  $2^{14} - 1 = 16\,383$  en vez de usar el máximo que permiten los 16 bits, 65 535, lo que hace que el valor obtenido del PSNR sea más ajustado.

$$\text{PSNR} = 10 \cdot \log_{10} \left( \frac{\max^2}{\text{MSE}} \right) = 10 \cdot \log_{10} \left( \frac{16\,383^2}{\text{MSE}} \right) \quad (5.1)$$

En el caso de la imagen del WTC, en que hay algunas bandas que alcanzan el valor 21 000, como ya se ha visto en el apartado anterior, se aplicará la fórmula del PSNR cambiando el valor máximo del componente a  $2^{15} - 1 = 32\,767$ .

Primeramente, se estudia la imperceptibilidad del método desarrollado teniendo en cuenta las diferentes imágenes descritas. La Tabla 5.2 muestra la transparencia obtenida con la imagen Cuprite para cada una de las 16 bandas con que se realiza el experimento: el PSNR (indicado en decibelios), los componentes modificados (CM), expresados en tanto por ciento respecto al total de cada banda ( $512 \times 512 = 262\,144$ ) y la diferencia media (DM) que se ha modificado cada uno de los componentes respecto a su valor original (en valor absoluto). La Tabla 5.3 muestra la media del PSNR de todas las bandas y los valores obtenidos para la imagen completa de 224 bandas, en la que se han modificado las 16 bandas descritas en la Sección 5.2.

De las tablas se puede destacar que este primer método consigue unos valores muy altos para el PSNR, ya sea de cada una de las bandas modificadas como también de la imagen completa, cosa que ya era de esperar por modificar únicamente pocas bandas y sólo el 1.83 % de los componentes que tiene la imagen. Es decir, que de los  $512 \times 512 \times 224 = 58\,720\,256$  componentes, sólo se han modificado 1 074 580. No obstante, hay que tener en cuenta

Tabla 5.2: PSNR de la imagen Cuprite marcada para cada banda.

#	PSNR (dB)	CM (%)	DM	#	PSNR (dB)	CM (%)	DM
1	60.93	25.54	22.39	9	58.84	25.91	27.79
2	63.29	25.53	15.96	10	64.47	25.44	15.01
3	63.76	25.58	15.76	11	64.38	25.57	15.17
4	63.87	25.63	15.70	12	61.31	25.82	21.54
5	64.34	25.61	14.74	13	62.17	25.78	19.33
6	64.65	25.53	14.26	14	61.67	25.65	19.89
7	64.55	25.55	14.47	15	61.52	25.53	20.92
8	63.74	25.66	16.06	16	59.42	25.89	26.35

Tabla 5.3: PSNR total de la imagen Cuprite marcada.

	PSNR (dB)	CM (%)	DM
Media (bandas marcadas)	62.68	25.64	18.45
Completa (224 bandas)	73.73	1.83	18.45

que, de media, los componentes modificados lo han hecho en una variación de 18.45 (sobre el máximo de 16 384).

El valor descrito como diferencia media (DM) representa la media de las modificaciones introducidas por los métodos descritos, es decir, entre los valores originales y los valores marcados de los componentes (en valor absoluto). Aunque en las tablas aparece un valor decimal, en realidad éste representa la media de un conjunto de números enteros, ya que todas las diferencias de las modificaciones son valores enteros.

En el caso de la imagen Indian Pines, los resultados, que se muestran en las Tablas 5.4 y 5.5, son muy parecidos a los obtenidos para la imagen Cuprite, aunque ligeramente mejores que para ésta. Como se puede observar, aunque se modifican más componentes, éstos lo hacen con menor diferencia respecto al valor original, por lo que la imagen marcada se aproxima más a la imagen original (en términos de PSNR).

Tabla 5.4: PSNR de la imagen Indian Pines marcada para cada banda.

#	PSNR (dB)	CM (%)	DM	#	PSNR (dB)	CM (%)	DM
1	60.22	30.28	20.26	9	71.08	29.97	6.61
2	60.91	31.13	20.23	10	65.24	30.77	12.27
3	60.39	30.68	21.78	11	69.08	30.26	8.12
4	60.41	31.17	21.51	12	72.18	29.65	5.97
5	64.45	30.93	13.84	13	73.39	29.51	5.17
6	62.04	31.09	17.80	14	71.96	29.64	5.88
7	62.47	31.05	16.95	15	72.88	29.55	5.35
8	68.56	30.40	8.82	16	75.65	29.13	4.60

Tabla 5.5: PSNR total de la imagen Indian Pines marcada.

	PSNR (dB)	CM (%)	DM
Media (bandas marcadas)	66.88	30.32	12.20
Completa (224 bandas)	75.62	2.20	12.20

En el caso de la imagen del WTC, la imagen es mucho más variada, ya que el terreno contiene agua, cemento, vegetación, etc. Las Tablas 5.6 y 5.7 muestra los resultados obtenidos. Se puede observar cómo los resultados son mucho mejores que para las otras dos imágenes: el PSNR es mayor y las diferencias introducidas por el método son considerablemente menores que en los dos casos anteriores.

Tabla 5.6: PSNR de la imagen del WTC marcada para cada banda.

#	PSNR (dB)	CM (%)	DM	#	PSNR (dB)	CM (%)	DM
1	75.30	19.22	7.31	9	91.82	1.73	6.61
2	75.58	18.85	8.48	10	84.93	2.96	12.27
3	76.60	19.75	7.54	11	86.84	2.52	8.12
4	75.63	20.47	7.96	12	91.94	1.71	5.97
5	81.46	19.44	4.51	13	84.16	3.21	5.17
6	80.06	19.92	5.01	14	83.07	3.56	5.88
7	83.21	19.12	3.61	15	86.81	2.51	5.35
8	81.49	19.61	4.48	16	90.12	1.92	4.60

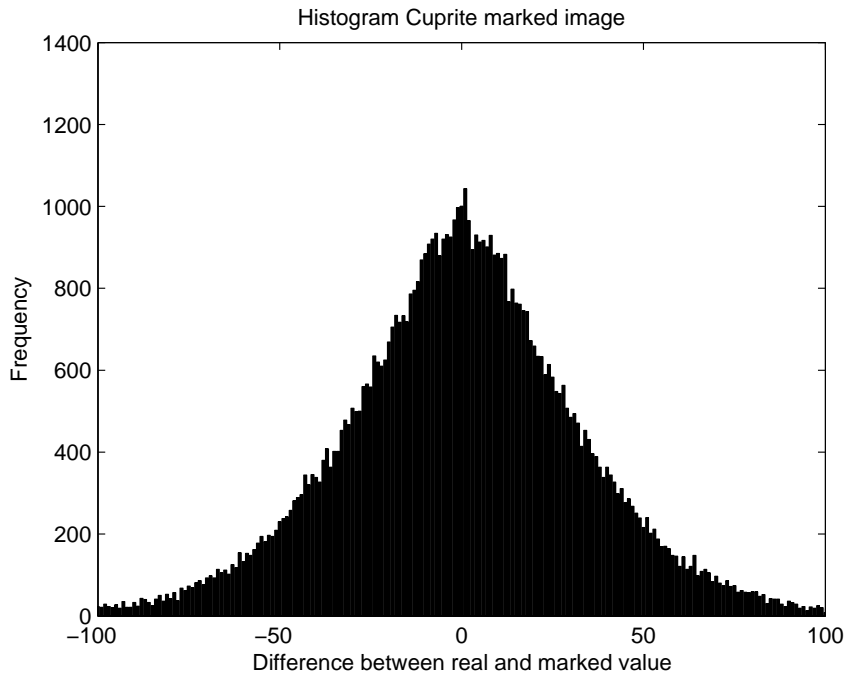
Tabla 5.7: PSNR total de la imagen del WTC marcada.

	PSNR (dB)	CM (%)	DM
Media (bandas marcadas)	83.07	18.62	4.31
Completa (224 bandas)	91.63	1.33	4.31

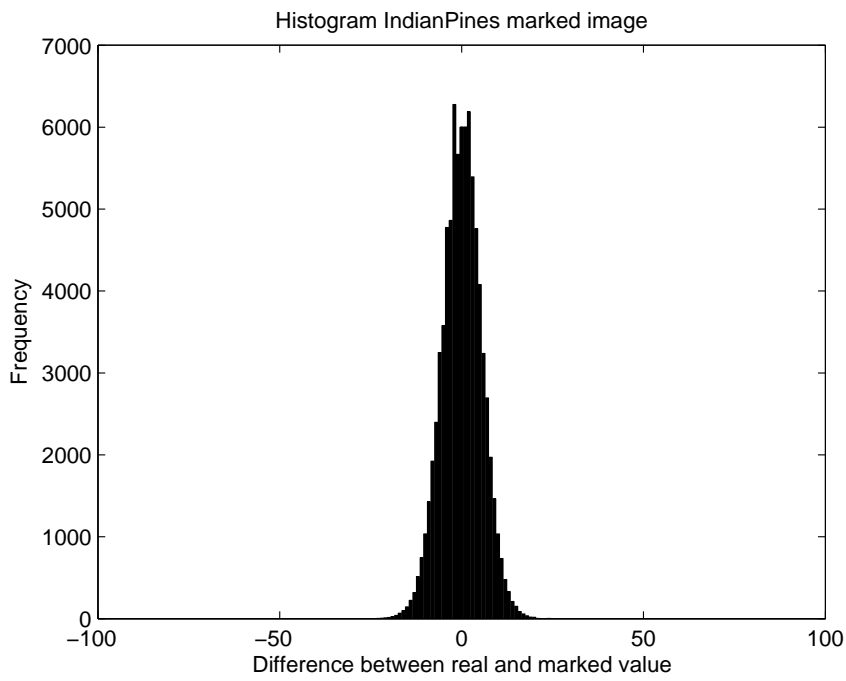
La Figura 5.8 muestra el histograma de las diferencias para cada una de las imágenes marcadas. Se puede apreciar, al igual que en las Tablas 5.2, 5.4 y 5.6, que la imagen que proporciona mejores resultados es la del WTC de Nueva York, ya que tiene más variedad de contenido y unas firmas espectrales con menos energía que el resto, lo que hace que el árbol TSVQ no tenga tanta profundidad.

La Figura 5.9 muestra cómo afecta este método a una firma espectral en concreto. La figura muestra la firma espectral del píxel  $150 \times 150$  de la imagen Cuprite, tanto la original como la marcada que, para poder ser diferenciada de la original, se muestra desplazada 200 unidades hacia abajo. Como se puede observar, ambas firmas son prácticamente idénticas y los métodos de clasificación (Plaza et al., 2005; Melgani and Bruzzone, 2004) descritos en la literatura que utilizan este tipo de imágenes no se verán afectados ya que, como se ha mostrado, el método no llega a afectar al 2% de los componentes de la imagen original.

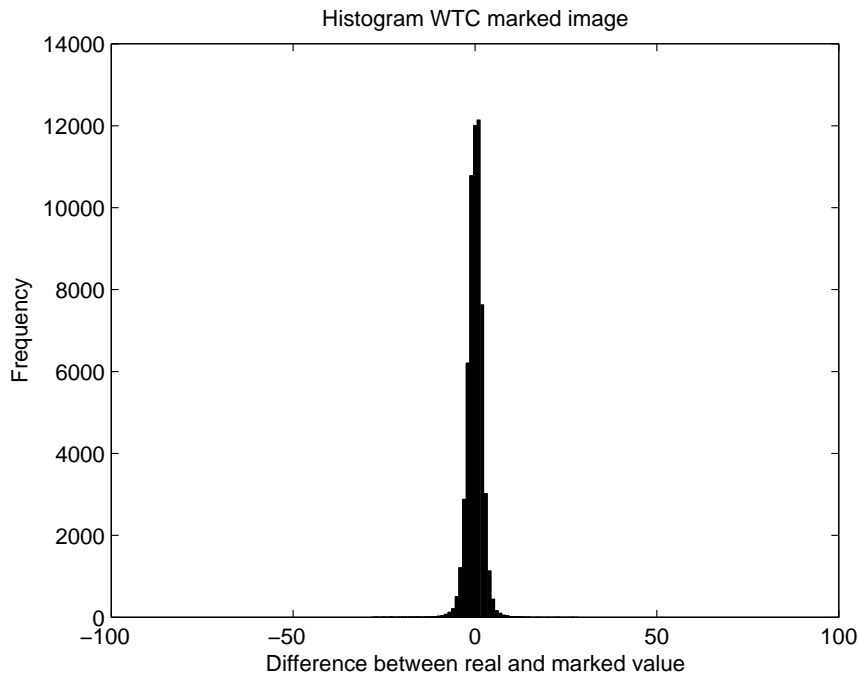
En la Figura 5.9 se puede apreciar que existen zonas en las que no hay ningún valor (están en blanco). Ello es debido a que se han eliminado esos valores ya que, o bien eran cero, o bien valores excesivamente altos, cercanos al 65 535, que claramente denotan un error del sensor en la captación de la luz. No se ha marcado ninguna de esas bandas, ya que se considera que en ellas no existe información válida de la imagen.



(a) Cuprite



(b) Indian Pines



(c) WTC

Figura 5.8: Histograma de las diferencias entre valores originales y marcados.

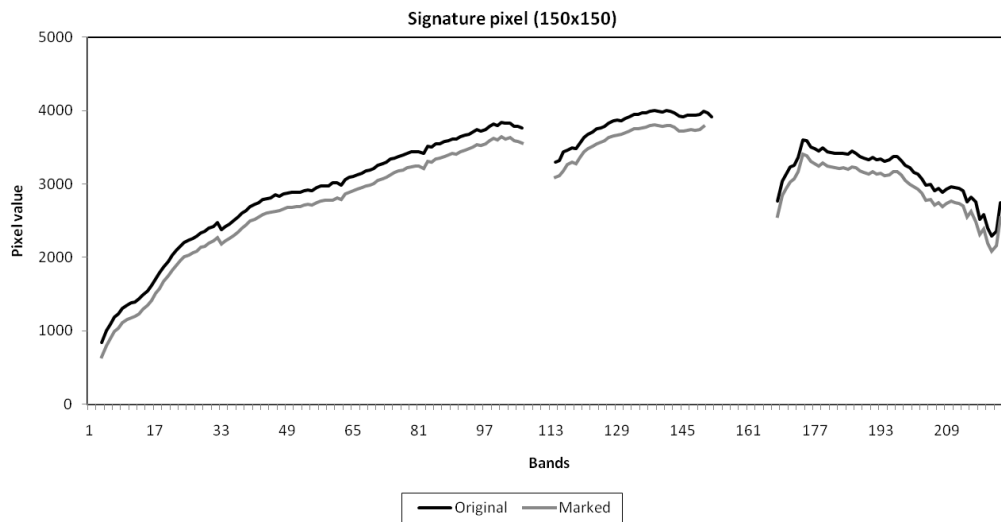


Figura 5.9: Diferencia entre la firma espectral original y marcada (desplazada) de un píxel marcado.

### 5.2.1.1. Capacidad

En general, para los métodos realizados en esta tesis, la capacidad se obtiene a partir de la cuantización vectorial que se aplica en ellos. La ima-

gen se divide en 64 bloques como máximo y la propiedad escogida para la cuantización también tomará, como máximo, 64 valores diferentes, ya que se hace coincidir el número de bloques existentes en la imagen con el número de intervalos a cuantizar de la propiedad escogida. Por tanto la capacidad máxima de los diferentes algoritmos es de 384 bits.

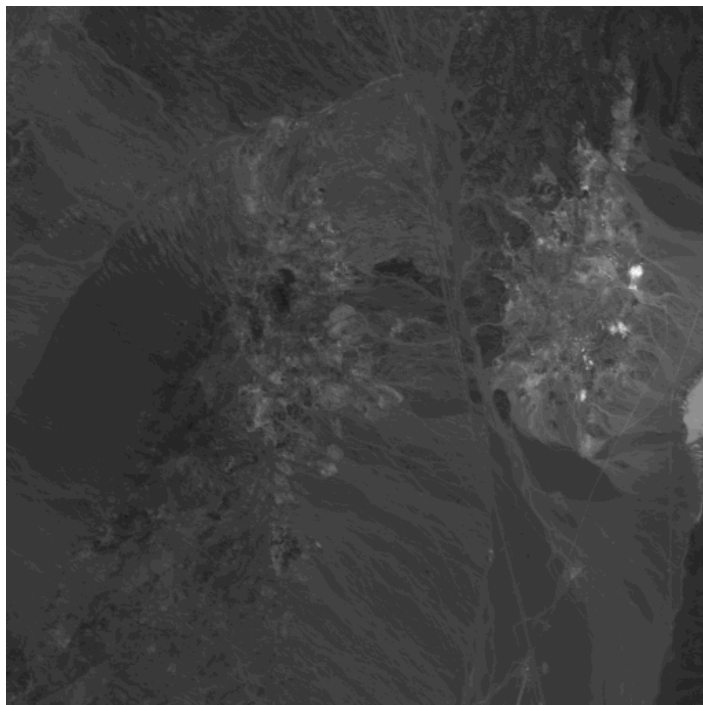
$$64 \text{ valores de la propiedad} = 2^6 \Rightarrow 6 \text{ bits} \times 64 \text{ bloques} = 384 \text{ bits.}$$

#### 5.2.1.2. Ataques de copia y reemplazo

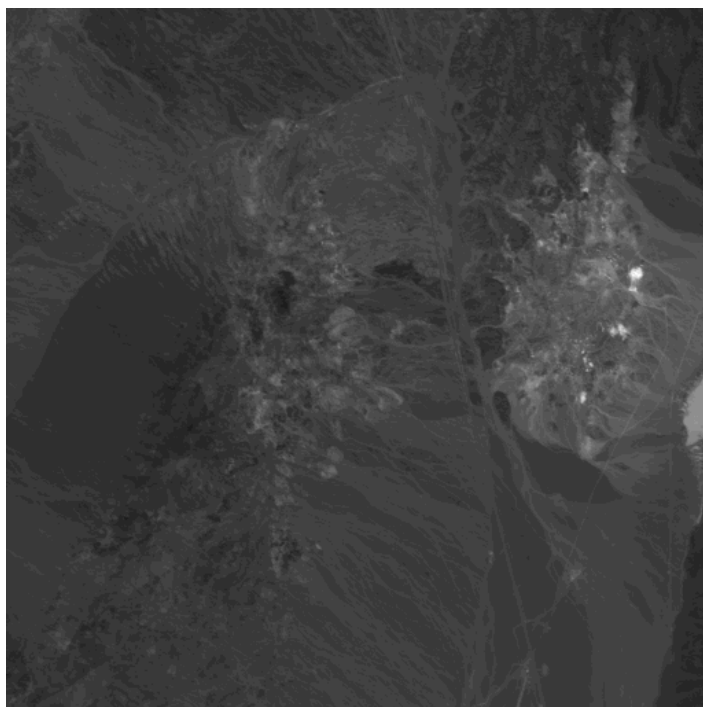
Para validar el esquema de marcado se han realizado pruebas de sustitución de áreas de la imagen por otras, seleccionando una zona y copiándola sobre otra diferente para falsificar esa parte de la imagen.

