



JPEG 2000

Técnicas informáticas de Imagen y Sonido

ÍNDICE

1. Definición y orígenes	3
2. Características	4
3. Proceso de compresión y descompresión	6
4. Transformación	7
5. Cuantificación	27
6. Codificación	29
7. JPEG 2000 vs JPEG	44
8. Problemática	51
9. Bibliografía	53

DEFINICIÓN Y ORÍGENES

- JPEG2000 es una norma de compresión creada en 1999 por el Joint Photographics Experts Group (JPEG) como alternativa mejorada de la norma de compresión JPEG, usada desde 1992.
- Figura como norma ISO (ISO/IEC 15444-1).

CARACTERÍSTICAS

- Transmisión progresiva y descompresión por calidad, resolución, localización espacial o por componentes a partir de una misma imagen fuente comprimida (***Comprime una vez, descomprime muchas***). El compresor marca la resolución máxima.

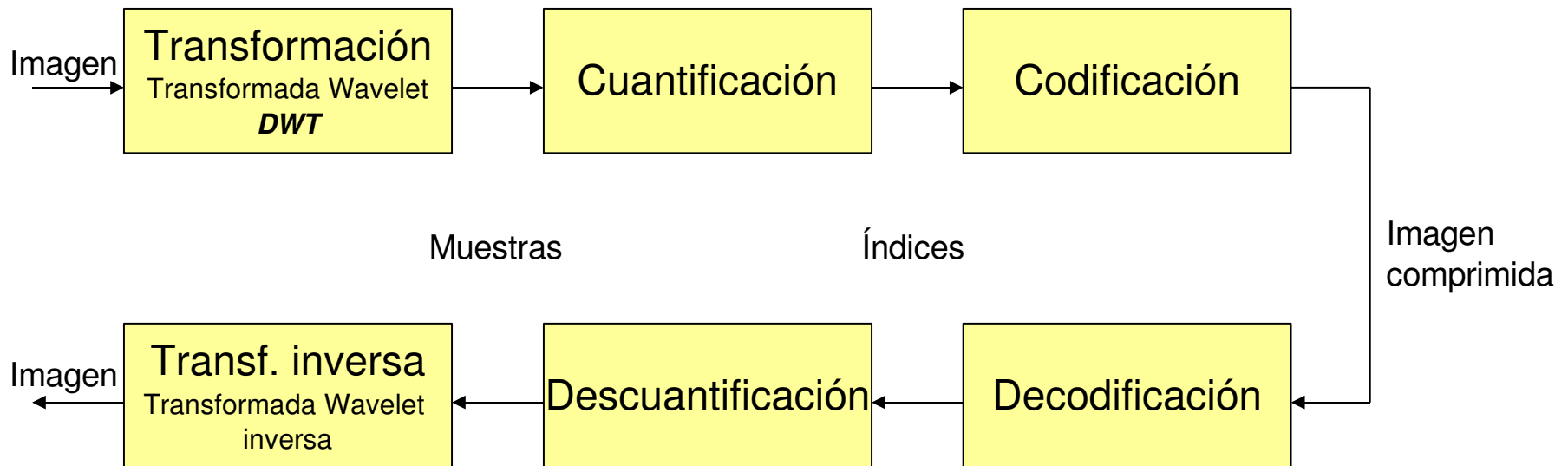
Es ideal para ir viendo la imagen mientras se descarga, sobre todo en conexiones lentas y para descargar imágenes aptas para dispositivos de bajas prestaciones.

- Más calidad en la imagen al comprimirla a ratios bajos.

CARACTERÍSTICAS

- Permite comprimir de forma Lossy (Pérdida de calidad con respecto al original) o Lossless (Sin pérdida de calidad al descomprimir)
- Robusto frente a errores en los bits producidos por el ruido de los canales de comunicaciones.
- Acceso y procesado aleatorios, permitiendo el acceso espacial sin descomprimir la imagen.

