

Compresión de datos con pérdida



INGENIERÍA DE SISTEMAS

Cursada 2017

COMPRESIÓN CON PÉRDIDA

- Se busca eliminar información redundante o menos relevante (tolerando cierta pérdida sin afectar sustancialmente la información)
- Los datos reconstruidos -al descomprimir- son una aproximación de los originales (que puede resultar imperceptible)
- Se alcanzan altas tasas de compresión (a costa de perder calidad)

Ejemplo



Imagen original



Imagen reconstruida (20:1)



Imagen reconstruida (36:1)

REDUNDANCIA EN IMÁGENES

| tipo de redundancia | asociada con |
|--------------------------|---|
| Estadística | frecuencia de aparición de los símbolos (píxeles) |
| Espacial (o inter-píxel) | correlación entre píxeles vecinos (para codificar o predecir otros) |
| Temporal (o inter-frame) | correlación entre cuadros adyacentes de video |
| Psico-visual | información "menos sensible" al sistema visual humano |

Ejemplo



MEDIDAS DE DISTORSIÓN

Si hay pérdida de información → medir error entre datos originales y reconstruidos



imagen original (I)

Imagen descomprimida (I')

Imagen de error

Algunas medidas:

- Error Cuadrático Medio (RMS)

$$e_{RMS} = \frac{1}{N \cdot M} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (I'_{ij} - I_{ij})^2$$

- Error Absoluto promedio

$$e_{abs} = \frac{1}{NM} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} |I'_{ij} - I_{ij}|$$

- Otras: PSNR (Peak Signal-to-Noise Ratio)
SSIM (Structural Similarity Index Metric)

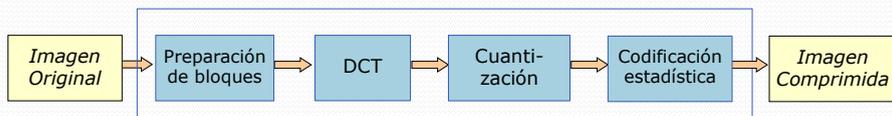
N filas x *M* columnas

...

JPEG

(Joint Photographic Experts Group)

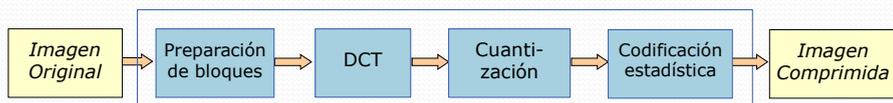
- Estándar de compresión de imágenes fijas (creado en 1992, con numerosas actualizaciones)
- Considera la mayor sensibilidad del ojo humano a:
 - cambios de brillo (luminancia) que de color (crominancia)
 - cambios de brillo en zonas homogéneas que en zonas de bordes entre objetos
- La especificación JPEG consiste de varias partes, incluyendo variantes
 - sin pérdida (modelo predictivo/adaptativo + Huffman)
 - con pérdida (transformación DCT + cuantización + Huffman/cod.aritm.-RLC)



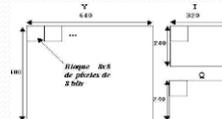
- Estándar JPEG2000 (transformada wavelets, región de interés, mejor adaptación a transmisión progresiva de las imágenes, uso limitado en navegadores web)

COMPRESIÓN:

JPEG



Preparación de bloques: La imagen se divide en bloques de 8x8 píxeles (gris). Si es RGB previamente se convierte al modelo de color YIQ (luminancia-crominancia) y se reducen espacialmente las componentes I y Q (crom.).



Transformación del coseno (DCT): Se procesan cada bloque y en cada posición se obtiene:

$$DCT(i, j) = \frac{1}{\sqrt{2 \cdot N}} \cdot c(i) \cdot c(j) \cdot \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} (pixel(x, y) \cdot \cos\left[\frac{(2x+1) \cdot i \cdot \pi}{2 \cdot N}\right] \cdot \cos\left[\frac{(2y+1) \cdot j \cdot \pi}{2 \cdot N}\right])$$

$$C(x) = \begin{cases} \frac{1}{\sqrt{2}} & \text{si } x = 0 \\ 1 & \text{si } x \neq 0 \end{cases}$$

En el bloque transformado (no se genera pérdida) el primer elemento es el más representativo y los demás pierden importancia a medida que se alejan de él

Cuantización: descarta información no relevante (pérdida):

