

Estándar JPEG

Marcos Martín

4 de mayo de 2004

1 Introducción

Las técnicas de compresión de imagen actualmente en desarrollo logran tasas de compresión desde 10:1 a 50:1 sin pérdida perceptible de calidad. Sin embargo, disponer de esta tecnología no es suficiente. Para que una aplicación del mercado que emplee almacenamiento y transmisión de imágenes digitales, es necesario disponer de un estándar de compresión que permite la interoperabilidad entre equipos de diferentes fabricantes.

Durante la última década se ha desarrollado el estándar JPEG (Joint Photographic Experts Group). Se ha estado trabajando para establecer el primer estándar internacional de compresión de imagen digital, tanto para imágenes con niveles de gris como en color. Este estándar ha sido desarrollado de forma conjunta por los comités CCITT e ISO. JPEG se propuso la ambiciosa tarea de desarrollar un estándar de compresión de imagen de propósito general para ajustarse a las necesidades de la mayor parte de las aplicaciones que utilizan imágenes digitales.

2 Requerimientos y Modos de Operación

El objetivo de JPEG es desarrollar un método de compresión de imagen que cumpla con los siguientes requisitos:

- Que logre tasas de compresión y calidad de imagen similares a las que son en la actualidad tecnológicamente viables. La calidad debe ser de “muy buena” a “excelente”. La compresión debe ser parametrizable de forma que el usuario puede elegir el compromiso deseado entre tasa de compresión y calidad.
- Que se pueda aplicar a cualquier tipo de imagen fuente.
- Que tenga una complejidad computacional tal que sea implementable en la práctica en el estado actual de la tecnología.

En cuanto a los modos de funcionamiento son los siguientes:

- Codificación secuencial: cada componente de la imagen se codifica siguiendo un patrón de izquierda a derecha y de arriba a abajo.

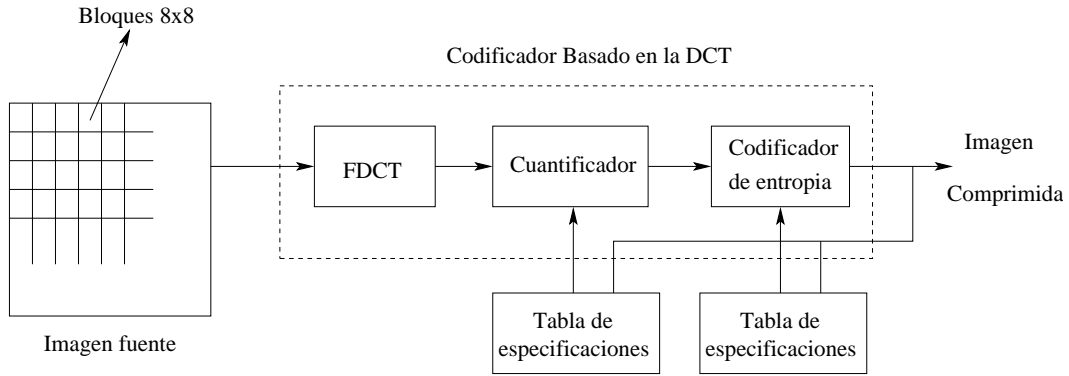


Figura 1: Esquema del codificador basado en la DCT.

- Codificación progresiva: la imagen se codifica en varias etapas para aplicaciones en las cuales el tiempo de transmisión sea elevado y el usuario prefiera ver cómo la imagen se va construyendo empezando por una imagen con baja calidad y aumentando progresivamente esta calidad.
- Codificación sin pérdidas: la imagen se codifica de tal forma que se garantice la recuperación exacta de la imagen fuente original.
- Codificación jerárquica: la imagen se codifica a múltiples resoluciones de forma que se pueda acceder a las versiones con menor resolución sin necesidad de descomprimir las versiones de mayor resolución.

3 Arquitectura del Estándar Propuesto

Como ya hemos dicho el estándar contiene cuatro modos de operación. Para cada modo se especifican uno o más codecs (codificadores/decodificadores). Los codecs dentro de un modo se diferencian en la precisión (número de bits) de las imágenes fuente y el método de entropía utilizado como parte de la compresión. Esta diversidad de codecs es necesaria para cumplir el objetivo de ser una herramienta de compresión de imagen genérica que soporte diferentes formatos de imágenes fuente. La mayor parte de implementaciones no soporta todos los modos ni todos los codecs, la mayoría de ellas sólo da soporte al codec secuencial básico (Baseline Sequential Codec). Este codec suele ser suficiente para la mayor parte de las aplicaciones.

4 Codificación Basada en la DCT

En las figuras 1 y 2 se puede ver el esquema del codificador y decodificador de los modos de funcionamiento basados en la DCT (Discrete Cosine Transform). En estas figuras se ilustra el caso de imágenes con una única componente (niveles de gris). Estos esquemas procesan bloques 8×8 de la imagen con niveles de gris. La compresión de imágenes en color se puede considerar similar a la compresión de varias imágenes con niveles de gris. En este caso se puede

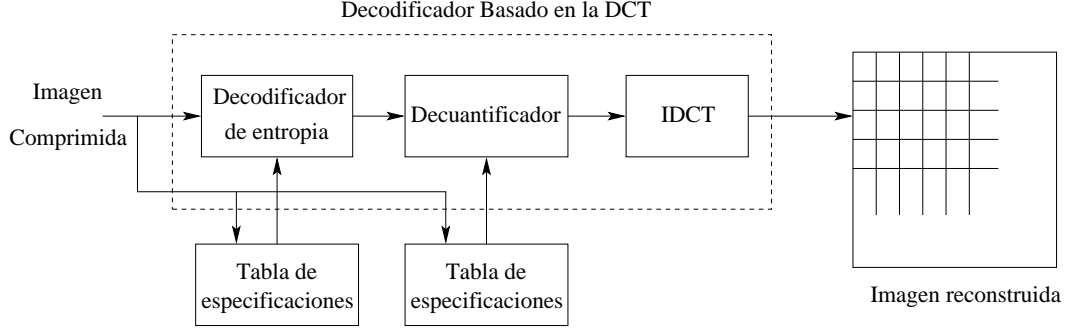


Figura 2: Esquema del decodificador basado en la DCT.

comprimir la primera componente de la imagen entera, después la segunda, etc. o bien entrelazando bloques 8×8 de cada componente.