PROCESO DE COMPRESIÓN Y DESCOMPRESION



TRANSFORMACIÓN

El proceso de transformación tiene el objetivo de preparar las muestras de la imagen a comprimir para que puedan ser cuantificadas y codificadas, capturando las dependencias estadísticas y separando la información relevante de la irrelevante.

JPEG 2000 sigue los siguientes pasos:

1. Desplazamiento de las muestras
2. Transformada de color
3. DWT (Transformada Wavelet Discreta)

TRANSFORMACIÓN

1. Desplazamiento de las muestras:

Es necesario situar las muestras en un rango que contenga al 0 como centro, si es necesario, para realizar la transformada de color. El desplazamiento se realiza independientemente en cada uno de los componentes sólo si lo necesita.

Si B es el número de bits de las muestras de un componente y están en un rango $0 < x_n(n_1, n_2) \leq 2^B$, se le restaría 2^{B-1} a cada muestra del componente, pasando a estar en el rango $2^{B-1} \leq x_n(n_1, n_2) \leq 2^{B-1}$.

TRANSFORMACIÓN

2. Transformada de color:

Se trata de aprovechar la redundancia de colores que se produce entre muestras próximas consiguiendo un espacio de color adecuado para la cuantificación.

Sólo se puede llevar a cabo cuando:

- La imagen tiene tres componentes o más. JPEG 2000 no obliga que los 3 primeros componentes sean los de RGB, pueden tener otro formato.
- Los tres primeros componentes tienen el mismo tamaño y sus muestras tienen el mismo número de bits.

TRANSFORMACIÓN

2. Transformada de color:

JPEG 2000 soporta los siguientes tipos de transformadas de color:

- Irreversible (*Irreversible Component Transformation, ICT*)
- Reversible (*Reversible Component Transformation, RCT*)
- No aplicar la transformación.

TRANSFORMACIÓN

2. Transformada de color:

Irreversible:

Sólo debe ser usada con la transformada wavelet irreversible.

Los valores de iluminancia (Y) y crominancia (C_b y C_r) se consiguen mediante la siguiente relación:

$$\begin{pmatrix} Y \\ C_b \\ C_r \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.16875 & -0.33126 & 0.5 \\ 0.5 & -0.41869 & 0.08131 \end{pmatrix} \cdot \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

TRANSFORMACIÓN

2. Transformada de color:

Reversible:

Sólo debe ser usada con la transformada wavelet reversible.

La RCT puede ser vista como una aproximación de la transformación YUV, similar a la YC_bC_r , pero que puede ser invertida casi exactamente.

$$\begin{pmatrix} Y \\ V \\ U \end{pmatrix} = \begin{pmatrix} \frac{R + 2G + B}{4} \\ R - G \\ B - G \end{pmatrix}$$

TRANSFORMACIÓN

2. Transformada de color:

Losas (Tiles):

Cada componente de color de la imagen es particionada en losas rectangulares no solapadas. Pueden tener cualquier tamaño, incluso la imagen entera, aunque todas las losas de un mismo color tienen el mismo tamaño excepto las de los bordes.

Cada losa se comprime individualmente.

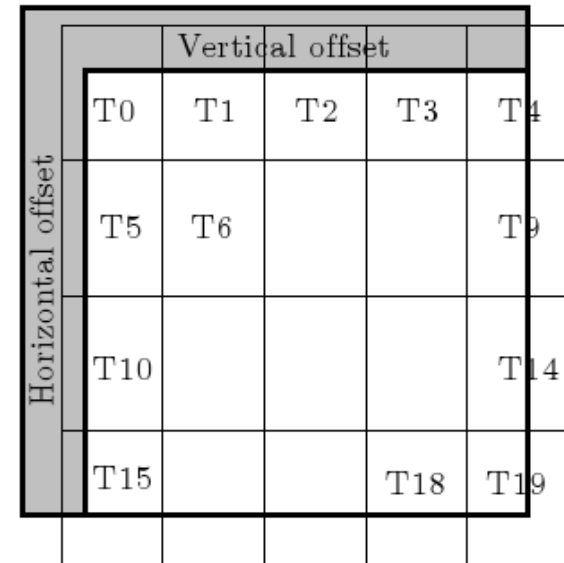
TRANSFORMACIÓN

2. Transformada de color:

Losas (Tiles):

JPEG 2000 permite que la imagen tengan offsets vertical y horizontal.