La detección de las zonas modificadas funciona correctamente para las tres imágenes. Sólo hay que aplicar el método de detección a la imagen para obtener la localización de la modificación realizada. En todos los casos, el sistema detecta estas modificaciones correctamente sin obtener falsos positivos o falsos negativos. Las Figuras 5.10 y 5.11 muestran este proceso. La Figura 5.10(a) muestra la banda 2 de la imagen Cuprite y la Figura 5.10(b) muestra la imagen marcada con el método descrito en la Sección 4.2.2. No se puede apreciar diferencia visual entre las dos imágenes, ya que como se ha visto, el PSNR es de 63.29 dB, que representa un valor relativamente alto. La Figura 5.11(a) muestra un ataque de copia y reemplazo, donde hay una zona que se ha sustituido por otra muy parecida y la Figura 5.11(b) muestra la detección y localización de la región modificada. El método de detección



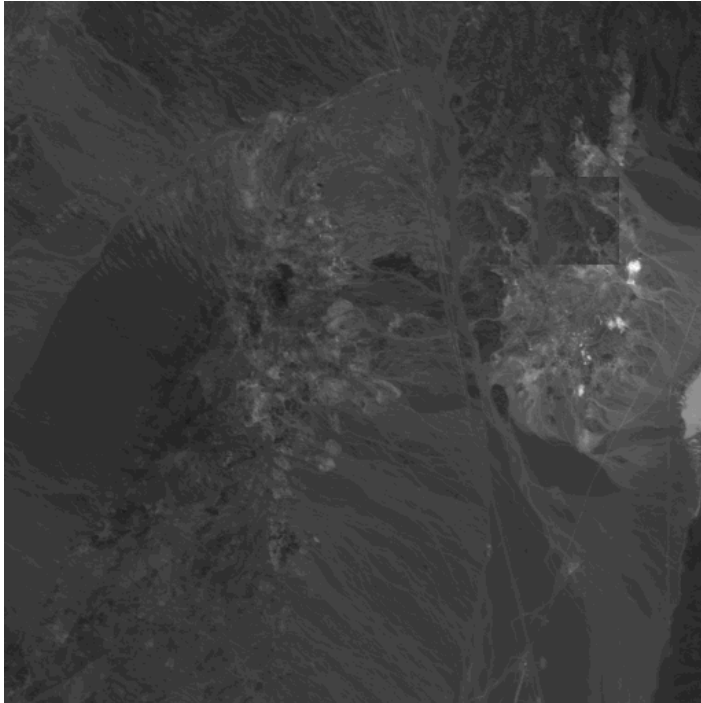


(a) Original

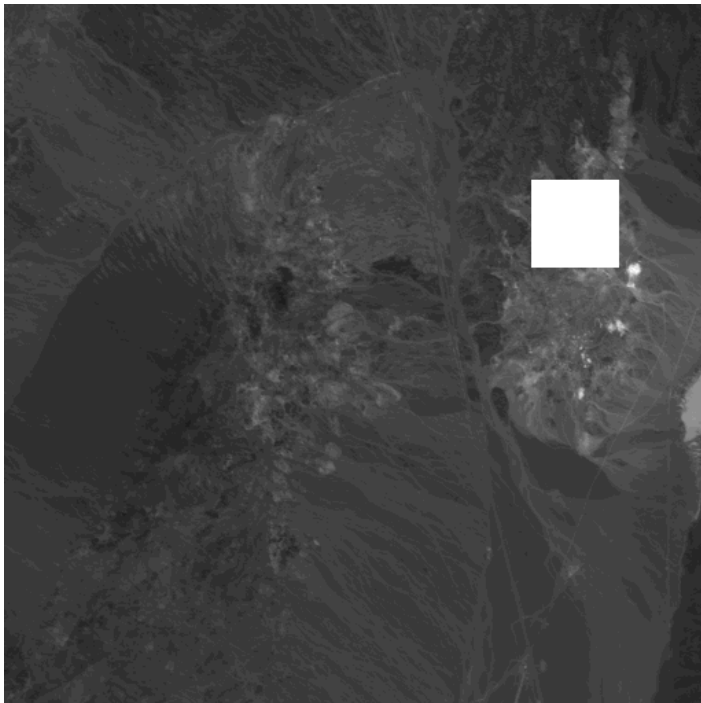


(b) Marcada

Figura 5.10: Imagen original (a) y marcada (b) de la banda 2.



(a) Modificación



(b) Localización

Figura 5.11: Imagen modificada (a) y localización de la modificación (b).

ha encontrado un valor erróneo de la entropía y, por consiguiente, señala en blanco el bloque que contiene la modificación.

### 5.2.1.3. Ataques de compresión

Para determinar la robustez del método frente a ataques de compresión se realizaron diferentes pruebas con el software Kakadu (Taubman, 2007), que realiza compresión JPEG2000 con diferentes parámetros de compresión. Las pruebas se han realizado siempre comprimiendo cada una de las 16 bandas marcadas separadamente. Se presentan los resultados para la compresión JPEG2000 que es la que mejores resultados da en cuanto a compresión y calidad de imagen, comparada con la compresión JPEG. Para otros métodos sí que se presentan los resultados de robustez frente a compresión JPEG. La Tabla 5.8 muestra, para las diferentes imágenes que se han utilizado, los valores de compresión a que se puede llegar (comprimiendo y descomprimiendo) sin eliminar la marca de las imágenes, relacionándola también con la alteración de ésta por el hecho de comprimir con pérdida de datos. En los tres casos, la imagen conserva la marca para compresión de hasta 8 bpp, mientras que la marca es eliminada si se comprime con menor calidad. La tabla muestra la información siguiente en sus columnas: la imagen con la que se ha realizado el ataque, la ratio de compresión del ataque, la resistencia de la marca al ataque, el PSNR para cada uno de los ataques, la distancia media en que se cambian los componentes, la distancia máxima de esta diferencia, el porcentaje de los componentes modificados y las bandas que no se modifican.

Se puede observar que la imagen que mejor comportamiento tiene en cuanto a degradación por compresión es la WTC, mientras que la peor es

Tabla 5.8: Resultados de ataques de compresión JPEG2000.

Imagen	Compr. (bpp)	Resistencia marca	PSNR (dB)	DM	Difer. máx.	CM %	Bandas no modificadas
Cuprite	8	Sí	85.06	1.19	5	50.47	1
	7	No	81.01	1.55	9	71.71	0
	6	No	76.34	2.37	17	83.75	0
	5	No	70.96	4.11	32	91.20	0
Indian Pines	8	Sí	83.61	1.31	7	26.42	9
	7	No	82.32	1.61	13	39.16	6
	6	No	80.77	2.07	23	56.69	2
	5	No	76.68	3.15	43	75.48	0
WTC	8	Sí	90.24	1.08	4	8.42	12
	7	No	85.16	1.23	8	23.23	9
	6	No	82.96	1.51	11	38.51	6
	5	No	81.89	1.90	23	57.18	2

la Cuprite. La diferencia media (DM) que se obtiene de la aplicación de este ataque a la imagen Cuprite es peor que la que se obtiene de la imagen WTC. Es decir, que la imagen WTC se modifica en menor medida que el resto. Además, si tomamos en cuenta los otros valores observados, encontramos que se modifican muchos menos componentes en esta última imagen que en las otras dos y, por otro lado, la diferencia máxima que se modifica un componente es ligeramente menor para la imagen WTC que en las otras dos. El algoritmo JPEG2000 deja exactamente igual algunas bandas después de comprimir y descomprimirla con una ratio de compresión suficientemente alto. En el caso de la imagen Cuprite esto pasa en un solo caso, con una ratio de compresión de 8 bpp y en una sola banda. En el caso de la imagen Indian Pines, en la compresión de 8 bpp, 7 bpp y 6 bpp para 9, 6 y 2 bandas respectivamente. En la imagen WTC, en todos los casos se puede observar que hay bandas en las que la compresión no afecta a los valores de los componentes.

### 5.2.2. Método basado en la extracción de los LSB

El segundo método (Serra-Ruiz and Megías, 2011) se basa en la extracción de los bits menos significativos (LSB) de todos los componentes de cada una de las bandas marcadas y construir los árboles TSVQ sobre los restantes bits más significativos (MSB). Los componentes que hasta ahora se representaban con 16 bits pasan a tener un número menor, que denotaremos como  $16 - n$ . Cabe destacar que el primer método mostrado en el apartado anterior es un caso particular de este segundo, con el valor de  $n = 0$

Como se comenta anteriormente, las imágenes realmente no utilizan todo el rango de valores posible. Por lo tanto, la elección de este parámetro  $n$  puede afectar considerablemente al proceso. Los métodos propuestos se basan en encontrar centroides de las regiones de Voronoi que satisfagan la propiedad escogida para cada uno de los árboles de los bloques que conforman la imagen global. Así, si se dispone de pocos bits de información y extraemos demasiados LSB, se corre el riesgo de no poder construir el árbol TSVQ por falta de información.

Veamos un caso concreto. Sea una imagen de 16 bits de resolución, en que una de las bandas tiene poca energía, con un valor máximo de 3 300, y se considera  $n = 4$ .

$$3\ 300 = 110011100100_2 \Rightarrow \downarrow 4 \Rightarrow 11001110_2 = 206. \quad (5.2)$$

Tras eliminar 4 LSB, el valor máximo de 3 300, como muestra la Ecuación 5.2, pasa a ser 206, lo que hace que todos los componentes de esa banda estén comprendidos entonces de 0 a 206, (aunque raramente las bandas tienen componentes muy cercanos al 0). Un caso como éste dificultaría la creación

de los árboles TSVQ, por lo tanto se deberá escoger el valor de  $n$  de acuerdo a las características de la imagen y al proceso de construcción de los árboles TSVQ.

Desde el punto de vista de la robustez, el parámetro  $n$ , a pesar de lo expuesto, interesa que sea lo mayor posible, ya que esto nos garantizará mejores resultados frente a los ataques de compresión y descompresión de la imagen. Cuanto más elevado sea el valor de  $n$  más robusto será el marcado para resistir a un ataque de compresión. Por otro lado, cuanto mayor sea el valor  $n$  en principio peor puede ser la transparencia (PSNR), ya que se modificarán los componentes con valores más altos al introducir los valores del árbol TSVQ a partir del bit  $n + 1$  de cada componente que se debe modificar. No obstante, los experimentos demuestran que el PSNR puede incluso aumentar pese a usar valores mayores del parámetro  $n$ .

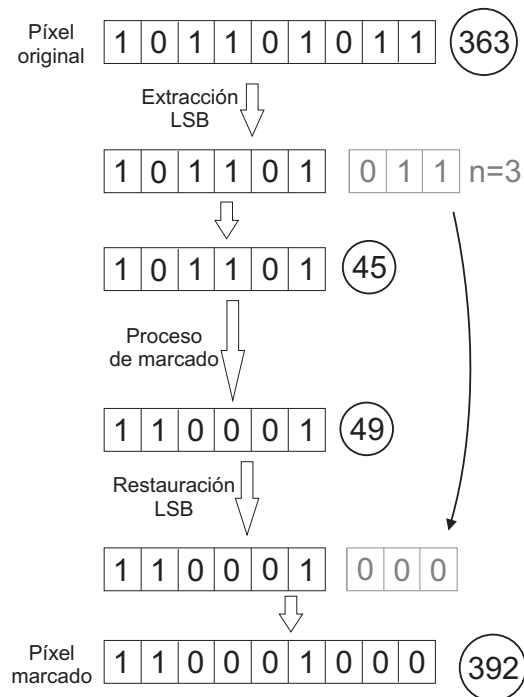


Figura 5.12: Proceso de marcado de un componente con  $n = 3$ .

La Figura 5.12 muestra cómo afecta el valor de  $n$  seleccionado. Para el caso de  $n = 3$  podemos ver cómo un valor inicial de 363 se puede convertir en 392, lo que representa una diferencia de 29 teniendo en cuenta que el proceso de marcado sólo ha cambiado el centroide en 4 unidades (de 45 a 49). En cambio si se hubiera escogido el valor de  $n = 2$  y con la misma diferencia de 4 unidades como valor de marcado, se hubiera obtenido el valor 376 como componente marcado, con lo que la diferencia pasaría de 29 a 13. Cuanto mayor sea  $n$ , más robusto será el esquema contra compresión, pero se introduce mayor distorsión en los componentes marcados. Sin embargo, el número de componentes modificados puede ser menor al aumentar el valor de  $n$ , por lo que no necesariamente el PSNR debe empeorar.

Por consiguiente, para encontrar el valor óptimo en cada imagen, se han de realizar experimentos con diversos valores de la variable  $n$  y de la clave de generación de los números pseudoaleatorios que hará cambiar el valor de cuantización para cada bloque.

Los resultados obtenidos para diversos valores de  $n$  se muestran más adelante, pero a continuación se describen, en detalle, los resultados obtenidos para el valor  $n = 2$ , es decir, en el que se extraen 2 LSB de la imagen original, siendo el procedimiento idéntico para otros valores del parámetro  $n$ . Se presentan los resultados únicamente para la imagen Cuprite, al ser ésta la imagen con la que el método anterior ( $n = 0$ ) obtiene peores resultados. En el caso de las otras dos imágenes (Indian Pines y WTC), los resultados obtenidos siempre mejoran los presentados para la imagen Cuprite.

El PSNR de la imagen Cuprite completa, incluyendo las 224 bandas, es de 74.12 dB y el porcentaje de componentes modificados es de 1.72%, lo que quiere decir que sólo 1 008 859 de los 58 720 256 componentes se han

modificado en el proceso de marcado con este método.

Tabla 5.9: PSNR de la imagen Cuprite marcada para cada banda.

#	PSNR (dB)	CM (%)	DM	#	PSNR (dB)	CM (%)	DM
1	61.224	24.002	22.368	9	59.110	24.868	27.485
2	63.657	23.548	15.788	10	64.957	23.701	14.602
3	64.215	23.811	15.393	11	64.851	23.875	14.787
4	64.335	23.841	15.344	12	61.688	24.583	21.122
5	64.884	23.616	14.415	13	62.598	24.353	18.880
6	65.221	23.487	13.894	14	62.101	24.356	19.366
7	65.068	23.645	14.104	15	61.882	24.429	20.349
8	64.208	23.899	15.656	16	59.746	24.835	25.877

La Tabla 5.9 muestra los valores del PSNR, los componentes modificados y la distancia media con que se modifican estos componentes. Como se puede apreciar, los valores son ligeramente mejores que en el caso del método anterior (Tabla 5.2). Se mejora el PSNR ya que se modifican menos componentes y los que se modifican lo hacen con una distancia media ligeramente menor.

Se puede apreciar también que el PSNR en algunas bandas (9 y 16) es ligeramente diferente al resto. Esto es debido a la poca información que tienen esas bandas lo que provoca que el árbol TSVQ sea menos profundo dando lugar a compresiones más altas y por tanto variaciones también más significativas en los componentes, dado que la diferencia media alcanza valores mayores. De media, el PSNR de las 16 bandas marcadas es de 63.11 dB, el porcentaje de componentes que se han modificado en este caso es de 24.05% de los  $512 \times 512 \times 16$  componentes de las bandas marcadas. El valor medio de la modificación es 18.09 (el rango de los valores de esta imagen es de 500 a 9 000).

Este quiere decir que el PSNR con  $n = 2$  mejoran respecto al caso  $n = 0$ . Este resultado, aparentemente contradictorio (ya que se han eliminado 2



LSB), es debido a que ahora se está marcando con valores que ya están comprimidos, es decir, que al haber eliminado 2 LSB ya hay  $2^2 = 4$  valores de componentes que se representan con el mismo valor en los árboles TSVQ.

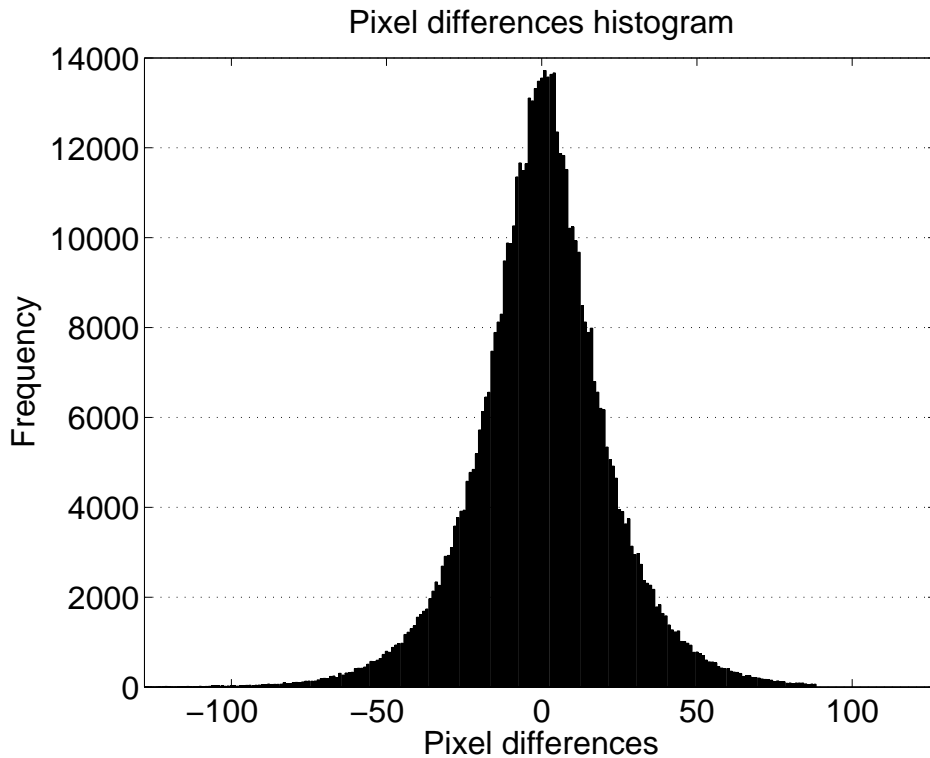


Figura 5.13: Histograma de las modificaciones del proceso de marcado con  $n = 2$ .

La Figura 5.13 muestra el histograma de las diferencias entre los valores originales y los marcados para la imagen Cuprite con  $n = 2$ . Se puede observar la diferencia de los componentes modificados respecto al valor original (no se muestran los casi 99% de componentes que no se han alterado). A simple vista, ya se observa que la modificación es menor con respecto a la Figura 5.8(a), ya que ahora se modifican menos los componentes y la gráfica está mucho más comprimida en torno al valor 0. Ahora encontramos cerca de 14 000 valores muy cercanos al 0, cuando para el anterior método eran sólo aproximadamente 1 000 los componentes que se modificaban tan poco.

La Figura 5.14 muestra la variación de la firma espectral introducida en el proceso de marcado. Se representa la firma espectral del píxel  $300 \times 300$ , tanto la original como la marcada, que para poder ser diferenciada de la original, se ha desplazado 200 unidades hacia abajo. En este caso el píxel  $150 \times 150$  no se ha modificado y por eso se muestra otro diferente. Como en los resultados presentados en la Sección 5.2.1, la imagen sólo se ve afectada en poco más del 1% de los componentes, por lo que los métodos de clasificación no verán alterado su resultado por usar la imagen marcada.