En el codec secuencial básico cada bloque de 8×8 se procesa por separado dando lugar a la salida comprimida. En el modo progresivo basado en DCT se dispone de un buffer a la entrada del codificador de entropía de forma que se almacenan los coeficientes cuantificados de toda la imagen completa para posteriormente irlos codificando en varias pasadas de forma que se vaya mejorando la calidad en cada pasada. En el modo jerárquico, los esquemas de las figuras 1 y 2 se integran como parte de un esquema global más grande que comienza comprimiendo la imagen a baja resolución para ir sucesivamente aumentando la resolución.

4.1 FDCT e IDCT 8×8

A la entrada del codificador las muestras de la imagen se agrupan en bloques de 8×8 . El valor de cada muestra está en el rango de enteros sin signo $[0, 2^P - 1]$. Este rango se desplaza para tener enteros con signo $[-2^{P-1}, 2^{P-1} - 1]$. Este bloque 8×8 sirve de entrada a la FDCT (Forward DCT) en el codificador. En el decodificador la IDCT (Inverse DCT) tiene como salida los bloques 8×8 que forman la imagen reconstruida. Las expresiones matemáticas para la FDCT y para la IDCT son:

$$F(u, v) = \frac{1}{4} C(u) C(v) \sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \quad (1)$$

y

$$f(x, y) = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 C(u) C(v) F(u, v) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \quad (2)$$

donde

$$C(u) = \begin{cases} 1\sqrt{2} & \text{para } u = 0 \\ 1 & \text{en otro caso} \end{cases} \quad (3)$$

La DCT está relacionada con la DFT. Además la DCT se puede ver como un analizador de armónicos y la IDCT como un sintetizador de armónicos. Los 64 coeficientes de la DCT representan las amplitudes de las frecuencias

espaciales o lo que es lo mismo el espectro del bloque 8×8 . El coeficiente para frecuencia cero se va a denominar coeficiente DC. Los restantes 63 coeficientes se van a denominar coeficientes AC. Ya que los valores del bloque 8×8 de la imagen fuente suelen variar muy lentamente, la información se va a concentrar en los coeficientes a baja frecuencia. Esto va a permitir lograr una tasa de compresión elevada ya que la mayor parte de los coeficientes de la DCT toman un valor cero o muy cercano a cero.

En el decodificador la IDCT parte de los 64 coeficientes y reconstruye el bloque correspondiente de la imagen. Matemáticamente la DCT es una aplicación uno a uno entre el dominio de la imagen y el dominio de la frecuencia espacial. Si los coeficientes no se cuantifican la IDCT recupera el bloque 8×8 original con total exactitud. Se puede decir que la DCT no introduce pérdidas en la calidad de la imagen.

Para el modo de funcionamiento basado en la DCT, el estándar JPEG especifica dos codecs distintos para imágenes fuente con 8 bits por componente e imágenes fuente con 12 bits por componente, respectivamente. Los codecs de 12 bits son necesarios para poder comprimir ciertos tipos de imágenes médicas y de otro tipo, y se requiere un mayor número de recursos computacionales para lograr la precisión requerida en la FDCT y en la IDCT.

4.2 Cuantificación

Tras la etapa FDCT cada uno de los coeficientes de la DCT se cuantifican de forma uniforme utilizando una tabla de cuantificación de 64 elementos que se debe especificar desde la aplicación (o usuario) como entrada del codificador. Cada elemento de la tabla es un entero de 1 a 255 que especifica el tamaño del escalón para cuantificar el coeficiente de la DCT correspondiente. El propósito de la cuantificación es aumentar la compresión representando cada coeficiente de la DCT con una precisión no mayor que la estrictamente necesaria para lograr la calidad deseada en la imagen reconstruida. En definitiva se trata de descartar aquella información que no es visualmente significativa. El proceso de cuantificación es irreversible, por lo que es un proceso que siempre introduce pérdidas.

La cuantificación se define como la división de cada coeficiente de la DCT entre el tamaño del escalón correspondiente y el posterior redondeo al entero más próximo:

$$F^Q(u, v) = \text{Round} \left(\frac{F(u, v)}{Q(u, v)} \right) \quad (4)$$

Cada coeficiente se normaliza con respecto al tamaño del escalón correspondiente. El proceso de decuantificación es el inverso y se elimina la normalización multiplicando el valor cuantificado por el tamaño del escalón:

$$F^{Q'}(u, v) = F^Q(u, v) \cdot Q(u, v) \quad (5)$$

Cuando el objetivo sea comprimir la imagen tanto como sea posible sin pérdida visual de calidad, se debe elegir el tamaño del escalón para cada coeficiente en el umbral de percepción de la función base correspondiente a ese

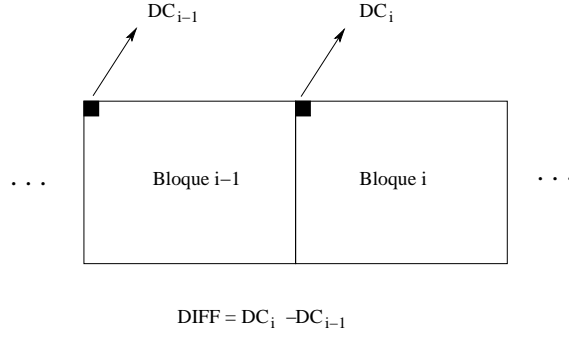


Figura 3: Codificación diferencial del coeficiente DC.