El particionado en losas tiene como objetivo el poder utilizar la característica de *Región de interés*, para poder descomprimir partes de la imagen sin tener que descomprimir toda la imagen.



Ejemplo de particionado

TRANSFORMACIÓN

3. DWT

Limitaciones de la transformada de Fourier:

Al pasar al dominio de la frecuencia, se pierde mucha información si la señal no es estacionaria (Cuando los parámetros estadísticos de la señal no constantes sobre el tiempo), como cambios bruscos, rupturas, etc.

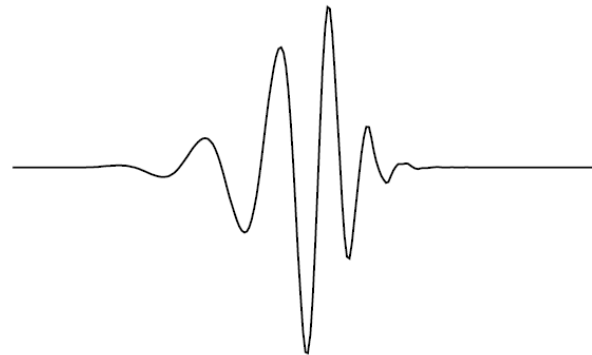
Es por ello por lo que se decide utilizar Wavelets en JPEG 2000.

TRANSFORMACIÓN

3. DWT

Wavelet:

Un wavelet es una onda limitada en tiempo (duración finita) con un valor medio de 0. Posee una forma irregular y asimétrica.



Ejemplo de wavelet

TRANSFORMACIÓN

3. DWT

El análisis con el uso de wavelets consiste en descomponer la señal a analizar en versiones de la wavelet, desplazadas en el tiempo y escaladas. Esta es la CWT (Transformada Wavelet Continua).

Los coeficientes se obtienen al comparar todas estas versiones con la señal a analizar. Estos coeficientes definen cuanto se parece la señal a la wavelet. Cuanto más se parezca la wavelet, mayor será el valor del coeficiente correspondiente.

TRANSFORMACIÓN

3. DWT

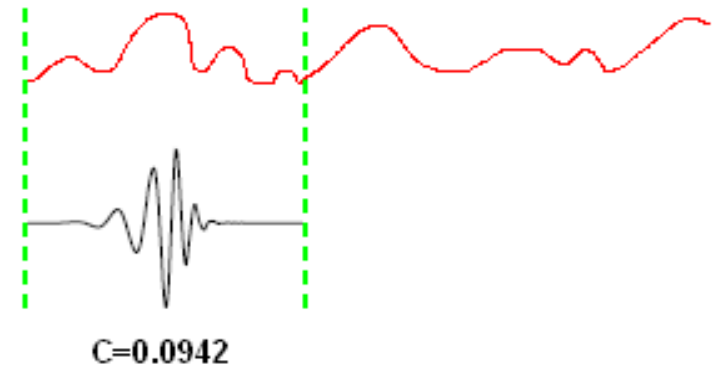
Los pasos de la CWT son los siguientes:

1. Comparar la wavelet con un intervalo al principio de la señal. Comer esto y lo otro.
2. Calcular C , que representa cuanto se parece la wavelet y el intervalo escogido.
3. Desplazar la wavelet a derecha y repetir los pasos 1,2 y 3 hasta completar toda la señal.
4. Estirar la wavelet y volver a repetir los pasos 1,2, 3 y 4, hasta que la wavelet sea tan alargada como la señal.