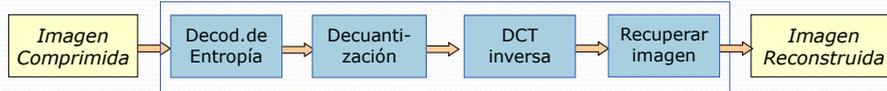
$$Valor_cuantiz(i, j) = \left\lfloor \frac{DCT(i, j)}{Quantum(i, j)} \right\rfloor$$

| | | | | | | | |
|-----|----|----|---|---|---|---|---|
| 150 | 60 | 20 | 4 | 1 | 0 | 0 | 0 |
| 92 | 35 | 10 | 3 | 1 | 0 | 0 | 0 |
| 25 | 10 | 3 | 1 | 0 | 0 | 0 | 0 |
| 7 | 2 | 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Codificación de Entropía: en cada bloque se aplica Huffman/Codif.Aritm. y RLC (para las secuencias de ceros), siguiendo un recorrido zig-zag

DESCOMPRESIÓN:

La decodificación consiste en el proceso inverso (algoritmo simétrico)



Decodificación de Entropía: decodificación Huffman/Codif.Aritm. y RLC

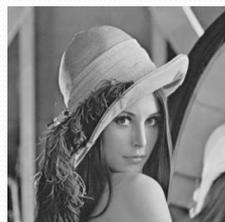
Decuantización: $Valor\ DCT'(i, j) = Valor_cuantiz(i, j) \cdot Quantum(i, j)$

Anti-transformada DCT:

$$pixel(x, y) = \frac{1}{\sqrt{2 \cdot N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \left(c(i) \cdot c(j) \cdot DCT(x, y) \cdot \cos\left[\frac{(2x+1) \cdot i \cdot \pi}{2 \cdot N}\right] \cdot \cos\left[\frac{(2y+1) \cdot j \cdot \pi}{2 \cdot N}\right] \right)$$

Recuperar imagen (por bloques): conformar la imagen a partir de los bloques decodificados y si es color recuperar los coeficientes RGB cambiando modelo de color

mayor tasa de compresión => menor calidad de reconstrucción



Original – grises (8 bpp)



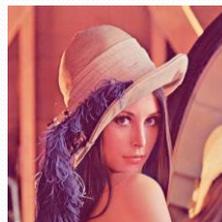
factor 10 (10:1)



factor 25 (50:1)



Original–color RGB(24 bpp)



factor 10 (26:1)

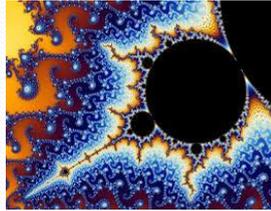


factor 25 (128:1)

COMPRESIÓN FRACTAL

Fractal: objeto matemático con propiedades de autosimilitud y detalle a toda escala

Ejemplos:

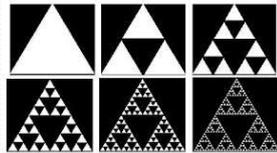


Set de Mandelbrot

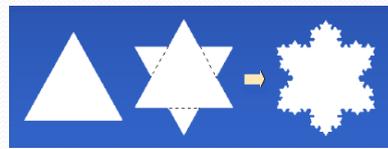


Set de Julia

Ciertos fractales se generan por aplicación iterativa de transformaciones afines –IFS- contractivas (rotaciones, cambios de escala, desplazamientos)



Triángulo de Sierpinski



copo de nieve de Koch

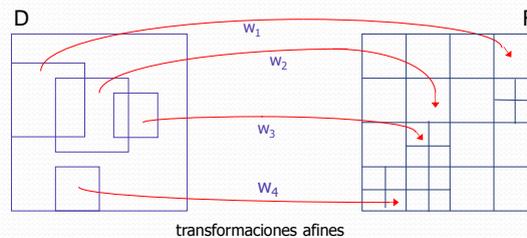
COMPRESIÓN FRACTAL

Codificación → Problema Inverso: Dada una imagen a comprimir, encontrar el conjunto de transformaciones que permitan describirla y luego re-crearla

- Esquema de compresión fractal semi-automático basado en *IFS* (Barnsley y Sloan, 1988)
- Enfoque automático de compresión fractal con IFS particionadas (Barnsley y Jacquin, 1991)

Algoritmo:

- Utilizar 2 versiones de la misma imagen a comprimir: Rango (R) y Dominio (D)
- Dividir la imagen R en bloques mediante partición quadtree
- Por cada bloque R_i en R (de $n \times n$ pixeles), buscar un bloque D_i en D (de $2n \times 2n$ pixeles) **similar** a R_i (al aplicar una transformación afín w_i)
 - si se encuentra: codificar según w_i
 - sino dividir el bloque en 4 y repetir con cada sub-bloque



COMPRESIÓN FRACTAL

Descompresión: Proceso iterativo más veloz que la compresión (alg. asimétrico)