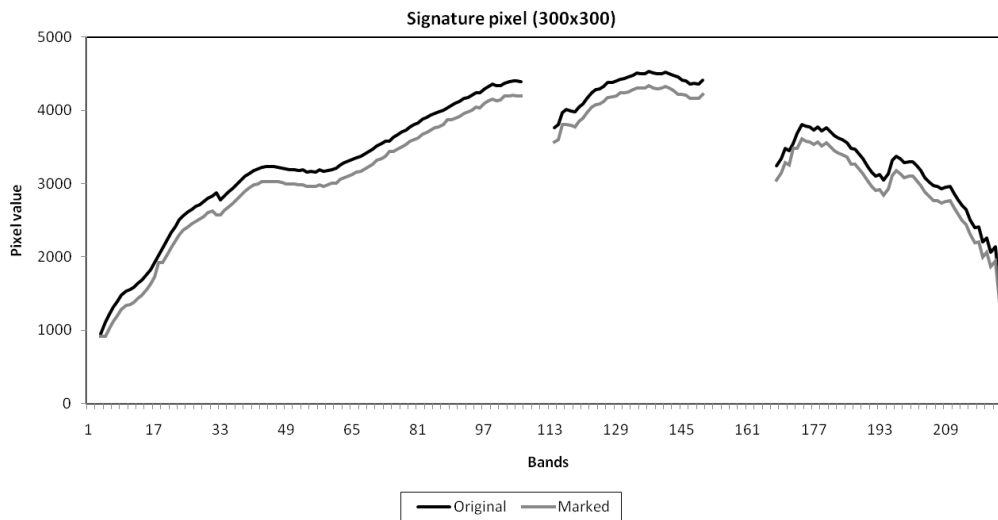


Figura 5.14: Diferencia entre la firma espectral original y marcada (desplazada) de un píxel marcado.

### 5.2.2.1. Análisis de la imperceptibilidad y reducción de las imágenes a 8 bpp

Como ya se ha comentado en el Capítulo 4 existen algunos métodos que trabajan con imágenes de una sola banda o marcan en las bandas de forma separada. Para poder comparar los resultados de este método con otros esquemas en los que se trabaja con imágenes de 8 bits por píxel, se ha generado

una versión de las imágenes con 8 bits por píxel, eliminando los bits menos significativos de cada componente y teniendo en cuenta el valor máximo de todos los componentes.

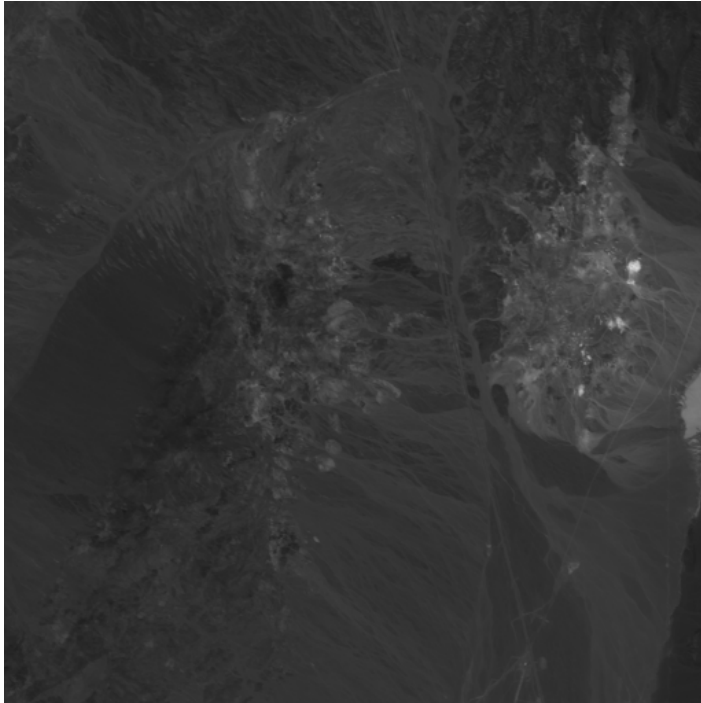
La Figura 5.15 muestra la imagen Cuprite original reducida a 8 bpp (5.15(a)) y la misma una vez ha sido marcada (5.15(b)) con el método descrito y  $n = 2$ . No se puede apreciar diferencia entre las dos imágenes, al igual que tampoco se observan diferencias significativas con las imágenes de 16 bpp.

Tabla 5.10: Valores de PSNR para diferentes imágenes y valores de  $n$ .

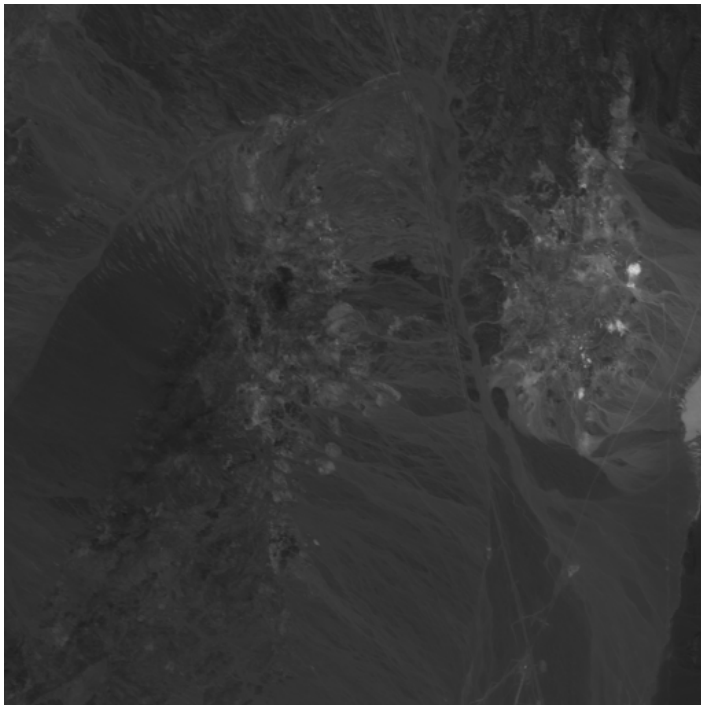
Imagen	$n$	PSNR (dB) Bandas marcadas	PSNR (dB) Imagen completa	DM	CM (%)
Cuprite (14 bpp)	2	66.25	77.34	16.84	0.96
	3	66.50	77.57	17.07	0.87
	4	67.58	78.42	17.20	0.71
Indian Pines (14 bpp)	2	70.87	79.43	11.13	1.12
	3	71.88	80.03	11.65	0.91
	4	73.22	80.92	13.01	0.63
WTC (15 bpp)	2	78.64	86.49	5.43	0.80
	3	80.11	87.35	6.47	0.50
	4	—	—	—	—
Cuprite (8 bpp)	0	58.74	69.93	1.14	0.46
	1	58.22	69.17	1.26	0.42
	2	61.06	71.56	1.43	0.17
Indian Pines (8 bpp)	0	60.38	70.22	1.12	0.45
	1	60.97	69.70	1.30	0.35
	2	65.25	72.38	1.53	0.12

La Tabla 5.10 muestra una comparativa de PSNR para diferentes valores del parámetro  $n$  y las diferentes imágenes con las que se ha marcado.

En el caso de la imagen WTC, la construcción de los árboles TSVQ no funciona bien con 8 bpp ya que, como se puede ver en la Figura 5.7, la mayoría de los componentes están por debajo de 10 000, lo que hace que



(a) Original



(b) Marcada

Figura 5.15: Imagen Cuprite reducida a 8 bpp (banda 2).

Tabla 5.11: Componentes de la imagen WTC reducida a 8 bpp.

Banda	1	2	3	4	5	6	7	8
Máx.	72	161	158	113	94	73	56	57
Mín.	6	4	1	1	0	0	0	0
Media	10.76	11.54	8.18	7.33	5.09	4.06	2.21	2.01
Banda	9	10	11	12	13	14	15	16
Máx.	58	53	46	37	47	46	45	35
Mín.	0	0	0	0	0	0	0	0
Media	1.28	1.18	0.85	0.64	0.58	0.86	0.56	0.29

una vez reducida a 8 bpp las diferentes bandas queden con valores muy bajos y con muy poca diferencia entre el valor máximo y el mínimo. Esto se puede comprobar en la Tabla 5.11, en la que para cada una de las bandas se muestran los valores máximo y mínimo de los componentes, así como la media aritmética de todos ellos. Por lo tanto, para la imagen WTC no se presentan resultados aplicando la reducción a 8 bpp.

Se puede observar, en la Tabla 5.10, que el número  $n$  de bits extraídos afecta a la diferencia entre el valor real y el valor marcado, ya que a medida que aumenta el valor de  $n$ , se incrementa el valor con que se construye el árbol TSVQ (el MSB) que sustituye al componente, es decir, que el centroide con el que se sustituyen los componentes que quedan dentro del rango del LSB extraído será mayor. Todos los componentes que se diferencien sólo por un valor comprendido dentro del rango de los LSB extraídos serán sustituidos por un número más alto al ser  $n$  mayor.

La Tabla 5.10 muestra, de izquierda a derecha, la imagen utilizada para realizar el experimento, el número de LSB extraídos, la media del PSNR de las bandas marcadas, el PSNR de toda la imagen, la media aritmética de los componentes modificados y el número de componentes modificados en porcentaje sobre la imagen completa.

Los valores del PSNR para la imagen Cuprite para  $n = 2, 3$  y  $4$  se muestran en las primeras filas de la tabla. Se puede ver que, extrayendo 4 LSB ( $n = 4$ ), el PSNR se incrementa con respecto a 2 y 3, de media, a 67.58 dB pero, en cambio, la media de las diferencias se incrementa de 16.84 (la media para  $n = 2$ ) hasta 17.20 . Esto se debe a que la marca afecta ahora a los bits más altos (*MSB*), del quinto bit en adelante. Con este valor de  $n$  se consigue una mejor robustez contra los ataques de compresión, pero se incrementa la diferencia entre el valor original del componente y el valor modificado y por tanto la imperceptibilidad del sistema de marcado puede verse afectada.

En general el PSNR se incrementa cuando  $n$  se incrementa. La razón de este comportamiento es que el TSVQ modifica menos componentes a medida que se incrementa el valor de  $n$ . Por ejemplo, para  $n = 2$  el esquema modifica el 0.96 % de todos los componentes, pero con  $n = 4$  modifica sólo el 0.71 %. Incrementando el valor de  $n$  reducimos el número de bits usados para construir los vectores, lo que revierte en árboles TSVQ más pequeños y, de este modo, el TSVQ modifica menos componentes. Sin embargo, valores muy altos de  $n$  provocan que el número de componentes para la construcción de los vectores del TSVQ no sean suficientes y por tanto no se pueda construir el árbol y marcar la imagen. Recordemos que son necesarios un número mínimo de vectores diferentes que depende de la dimensión de éstos.

El valor del PSNR mostrado en la tabla para las imágenes Cuprite e Indian Pines se calcula a partir de la fórmula del PSNR para 14 bits, ya que los dos bits más significativos son cero, dividiendo  $(2^{14} - 1)^2 = 16\,383^2$  por el MSE. Para la imagen WTC se calcula a partir de 15 bits, ya que solo el bit más significativo es cero para todos los componentes. Para las imágenes de 8 bpp el PSNR resultante se obtiene dividiendo  $(2^8 - 1)^2 = 255^2$  por el

MSE. En principio el PSNR debe ser mayor para las imágenes de 14 y 15 bpp que con las de 8 bpp, ya que las modificaciones alteran a los bits menos significativos.

Para las imágenes de 8 bpp, los valores de  $n$  son 0, 1 y 2. Valores más altos no permiten construir el árbol de compresión TSVQ, ya que quedan pocos bits para representar el valor del componente. Éste es el caso de la imagen WTC reducida a 8 bpp que no permite construir el árbol TSVQ ni siquiera para  $n = 0$ .

Como era de esperar, el PSNR de las imágenes de 8 bpp es un poco menor que las de 14 bpp. Con 8 bpp el PSNR está alrededor de 60 dB para las bandas marcadas y 70 dB si se tiene en cuenta toda la imagen. En cambio, para las imágenes con 14 bpp el PSNR aumenta hasta 70 dB para las bandas marcadas y 80 dB en el caso de la imagen entera. Cabe destacar que los resultados son bastante parecidos entre las imágenes del mismo tipo, ya que los valores retornados por el sensor son muy parecidos y el sistema se comporta de forma muy similar para todas las firmas espectrales.

Para las imágenes de 8 bpp, debido a la menor información que se dispone ahora, hay que realizar más intentos de marcado de la imagen de 8 bpp ya que no todos los bloques generan el árbol TSVQ con cualquier valor de la entropía que le asigna el esquema a partir de la secuencia de números pseudoaleatorios.

#### 5.2.2.2. Ataques de copia y reemplazo

Al igual que en el caso de los anteriores métodos y para las imágenes de 14 bpp, el método detecta todas las modificaciones de copia y reemplazo de

Tabla 5.12: Robustez del esquema contra ataques de compresión JPEG2000 para la imagen Cuprite de 14 bpp.

$n$ bits	Compr. (bpp)	Resistencia marca	PSNR (dB)	DM	Máx. difer.	CM (%)
2	8	Sí	85.08	1.19	5	53.75
	7	No	81.03	1.55	10	71.66
	6	No	76.35	2.37	16	83.69
3	8	Sí	85.09	1.19	5	53.69
	7	Sí	81.04	1.55	9	71.61
	6	No	76.36	2.37	17	83.69
	5	No	70.99	4.11	40	91.19
4	8	Sí	85.10	1.19	6	53.60
	7	Sí	81.04	1.55	10	71.59
	6	Sí	76.37	2.36	17	83.69
	5	No	70.98	4.11	32	91.15

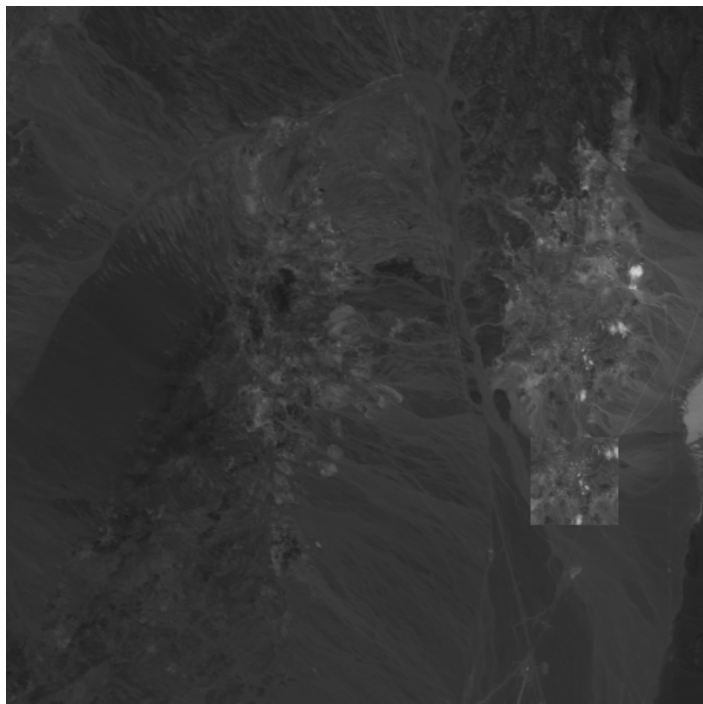
una zona de la imagen sobre otra. Los resultados obtenidos son idénticos a los mostrados en las Figuras 5.10 y 5.11 con los mismos resultados.

La Figura 5.16 muestra la imagen Cuprite original reducida a 8 bpp (5.16(a)) y la misma una vez ha sido identificada la zona modificada (5.16(b)) con el método descrito y  $n = 2$ .

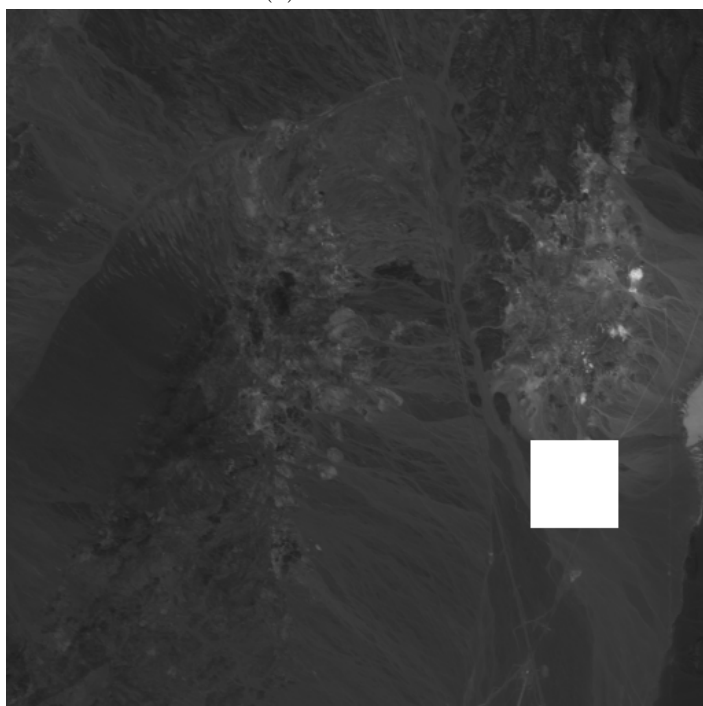
### 5.2.2.3. Ataques de compresión

Al igual que el método anterior, se han realizado experimentos de compresión y descompresión JPEG2000 de la imagen Cuprite de 14 bpp marcada con el software Kakadu (Taubman, 2007), para cada una de las 16 bandas marcadas por separado. La Tabla 5.12 muestra, para los diferentes valores del parámetro  $n$ , los valores de compresión a que se puede llegar comprimiendo y descomprimiendo y en qué casos se mantiene o elimina la marca de las bandas. La tabla muestra la media y el valor máximo de la diferencia de





(a) Modificación



(b) Detección de la modificación

Figura 5.16: Imagen Cuprite de 8 bpp (banda 2).

valores entre el valor marcado y el comprimido de los componentes, así como el PSNR del ataque. El parámetro  $n$  determina la robustez del sistema, ya que desviaciones del valor del componente de aproximadamente  $2^n - 1$  serán permitidas por el esquema. Gracias a la cuantización de la entropía en el proceso de marcado y detección para cada uno de los bloques, con valores ligeramente superiores a este límite todavía se puede recuperar la marca. Así, para  $n = 2$  la compresión sólo se permite hasta 8 bpp (la imagen original tiene 14 bpp). Para  $n = 3$  y  $n = 4$ , se permite una compresión de hasta 7 y 6 bpp respectivamente, con una desviación de 9 y 17 en los valores de los componentes, para cada uno. Este ejemplo demuestra claramente cómo el parámetro  $n$  del esquema de marcado y el paso de cuantización de la entropía para cada bloque pueden ser usados para sintonizar la robustez y fragilidad del método.