TRANSFORMACIÓN

3. DWT

Existe una correspondencia entre la escala del wavelet y el análisis de las frecuencias que se extraen de los coeficientes:



Comparación con la Wavelet

Escala baja → Wavelet encogida → Detalles → Alta frecuencia.

•Escala alta → Wavelet estirada → Visión general → Baja frecuencia

TRANSFORMACIÓN

2. Transformada Wavelet Discreta:

Existen dos tipos de transformadas wavelets:

- Wavelet de punto flotante (Floating-Point Wavelet)
- Wavelet de enteros (Integer Wavelet)

Ambas permiten transmisión progresiva, pero sólo la Wavelet de enteros es reversible, por lo que es usada para la compresión lossless.

TRANSFORMACIÓN

2. Transformada Wavelet Discreta:

Denotamos una fila de pixels de una losa como P_k, P_{k+1}, \dots, P_m . Por la naturaleza de las funciones wavelets que usa JPEG 2000, los pixeles menores que k y mayores que m quizás no tengan que ser usados. Por lo tanto, antes de aplicar la transformada, los pixeles de la losa deban de ser extendidos.

Para ello, JPEG 2000 utiliza el método de la *extensión simétrica periódica*.

TRANSFORMACIÓN

2. Transformada Wavelet Discreta:

Cálculo de la transformada Wavelet de enteros:

Esta transformación calcula los coeficientes $C(i)$, calculando primero los valores impares $C(2i+1)$, y después los valores pares a partir de los impares. Estas son las fórmulas:

$$C(2i+1) = P(2i+1) - [(P(2i) + P(2i+2)) / 2], \text{ para } k-1 \leq 2i+1 < m+1$$

$$C(i) = P(2i) + [(C(2i-1) + C(2i+1) + 2) / 4], \text{ para } k \leq 2i < m+1$$

TRANSFORMACIÓN

2. Transformada Wavelet Discreta:

Cálculo de la transformada Wavelet de punto flotante:

Consta de 6 pasos de cálculo:

1. $C(2i+1) = P(2i+1) + \alpha[P(2i) + P(2i+2)]$, para $k-3 \leq 2i+1 < m+3$
2. $C(2i) = P(2i) + \beta[C(2i-1) + C(2i+1)]$, para $k-2 \leq 2i < m+2$
3. $C(2i+1) = C(2i+1) + \gamma[C(2i) + C(2i+2)]$, para $k-1 \leq 2i+1 < m+1$
4. $C(2i) = C(2i) + \delta[C(2i-1) + C(2i+1)]$, para $k \leq 2i < m$
5. $C(2i+1) = -K \times C(2i+1)$, para $k \leq 2i + 1 < m$
6. $C(2i) = (1/K) \times C(2i)$, para $k \leq 2i < m$

TRANSFORMACIÓN

2. Transformada Wavelet Discreta:

Cálculo de la transformada Wavelet de punto flotante:

Las cinco constantes anteriores son los coeficientes de filtrado que utiliza JPEG 2000 para la transformada Wavelet. Sus valores son:

- $\alpha = -1.586134342$
- $\beta = -0.052980118$
- $\gamma = 0.882911075$
- $\delta = 0.443506852$
- $K = 1.230174105$

TRANSFORMACIÓN

2. Transformada Wavelet Discreta:

Estas transformaciones unidimensionales se realizan L veces, donde L es el número de niveles o resoluciones de subbandas. L es un parámetro que se puede definir por el usuario o dejarlo que lo defina el compresor.

El nivel $L-1$ corresponde a la imagen original (Máxima resolución) y el nivel 0 a la subbanda de más baja frecuencia.

CUANTIFICACIÓN

Cuantificación escalar uniforme:

El proceso de cuantificación tiene como objetivo convertir los coeficientes producidos por el proceso de transformación en índices de cuantificación.

La cuantificación produce distorsión, por lo que para la compresión sin pérdidas no se realiza.

Cada subbanda tiene un tamaño de paso de cuantificación diferente por el cual se dividen los coeficientes producidos por la transformación Wavelet y que están en la misma subbanda. Al final, se trunca el resultado.