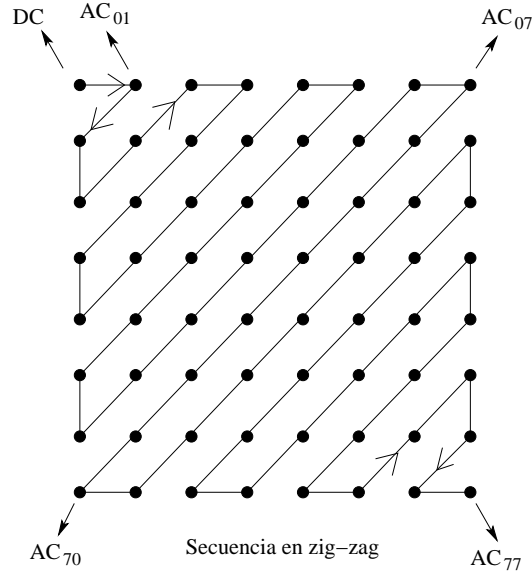


Figura 4: Ordenación en zig-zag de los 63 coeficientes AC.

coeficiente. Estos umbrales son función también de las características de la imagen fuente, de las características del equipo de presentación y de la distancia de observación.

4.3 Codificación DC y Secuencia en Zig-Zag

Tras la cuantificación, el coeficiente DC se trata de forma diferente los 63 coeficientes AC. El coeficiente DC es una medida del valor medio del bloque 8×8 de la imagen original. Debido a que existe una correlación muy elevada entre los coeficientes DC de bloques adyacentes, el coeficiente DC se codifica como la diferencia entre el coeficiente DC del bloque actual y el coeficiente DC del bloque anterior del orden de la secuencia de bloques (que se definirá más adelante) como puede verse en la figura 3. Este trato especial se puede justificar debido a que el coeficiente DC frecuentemente contiene una fracción significativa de la energía total del bloque 8×8 .

Finalmente, el resto de coeficientes AC ya cuantificados se ordenan siguiendo el patrón en zig-zag de la figura 4. Esta ordenación va a simplificar la

etapa de codificación de entropía colocando los coeficientes de baja frecuencia espacial antes que los de alta frecuencia.

4.4 Codificación de Entropía

El paso final del codificador basado en la DCT es la codificación de entropía. Esta etapa logra una compresión adicional sin pérdidas codificando los coeficientes de la DCT de la forma más compacta posible basándose en su estadística. JPEG especifica dos métodos de codificación de entropía: la codificación Huffman y la codificación aritmética. El codec secuencial básico utiliza codificación Huffman, pero se permiten ambos tipos de codificadores de entropía para todos los modos de funcionamiento.

El proceso de codificación se puede considerar como un proceso en dos etapas:

- En la primera etapa se convierte la secuencia en zig-zag de coeficientes cuantificados en una secuencia intermedia de símbolos.
- En la segunda etapa se convierte los símbolos en una secuencia de bits en la cual los símbolos externamente no tienen fronteras identificables.

La definición y forma de los símbolos intermedios depende tanto del modo de operación basado en la DCT como en el método de codificación de entropía empleado.

La codificación Huffman requiere que la aplicación (o el usuario) especifique uno o más conjuntos de tablas de códigos Huffman. Estas tablas serán necesarias tanto en el codificador como en el decodificador. Las tablas Huffman pueden estar predefinidas por defecto dentro de una aplicación o se pueden determinar específicamente para cada imagen a partir de la estadística de ésta.

En el método de codificación aritmética especificado en el estándar JPEG, no son necesarias tablas externas puesto que este tipo de codificación se adapta a la estadística de la imagen según se va codificando. Si se desea se puede facilitar estas tablas para mejorar el funcionamiento del codificador aritmético, pero no son estrictamente necesarias. La codificación aritmética da lugar a una compresión del 5 al 10% mejor que la Huffman, sin embargo la carga computacional es mayor.

4.5 Tasa de Compresión y Calidad

Para imágenes en color y con escenas con complejidad moderada las tasas de compresión y la calidad lograda se puede resumir como sigue (incluyen las componentes de luminancia y crominancia):

- 0.25 – 0.5 bits/pixel: calidad moderada a buena, suficiente para algunas aplicaciones.
- 0.5 – 0.75 bits/pixel: calidad buena a muy buena, suficiente para muchas aplicaciones.

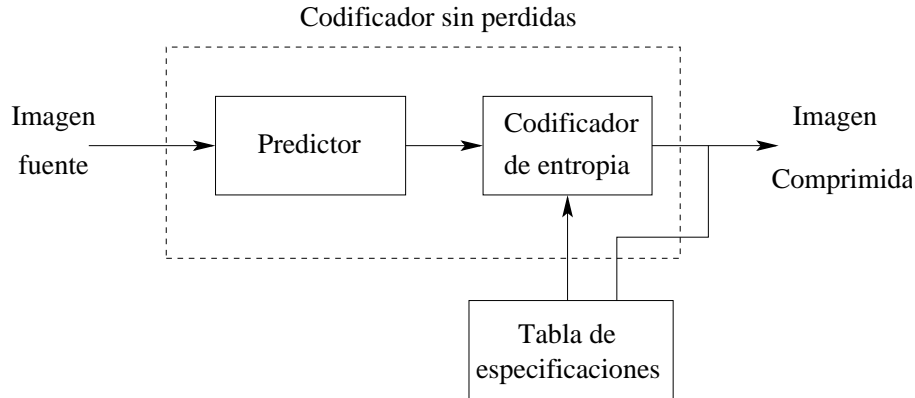


Figura 5: Esquema del codificador predictivo sin pérdidas.

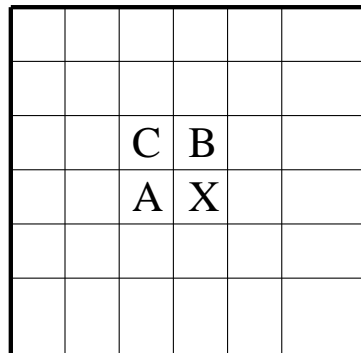


Figura 6: Configuración del vecindario para determinar la predicción.

- 0.75 – 1.5 bits/pixel: calidad excelente, suficiente para la mayoría de aplicaciones.
- 1.5 – 2.0 bits/pixel: normalmente indistinguible de la imagen original, suficiente para casi la completa mayoría de aplicaciones demandadas.