Se puede observar que el caso que mejor funciona es el de  $n = 4$ , ya que permite más compresión a la imagen original, pero representa una distorsión más elevada de la imagen. El método modifica los MSB, por lo tanto cuanto mayor sea el parámetro  $n$ , mayor es la modificación que realiza éste. Si se escoge un valor menor de  $n$  menor, por ejemplo 2, los centroides se calcularán a partir del tercer bit, y, por tanto, el sistema no será tan robusto ya que las compresiones JPEG2000 afectaran a bits más allá del segundo.

También se han realizado ataques de compresión mediante el algoritmo estándar JPEG, de menor calidad que el JPEG2000 pero mucho más utilizado. JPEG sólo funciona para 8 bpp, ya sea para imágenes en color o en escala de grises. Por tanto, para poder realizar ataques de compresión y comparar los resultados obtenidos con otros métodos de la literatura, se ha optado por utilizar el compresor JPEG con diferentes valores de calidad sobre las imágenes reducidas a 8 bpp.

La Tabla 5.13 muestra los resultados obtenidos para los ataques JPEG sobre la imagen Cuprite reducida a 8 bpp marcada con el algoritmo de extracción de los LSB. Se puede observar que los resultados de PSNR, diferencia media y diferencia máxima varían muy poco entre los diferentes valores de  $n$ . Para la imagen Cuprite de 8 bpp no se puede aplicar un valor de  $n$  superior a 3, ya que no se puede construir el árbol de TSVQ con tan poco rango de datos (al quedarnos sólo con 4 bits por cada componente de la banda). Para obtener una buena robustez del método, el parámetro  $n$  se denota como un elemento principal, ya que hace que el sistema sea o no robusto para un determinado ataque. Otro parámetro que influye en la robustez es el paso de cuantización de la entropía, que permite una cierta tolerancia en la detección de la marca.

### 5.2.3. Método de marcado sobre los coeficientes de la DWT

El tercer método desarrollado (Serra-Ruiz and Megías, 2010a) se basa en la utilización de la transformada *wavelet* discreta para marcar los coeficientes de ésta en lugar de marcar sobre los valores reales de los componentes (la firma espectral).

Por lo tanto, se aplica la DWT y se guardan todas las bandas de coeficientes resultantes para realizar el marcado de las bandas seleccionas y posteriormente la construcción de la imagen marcada. El método descrito en la Sección 4.2.4 propone marcar las bajas frecuencias, es decir, la banda LL de la transformada discreta *wavelet*, dejando las bandas más altas, (HL, LH

Tabla 5.13: Robustez del esquema LSB contra ataques de compresión JPEG para la imagen Cuprite de 8 bpp.

$n$ bits	Compr. (calidad)	Resistencia marca	PSNR (dB)	DM	Diferencia máxima	CM (%)
1	100	Sí	58.91	1	1	8.35
	99	Sí	54.72	1.01	2	21.85
	98	No	50.58	1.08	4	46.03
	97	No	48.41	1.21	6	56.69
	96	No	47.09	1.32	7	62.04
	95	No	46.05	1.42	8	65.67
	94	No	45.38	1.49	10	67.80
	93	No	44.82	1.56	11	69.43
	92	No	44.37	1.62	13	70.88
	91	No	44.01	1.66	13	71.67
	90	No	43.69	1.71	14	72.53
	85	No	42.44	1.89	18	75.69
	80	No	41.60	2.04	21	77.67
75	No	40.94	2.16	26	79.16	
2	100	Sí	58.90	1	1	8.36
	99	Sí	54.73	1.01	2	21.83
	98	Sí	50.58	1.08	4	46.07
	97	No	48.41	1.21	6	56.73
	96	No	47.08	1.32	7	62.06
	95	No	46.05	1.42	8	65.69
	94	No	45.37	1.49	10	67.8
	93	No	44.82	1.56	11	69.45
	92	No	44.35	1.62	13	70.90
	91	No	44.01	1.66	13	71.70
	90	No	43.69	1.71	14	72.57
	85	No	42.44	1.89	18	75.81
	80	No	41.59	2.04	21	77.75
75	No	40.93	2.16	26	79.22	
3	100	Sí	58.92	1	2	8.34
	99	Sí	54.73	1.01	2	21.80
	98	Sí	50.58	1.08	4	46.06
	97	Sí	48.40	1.21	6	56.73
	96	Sí	47.07	1.32	7	62.05
	95	Sí	46.04	1.42	9	65.69
	94	No	45.37	1.49	10	67.83
	93	No	44.81	1.56	11	69.45
	92	No	44.34	1.62	13	70.92
	91	No	44.01	1.66	13	71.70
	90	No	43.68	1.71	13	72.59
	85	No	42.43	1.89	17	75.78
	80	No	41.59	2.04	21	77.77
75	No	40.93	2.16	25	79.20	

y HH) intactas. El sistema de marcado se ha adaptado para poder funcionar acorde con las características de estas bandas, ya que si se aplica la DWT a un bloque de  $64 \times 64$  píxeles, cada una de las bandas (LL, HL, LH y HH) tiene ahora  $32 \times 32$  coeficientes.

Los valores que se usan ahora ya no representan valores reales de componentes, sino coeficientes de la DWT, aunque el sistema de marcado funcionará de la misma forma. Posteriormente se aplica la transformada inversa a partir de los coeficientes modificados (LL') y los valores de las bandas más altas (LH, HL y HH) intactos de la transformada *wavelet* del bloque original. Cabe destacar que las bandas de la transformada *wavelet* no tienen nada que ver con las bandas de las imágenes hiperespectrales. En este caso, las “bandas” representan a las cuatro regiones en las que se dividen los coeficientes de la transformada *wavelet* (LL, LH, HL y HH).

Tabla 5.14: PSNR de la imagen Cuprite marcada para cada banda.

#	PSNR (dB)	CM (%)	DM	#	PSNR (dB)	CM (%)	DM
1	62.36	28.93	16.70	9	59.32	29.17	23.87
2	62.94	28.76	14.73	10	64.24	28.79	13.35
3	63.31	28.88	14.71	11	64.10	28.82	13.73
4	63.40	28.79	14.67	12	62.43	28.96	16.98
5	63.81	28.65	13.83	13	63.07	28.85	15.40
6	64.01	28.78	13.50	14	61.60	28.93	17.25
7	64.06	28.76	13.46	15	62.22	28.99	16.67
8	63.70	28.78	14.13	16	60.27	29.14	21.27

La Tabla 5.14 muestra los valores obtenidos con la aplicación de la transformada Daubechies 1 para cada una de las bandas. Se muestra el PSNR, el número de componentes modificados y la distancia media con que se ven afectados para el caso de la imagen Cuprite (de 14 bpp). El resto de imágenes proporcionan resultados ligeramente mejores.

Tabla 5.15: PSNR de la imagen Cuprite marcada para DB1, DB3 y DB5.

	PSNR (dB)	CM (%)	DM
DB1	74.02	2.06	15.91
DB3	71.34	4.21	12.68
DB5	72.50	4.62	10.67

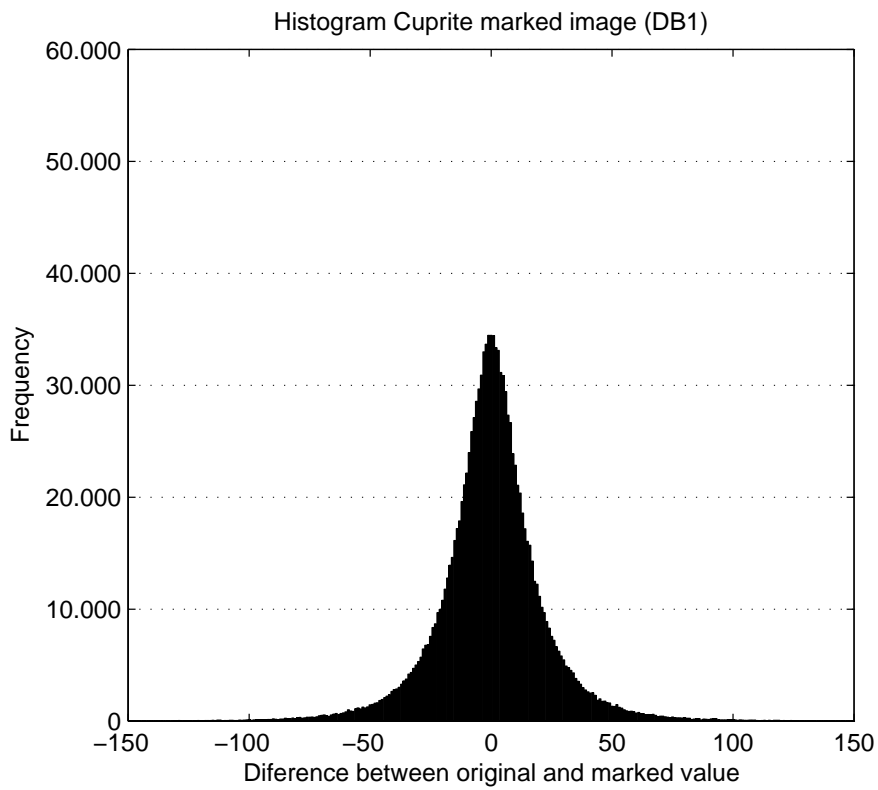
En la Tabla 5.15 se puede ver la comparación entre diferentes variantes de la transformada *wavelet*, concretamente tres valores de la familia de transformadas Daubechies (1, 3 y 5). A partir del valor 5, la construcción del árbol TSVQ tiene problemas y no se pueden construir los árboles para todos los bloques de la imagen.

La Figura 5.17 muestra el histograma de la imagen Cuprite para los tres parámetros descritos (DB1, DB3 y DB5). Para la DB5 los valores de las modificaciones están más concentrados.

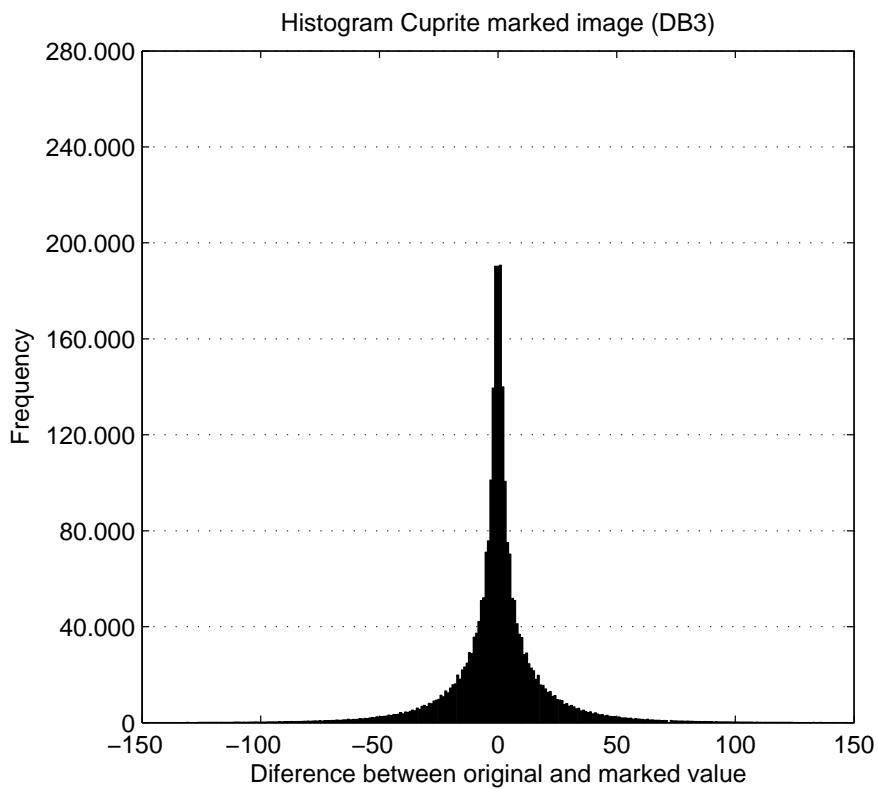
Se puede observar, a partir de la Tablas 5.14, y 5.9 que los resultados de este método son muy parecidos en cuanto a PSNR a los obtenidos en el apartado anterior (algo inferiores), pero este esquema es mucho más robusto frente a ataques de compresión JPEG2000.

### 5.2.3.1. Ataques de copia y reemplazo

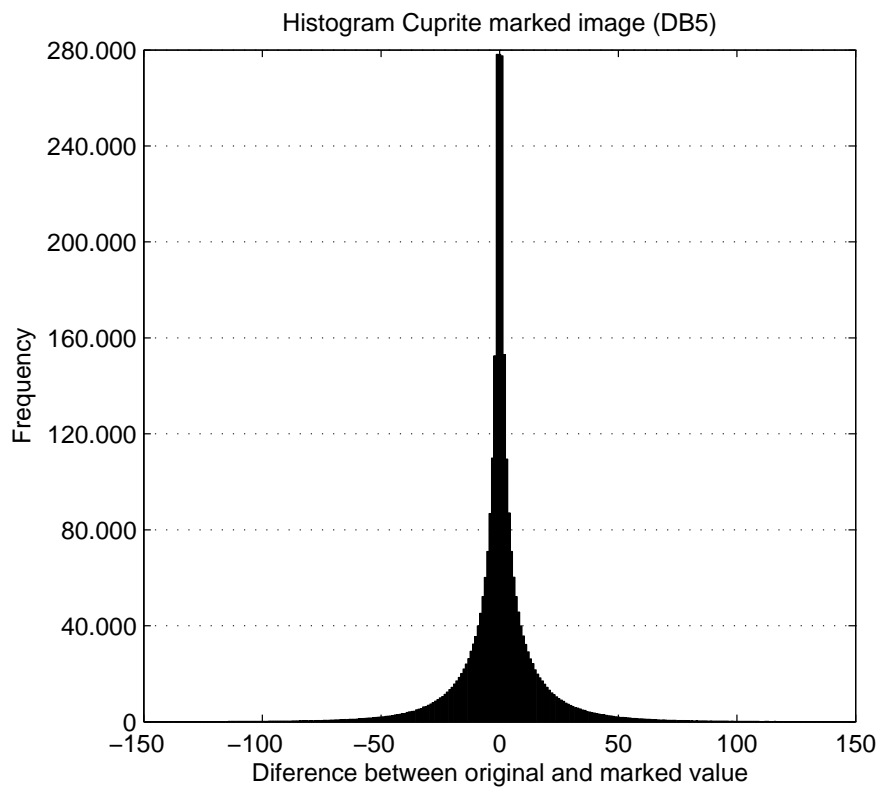
Al igual que en los dos métodos anteriores, éste detecta correctamente todos los ataques de copia y reemplazo. El esquema detecta los cambios realizados en la imagen ya sea con partes de la misma imagen o por la variación de la firma espectral. Los resultados son idénticos a los mostrados en la Figura 5.11 y las diferencias visuales entre las dos figuras no son apreciables.



(a) Daubechies 1



(b) Daubechies 3



(c) Daubechies 5

Figura 5.17: Histograma de las diferencias entre valores originales y marcados con la transformada *wavelet*.



## 5.2.3.2. Ataques de compresión

La Tabla 5.16 resume los ataques de compresión JPEG2000 contra el método. Dicha tabla muestra una mejora en cuanto a la compresión que se puede alcanzar sin eliminar la marca de la imagen. Se puede comprimir la imagen hasta 5 bpp sin perder la marca (o propiedad) de cada uno de los bloques.

Tabla 5.16: Resultados de los ataques de compresión con JPEG2000 para la imagen Cuprite de 14 bpp.

DWT	Compr. (bpp)	Resistencia marca	PSNR (dB)	DM	Diferencia máxima	CM (%)
DB1	8	Sí	85.09	1.19	5	53.80
	7	Sí	80.97	1.56	9	71.89
	6	Sí	76.30	2.38	16	83.80
	5	Sí	70.91	4.14	29	91.25
DB3	8	Sí	85.05	1.19	6	53.97
	7	Sí	80.66	1.59	9	73.16
	6	Sí	75.95	2.45	17	84.62
	5	Sí	70.53	4.27	40	91.75
DB5	8	Sí	85.10	1.19	5	53.71
	7	Sí	80.70	1.58	10	73.02
	6	Sí	75.98	2.44	15	84.56
	5	Sí	70.57	4.25	40	91.72

Al igual que para el método basado en la extracción de los LSB, también se ha realizado un estudio sobre la robustez de este método contra ataques de compresión JPEG. La Tabla 5.17 muestra los resultados de estos ataques para la imagen Cuprite reducida a 8 bpp. Se puede apreciar como los resultados obtenidos son ligeramente peores en cuanto a PSNR y número de componentes modificados, pero mejores teniendo en cuenta que este método es mucho más robusto, ya que detecta la marca hasta compresiones más elevadas, alcanzando el 90% de calidad. La diferencia entre las diferentes variantes de la transformada *wavelet* es mínima, ya que apenas varía el PSNR medio de

las bandas modificadas, ni la distancia media entre los valores marcados y atacados. No obstante, sí se puede apreciar una ligera ventaja de la variante *Daubechies* 1 respecto a las otras dos teniendo en cuenta la robustez de la marca, permitiendo alcanzar el 90 % de calidad mientras que las variantes *Daubechies* 3 y 5 lo hacen hasta el 91 % y 92 % respectivamente.

#### 5.2.4. Método de marcado sobre DWT y aplicación de regiones de tamaño variable

Del método anterior cabe destacar el buen comportamiento frente a compresiones y, aunque se obtienen valores ligeramente peores de PSNR, se mejora en cuanto a la distancia media que se modifican los componentes. Aún así, se pueden encontrar dos problemas que se pueden solucionar a la vez con la misma estrategia. El problema surge en la división en bloques de exactamente el mismo tamaño ( $64 \times 64$  píxeles).

Las regiones de las imágenes no serán nunca (sería muy raro) apropiadas a la realidad, por lo que dividir en regiones fijas de  $64 \times 64$  producirá que partes de la imagen que sean muy similares estén ubicadas en regiones o bloques diferentes y, a la vez, que zonas muy diferentes estén ubicadas en un mismo bloque.

También existe el problema de que, al ser divisiones exactas de la imagen, un ataque muy sencillo consistiría en el cálculo de la entropía de un bloque para sustituirlo por otro (ataque de copia y reemplazo) y forzar la entropía del nuevo bloque para que sea idéntica a la que tenía el anterior. La sustitución de estos bloques no se detectaría como alteración en la imagen marcada.

Tabla 5.17: Robustez del esquema DWT frente a ataques de compresión JPEG para la imagen Cuprite de 8 bpp.