CUANTIFICACIÓN

El parámetro de cuantificación debe de ser determinado iterativamente a fin de conseguir el bitrate deseado.

CODIFICACIÓN

El proceso de codificación busca preparar la información obtenida de los anteriores procesos, eliminando la redundancia estadística que pudiera haber a fin de minimizar la cantidad de bits sin perder datos.

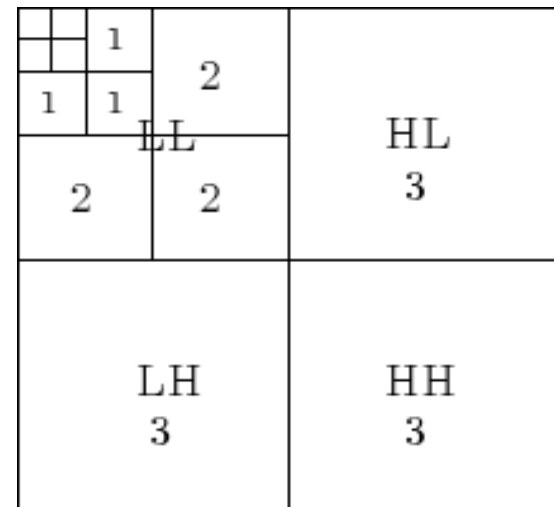
La codificación es un proceso reversible, lo que hace que no haya que diferenciar esta vez si estamos comprimiendo con pérdida o sin pérdida.

CODIFICACIÓN

Bloques de código y precintos:

Antes de pasar a codificar, es necesario crear otras particiones llamadas bloques de código (*code-blocks*) y precintos.

Dada una losa de una componente de color, esta se compone de las diferentes subbandas obtenidas tras realizar la transformada Wavelet. El tamaño total de las subbandas equivale al de la losa.

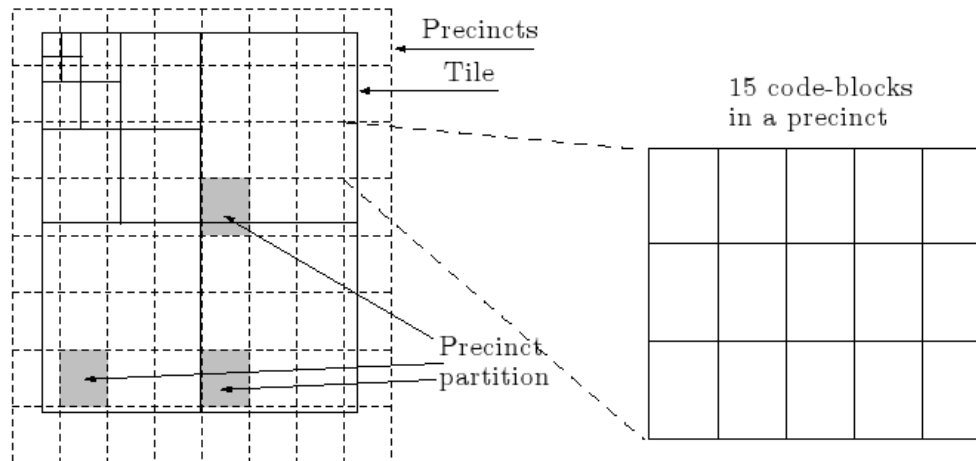


4 niveles de resolución
en una losa

CODIFICACIÓN

Bloques de código y precintos:

Establecemos entonces una serie de rejillas de rectángulos en toda la imagen, que son los precintos.



Las dimensiones (Altura y anchura) de los precintos son potencias de 2.

CODIFICACIÓN

Bloques de código y precintos:

Una partición de precintos de un nivel la constituyen los 3 precintos localizados en la misma región de cada subbanda que forma el nivel.