→ consiste en encontrar el *atractor* (o punto fijo) del conjunto de transformaciones

- a partir de **cualquier** imagen (aún de distinto tamaño que la original → **independencia de la resolución**)
- aplicar reiteradamente el conjunto de transformaciones (la imagen generada en una iteración es imagen de entrada en la siguiente)

Ejemplo:



Iteración 0 Iteración 1 Iteración 2 Iteración 3 Iteración 10

COMPRESIÓN DE VIDEO

Un video se compone a partir de una secuencia de cuadros fijos (frames)

Frame rate: 25 ó 30 frames/seg (o más)

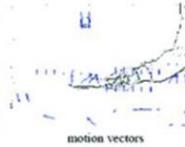
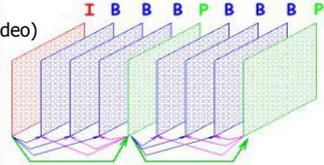


Requerimientos:

- acceso aleatorio → puntos de acceso
- reproducción y búsqueda hacia adelante/atrás
- sincronización audio-visual (y mecanismo de resincronización)
- robustez frente a errores
- controlar el *delay* codificación/decodificación (ej: aplicaciones de videoconferencia)

MPEG (Moving Picture Experts Group)

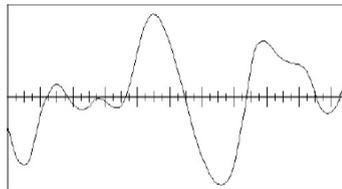
- Incluye un grupo de estándares de codificación de audio y video (1988, con numerosas revisiones y/o nuevas ediciones)
- Se aprovecha la redundancia espacial, temporal y psico-visual - alcanza altas tasas de compresión
- Codifica 3 tipos de frames:
 - Intra-frames (I) → codificación JPEG (puntos de acceso al video)
 - Inter-frame causales (P) → codificación predictiva a la imagen I ó P anterior (por compensación de movimiento)
 - Inter-frame bidireccionales (B) → codificación predictiva a la imagen I ó P anterior y posterior (compensación mov.)
- Compensación de movimiento: busca codificar el "error" entre cuadros → ocupa menos espacio que los cuadros mismos (fase costosa)



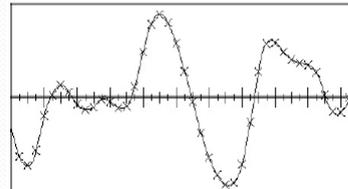
COMPRESIÓN DE SONIDO

Muestreo - sonido digital:

- Se toman muestras de la señal a intervalos regulares de tiempo
- el rango de valores se ajusta alrededor del cero (ej. -250mv and +250mv) y se escala para su almacenamiento (ej. en 8-bits: -128 to +127)



A typical audio waveform.



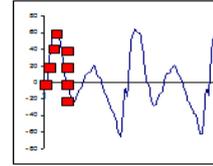
A typical audio waveform being sampled at 8KHz.

De: The Data Compression Book (Nelson)

COMPRESIÓN DE SONIDO

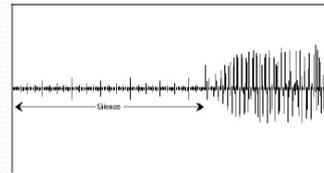
Codificación diferencial:

- Cuando la señal presenta alta correlación entre muestras sucesivas → codificar diferencias entre las muestras (la diferencia ocupa menos bits que el valor original)
- Si algún valor de diferencia es mayor que la que se puede representar → pérdida de información



Codificación de silencios:

- Similar a RLC considerando secuencias de silencios relativos (los valores en *cero ± tolerancia*)
- Se antepone un bit de *flag* que indica si sigue codificación del valor o una corrida de silencios (RLC)



A typical sound sample with a long sequence of silence.

MPEG-1 audio layer 3 (o MP3) :

- Formato de audio digital comprimido con pérdida, basado en la capacidad perceptiva del oído humano a las ondas de audio
- El ciclo interno de codificación se basa en cuantificación y codificación Huffman

BIBLIOGRAFÍA

Cover T., Thomas J., *Elements of Information Theory*, 2nd ed., John Wiley & Sons, 2006

Nelson M., Gailly J., *The Data Compression Book*, 2nd ed., M&T Books, 1996

Gonzalez R., Woods R., *Digital Image Processing*, 2nd ed., Prentice Hall, 2002

Sayood K., *Introduction to Data Compression*, 3rd ed., Elsevier, 2006

JPEG Standard, <http://www.jpeg.org>

MPEG standard, <http://www.mpeg.org>