5 Codificación Predictiva sin Pérdidas

Un método sin pérdidas basado en la DCT era difícil de definir como un estándar práctico en el que los codificadores y decodificadores se implementaran independientemente. JPEG ha adoptado como método de compresión sin pérdidas un método predictivo simple que es totalmente independiente del codificador basado en la DCT. La elección de este método no es el resultado de un análisis riguroso como lo fue en el caso de la codificación basada en la DCT, sin embargo los resultados que se obtienen con este método sin pérdidas son bastante similares a los resultados conseguidos con la tecnología actual.

En la figura 5 se muestra el esquema del codificador predictivo sin pérdidas para una imagen con una componente. El predictor combina los valores de hasta tres píxeles vecinos (A , B y C) para predecir el valor del píxel X según la configuración que se muestra en la figura 6. Ahora esta predicción se resta del valor real para el píxel X y la diferencia se codifica sin pérdidas usando

Valor de selección	Predicción
0	sin predicción
1	A
2	B
3	C
4	$A+B-C$
5	$A+(B-C)/2$
6	$B+(A-C)/2$
7	$(A+B)/2$

Tabla 1: Predictores para la codificación sin pérdidas.

el codificador de entropía: codificador Huffman o codificador aritmético. Se puede elegir cualquiera de los predictores de los que se recogen en el cuadro 1 en función del valor de selección.

La selección 1, 2 y 3 del cuadro 1 son predictores unidimensionales, mientras que la selección 4, 5, 6 y 7 son predictores bidimensionales. La selección 0 sólo se puede utilizar para el codificador diferencial en el modo de funcionamiento jerárquico. La codificación de entropía es casi idéntica a la que se emplea para el coeficiente DC en el codificador basado en la DCT como se explica más adelante para el codificador Huffman.

El modo de operación sin pérdidas tiene dos codecs diferentes dependiendo del método de codificación de entropía empleado: codificación Huffman y codificación aritmética. Pueden comprimir imágenes para varios tipos de precisión de la imagen fuente: de 2 a 16 bits. Además, pueden utilizar cualquiera de los predictores del cuadro 1 excepto la selección 0. La tasa de compresión lograda para imágenes en color y con escenas con complejidad moderada es de 2:1.

6 Imágenes con Múltiples Componentes

Uno de los objetivos de JPEG es comprimir imágenes en color o con múltiples componentes. La propuesta JPEG debe permitir comprimir una gran variedad de formatos de imágenes fuente.

6.1 Formatos de Imagen Fuente

En la figura 7 se ilustra el modelo de imagen fuente. Una imagen fuente contiene de 1 a 255 componentes de imagen, a menudo denominadas bandas o colores espectrales o canales. Cada componente consiste en una matriz de pixels. Un pixel viene definido mediante un entero sin signo con precisión de P bits en el rango $[0, 2^P - 1]$. Todos los pixels de todas las componentes de la

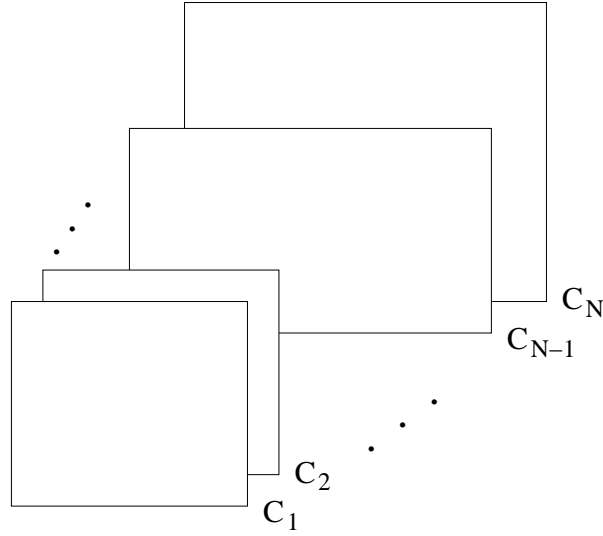


Figura 7: Imagen fuente con múltiples componentes.

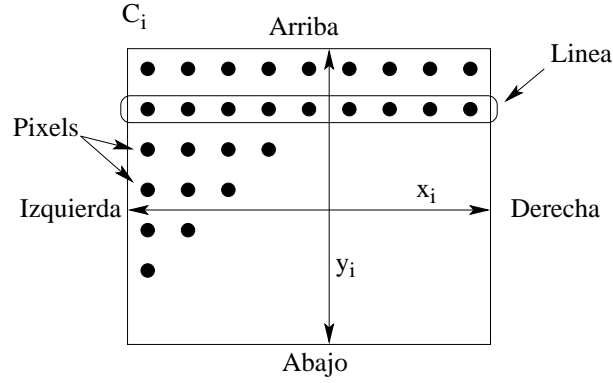


Figura 8: Características de cada componente de la imagen fuente.

imagen fuente tienen la misma precisión de P bits. P puede ser 8 ó 12 para los codecs basados en la DCT y de 2 a 16 para los codecs predictivos.

La componente i de la imagen fuente tiene dimensiones x_i e y_i . Para el poder comprimir imágenes en las que se han muestreado algunas componentes de la imagen a tasas diferentes que otras, cada componente puede tener dimensiones diferentes. Las dimensiones deben estar relacionadas mediante números enteros H_i y V_i , es decir, se deben especificar los factores relativos de muestreo. La dimensión de la imagen fuente X e Y se define como el valor máximo de las dimensiones x_i e y_i de las componentes y pueden tomar cualquier valor hasta 2^{16} . Los factores H_i y V_i sólo pueden tomar valores enteros de 1 a 4. Los parámetros que se codifican son las dimensiones X e Y y los factores H_i y V_i . El decodificador reconstruye las dimensiones x_i e y_i de cada componente mediante la expresión:

$$x_i = \text{Ceil} \left(X \cdot \frac{H_i}{H_{\max}} \right) \quad (6)$$

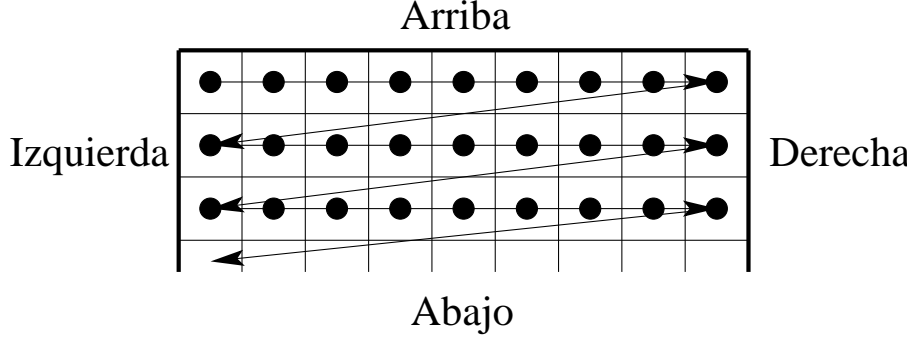


Figura 9: Orden de los datos sin entrelazado.

