

Compressão de Imagem

- Os recursos necessários para armazenar e transmitir imagens são imensos, o que torna atractiva a compressão de imagem
- A compressão de imagem baseia-se na remoção de informação redundante existente nas imagens
- Existem duas categorias de compressão de imagem:
 - **não destrutiva** – é possível reconstruir EXACTAMENTE a imagem original antes de ter sido efectuada a compressão
 - **destrutiva** – no processo de compressão são perdidas características das imagens, o que permite obter graus de compressão mais elevados

● Fundamentos

- Se n_1 e n_2 forem duas representações da mesma informação, o grau de compressão (C_R) é dado por:

$$C_R = \frac{n_1}{n_2}$$

- A redundância relativa R_D pode ser definida por:

$$R_D = 1 - \frac{1}{C_R}$$

- Existem três tipos de redundância nas imagens que são explorados pelos mecanismos de compressão:
 1. **codificação** – A forma como a imagem é representada (codificada) introduz redundância
 2. **inter-pixel** – A imagem apresenta repetições de padrões de pixels
 3. **psico-visual** – A imagem inclui informação que visualmente não é relevante

Compressão de Imagem

• Redundância de codificação

- Se os tons de pixels de uma imagem não ocorrem com a mesma frequência (probabilidade) os tons mais frequentes podem ser codificados com menos bits
- O número médio de bits necessários para codificar uma imagem é dado pela soma do número de bits utilizados para representar cada tom $l(r_k)$ multiplicado pela frequência desse mesmo tom $p_r(r_k)$:

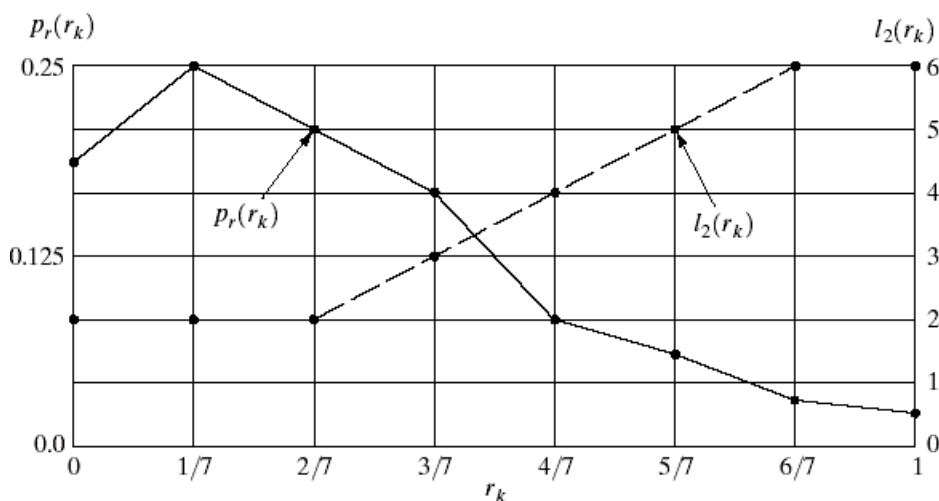
$$L_{avg} = \sum_{k=0}^{L-1} l(r_k) p_r(r_k) \quad L \text{ é o número de tons na imagem}$$

•Exemplo:

r_k	$p_r(r_k)$	Code 1	$l_1(r_k)$	Code 2	$l_2(r_k)$
$r_0 = 0$	0.19	000	3	11	2
$r_1 = 1/7$	0.25	001	3	01	2
$r_2 = 2/7$	0.21	010	3	10	2
$r_3 = 3/7$	0.16	011	3	001	3
$r_4 = 4/7$	0.08	100	3	0001	4
$r_5 = 5/7$	0.06	101	3	00001	5
$r_6 = 6/7$	0.03	110	3	000001	6
$r_7 = 1$	0.02	111	3	000000	6

O code 2 requer uma média de 2,7 bits/pixel

Representação gráfica do processo de compressão do exemplo anterior:

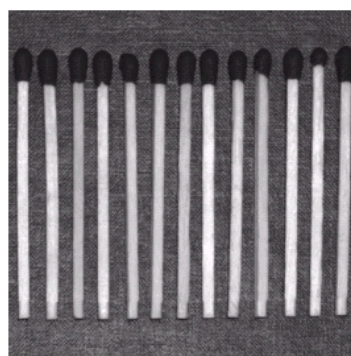


- O processo de codificação utilizado no exemplo anterior é designado por codificação de comprimento variável, uma vez que cada tom é codificado com um número de bits diferente.