Cada precinto se compone de 15 bloques de códigos (*code-blocks*), que son las unidades mínimas que se usarán para la codificación. Un bloque de código es un rectángulo de tamaño $2^{x_{cb}} \times 2^{y_{cb}}$, donde $2 \leq x_{cb}, y_{cb} \leq 10$, y $x_{cb} + y_{cb} \leq 12$.

CODIFICACIÓN

Bloques de código y precintos:

Con este particionado en losas, precintos y bloques de códigos conseguimos distintos tamaños de partición, que nos permitirá que se puedan crear implementaciones para codificadores que sean más eficientes a la hora de gestionar la memoria, facilitar el streaming, poder acceder a la región que nos interese y descomprimirla independientemente.

CODIFICACIÓN

Codificación:

Los coeficientes wavelet de un bloque de código se codifican individualmente por plano de bits. Se va haciendo desde el plano de bits más significativos hasta el plano de bits menos significativos.

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

CODIFICACIÓN

Codificación:

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

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

CODIFICACIÓN

Codificación:

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

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

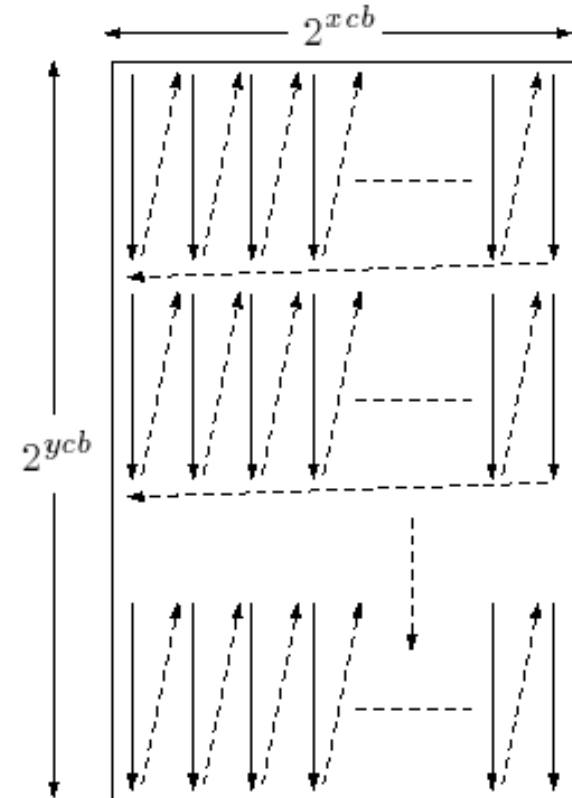
El primer plano de bits más significativo no nulo sólo tendrá que aplicar el 3er paso. El resto, seguirá los 3 pasos.

CODIFICACIÓN

Codificación:

Un plano de bits se recorre de la siguiente manera a la hora de aplicar el proceso, por franjas verticales.

Hay cuatro filas de franjas.



CODIFICACIÓN

Codificación:

Propagación de la significancia:

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

1. El coeficiente wavelet asociado es insignificante (Significancia=0)
2. Al menos uno de los 8 bits vecinos son significantes.

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

CODIFICACIÓN

Codificación:

Propagación de la significancia:

Subsecuentemente, los bits codificados en este paso (y los siguientes dos pasos) consideraran el nuevo valor de la significancia del coeficiente wavelet asociado.

Este paso muestra que al menos es necesario que algunos coeficientes wavelet sean declarados significativos para que haya codificación. Es por ello por lo que el primer plano de bit más significativo sólo hace el 3er paso.

CODIFICACIÓN

Codificación:

Refinamiento de magnitud:

Este segundo paso codifica todos los bits de los coeficientes wavelets que se convirtieron en significativos en el plano de bits anterior.

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

CODIFICACIÓN

Codificación:

Limpieza:

Por último, en este paso se codifican los bits que no se han codificado en los pasos 1 y 2.

Si el bit a codificado vale 1, el coeficiente wavelet correspondiente pasa a tener el coeficiente de significancia a 1.