$$y_i = \text{Ceil} \left(Y \cdot \frac{V_i}{V_{\max}} \right) \quad (7)$$

6.2 Orden de Codificación y Entrelazado

Un estándar de compresión de imagen debe decir cómo los sistemas deben manejar los datos durante el proceso de descompresión. Muchos sistemas de visualización e impresión necesitan disponer de todas las componentes de la imagen al mismo tiempo. Esto sólo es posible si las componentes se entrelazan dentro de la imagen comprimida.

Para que se pueda aplicar el mismo mecanismo de entrelazado tanto para codecs basados en la DCT como para codecs predictivos, el estándar JPEG ha definido el concepto de unidad de datos. Una unidad de datos es un pixel en el caso de codecs predictivos y un bloque 8×8 para los codecs basados en la DCT.

El orden en el que se colocan las unidades de datos en la imagen comprimida es una generalización del patrón de escaneado de línea. Generalmente, las unidades de datos se ordenan de izquierda a derecha y de arriba a abajo según la figura 9. Si un componente de la imagen no se entrelaza con otros componentes, las unidades de datos comprimidos siguen el orden de la figura 9.

Cuando se entrelazan dos o más componentes, cada componente C_i se divide en regiones rectangulares de H_i y V_i unidades de datos, como se muestra en el ejemplo de la figura 10. Las regiones se ordenan dentro de cada componente de izquierda a derecha y de arriba a abajo. Dentro de cada región las unidades de datos se ordenan de izquierda a derecha y de arriba a abajo. JPEG define el término MCU (Minimum Coded Unit) como el grupo más pequeño de unidades de datos entrelazadas.

Los datos entrelazados son una secuencia ordenada de MCUs. Las unidades de datos contenidas en cada MCU vienen determinadas por el número de componentes entrelazados y sus correspondientes factores de muestreo relativos. El número máximo de componentes entrelazadas es 4 y el número máximo de unidades de datos por cada MCU es de 10. Esto se puede expresar según la ecuación:

$$\sum_i H_i \cdot V_i \leq 10 \quad (8)$$

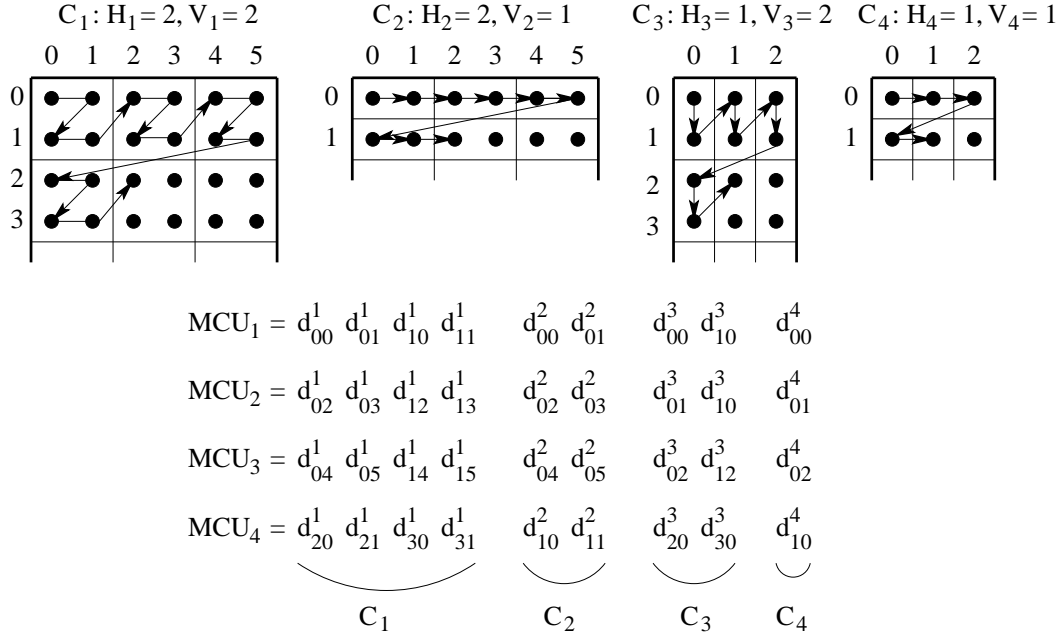


Figura 10: Ejemplo de orden los de datos en el entrelazado generalizado.

donde el índice i representa cada las componente entrelazada. Debido a esta restricción no se pueden entrelazar todas las posibles combinaciones de 4 componentes. Hay que tener en cuenta que JPEG permite que algunas componentes estén entrelazadas, mientras que otras no.

6.3 Múltiples tablas

Los codecs JPEG deben controlar la aplicación de las tablas apropiadas a las componentes apropiadas de la imagen. Se debe usar la misma tabla de cuantificación y la misma tabla de codificación de entropía (o el mismo conjunto de tablas) para codificar todos los pixels de la misma componente de la imagen.

Los decodificadores JPEG pueden almacenar hasta 4 tablas de cuantificación diferentes y hasta 4 (conjuntos de) tablas diferentes de codificación de entropía simultáneamente. La codificación secuencial básica es una excepción: sólo puede almacenar hasta 2 (conjuntos de) tablas de codificación de entropía. Va a ser necesario conmutar entre diferentes tablas durante la descompresión si se utilizan varios componentes entrelazados para aplicar las tablas apropiadas a las componentes apropiadas de la imagen. Durante la descompresión de una componente (o componentes entrelazadas) no se pueden cargar tablas, pero sí cuando se termina de descomprimir una componente y se pasa a la siguiente (caso no entrelazado).