DWT	Compr. (calidad)	Resistencia marca	PSNR (dB)	DM	Máx. difer.	CM (%)
DB1	100	Sí	58.90	1	1	8.35
	99	Sí	54.73	1.01	2	21.84
	98	Sí	50.57	1.08	4	46.15
	97	Sí	48.38	1.21	6	56.89
	96	Sí	47.05	1.32	7	63.31
	95	Sí	46.01	1.42	9	65.98
	94	Sí	45.33	1.49	10	68.10
	93	Sí	44.78	1.56	11	69.69
	92	Sí	44.30	1.62	13	71.15
	91	Sí	43.97	1.66	13	71.96
	90	Sí	43.64	1.71	13	72.82
	85	No	42.39	1.90	18	76.03
	80	No	41.55	2.04	21	77.99
75	No	40.89	2.17	26	79.42	
DB3	100	Sí	58.91	1	1	8.36
	99	Sí	54.69	1.01	2	21.97
	98	Sí	50.43	1.08	4	47.05
	97	Sí	48.16	1.22	6	58.21
	96	Sí	46.79	1.34	7	63.69
	95	Sí	45.73	1.45	8	67.30
	94	Sí	45.04	1.53	10	66.47
	93	Sí	44.48	1.59	11	71.07
	92	Sí	44.00	1.66	13	72.43
	91	Sí	43.66	1.71	13	73.24
	90	No	43.33	1.75	14	74.10
	85	No	42.09	1.95	18	77.15
	80	No	41.25	2.10	21	79.03
75	No	40.60	2.23	26	80.37	
DB5	100	Sí	58.91	1	2	8.35
	99	Sí	54.65	1.01	2	22.17
	98	Sí	50.23	1.09	4	48.35
	97	Sí	47.87	1.24	6	59.89
	96	Sí	46.50	1.37	7	65.36
	95	Sí	45.46	1.48	9	68.91
	94	Sí	44.80	1.56	9	70.85
	93	Sí	44.25	1.63	12	72.36
	92	Sí	43.81	1.69	13	73.61
	91	No	43.49	1.73	13	74.30
	90	No	43.18	1.78	13	74.99
	85	No	42.07	1.21	19	77.44
	80	No	41.37	2.08	24	78.87
75	No	40.83	2.18	23	79.86	

Teniendo en cuenta estos dos problemas, se introduce un cambio en los métodos propuestos y se propone el método basado en la transformada *wavelet* y la elección manual de bloques de tamaño variable (Serra-Ruiz and Megías, 2010b) a marcar. La división de los bloques de las imágenes no se realizará por regiones exactas de  $64 \times 64$  píxeles, sino que se agruparán los diferentes valores de las zonas de la imagen para dar lugar a bloques más uniformes y de diferentes tamaños. Con esto, se consigue que las firmas espectrales que son parecidas (que denoten el mismo elemento predominante) estén en el mismo bloque de marcado y otras firmas espectrales diferentes se agrupen para formar otro bloque diferente. Los bloques propuestos tendrán tamaños diferentes:  $64 \times 64$ ,  $32 \times 96$ ,  $96 \times 64$ ,  $128 \times 128$ , ...

Los bloques se van a seleccionar con medidas ya prefijadas para simplificar los cálculos, aunque podrían ser completamente diferentes. Se han escogido tamaños en que las dimensiones sean múltiplo de 32, así tendremos bloques de forma rectangular con lados comprendidos entre 32, 64, 96 y 128 píxeles. Se ha limitado el esquema a estas cuatro posibilidades, como se ha descrito, para simplificar el proceso, pero se podrían asignar también regiones irregulares, lo que dificultaría a posteriori el tratamiento de estos bloques y la detección de las modificaciones, a la vez que complicaría significativamente los ataques de copia y reemplazo.

La Figura 5.18 muestra un ejemplo de la selección de las regiones escogidas manualmente para la imagen Cuprite. Se puede observar que las regiones son más grandes que antes, ahora hay regiones desde  $64 \times 64$  hasta  $128 \times 128$  píxeles y, por lo tanto, si en los métodos anteriores se marcaba en  $8 \times 8 = 64$  bloques uniformes, ahora se hace en 37 bloques diferentes que agrupan muchas más firmas espectrales.

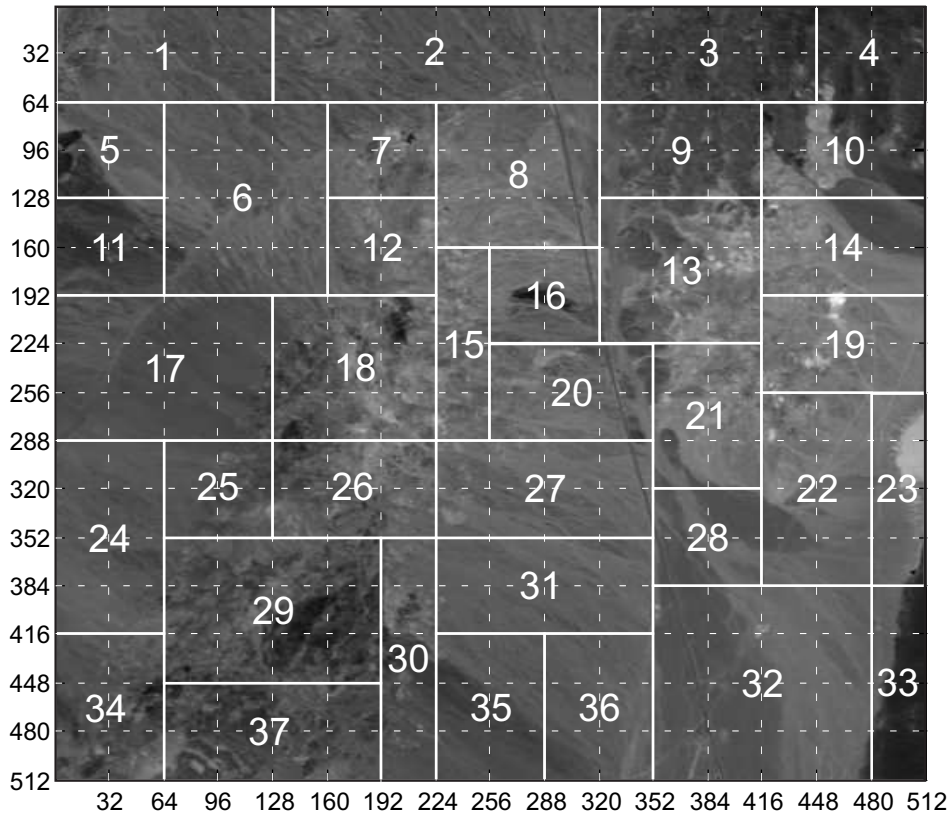
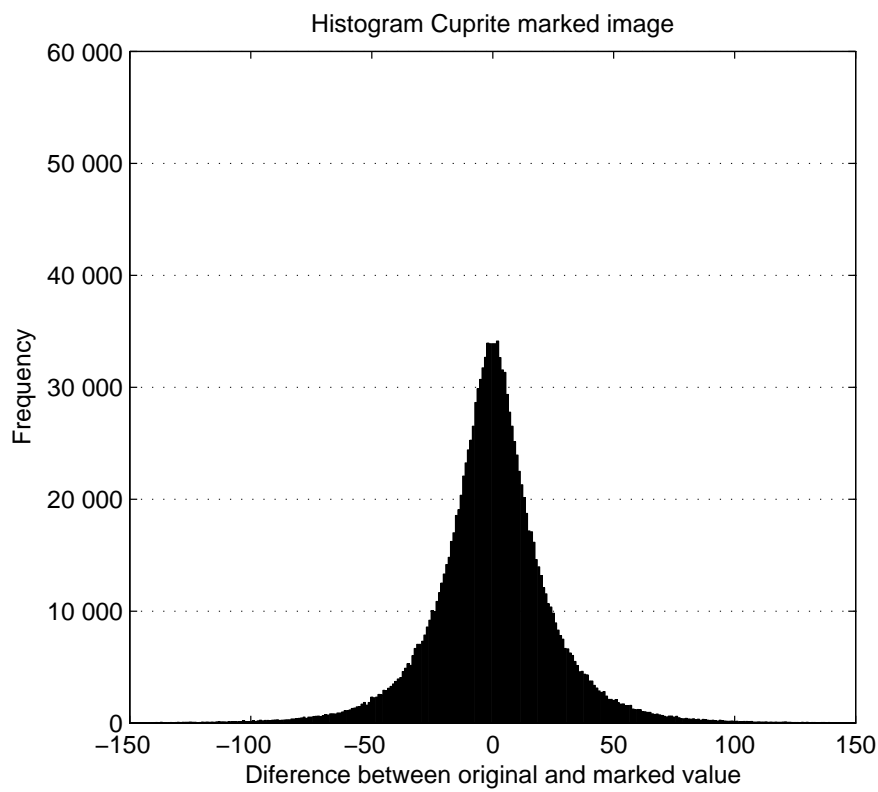


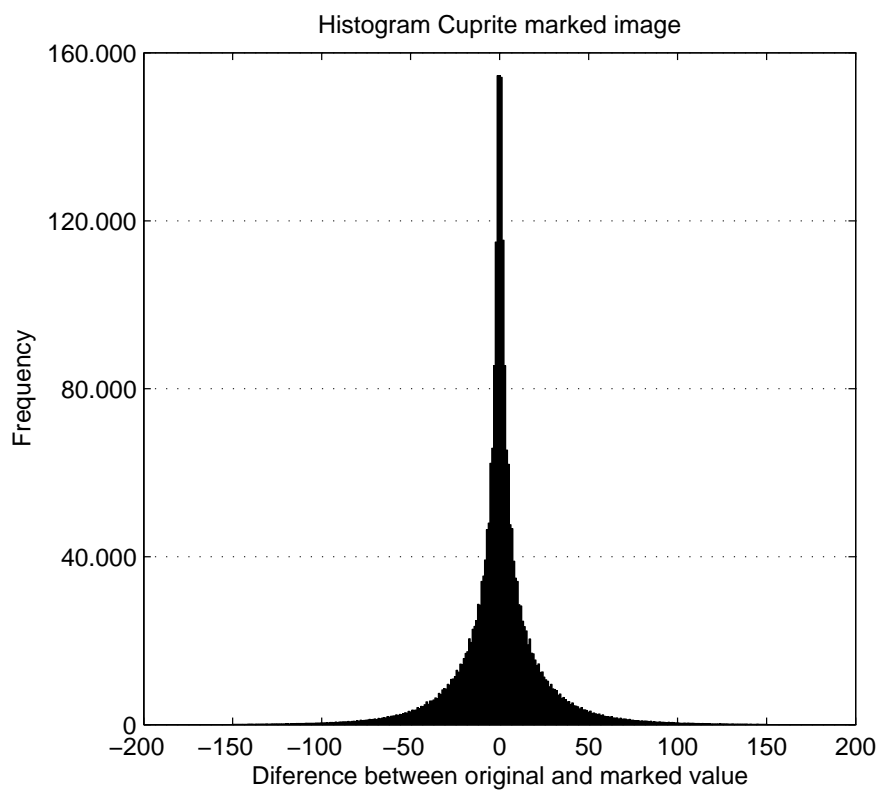
Figura 5.18: Proceso de división de los bloques.

La Figura 5.19 muestra el histograma de las diferencias entre los componentes originales de la firma espectral y los componentes marcados, aplicado la transformada discreta *wavelet* DB1 (a) y DB3 (b). Como antes, el histograma muestra cómo en el caso de la DB1 se modifican muchos menos componentes, aunque la diferencia es ligeramente mayor.

El método de marcado introduce muy poca variación en el comportamiento de la firma espectral, como ya se ha comentado en los métodos anteriores. Para este método, la Figura 5.20 muestra la diferencia entre las dos, donde la firma espectral marcada nuevamente se ha desplazado 200 unidades para poder apreciar la casi nula diferencia entre las dos curvas. En gris se puede ver cómo la firma espectral marcada (en negro la original) varía muy poco y la forma de ésta se mantiene a lo largo de todas las bandas, lo que hace que



(a) Daubechies 1



(b) Daubechies 3

Figura 5.19: Histograma de las diferencias entre los valores originales y marcados para la imagen Cuprite aplicando la transformada DWT.

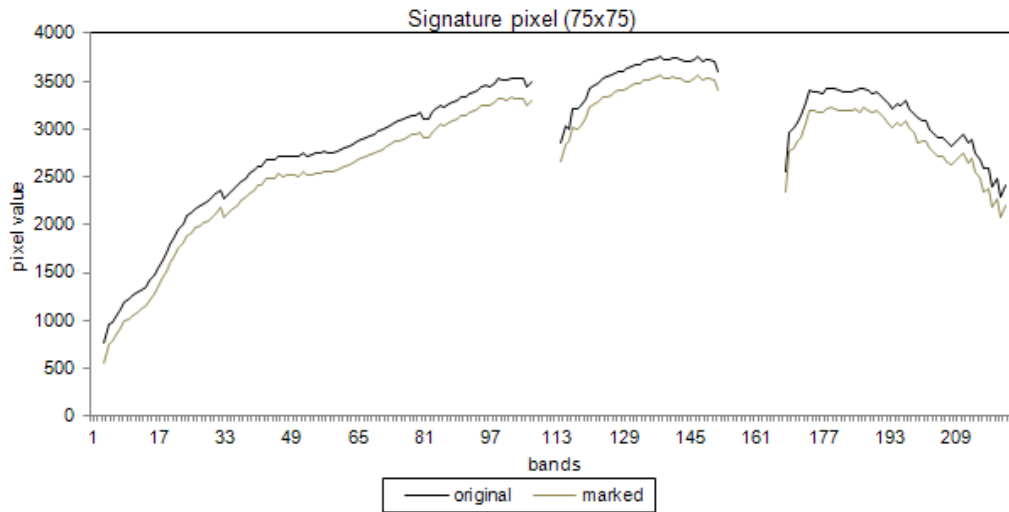


Figura 5.20: Diferencia entre las firma espectrales, original y marcada.

los métodos de clasificación no vean alterados sus resultados.

Tabla 5.18: Imperceptibilidad del esquema de marcado para las bandas marcadas (usando la transformada Daubechies 1 entera).

#	PSNR (dB)	CM (%)	DM	#	PSNR (dB)	CM (%)	DM
1	60.97	32.72	18.45	9	57.94	32.94	24.80
2	61.30	32.48	16.75	10	62.78	32.50	15.11
3	61.98	32.53	16.03	11	62.70	32.54	15.36
4	62.01	32.61	16.47	12	61.08	32.70	18.79
5	62.14	32.44	16.04	13	61.62	32.68	17.27
6	62.18	32.52	16.06	14	59.86	32.69	19.89
7	62.53	32.50	15.07	15	60.65	32.75	18.96
8	62.08	32.54	15.84	16	58.87	32.87	23.95

La Tabla 5.18 muestra los valores del PSNR, los componentes modificados y las diferencias medias para cada una de las bandas modificadas, obtenidos a partir de aplicar el método de marcado usando la transformada discreta *wavelet* Daubechies 1. Como se puede observar los valores obtenidos con este método son muy parecidos al método anterior. La ventaja de este método radica en la mejor seguridad que ofrecen las regiones de tamaño variable y no en cuanto a la calidad de la imagen marcada se refiere.

Si comparamos los resultados de imperceptibilidad con respecto a las posibles transformadas, vemos, en la Tabla 5.19, que se obtienen mejores resultados para la Daubechies 5, si tenemos en cuenta el PSNR, aunque en este caso se modifican muchos más componentes que con la transformada Daubechies 1.

Tabla 5.19: PSNR de la imagen Cuprite marcada para DB1, DB3 y DB5.

	PSNR (dB)	CM (%)	DM
DB1	67.13	2.38	23.56
DB3	69.77	4.07	15.73
DB5	72.27	4.58	11.11

#### 5.2.4.1. Capacidad

Para los métodos en que la división por bloques es variable, la capacidad será menor que en los métodos anteriores, ya que se obtendrá un número menor de bloques para dividir la imagen. Por lo tanto no se llegará a disponer de 64 bloques y 6 bits de información por bloque.

La capacidad en este caso variará en función de los bloques resultantes. Como ejemplo para el caso de 37 bloques tenemos  $h = \frac{1}{37}$ , de manera que:

$$37 \text{ valores de cuantización} \Rightarrow 2^6 \Rightarrow 6 \text{ bits} \times 37 \text{ bloques} = 222 \text{ bits.}$$



#### 5.2.4.2. Ataques de copia y reemplazo

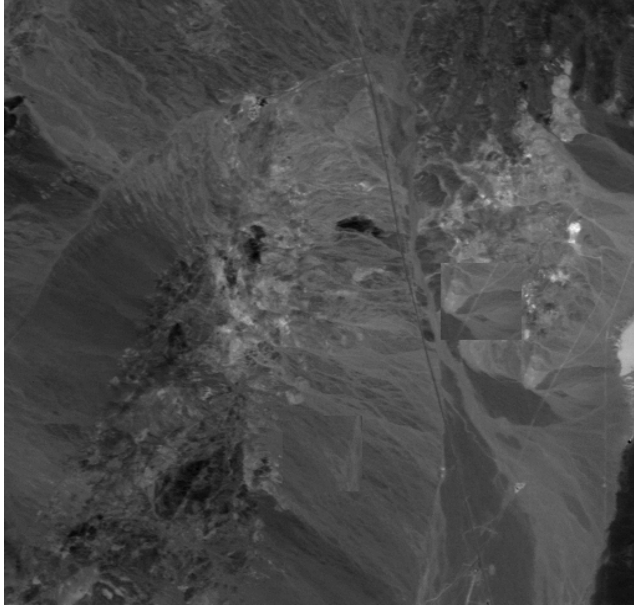
Para determinar la eficacia de este método, se aplica una serie de ataques de sustitución. Se ha copiado una parte de la imagen y se ha sustituido en otra parte de la misma. La Figura 5.21(a) muestra la modificación de la imagen marcada, un ataque de copia de una parte sobre otra y la Figura 5.21(b) muestra la detección de la modificación. Como se puede apreciar, la detección identifica el bloque donde se ha producido la manipulación (correspondiente al bloque etiquetado como 21 en la Figura 5.18).

#### 5.2.4.3. Ataques de compresión

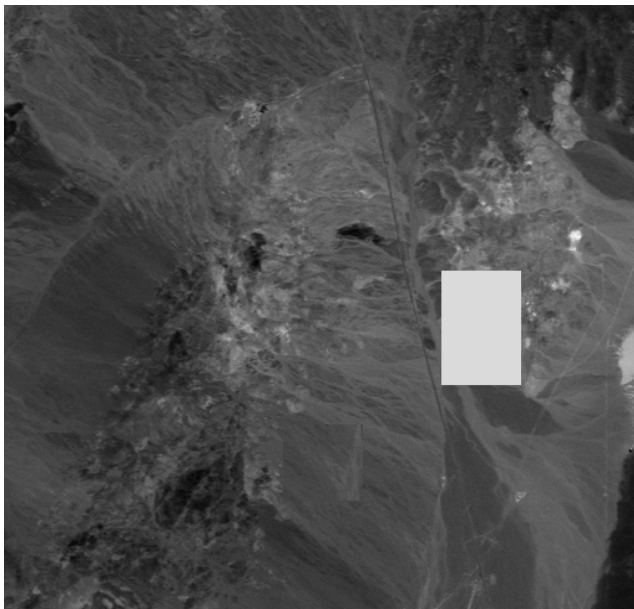
Por último, se muestra la robustez de este método aplicando nuevamente una serie de ataques de compresión JPEG2000 que determinarán si el método resiste o no a las compresiones de las imágenes.

La Tabla 5.20 muestra los resultados obtenidos para la compresión de las 16 bandas de la imagen Cuprite aplicándola a las diferentes transformadas que se han usado.