CODIFICACIÓN

Paquetes y layers:

Una vez que todos los bits de todos los coeficientes de todos los bloques de códigos de una partición de precintos han sido codificados en el flujo de bits correspondiente, se le añade un cabecero a ese flujo con la información necesaria para descodificar. Ese flujo con la cabecera conforma un paquete.

Si todos los bloques de código de un paquete son cero, el cuerpo del paquete estará vacío

CODIFICACIÓN

Paquetes y layers:

Un layer es un conjunto compuesto de un paquete por cada nivel de resolución de la misma partición de precintos. Se trata de un incremento de calidad sobre esa región al máximo.

Por tanto, cada paquete supone el incremento de su nivel de calidad al que corresponde.

JPEG 2000 vs JPEG

En cuanto al nuevo estándar JPEG 2000, ha supuesto grandes mejoras con respecto al antecesor JPEG a la hora de la compresión en modo Lossy (Con pérdida) a ratios bajos.

Algunas de las diferencias más notables entre ambos son:

- Defectos por bloqueo
- Distorsión de color
- Defectos en bordes
- Defectos de definición

JPEG 2000 vs JPEG

Defectos por blocaje:

A la hora de comprimir, JPEG sólo trabaja con bloques de 8x8, causando que se distingan pequeños cuadros:

JPEG 2000 0.5 bits/pixel
Ratio de comp. 1:48



Original



JPEG 0.5 bits/pixel
Ratio de comp. 1:48



JPEG 2000 no lo padece debido a que al realizar la DWT no está limitado a un tamaño fijo de bloque

JPEG 2000 vs JPEG

Defectos por blocaje:

A la hora de comprimir, JPEG sólo trabaja con bloques de 8x8, causando que se distingan pequeños cuadros:

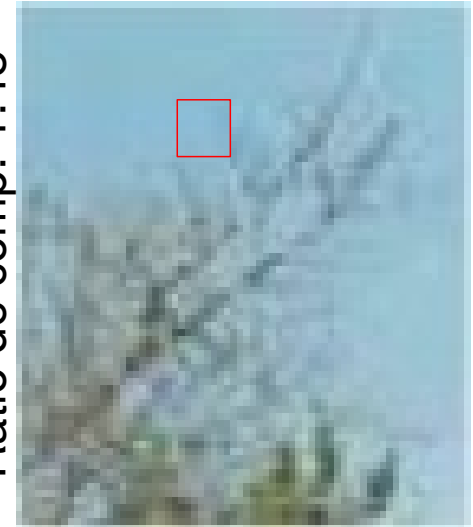
JPEG 2000 0.5 bits/pixel
Ratio de comp. 1:48



Original



JPEG 0.5 bits/pixel
Ratio de comp. 1:48



JPEG 2000 no lo padece debido a que al realizar la DWT no está limitado a un tamaño fijo de bloque

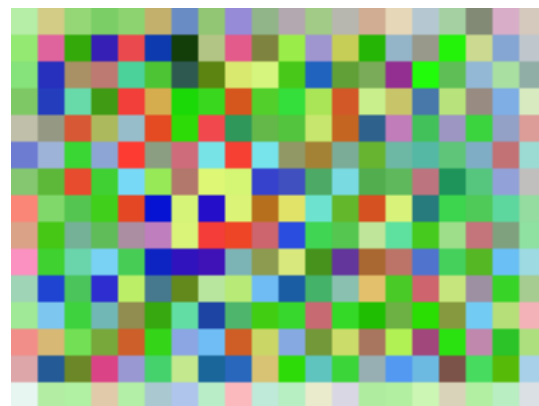
JPEG 2000 vs JPEG

Distorsión de color:

Ambos estándares obtienen distorsión tras hacer la transformada de color cuando hay cambios bruscos de color y brillo en los píxeles cercanos.



JPEG



Original



JPEG 2000

JPEG 2000 lo hace mejor que JPEG en este aspecto.