7 Codecs Secuenciales: Básico y Otros Basados en la DCT

El modo de funcionamiento secuencial basado en la DCT consiste en la etapas de FDCT y de cuantificación, y el control de múltiples componentes si es el caso. Además del codec secuencial básico, se definen otros codecs secuenciales basados en la DCT para permitir diferentes precisiones de la imagen fuente (8 ó 12 bits) y los dos métodos diferentes de codificación de entropía: Huffman y aritmética.

La codificación secuencial básica está diseñada para imágenes fuente con 8 bits de precisión y emplea codificación de entropía de Huffman. Además se diferencia de otros tipos de codecs secuenciales basados en la DCT en que sólo puede almacenar dos conjuntos de tablas de códigos Huffman (una tabla para los coeficientes AC y otra para el coeficiente DC para cada conjunto). Para imágenes con tres o cuatro componentes se tienen que utilizar al menos uno de los conjuntos de tablas de códigos Huffman para codificar dos de las componentes. Como ya hemos dicho esto no representa ninguna restricción para el caso no entrelazado puesto que siempre se pueden cargar tablas nuevas antes de comenzar a descomprimir cada componente.

Para muchas aplicaciones que necesiten entrelazar tres componentes esto no representa una restricción demasiado seria. Los sistemas de color (HSI, YUV, etc.) que representan la información cromática (de color) en dos componentes y la información acromática (intensidad) en una tercera componente son más mucho más eficientes que el sistema RGB. Se puede utilizar un conjunto de tablas de Huffman para la componente acromática y otro conjunto para las componentes de crominancia. La estadística de los coeficientes DCT para ambas componentes de crominancia son similares para la mayor parte de las imágenes por lo que un conjunto de tablas de Huffman funcionarán prácticamente igual que dos.

Tras la etapa de cuantificación normalmente suele pocos coeficientes distintos de cero. La tarea del codificador de entropía es codificar esta información de la forma más eficiente posible. La codificación se realiza en dos etapas como ya dijimos: la conversión de los coeficientes DCT cuantificados en una secuencia de símbolos intermedios y la asignación de códigos Huffman de longitud variable a estos símbolos.

7.1 Representación de los Símbolos Intermedios

En la secuencia de símbolos intermedios se representa cada coeficiente AC diferente de cero en combinación con el número de coeficientes AC consecutivos nulos que lo preceden en la secuencia en zig-zag. A este número de coeficientes nulos se lo denomina RL (Runlength). Cada combinación de RL y coeficiente AC distinto de cero se lo representa como un par de símbolos:

		TAMAÑO							
		0	1	2	.	.	.	9	10
RL	0	EOB							
	1	X							
	.	X	Valores						
	.	X	RL+TAMAÑO						
	.	X							
	14	X							
	15	ZRL							

Tabla 2: Estructura del Símbolo-1.

Símbolo-1	Símbolo-2
(RL, TAMAÑO)	(AMPLITUD)

Símbolo-1 representa dos entidades de información denominadas RL y TAMAÑO. Símbolo-2 representa una entidad de información denominada AMPLITUD, que es simplemente la amplitud del coeficiente AC distinto de cero. El RL es el número de coeficientes AC nulos consecutivos que preceden al coeficiente no nulo en la secuencia en zig-zag. El TAMAÑO representa el número de bits utilizados para representar la AMPLITUD, esto es, para codificar el símbolo-2.

El RL representa el número de coeficientes AC nulos y puede tomar cualquier valor entre 0 y 15. Sin embargo, la secuencia de coeficientes AC nulos puede ser mayor que 15, de forma que el Símbolo-1 con valor (15,0) se utiliza para representar longitudes de secuencias de tamaño 16. Puede haber hasta 3 extensiones de valor (15,0) consecutivas antes del Símbolo-1 de terminación que completa la longitud real de la secuencia de coeficientes AC nulos. El Símbolo-1 de terminación siempre le sigue un único Símbolo-2, excepto en el caso en el que la secuencia de ceros incluya el último coeficiente AC (el número 63). Para este caso (bastante frecuencia), el Símbolo-1 especial (0,0) se utiliza como EOB (End of Block).

Para cada bloque de 8×8 pixels, la secuencia en zig-zag de los 63 coeficientes AC se representa mediante una secuencia de: pares de símbolos Símbolo-1 y Símbolo-2, hasta 3 extensiones de la secuencia de ceros seguido del Símbolo-1 de terminación y el Símbolo-2, y simplemente un Símbolo-1 en el caso de EOB.

El rango posible de los coeficientes AC cuantificados determina al rango de los valores de TAMAÑO y AMPLITUD que los representan. Un análisis de la ecuación de la FDCT muestra que si el bloque de 64 elementos de entrada (8×8) se representa con N bits, entonces la parte no fraccionaria de la salida (los coeficientes DCT) puede crecer hasta $N + 3$ bits. Este será por tanto el

TAMAÑO	AMPLITUD	Código VLI
1	-1,1	0,1
2	-3,-2,2,3	00,01,10,11
3	-7,...,-4,4,...,7	000,...,011,100,...,111
4	-15,...,-8,8,...,15	0000,...,0111,1000,...,1111
5	31,...,-16,16,...,31	00000,...,01111,10000,...,11111
6	63,...,-32,32,...,63	000000,...,011111,100000,...,111111
7	-127,...,-64,64,...,127	.
8	-255,...,-128,128,...,255	.
9	-511,...,-256,256,...,511	.
10	-1023,...,-512,512,...,1023	.

Tabla 3: Estructura del Símbolo-2 y VLIs.

valor mayor de los coeficientes de la DCT cuantificados, cuando se utiliza el tamaño de escalón más pequeño posible, que era la unidad.