Como se puede apreciar, los resultados son muy parecidos a los obtenidos en la Sección 5.2.3.2 anterior, en el que las regiones son completamente cuadradas y todas tienen un tamaño idéntico de  $64 \times 64$  píxeles. Por lo tanto, este método aporta mayor seguridad contra ataques pero no obtiene mejores resultados en cuanto a la calidad de las imágenes marcadas o la robustez frente a ataques de compresión.



(a) Modificación



(b) Detección

Figura 5.21: Imagen modificada (a) y detectada (b) (para la banda 9).

Tabla 5.20: Resultados de los ataques de compresión con JPEG2000 para la imagen Cuprite.

DWT	Compr. (bpp)	Resistencia marca	PSNR (dB)	DM	Diferencia máxima	CM (%)
DB1	8	Sí	85.06	1.19	7	53.92
	7	Sí	80.96	1.56	9	71.89
	6	Sí	76.28	2.39	18	83.84
	5	Sí	70.89	4.14	31	91.27
DB3	8	Sí	85.02	1.12	5	50.75
	7	Sí	80.92	1.57	9	72.01
	6	Sí	76.23	2.40	18	83.90
	5	Sí	70.84	4.17	36	91.32
DB5	8	Sí	85.08	1.12	5	50.45
	7	Sí	80.99	1.56	10	71.77
	6	Sí	76.30	2.38	17	83.78
	5	Sí	70.92	4.14	40	91.25

### 5.2.5. Método de marcado sobre DWT y automatización de la selección de las regiones a marcar

A partir del método anterior, descrito en la Sección 4.2.5, nace la necesidad de automatizar el proceso de selección de los bloques, ya que éstos se seleccionaban manualmente a partir de la parte visible de cada banda.

Se parte del método anterior para incluir un paso previo donde se obtendrán los bloques a marcar. Para eso se parte de los valores de las bandas y se crean pequeños grupos de forma cuadrada de  $32 \times 32$  píxeles, los cuales se irán agrupando en función de la semejanza que tengan con los bloques contiguos. Tal y como se explica en la Sección 4.2.5, para todas las bandas y para cada uno de los bloques de  $32 \times 32$  píxeles, se calculan los posibles bloques a formar.

La Figura 5.22 muestra el resultado de este proceso de agrupación. Como se puede ver, el proceso selecciona grandes áreas donde la variación de las

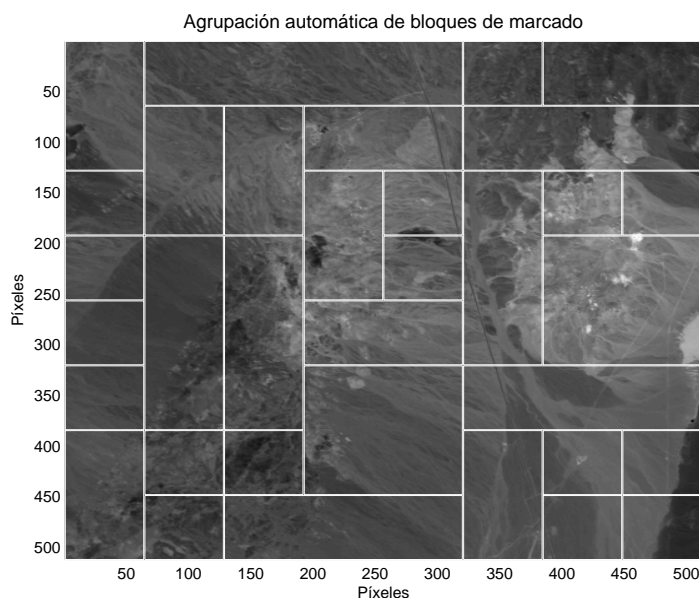


Figura 5.22: Bloques resultantes de la automatización de la selección de las regiones para la imagen Cuprite.

firmas espectrales es mínima y zonas muy pequeñas que no se pueden agrupar con la contigua por representar una firma espectral diferente.

Una vez creados los bloques automáticamente, el método continúa igual que en el caso del esquema presentado en la Sección 5.2.4. Como era de esperar, los resultados obtenidos del PSNR y diferencias medias de este método son casi idénticos al método anterior, donde el proceso de selección de los bloques se realizaba de forma manual. Aunque el resultado final es bastante parecido, se aprecia una ligera mejora respecto al PSNR, ya que la selección de las zonas es mucho más precisa que en el caso del método anterior. Así pues, la Tabla 5.21 muestra los resultados obtenidos para este método para la imagen Cuprite. Se puede observar que, en los tres casos, el PSNR es un poco mejor, ya que se modifican menos componentes y la diferencia media con que se modifican es menor con respecto a los resultados obtenidos con el método anterior.

Tabla 5.21: PSNR de la imagen Cuprite marcada para DB1, DB3 y DB5.

	PSNR (dB)	CM (%)	DM
DB1	74.52	1.79	15.98
DB3	71.93	3.74	12.42
DB5	74.55	4.15	8.66

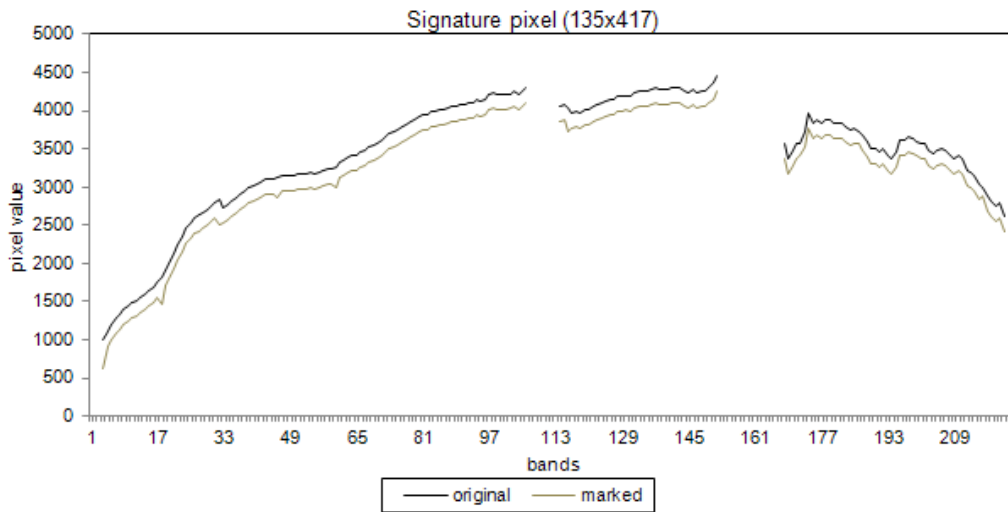
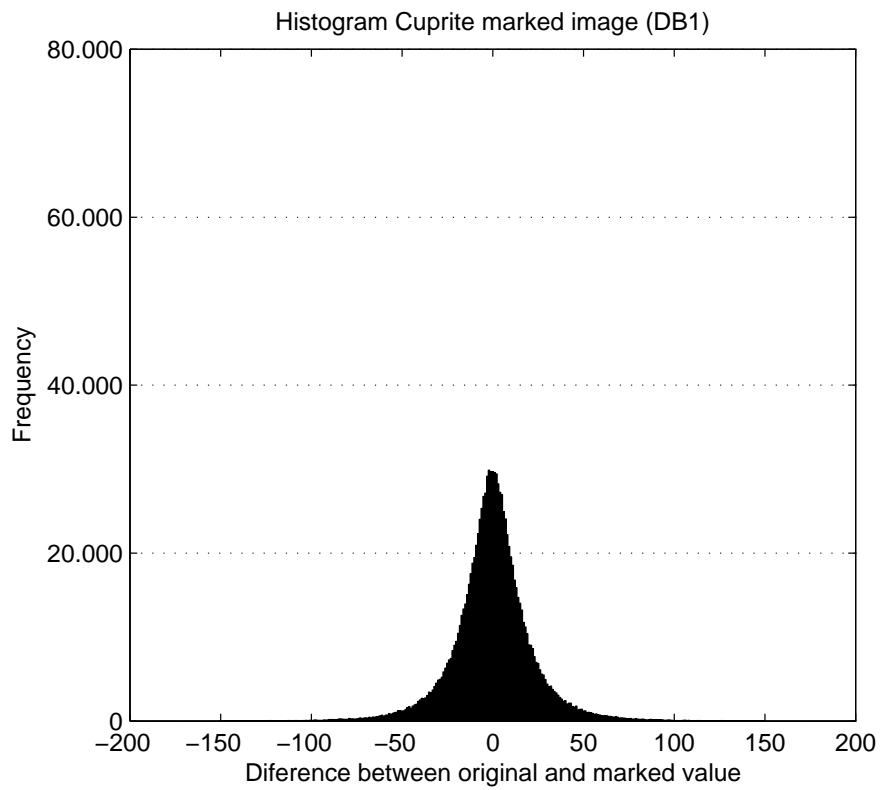


Figura 5.23: Diferencia entre las firma espectrales, original y marcada.

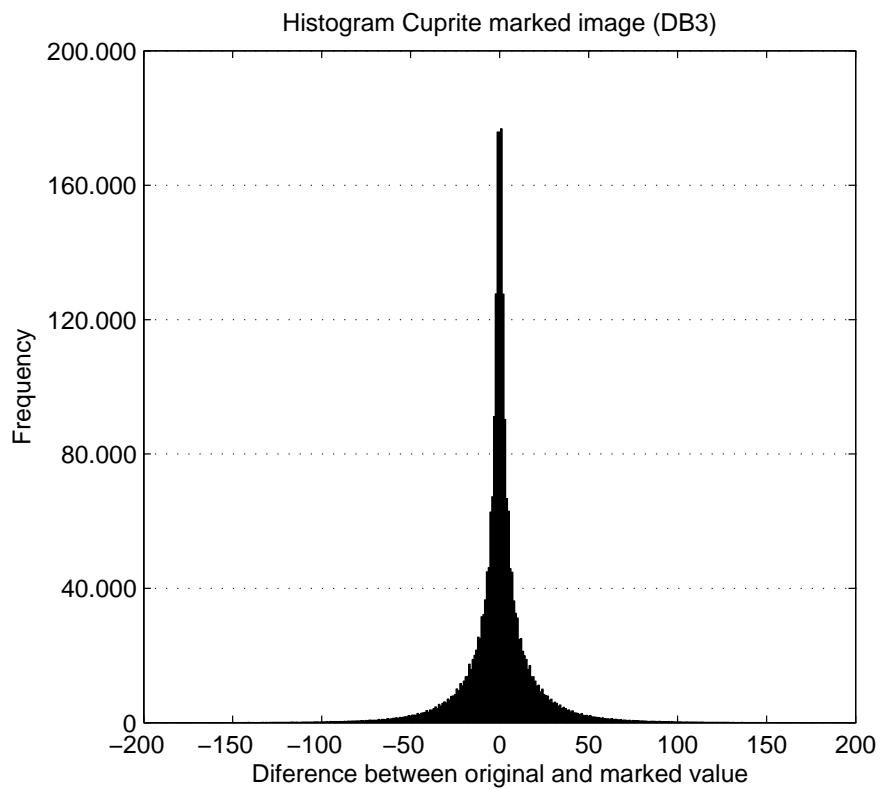
Para el caso de la imagen reducida a 8 bpp y aplicando este mismo método se obtiene un PSNR de la imagen de 224 bandas de 65.21 dB, modificando sólo 1.14% de los componentes con una distancia media de 1.19 de los 255 posibles para las imágenes de 1 byte.

La Figura 5.23 muestra la firma espectral del píxel ( $135 \times 417$ ) con los valores marcados desplazados 200 unidades. Como se puede ver la diferencia es mínima y casi inapreciable.

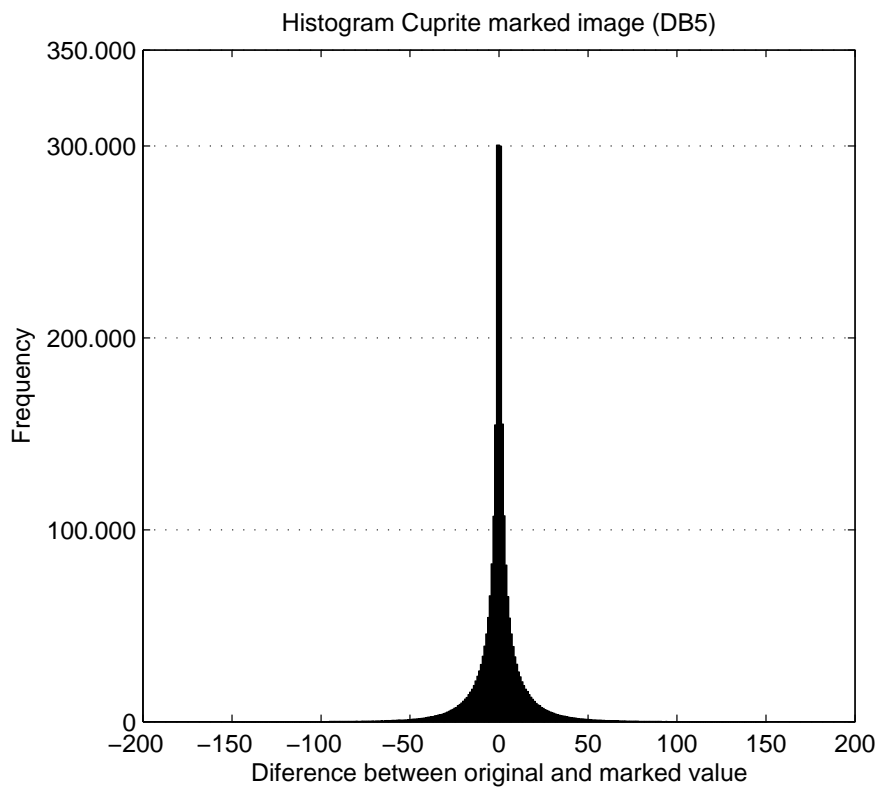
Finalmente, en la Figura 5.24 se muestran los histogramas de las diferencias entre los valores originales y los marcados. Se puede ver que los componentes varían menos con la transformada DB1, aunque con DB3 y DB5 se modifica menos cada componente, pero se cambian muchos más.



(a) Daubechies 1



(b) Daubechies 3



(c) Daubechies 5

Figura 5.24: Histograma de las diferencias entre valores reales y marcados para la imagen Cuprite.

Tabla 5.22: Resultados de los ataques de compresión con JPEG2000 para la imagen Cuprite de 14 bpp.

DWT	Compr. (bpp)	Resistencia marca	PSNR (dB)	DM	Diferencia máxima	CM (%)
DB1	8	Sí	79.76	1.19	5	53.80
	7	Sí	80.98	1.56	9	71.85
	6	Sí	76.29	2.38	17	83.81
	5	Sí	70.91	4.13	31	91.26
DB3	8	Sí	85.06	1.19	5	53.90
	7	Sí	80.94	1.56	9	71.94
	6	Sí	76.26	2.39	17	83.89
	5	Sí	70.87	4.15	40	91.28
DB5	8	Sí	85.11	1.19	5	53.58
	7	Sí	81.01	1.55	9	71.72
	6	Sí	76.33	2.38	16	83.74
	5	Sí	70.94	4.13	40	91.23

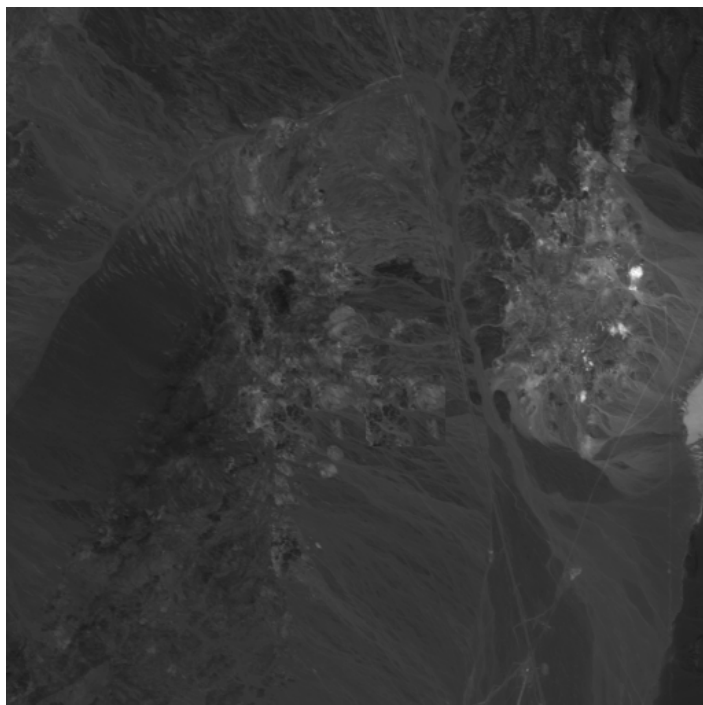
### 5.2.5.1. Ataques de copia y reemplazo

Nuevamente para determinar la eficacia de este último método propuesto, se aplica una serie de ataques de copia y reemplazo copiando una parte de la imagen marcada, de todas las firmas espectrales, y se ha sustituido en otra parte de la imagen. La Figura 5.25 muestra un ataque de copia de una parte sobre otra (a) y la detección de la modificación (b). La detección identifica como modificado el bloque completo de la Figura 5.22

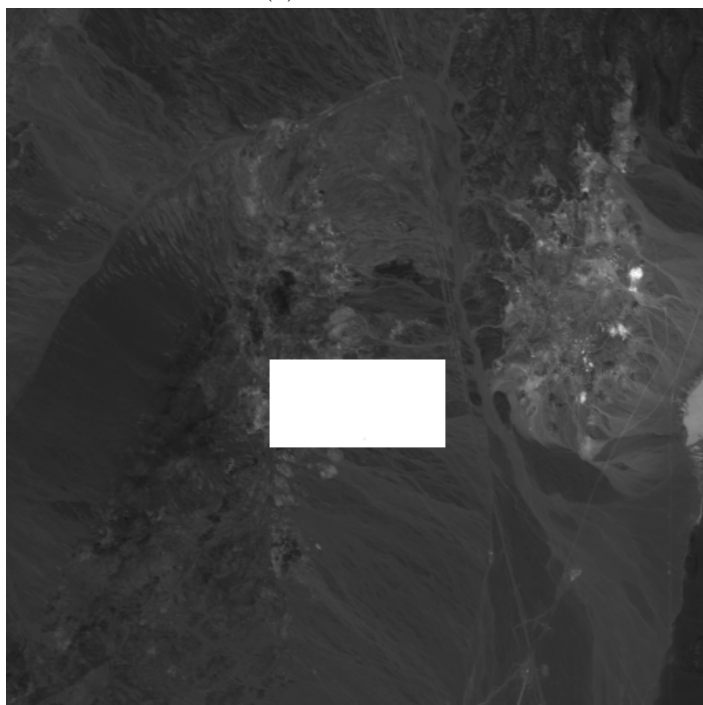
### 5.2.5.2. Ataques de compresión

Nuevamente se describen brevemente los ataques de compresión que se han realizado para este método. La Tabla 5.22 muestra los resultados obtenidos para cada una de las diferentes transformadas analizadas y los diferentes ratios de compresión que se han aplicado. Como se puede observar, el método





(a) Modificación



(b) Detección

Figura 5.25: Imagen modificada y detectada (banda 2).

resiste a las compresiones descritas.