JPEG 2000 vs JPEG

Defectos en bordes:

JPEG 2000 suaviza más los detalles consiguiendo un mejor resultado que JPEG, que remarca más los bordes pero tiene problemas con las líneas curvas.

not be inferred from
established approach
ts generalizations as
McGill's interaction

JPEG
0,72 bits/pixel

not be inferred from
established approach
ts generalizations as
McGill's interaction

Original
Escala de grises
de 16 niveles a
0,64 bits/pixel

not be inferred from
established approach
ts generalizations as
McGill's interaction

JPEG 2000
0,72 bits/pixel

JPEG 2000 vs JPEG

Defectos de definición:

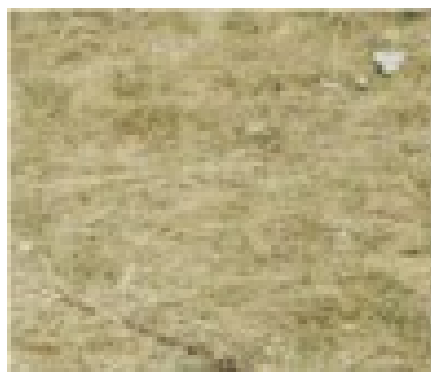
Como se ha visto en los defectos en bordes, JPEG 2000 tiende a suavizar las imágenes perdiéndose bastantes detalles y texturas de la imagen original.



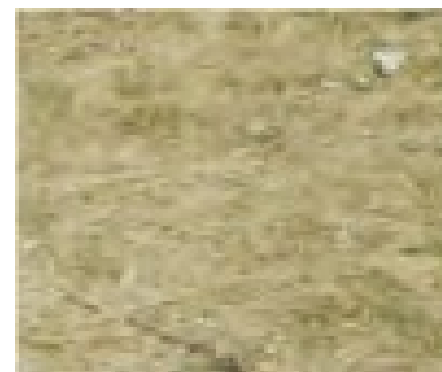
JPEG

2.0 bits/pixel

Ratio de compresión 1:12



Original



JPEG 2000

2.0 bits/pixel

Ratio de compresión 1:12

En este caso, JPEG obtiene mejor definición.

JPEG 2000 vs JPEG

Conclusión:

Como se ha visto, a bajos ratios JPEG 2000 solventa algunos de los problemas que tiene JPEG, como la percepción de los bloques de 8x8, y mejora otros como la distorsión de color.

No obstante, a mayores ratios (≥ 3.0 bits/pixel) no hay diferencias visibles entre ambos estándares.

Por ello, es recomendable comprimir con JPEG 2000 para ratios bajos para obtener mayor calidad y prestaciones.

PROBLEMÁTICA

Pese a que, como hemos visto, JPEG 2000 posee unas grandes ventajas técnicas, ha tenido poquísimo éxito como estándar abierto, sobretodo en internet.

Esto es debido a que sobre el estándar pesan numerosas patentes de software (Algunas “sumergidas”) sobre los algoritmos matemáticos que se utilizan, creando temor a los desarrolladores de software y sobretodo a los desarrolladores de navegadores de encontrarse con estas. Por ello, no dan soporte al formato JPEG2000 y siguen con los formatos clásicos (JPEG, GIF, ...)

PROBLEMÁTICA

La poca aceptación hizo que las compañías y organizaciones contribuyentes establecieran licencias libres para las partes básicas del estándar. El propio comité JPEG advierte sobre ello.

El desarrollador podría obtener problemas legales en caso de no haber pagado todas las licencias de patentes.

Aun así, se pueden encontrar herramientas que permiten su uso como Kakadu.

BIBLIOGRAFÍA

1. *Data compression – The Complete Reference. Fourth Edition.* D. Salomon.
2. *An Overview of JPEG-2000.* M. W. Marcellin, M. J. Gormish, A. Bilgin, M. P. Boliek
3. *Implementación y Prueba del estándar JPEG 2000.* R. Villaverde
4. *Baseline JPEG and JPEG2000 Artifacts Illustrated.* A. Jakaulin - <http://ai.fri.uni-lj.si/~aleks/jpeg/artifacts.htm>