En la codificación secuencial básica se utiliza una imagen fuente con 8 bits por lo que los valores a la entrada de la FDCT puede variar en el rango $[-2^7, 2^7 - 1]$. Entonces la parte no fraccionaria de los coeficientes AC puede variar en el rango $[-2^{10}, 2^{10} - 1]$. Se utiliza para el Símbolo-2 una codificación de entero con signo con longitud de 1 a 10 bits. Así TAMAÑO es un entero que va desde 1 hasta 10. Como ya hemos dicho RL puede variar de 0 a 15. En los cuadros 2 y 3 se ilustra la representación utilizada para los símbolos Símbolo-1 y Símbolo-2.

La representación intermedia del coeficiente DC diferencial se estructura de forma similar. Sin embargo, el Símbolo-1 sólo representa la información de TAMAÑO y el Símbolo-2 de AMPLITUD. La información RL carece de sentido en el caso del coeficiente DC. Es decir, la representación es:

$$\begin{array}{cc}
\text{Símbolo-1} & \text{Símbolo-2} \\
(\text{TAMAÑO}) & (\text{AMPLITUD})
\end{array}$$

Debido a que el coeficiente DC se codifica de forma diferencial su rango de variación es el doble que para los coeficientes AC, es decir, $[-2^{11}, 2^{11} - 1]$, por lo que será necesario añadir una fila más al final del cuadro 3. En este caso el TAMAÑO del Símbolo-1 puede tomar cualquier valor entre 1 y 11.

7.2 Codificación de Entropía de Longitud Variable

Una vez que la secuencia de coeficientes DCT del bloque 8×8 se ha representado utilizando una secuencia símbolos intermedios según se ha dicho, a

TAMAÑO	VLC
0	00
1	010
2	011
3	100
4	101
5	110
6	1110
7	11110
8	111110
9	1111110
10	11111110
11	111111110

Tabla 4: VLCs para el TAMAÑO del coeficiente DC.

continuación se asignarán códigos de longitud variable. Para cada bloque 8×8 , la representación intermedia para el coeficiente DC se codifica y se presenta a la salida en primer lugar.

Tanto para el coeficiente DC como para los AC, cada Símbolo-1 se codifica con un VLC (Variable Length Code) del conjunto de tablas de Huffman asignadas a la componente de la imagen correspondiente. Cada Símbolo-2 se codifica con un código VLI (Variable Length Integer) cuya valor se puede ver en el cuadro 3. Los VLCs y VLIs son códigos con longitud variable, pero los VLIs no son códigos Huffman. Una diferencia importante es que la longitud de un VLC (código Huffman) no se conoce hasta que no se decodifica, mientras que la longitud de un VLI está almacenada en el VLC que le precede. Los VLCs deben venir especificados externamente por la aplicación (usuario) como entrada al codec. En el cuadro 4 se recogen los VLCs para el coeficiente DC y en el cuadro 5 se recogen los primeros VLCs para los coeficientes AC.

7.3 Otros Codecs Secuenciales DCT

El codec que soporta imágenes fuente con 12 bits con codificación de entropía de Huffman es una extensión directa del método de codificación descrito. En este caso los coeficientes cuantificados se deben pueden extender hasta 4 bits, por lo que los valores del TAMAÑO y AMPLITUD de los códigos intermedios habrá que ampliarlos adecuadamente. Existen también versiones de 8 y 16 bits que emplean codificación de entropía aritmética.

(RL,TAMAÑO)	VLC
(0,0)	1010
(0,1)	00
(0,2)	01
(0,3)	100
(0,4)	1011
(0,5)	11010
(0,6)	1111000
(0,7)	11111000
(0,8)	1111110110
(0,9)	1111111110000010
(0,10)	1111111110000011
(1,1)	1100
(1,2)	11011
(1,3)	1111001
(1,4)	111110110
(1,5)	11111110110
(1,6)	1111111110000100
(1,7)	1111111110000101
(1,8)	1111111110000110
(1,9)	1111111110000111
(1,10)	1111111110001000
(2,1)	11100
(2,2)	11111001
(2,3)	1111110111
(2,4)	111111110100
(2,5)	1111111110001001
(2,6)	1111111110001010
(2,7)	1111111110001011
(2,8)	1111111110001100
(2,9)	1111111110001101
(2,10)	1111111110001110

Tabla 5: Primeros VLCs para el Símbolo-2 de los coeficientes AC.

8 Modo DCT Progresivo

El modo DCT progresivo tiene las mismas etapas de FDCT y de cuantificación que el modo secuencial. La diferencia estriba en que cada componente de la imagen se codifica en varias pasadas en lugar de una única. La primera pasada codifica una versión de mala calidad, pero reconocible, de la imagen que puede transmitirse rápidamente en comparación con la imagen comprimida entera. Las sucesivas pasadas van añadiendo más detalles a la imagen aumentando su calidad hasta el nivel deseado.

Para lograr este objetivo, es necesario añadir un buffer a la salida del cuantificador antes de pasar los coeficientes cuantificados al codificador. El tamaño del buffer debe ser suficiente como para almacenar los coeficientes cuantificados de toda la imagen (que como mucho se podrán almacenar con 3 bits más que la imagen original). Cada vez que se cuantifican los coeficientes de cada bloque se almacenan en el buffer. Cuando ya están almacenados todos los coeficientes se van codificando parcialmente en varias pasadas.

Hay dos métodos complementarios para codificar parcialmente los coeficientes almacenados en el buffer:

- En el primer método sólo una banda especificada de la secuencia en zig-zag se codifica en cada pasada. Este procedimiento se denomina selección espectral debido a que cada banda típicamente contiene aquellos coeficientes que ocupan una zona de las bajas a las altas frecuencias del espectro en frecuencia del bloque 8×8 .
- En el segundo método los coeficientes se van codificando con una precisión mayor en cada pasada hasta llegar a la precisión final. En la primera pasada sólo los bits más significativos de los coeficientes se codifican. En pasadas sucesivas se van codificando el resto de bits menos significativos. Este procedimiento se denomina aproximación sucesiva.

Ambos métodos se puede utilizar por separado y mezclados en ciertas combinaciones.