El comportamiento de este método es muy parecido al anterior, casi idéntico. En el caso de los ataques de compresión hay una muy pequeña mejora en las diferencias máximas que se aplican a los componentes, en algunos casos la distancia media es un poco menor, y se modifican ligeramente menos componentes. Por lo tanto, este último método se demuestra mejor que los anteriores, ya que es el que mejores resultados de PSNR tiene a la vez que es el más robusto contra ataques de compresión.

La diferencia visual entre las imágenes marcadas y las originales es prácticamente nula. No se pueden apreciar cambios en la imagen, como tampoco son significativos los cambios que se realizan en la firma espectral, teniendo en cuenta que de los pocos componentes que se modifican (del 2 al 4%) lo hacen en una diferencia reducida (entre 8 y 16 unidades, de un máximo de 16 383).

Como en los métodos anteriores (excepto para el descrito en la Sección 5.2.5 donde los resultados son idénticos a los presentados aquí) se presenta un estudio de la robustez del método frente a ataques de compresión JPEG. La Tabla 5.23 muestra los resultados de estos ataques. Se puede apreciar como los resultados obtenidos son muy similares a los obtenidos con los otros métodos, ya que el ataque de compresión es el mismo para todos los casos y, por tanto, los valores de PSNR entre la imagen marcada y la imagen atacada son prácticamente idénticos. Este método mejora la robustez frente a ataques de sustitución de zonas de la imagen, pero no mejora sustancialmente los resultados frente a ataques de compresión.

Tabla 5.23: Robustez del esquema DWT con selección de áreas automática frente a ataques de compresión JPEG para la imagen Cuprite de 8 bpp.

DWT	Compr. (calidad)	Resistencia marca	PSNR (dB)	DM	diferencia. máxima	CM (%)
DB1	100	Sí	58.91	1	2	8.34
	99	Sí	54.73	1.01	2	21.81
	98	Sí	50.57	1.08	4	46.16
	97	Sí	48.39	1.21	6	56.89
	96	Sí	47.05	1.32	7	62.34
	95	Sí	46.01	1.42	8	65.98
	94	Sí	45.33	1.49	10	68.12
	93	Sí	44.78	1.56	11	69.74
	92	Sí	44.30	1.62	13	71.16
	91	Sí	43.97	1.66	13	71.99
	90	No	43.64	1.71	14	72.82
	85	No	42.39	1.90	16	76.02
	80	No	41.54	2.05	21	78.00
75	No	40.89	2.17	26	79.44	
DB3	100	Sí	58.91	1	2	8.35
	99	Sí	54.69	1.01	2	22.01
	98	Sí	50.43	1.08	4	47.07
	97	Sí	48.16	1.22	6	58.18
	96	Sí	46.79	1.34	8	63.72
	95	Sí	45.73	1.45	8	67.36
	94	Sí	45.04	1.53	10	69.45
	93	Sí	44.47	1.59	13	71.09
	92	Sí	44.00	1.66	13	72.49
	91	No	43.66	1.71	14	73.25
	90	No	43.33	1.75	14	74.10
	85	No	42.08	1.95	18	77.17
	80	No	41.24	2.10	22	79.06
75	No	40.60	2.23	26	80.37	
DB5	100	Sí	58.91	1	2	8.35
	99	Sí	54.66	1.01	2	22.15
	98	Sí	50.23	1.09	4	48.31
	97	Sí	47.88	1.24	6	59.83
	96	Sí	46.50	1.36	7	65.31
	95	Sí	45.47	1.47	9	68.84
	94	Sí	44.81	1.56	10	70.76
	93	Sí	44.27	1.62	12	72.24
	92	Sí	43.83	1.68	13	73.48
	91	Sí	43.52	1.73	13	74.17
	90	No	43.22	1.77	13	74.83
	85	No	42.13	1.94	19	77.23
	80	No	41.46	2.06	24	78.53
75	No	40.96	2.15	23	79.47	

### 5.3. Análisis comparativo con otros esquemas de watermarking

Existen algunos métodos de marcado en la literatura, como se describe en el Capítulo 3, que utilizan imágenes de obtención remota, algunas hiperespectrales y otras no. En esta sección compararemos algunos métodos descritos en la Sección 3.4 con los métodos propuestos en este trabajo.

Todos los métodos seleccionados son ciegos –excepto (Tamhankar et al., 2003)–, semifrágiles y por lo tanto permiten cierta compresión de las imágenes tratadas. Los métodos seleccionados son aquellos que se utilizan para detección de falsificaciones pero no todos los métodos tienen la localización de las modificaciones producidas, es decir, que determinan si la imagen se ha modificado en algún lugar sin concretar dónde.

La Tabla 5.24 muestra los resultados obtenidos para cada uno de los métodos estudiados. En la primera columna tenemos la referencia del método, en la segunda columna se muestra el tipo de imagen remota que considera el esquema, la tercera columna describe dónde se aplica el watermarking (en una banda, en un conjunto de bandas o en toda la firma espectral), la cuarta columna muestra el PSNR obtenido para cada uno de los métodos estudiados, si es que está descrito en el artículo seleccionado y, finalmente, la quinta columna indica si el método sirve o no para la localización de las modificaciones, mostrando el área que identifica como modificada o “No” si sólo detecta modificaciones pero no las localiza.

Entre los métodos que se aplican a las imágenes hiperespectrales, se ha de notar que (Sal and Graña, 2008) se aplica a cada banda por separado, lo que significa que la firma espectral no se modifica de manera uniforme y por tanto

Tabla 5.24: Comparación de los métodos presentados con otros sistemas de watermarking

Esquema	Tipo imagen	Estrategia marcado	PSNR	Localización modificación
(Tamhankar et al., 2003)	Hiperespectral	Firmas espec. seleccionadas	No reportado	No
(Qin et al., 2004)	RGB	RGB	No reportado	Área modificación
(Wang et al., 2005)	Pancromática (1 banda)	1 banda	$\sim 55$ dB (8 bpp)	No
(Ho et al., 2005)	Grisés (1 banda)	1 banda	$\sim 40$ dB (8 bpp)	Bloques $8 \times 8$
(Caldelli et al., 2006)	Grisés (1 banda)	1 banda	$\leq 45$ dB (8 bpp)	Bloques $16 \times 8$
(Sal and Graña, 2008)	Hiperespectral	Banda a banda	No reportado	No
(Serra-Ruiz et al., 2006)	Hiperespectral	16 bandas	$\sim 73$ dB (14 bpp)	$64 \times 64$ (o $32 \times 32$ )
(Serra-Ruiz and Megías, 2011)	Hiperespectral	16 bandas	$\sim 80$ dB (14 bpp) $\sim 70$ dB (8 bpp)	$64 \times 64$ (o $32 \times 32$ )
(Serra-Ruiz and Megías, 2010a)	Hiperespectral	16 bandas	$\sim 74$ dB (14 bpp) $\sim 67$ dB (8 bpp)	$64 \times 64$ (o $32 \times 32$ )
(Serra-Ruiz and Megías, 2010b)	Hiperespectral	16 bandas	$\sim 70$ dB (14 bpp) $\sim 65$ dB (8 bpp)	Variable $32 \times 32$   $128 \times 128$
Sección 4.2.6	Hiperespectral	16 bandas	$\sim 75$ dB (14 bpp) $\sim 65$ dB (8 bpp)	Variable $32 \times 32$   $128 \times 128$

la curva de la firma puede ser alterada significativamente. El sistema descrito por (Sal and Graña, 2008) se puede utilizar para detectar alteraciones, pero no dispone de localización de las firmas espectrales manipuladas. Aunque el esquema presentado en (Tamhankar et al., 2003) trabaja con la firma espectral de un conjunto de píxeles, no proporciona información sobre la localización de las manipulaciones de la imagen. En cuanto al tamaño de la localización se refiere, (Qin et al., 2004; Ho et al., 2005; Caldelli et al., 2006) detectan áreas más pequeñas en comparación con los esquemas propuestos en esta tesis, pero los métodos se aplican a cada banda por separado y producen peores resultados de PSNR.

Se puede apreciar que los métodos propuestos en esta tesis disponen de localización de las modificaciones de las imágenes, trabajan con la firma espectral completa y obtienen unos resultados extremadamente altos de calidad de la imagen, con el PSNR por encima de todos los demás métodos: alrededor de 65 dB para las imágenes reducidas a 8 bpp, contra los 55 dB del mejor de los otros esquemas de marcado, todo y que ésta no implementa localización. Aún y considerando únicamente el PSNR de las 16 bandas modificadas, éste alcanza el valor de 60 dB, también superior a los métodos comparados. En el caso de considerar las imágenes con 14-15 bpp se llega a unos valores realmente muy altos de calidad, entre 70 y 80 dB en PSNR según el método.

Los diferentes métodos no se han podido probar exactamente con las mismas imágenes o con los mismos parámetros por diferentes razones: algunos de ellos no proporcionan todos los datos para poder reproducir con exactitud los métodos de marcado, no se dispone de exactamente las mismas imágenes o no son hiperespectrales (son de obtención remota monobanda) y algunos tienen métodos muy sofisticados (por ejemplo algoritmos genéticos que requieren una amplia experiencia para poderlos implementar). Sin embargo,

los resultados mostrados en la Tabla 5.10 muestran que los métodos propuestos tienen valores muy similares de PSNR para diferentes imágenes. Así, la comparación para las imágenes de 8 bpp mostradas en la Tabla 5.24 es suficientemente válida.





# Capítulo 6

## Conclusiones

### 6.1. Conclusiones destacadas

La libre distribución de los contenidos digitales está en auge. La gran facilidad que proporciona Internet para distribuir estos contenidos hace que las personas compartan las imágenes, música o videos con un simple “clic” del ratón de su ordenador. Esto hace que las técnicas de protección del contenido digital sean de gran importancia para autenticarlo o para proteger los derechos de autor.

Dejando de lado la protección de los derechos de autor, la tesis se centra en el desarrollo de métodos para la autenticación y la detección de modificaciones para imágenes de obtención remota. Éstas son imágenes caras de obtener, que requieren de un sensor especial que capta la reflexión de la luz para diferentes longitudes de onda y que se instala en un avión o, en algunos casos, en satélites, como los LANDSAT o los enviados a otros planetas. Esto provoca que tome mucha importancia la autenticación del contenido. Las imágenes que se obtienen de la superficie terrestre o de otros planetas se deben poder validar para estar seguros de que no se ha modificado ninguna parte de éstas.

Al inicio de la tesis se establece como objetivo principal de la misma el diseño e implementación de nuevos métodos de watermarking semifrágil de imágenes multibanda. Los sistemas desarrollados debían permitir cierta compresión con pérdida de las imágenes y ser capaces de detectar modificaciones del contenido, localizando las zonas en la que se haya producido dicha manipulación. Además, los métodos debían ser ciegos, es decir, que para la localización de las modificaciones no se debería tener que utilizar la versión original de las imágenes.

Se ha conseguido desarrollar una familia de métodos que incrustan una marca en el contenido para la localización de las modificaciones permitiendo, a la vez, cierta compresión con pérdida de la imagen marcada. Estos métodos se han probado para tres imágenes hiperespectrales diferentes: la imagen Cuprite (de una mina del estado de Nevada en EE.UU.), la imagen Indian Pines (de unos campos de cultivo del estado de Indiana, también en EE.UU.) y la imagen del WTC del Nueva York tomada el 16 de septiembre de 2001. Todas ellas son imágenes hiperespectrales de tipo AVIRIS con 224 (ó 220) bandas y una profundidad de color de 2 bytes por píxel.

Con la aplicación de los métodos desarrollados, el proveedor del contenido sólo necesita marcar una vez la imagen y puede distribuir ésta con diferentes calidades, atendiendo a las diferentes ratios de compresión que se puede aplicar a la imagen sin que la marca desaparezca. Esto permite simplificar considerablemente el sistema de distribución de las imágenes, ya que el proveedor puede tener una única imagen marcada y distribuirla con diferentes calidades (ratios de compresión). Con esto se ahorra el coste que supondría marcar diferentes versiones de la imagen comprimidas con diferentes calidades para clientes diferentes.

## 6.2. Resultados de la tesis

Se han desarrollado cinco métodos diferentes para autenticar el contenido de las imágenes hiperespectrales. Todos ellos utilizan las técnicas de compresión TSVQ para marcar las diferentes áreas (bloques) que se quiere detectar si se han modificado. Mediante el uso de los árboles de compresión TSVQ y la cuantización de la entropía se consigue marcar las imágenes multibanda. Concretamente, se modifican los valores de las firmas espectrales para garantizar que se conserva la forma de éstas, de manera que no se pueda alterar el contenido de las firmas espectrales intentando falsificar el material que representa el píxel de la imagen.

El primer método de marcado se basa en la modificación de los componentes de las bandas de estas imágenes forzando a que el árbol TSVQ tenga una cierta entropía. El método divide la imagen en pequeños bloques tridimensionales (con dos dimensiones espaciales y una espectral) y fuerza a que cada bloque de  $64 \times 64$  píxeles de la imagen tenga una entropía ligeramente diferente. Una secuencia pseudoaleatoria sirve para determinar la entropía que ha de tener cada uno de los bloques de la imagen.

El segundo método se basa en el anterior, pero para hacerlo más robusto frente a ataques de compresión se le extraen un cierto número  $n$  de bits, los menos significativos, a los componentes y se marca la imagen obtenida con los bits más significativos de los mismos, usando los árboles de compresión TSVQ y la cuantización de la entropía. El proceso finaliza con la agregación de los LSB extraídos, minimizando la distancia entre el componente marcado y el original. Al igual que en el caso anterior, se cuantiza la entropía para poder detectar las modificaciones de la imagen, usando también una secuencia pseudoaleatoria generada a partir de una semilla que forma parte de la clave

secreta del método.

El tercer método se basa en la aplicación de la transformada *wavelet* en el marcado de las imágenes. Teniendo en cuenta que el algoritmo de compresión JPEG2000 utiliza la transformada *wavelet* para comprimir, almacenando los coeficientes de ésta, se propone un nuevo esquema de watermarking para que las marcas se incrusten en los coeficientes de la transformada *wavelet* de los diferentes bloques de  $64 \times 64$  píxeles en que se divide la imagen hiperespectral. Al igual que en los métodos anteriores, la entropía se cuantiza y se usa una secuencia pseudoaleatoria para escoger el valor de ésta en cada bloque.

Todos los métodos anteriores se basan en la división de la imagen hiperespectral de forma regular. Para poder garantizar la integridad y dificultar el proceso de manipulación, se diseña un nuevo método, a partir del anterior, en el que se introduce la división en bloques de diferentes tamaños, seleccionados de forma manual. Este cuarto método es más seguro que los anteriores, ya que no todos los bloques son de  $64 \times 64$  píxeles, sino que pueden tener diferentes medidas (e incluso formas). La entropía también se cuantiza y su valor en cada bloque se determina mediante una secuencia pseudoaleatoria. Como el número de bloques a marcar es diferente, se elige otro valor del paso de cuantización que dependerá del número de bloques. En este caso, la forma con que se divide la imagen en bloques es importante y se deberá recuperar en el momento de requerir la detección de las manipulaciones.

El quinto método surge de la necesidad de automatizar el proceso de división en bloques de diferentes tamaños. Así, éste se basa en el método anterior, pero la selección del tamaño de cada uno de los bloques a marcar se realiza de forma automática. Este proceso tiene en cuenta los diferentes valores de los componentes de todas las bandas y distribuye los bloques

agrupando componentes de valores similares. De este modo, zonas en las que el contenido sea similar estarán comprendidas en bloques grandes y a las zonas en las que el contenido de la superficie sea muy diferente se les designará un bloque mucho más pequeño. En este caso el tamaño y la posición de los bloques son necesarios para realizar la detección de modificaciones.

Los resultados obtenidos al respecto de la imperceptibilidad de la marca insertada en los contenidos para todos los métodos son muy buenos, estando el PSNR comprendido entre 60 dB (para imágenes de 8 bpp) y hasta 80 dB (para imágenes con 2 bytes de profundidad de color), lo que demuestra que los métodos afectan de forma mínima a las imágenes y, por tanto, permiten que los algoritmos de clasificación de materiales que se usan para estas imágenes funcionen correctamente con la versión de la imagen marcada. Al margen del PSNR también se ha analizado cómo afecta el método de incrustación a la forma de las curvas de las firmas espectrales, mostrando que la variación de éstas es mínima.

Respecto a la eficacia de los diferentes esquemas, éstos se han probado ampliamente para ataques de copia y reemplazo. Los métodos responden correctamente a este tipo de ataques y determinan la posición donde se ha realizado una modificación de la imagen marcada. También se ha comprobado que la imagen, una vez marcada, permite cierta compresión sin que por ello se elimine la marca y, por tanto, se puedan tener diferentes versiones de la imagen ya marcada a partir de las compresiones permitidas. Por el hecho de haber usado la transformada *wavelet* en los últimos métodos, éstos soportan mejor las compresiones JPEG2000 aunque no toleran tanto la compresión JPEG, que se basa en la transformada DCT.

La comparativa de los métodos de la tesis con otros de la literatura de-

talla cómo se mejoran los resultados obtenidos para este tipo de imágenes. Comparando los resultados se puede comprobar que se mejora, en todos los casos, la imperceptibilidad de las imágenes marcadas además de implementar la localización de las modificaciones, cosa que no todos los métodos de watermarking semifrágil sobre imágenes hiperespectrales de la literatura permiten.

### 6.3. Trabajo futuro

A partir de los experimentos realizados con los diferentes métodos propuestos en la tesis se contempla la necesidad de reducir el tamaño de los bloques a marcar. Las pequeñas modificaciones en las imágenes se detectan con una resolución igual al tamaño de los bloques. Una primera idea a desarrollar sería la inclusión de píxeles de los bloques vecinos a la hora de construir los árboles TSVQ pero sólo introducir la marca en el bloque tratado. Esto permitiría que se pudiera disponer de más vectores para la construcción de los árboles de compresión TSVQ y, por lo tanto, el tamaño del bloque podría reducirse considerablemente. Sin embargo esto presentaría ciertas complicaciones, ya que los bloques marcados y los originales no tienen los mismos vectores, por lo que el proceso tendría que ser iterativo hasta garantizar que la detección de la marca se produce de manera correcta.

En los últimos métodos propuestos se ha usado la transformada *wavelet* para el marcado sobre los coeficientes de ésta. Como posible línea de continuación se podría analizar el uso de las transformadas DCT, que es la que se utiliza en la compresión JPEG, o *Karhunen-Loève* (KLT), que aprovecha mucho mejor la redundancia de datos que tienen las imágenes hiperespectrales. El uso de estas transformadas pueden favorecer el comportamiento de los esquemas de marcado frente a compresiones JPEG y JPEG2000.

También podría ser interesante que el esquema permita además modificaciones diferentes a la compresión con pérdida, como por ejemplo rotaciones, filtros, recortes, etc. Una opción podría ser incrustar algún tipo de marca en el contenido para poder detectar estas rotaciones o recortes y encontrar la división de los bloques realizada en el proceso de marcado de la imagen.