9 Modo Jerárquico

El modo jerárquico proporciona una codificación piramidal de la imagen a múltiples resoluciones, cada una de las cuales se diferencia de la codificada anterior o posteriormente en un factor de 2 horizontalmente, verticalmente o ambos. El procedimiento de codificación se puede resumir como sigue:

- Filtrado y submuestreo de la imagen original por el número adecuado de múltiplos de 2 en cada dimensión.
- Codificar esta imagen de tamaño reducido utilizando uno de los métodos vistos: DCT secuencial, DCT progresivo o sin pérdidas.

Imagen Número	Tasa de compresión
1	42:1
2	32:1
3	26:1
4	22:1
5	18:1
6	16:1
7	14:1
8	8:1
9	2:1

Tabla 6: Tasas de compresión JPEG para la radiografía de la mano.

- Decodificar esta imagen de tamaño reducido e interpolarla y sobremuestrearla por un factor de 2 horizontalmente, verticalmente o en ambas direcciones, utilizando la misma técnica que se empleará en el decodificador.
- Utilizar esta versión sobremuestreada como una predicción de la imagen a la nueva resolución y codificar la imagen diferencia utilizando uno de los métodos vistos: DCT secuencial, DCT progresivo o sin pérdidas.
- Repetir los dos pasos anteriores hasta que se alcance la resolución final de la imagen.

Los pasos 2 y 4 se pueden hacer utilizando para cada componente:

- Sólo codificación DCT.
- Sólo codificación sin pérdidas.
- Codificación DCT excepto para la última resolución que se utilizará codificación sin pérdidas.

La codificación jerárquica es útil en aplicaciones en las que se tiene que presentar una imagen con mucha resolución en un dispositivo con resolución menor o variable.

10 Ejemplo de Compresión JPEG

En la figura 11 podemos ver la una región de interés de una radiografía de una mano comprimida con JPEG para varias calidades y tasas de compresión. Vamos a numerar las imágenes de izquierda a derecha y de arriba a abajo. Están ordenadas de menor calidad y mayor tasa de compresión a mayor calidad y menor tasa de compresión. La última imagen corresponde a JPEG sin

pérdidas mientras que las otras 8 son para JPEG secuencial DCT básico con diferente precisión en los coeficientes de la DCT. En el cuadro 6 se recogen las tasas de compresión logradas redondeadas al entero más próximo.

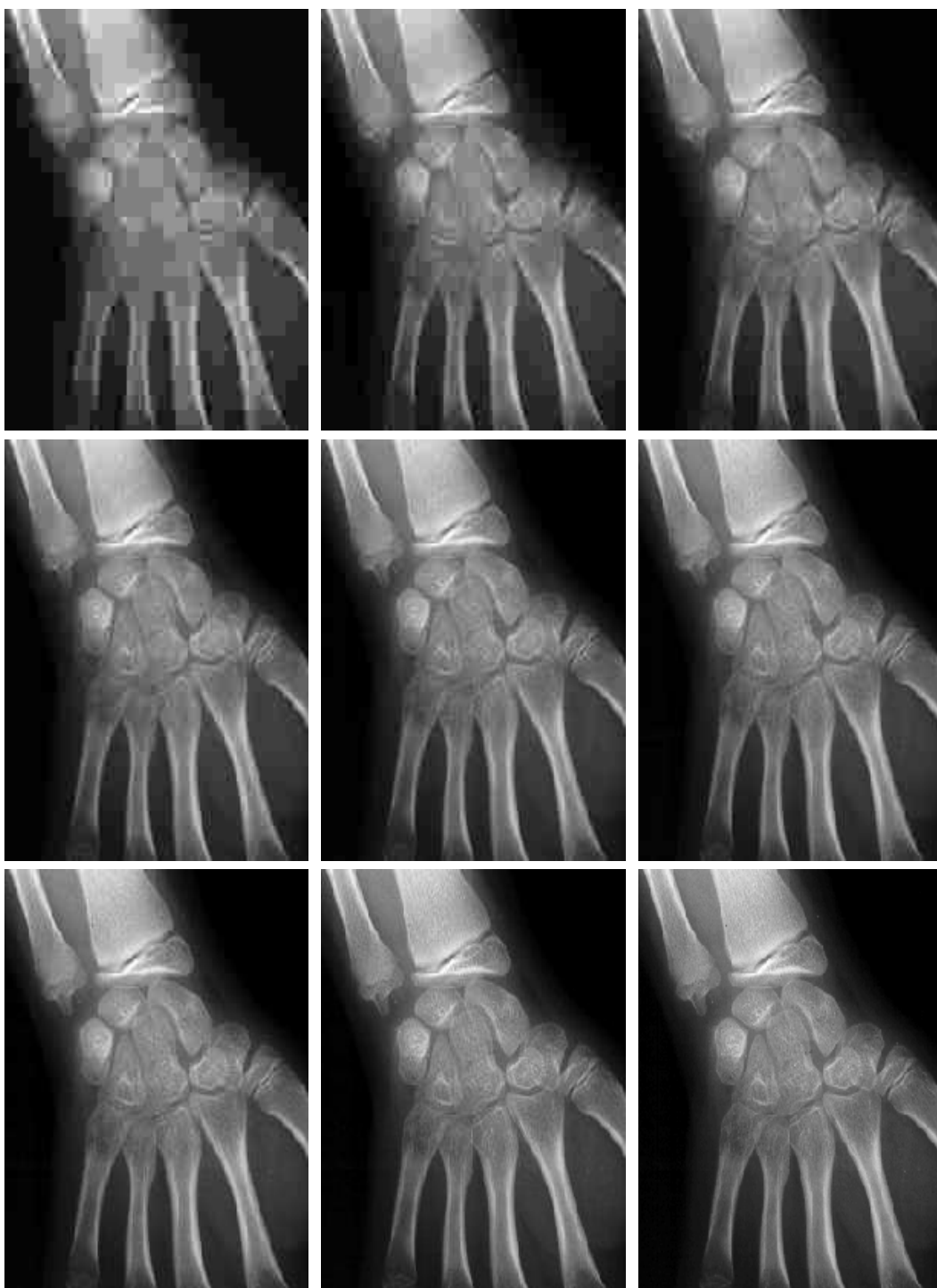


Figura 11: Radiografía de una mano comprimida con JPEG con varias calidades y tasas de compresión.