Otra línea de investigación futura son los métodos de watermarking semifrágiles reversibles, que permitirían recuperar la imagen original una vez se haya autenticado que ésta no ha sido sometida a manipulaciones fraudulentas. Se debería marcar la imagen de forma que, una vez recuperada la marca, se pueda eliminar ésta de la imagen obteniéndose como resultado la imagen original sin variación alguna.





# Bibliografía

- Aiazzi, B., Alparone, L., Baronti, S., Lastrì, C., and Santurri, L. (2006). *Hyperspectral Data Compression*, chapter Near-Lossless Compression of Hyperspectral Imagery Through Crisp/Fuzzi Adaptive DPCM, pages 147–178. Springer Science+Business Media, Inc.
- Alford, H. (2005). Fictitious entry. [http://www.newyorker.com/archive/2005/08/29/050829ta\\_talk\\_alford](http://www.newyorker.com/archive/2005/08/29/050829ta_talk_alford), (Accedido 19 de junio de 2011).
- Arjun, S. and Rao, N. (2008). An approach to reversible information hiding for images. In *TENCON 2008 - 2008 IEEE Region 10 Conference*, pages 1–6.
- Arnold, M., Schmucher, M., and Wolthunsen, S. (2003). *Techniques and Applications of Digital Watermarking and Content Protection*. Computer security series. Artech House.
- Arnold, N. and Huang, Z. (2001). Blind detection of multiple audio watermarks. In *Web Delivering of Music, 2001. Proceedings. First International Conference on*, pages 4–11.
- Balmer, L. (1996). *Signal and systems: An Introduction*. Prentice Hall.
- Bell, A. (1999). The dynamic digital disk. *IEEE Spectrum*, 36(10):28–35.
- Bloom, J., Cox, I., Kalker, T., Linnartz, J.-P., Miller, M., and Traw, B.

- (1999). Copy protection for DVD video. In *Proceedings of the IEEE*, volume 87, pages 1267–1276. IEEE.
- Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Wadsworth International Group.
- Brown, J. (1997). Playmate meets geeks who made her a net star. <http://www.wired.com/culture/lifestyle/news/1997/05/4000> (Accedido 19 de junio de 2011).
- Caldelli, R., Filippini, F., and Barni, M. (2006). Joint near-lossless compression and watermarking of still images for authentication and tamper localization. *Signal Processing: Image Communication*, 21:10(10):890–903.
- Caldelli, R., Macaluso, G., Barni, M., and Magli, E. (2004). Joint near-lossless watermarking and compression for the authentication of remote sensing images. In *Proceedings of the 24th International Geoscience and Remote Sensing Symposium*, volume 1.
- Camps-Valls, G., Bandos Marsheva, T., and Zhou, D. (2007). Semi-supervised graph-based hyperspectral image classification. *Geoscience and Remote Sensing, IEEE Transactions on*, 45(10):3044–3054.
- Cayre, F., Fontaine, C., and Furon, T. (2004). Watermarking attack: Security of WSS techniques. In *IWDW'04*, pages 171–183.
- Chang, C.-I., Du, Q., Sun, T.-L., and Althouse, M. (1999). A joint band prioritization and band-decorrelation approach to band selection for hyperspectral image classification. *Geoscience and Remote Sensing, IEEE Transactions on*, 37(6):2631–2641.
- Clark, R. N. (2001). Environmental studies of the world trade center area after the september 11, 2001 attack. *U.S. Geological Survey*.

- Craver, S. and Stern, J. (2001). Lessons learned from SDMI. In *Multimedia Signal Processing, 2001 IEEE Fourth Workshop on*, pages 213–218.
- Cristophe, E., Léger, D., and Mailhes, C. (2005). Quality criteria benchmark for hyperspectral imagery. *GRS*, 43(9):2103–2114.
- Dittmann, J., Megías, D., Lang, A., and Herrera-Joancomartí, J. (2006). Theoretical framework for a practical evaluation and comparison of audio watermarking schemes in the triangle of robustness, transparency and capacity. *T. Data Hiding and Multimedia Security*, pages 1–40.
- Eggers, J. and Girod, B. (2001). Blind watermarking applied to image authentication. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, volume 3, pages 1977–1980 vol.3.
- Ekici, O., Sankur, B., Naci, U., Coskun, B., and Akcay, M. (2004). Comparative assessment of semifragile watermarking methods. *Journal of Electronic Imaging*, 13(1):209–216.
- El-Taweel, G., Onsi, H., Samy, M., and Darwish, M. (2007). Secure and non-blind watermarking scheme for color images based on DWT. In *ICGST International Journal on Graphics, Vision and Image Processing*, pages 1–5.
- Fallahpour, M., Megías, D., and Ghanbari, M. (2009). High capacity, reversible data hiding in medical images. In *EEE International Conference on Image Processing (ICIP)*, pages 4241–4244, Cairo.
- Fridrich, J. (1998). Applications of data hiding in digital images.
- Fridrich, J. (2002). Security of fragile authentication watermarks with lo-

- calization. In *Security and Watermarking of Multimedia Contents Proc. SPIE*, volume 4675, pages 691–700, San Jose, CA.
- Gantasala, R. and Prasad, M. (2009). New quantization technique in semi-fragile digital watermarking for image authentication. In Prasad, S. K., Routray, S., Khurana, R., and Sahni, S., editors, *Information Systems, Technology and Management*, volume 31, pages 244–255. Springer Berlin Heidelberg.
- Gersho, A. and Gray, R. M. (1992). *Vector Quantization and Signal Compression*. Communications and Information Theory. Kluwer Academic Publishers, Norwell, MA, USA.
- Gray, R. M. and Neuhoff, D. L. (1998). Quantization. *IEEE Transactions on Information Theory*, 44(6):2325–2383.
- Ho, A., Zhu, X., and Woon, W. (2005). A semi-fragile pinned sine transform watermarking system for content authentication of satellite images. *IEEE International Geoscience and Remote Sensing Symposium*, 1-8.
- Huffman, D. A. (1952). A method for the construction of minimum-redundancy codes. In *Proceedings of the IRE*, volume 40, pages 1098–1101.
- Image Power Inc. (2003). JASPER JPEG-2000 transcoder. <http://www.ece.uvic.ca/~mdadams/jasper/> (Accedido 19 de junio de 2011).
- Jähne, B. (2005). *Digital Image Processing*. Springer-Verlag Berlin Heidelberg.
- Johnson, N., Duric, Z., and Jajodia, S. (2000). *Information Hiding: Steganography and Watermarking - Attacks and Countermeasures*. Kluwer Academic Publishers.

- Kasner, J., Marcellin, M., and Hunt, B. (1999). Universal trellis coded quantization. In *IEEE Trans. Image*, number 8, pages 1677–1687.
- Katzenbeisser, S. and Petitcolas, F. (1999). *Information hiding techniques for steganography and digital watermarking*. Artech House Publishers.
- Landgrebe, D. (1992). AVIRIS NW indianas indian pines 1992. [Online]. Available: <ftp://ftp.ecn.purdue.edu/biehl/MultiSpec/92AV3C.1an> (original files) and [ftp://ftp.ecn.purdue.edu/biehl/PC\\_MultiSpec/ThyFiles.zip\(groundtruth\)](ftp://ftp.ecn.purdue.edu/biehl/PC_MultiSpec/ThyFiles.zip(groundtruth)).
- Lang, A., Dittmann, J., Spring, R., and Vielhauer, C. (2005). Audio watermarking attacks: from single to profile attacks. In *Multimedia and Security, Workshop New York, NY, USA*, pages 39–50, New York. ACM.
- Lee, S., Yoo, C., and Kalker, T. (2007). Reversible image watermarking based on integer-to-integer wavelet transform. *Information Forensics and Security, IEEE Transactions on*, 2(3):321–330.
- Li, C.-T. (2004). Digital fragile watermarking scheme for authentication of JPEG images. *Vision, Image and Signal Processing, IEE Proceedings -*, 151(6):460–466.
- Lin, C.-C., Tai, W.-L., and Chang, C.-C. (2008). Multilevel reversible data hiding based on histogram modification of difference images. *Pattern Recognition*, 41(12):3582–3591.
- Lin, E., Podilchuk, C. I., and Delp, E. J. (2000). Detection of image alterations using semi-fragile watermarks. In *Storage and Retrieval for Image and Video Databases*.
- Linde, Y., Buzo, A., and Gray, R. M. (1980). An algorithm for vector quantizer design. *COM, COM-28(1)*:84–95.

- Liu, K. R., Trappe, W., Wang, Z. J., Wu, M., and Zhao, H. (2005). *Multi-media Fingerprinting forensics for Traitor Tracing*. Hindawi.
- Mallat, S. (1998). *A wavelet tour of signal processing*. Academic Press.
- Markas, T. and Reif, J. (1993). Multispectral image compression algorithms. In *Data Compression Conference, 1993. DCC '93.*, pages 391 – 400.
- Mathison (1779). *Historical Chonicle*, volume 49, page 374. Gentlemen's Magazine.
- Megías, D., Serra-Ruiz, J., and Fallahpour, M. (2010). Efficient self-synchronised blind audio watermarking system based on time domain and FFT amplitude modification. *Signal Processing*, 90(12):3078–3092.
- Melgani, F. and Bruzzone, L. (2004). Classification of hyperspectral remote sensing images with support vector machines. *IEEE Transactions on Geoscience and Remote Sensing*, 42(8):1778–1790.
- Menezes, A., van Oorschot, P., and Vanstone, S. (1996). *Handbook of Applied Cryptography*. Discrete Mathematics and ITS Applications. CRC Press.
- Miano, J. (1999). *Compressed image file formats : JPEG, PNG, GIF, XBM, BMP*. Addison-Wesley, cop.
- Mielikainen, J. and Toivanen, P. (2006). *Hyperspectral Data Compression*, chapter Lossless Hyperspectral Image Compression via Linear Prediction, pages 57–74. Springer Science+Business Media, Inc.
- Minguillón, J., Pujol, J., Serra, J., and Ortuño, I. (2000a). Influence of lossy compression on hyperspectral image classification. In *Proceedings of Data Mining'2000*, pages 545–554, Cambridge, UK.

- Minguillón, J., Pujol, J., Serra, J., Ortuño, I., and Guitart, P. (2000b). Adaptive lossy compression and classification of hyperspectral images. In *Proceedings of Image and Signal Processing for Remote Sensing VI*, volume 4170, pages 214–225, Barcelona, Spain.
- Motta, G., Rizzo, F., and Storer, J. A. (2005). *Hyperspectral Data Compression*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.
- Moulin, P. and Koetter, R. (2005). Data-hiding codes. *Proceedings of IEEE*, vol. 93(12):pp: 2083–2126.
- Petitcolas, F. and Anderson, R. (1999). Evaluation of copyright marking systems. In *Proceedings of IEEE Multimedia Systems'99*, pages 574–579.
- Petitcolas, F., Anderson, R., and Kuhn, M. (1998). Attacks on copyright marking systems. In *2nd Workshop on Information Hiding*, LNCS 1525, pages 219–239, Portland, Oregon, USA. Springer-Verlag.
- Plaza, A., Martinez, P., Plaza, J., and Perez, R. (2005). Dimensionality reduction and classification of hyperspectral image data using sequences of extended morphological transformations. *Geoscience and Remote Sensing, IEEE Transactions on*, 43(3):466 – 479.
- Qin, Q., Wang, W., and Chen, S. (2004). Research of digital semi-fragile watermarking of remote sensing image based on wavelet analysis. *IEEE International Geoscience and Remote Sensing Symposium*, 1-7.
- Ramasubramanian, M., Pattanaik, S. N., and Greenberg, D. P. (1999). A perceptually based physical error metric for realistic image synthesis. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 73–82, New York, USA. ACM Press/Addison-Wesley Publishing Co.

- Raudys, S. J. (1997). On dimensionality, sample size, and classification error of nonparametric linear classification algorithms. *PAMI*, 19(6):667–671.
- Ren, H., Chang, C., and Zhang, J. (2009). Reversible image hiding algorithm based on pixels difference. In *Automation and Logistics, 2009. ICAL '09. IEEE International Conference on*, pages 847–850.
- Rey, C. and Dugelay, J. (2002). A survey of watermarking algorithms for image authentication. *Journal on Applied Signal Processing*, 6:613–621.
- Sal, D. and Graña, M. (2008). *Studies in Computational Intelligence*, volume 133/2008, chapter A Multiobjective Evolutionary Algorithm for Hyperspectral Image Watermarking, pages 63–78. Springer Berlin / Heidelberg.
- Serra-Ruiz, J. and Megías, D. (2010a). DWT and TSVQ-based semi-fragile watermarking scheme for tampering detection in remote sensing images. In *Image and Video Technology (PSIVT), 2010 Fourth Pacific-Rim Symposium on*, pages 331–336.
- Serra-Ruiz, J. and Megías, D. (2010b). Watermarking scheme for tampering detection in remote sensing images using variable size tiling and DWT. In Huang, B., Plaza, A., Serra-Sagrista, J., Lee, C., Li, Y., and Qian, S., editors, *Proceedings of Satellite Data Compression, Communications, and Processing VI*, volume 7810, page 78100A, San Diego. SPIE.
- Serra-Ruiz, J. and Megías, D. (2011). A novel semi-fragile forensic watermarking scheme for remote sensing images. *International Journal of Remote Sensing*. In press.
- Serra-Ruiz, J., Megías, D., J.Herrera-Joancomartí, and Minguillón, J. (2006). Multiband semigrfile watermarking for multi and hyperspectral images based on iterative tree structured vector quantization. In Bruzzone, L.,



- editor, *Proceedings of Image and signal processing for remote sensing XII*, volume 6365, pages 6365001–6365010, Stockholm. SPIE.
- Shannon, C. E. (1948). A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423 and 623–656.
- Silvestre, G., Hurley, N., Hanau, G., and Dowling, W. (2001). Informed audio watermarking scheme using digital chaotic signals. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, volume 3, pages 1361–1364 vol.3.
- Smith, R. B. (2006). Introduction to hyperspectral imaging. <http://www.microimages.com/getstart/hyprspec.htm> (Accedido 19 de junio de 2011).
- Swanson, M., Zhu, B., Tewfik, A., and Boney, L. (1998). Robust audio watermarking using perceptual masking. *Elsevier Signal Processing, Special Issue on Copyright Protection and Access Control*, 66(3):337–355.
- Sweldens, W. (1998). The lifting scheme: a construction of second generation wavelets. *SIAM J. Math. Anal.*, 29:511–546.
- Tamhankar, H., Bruce, L., and Younan, N. (2003). Watermarking of hyperspectral data. *Geoscience and Remote Sensing Symposium, 2003. IGARSS '03. Proceedings. 2003 IEEE International*, 6:3574–3576 vol.6.
- Taubman, D. (2007). Kakadu JPEG-2000 encoder. <http://www.kakadusoftware.com>. (Accedido 19 de junio de 2011).
- Taubman, D. S. and Marcellin, M. W. (2002). *JPEG2000 : image compression fundamentals, standards, and practice*. Kluwer Academic Publishers, Boston.

- NASA, Jet Propulsion Laboratory (2004). Airborne visible / infrared imaging spectrometer (AVIRIS). <http://aviris.jpl.nasa.gov> (Accedido 19 de junio de 2011).
- NASA, U.S. Geological Survey (1972). Landsat program. <http://landsat.gsfc.nasa.gov> (Accedido 19 de junio de 2011).
- Tian, J. (2002). Wavelet-based reversible watermarking for authentication. In *Security and Watermarking of Multimedia Contents IV*, volume 4675, pages 679–690. SPIE.
- Voronoi, G. (1907). Nouvelles applications des paramètres continus à la théorie des formes quadratiques. *Journal für die Reine und Angewandte Mathematik*, 133:97–178.
- Wang, X., Guan, Z., and Wu, C. (2005). Advanced data mining and applications. 3584/2005:423–430.
- Yang, Q., Gao, T., and Li, F. (2010). A novel semi-fragile authentication watermarking scheme of color JPEG image in compressed domain. In *Multimedia Information Networking and Security (MINES), 2010 International Conference on*, pages 666 –670.
- Yu, H., Kundur, D., and Lin, C.-Y. (2001). Spies, thieves, and lies: The battle for multimedia in the digital era. *IEEE Multimedia*, 8(3):8–12.
- Zhu, X., Ho, A. T., and Marziliano, P. (2007). A new semi-fragile image watermarking with robust tampering restoration using irregular sampling. *Signal Processing: Image Communication*, 22(5):515 – 528.

# Publicaciones

## Derivadas de la tesis

### Revistas:

- J. Serra-Ruiz, D. Megías. *A novel semi-fragile forensic watermarking scheme for remote sensing images*, International Journal of Remote Sensing, Taylor and Francis. ISSN: 0143-1161 (factor de impacto 1,089 en 2009. Segundo cuartil en categoría *Remote Sensing*). Aceptado en mayo de 2010, pendiente de publicar.

### Proceedings:

- J. Serra-Ruiz, D. Megías. *DWT and TSVQ-based Semi-fragile Watermarking Scheme for Tampering Detection in Remote Sensing Images*. Proceedings of the PSIVT 2010. (Noviembre 2010). Pág. 331-336 **CORE B**.
- J. Serra-Ruiz, D. Megías. *Watermarking scheme for tampering detection in remote sensing images using variable size tiling and DWT*. Proceedings of the SPIE 2010. Optical Engineering + Applications conference. Volumen: 7810 Pág. 78100A (Agosto 2010)
- J. Serra-Ruiz, D. Megías, J.Herrera-Joancomartí, J. Minguillón. *Mul-*

*tiband semifragile watermarking for multi and hyperspectral images based on iterative tree structured vector Quantization*. Proceedings of the SPIE Remote Sensing Conference 2006. Volumen: 6365. Página inicial: 63650O-1, final: 63650O-10 (septiembre 2006).

## Otras publicaciones de watermarking

### Revistas:

- D. Megías, J. Serra-Ruiz, M. Fallahpour. *Efficient self-synchronised blind audio watermarking system based on time domain and FFT amplitude modification*, Journal of Signal Processing, ISSN: 0165-1684 (factor de impacto 1,135 en 2009, Segundo cuartil en categoría *Engineering, Electrical & Electronic*). Volumen: 90 (12). Página inicial: 3078, final: 3092 (Diciembre 2010).

### Proceedings:

- D. Megías, J. Herrera Joancomartí, J. Serra-Ruiz, J. Minguillón. *A benchmark assessment of the WAUC watermarking audio algorithm*. Proceedings of the SPIE 18th Annual Symposium Electronic Imaging 2006. Volumen: 6072. Página inicial: 60721K-1, final: 60721K-10. (enero 2006).
- J. Minguillón, J. Herrera Joancomartí, D. Megías, J. Serra-Ruiz, J. Serra-Sagrístà, F. García. *The Influence of Mark Embedding Strategies on Lossless Compression of Ultraspectral Images*. Proceedings of the IGARSS 2005. Página inicial: 144, final: 147. (agosto 2005). **CORE C**.