Compressão de Imagem

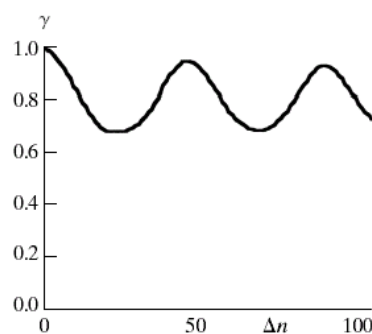
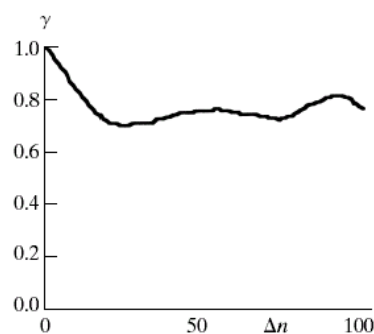
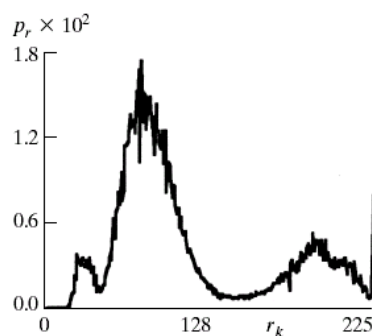
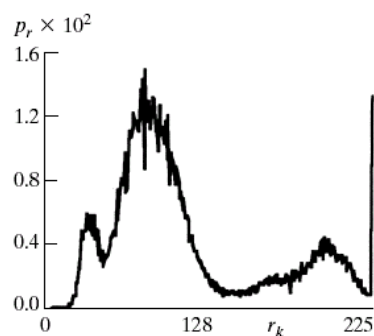
• Redundância inter-pixels

- Em algumas imagens existem padrões de pixels que se repetem, implicando que um pixel introduz pouca informação, relativamente aos seus vizinhos, porque o valor do pixel pode ser previsto a partir do valor dos vizinhos.
- Neste tipo de imagens a correlação entre blocos de pixels (γ) é mais elevada:



a	b
c	d
e	f

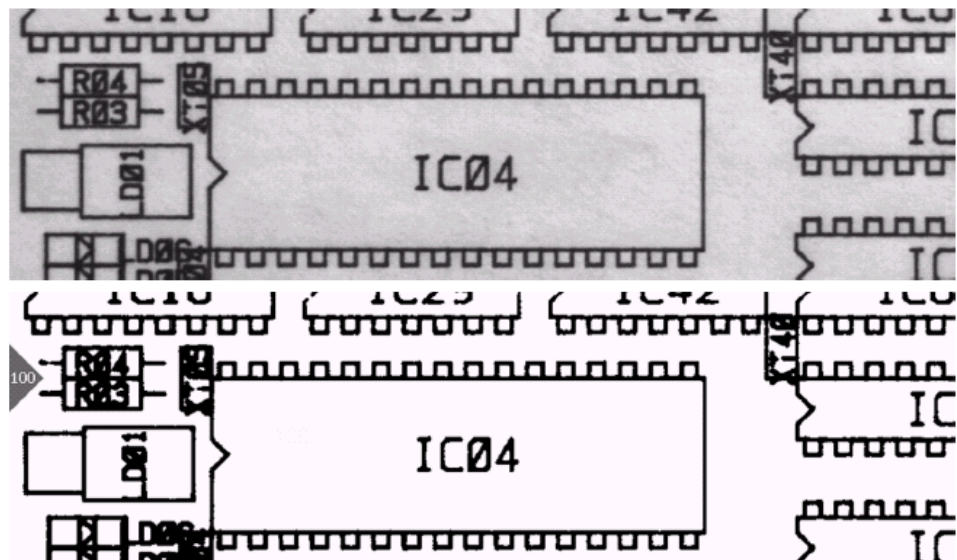
FIGURE 8.2 Two images and their gray-level histograms and normalized autocorrelation coefficients along one line.



Compressão de Imagem

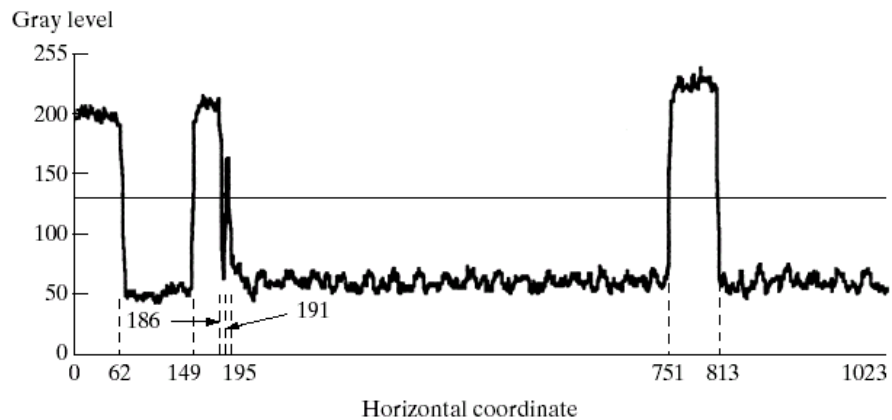
• Redundância inter-pixels (continuação)

- A redundância inter-pixels é normalmente removida através da transformação para um formato mais eficiente (geralmente “não visual”), por exemplo utilizando a diferença entre pixels adjacentes da imagem
- Exemplo – Utilizar o tom duração para representar a imagem



a
b
c
d

FIGURE 8.3
Illustration of run-length coding:
(a) original image.
(b) Binary image with line 100 marked.
(c) Line profile and binarization threshold.
(d) Run-length code.



Line 100: (1, 63) (0, 87) (1, 37) (0, 5) (1, 4) (0, 556) (1, 62) (0, 210)

O tom 1 aparece 63 vezes, depois, o tom 0 aparece 87 vezes, etc.

Neste exemplo, a codificação tom-duração permite codificar uma linha de 1024 bits em apenas 88 bits.

Nesta imagem o grau de compressão é 2,63 (considerando toda a imagem)

Compressão de Imagem

• Redundância psico-visual

- Existe certa informação nas imagens que é relativamente menos importante que outra para os sistemas de visão
- A informação relativamente menos importante pode ser removida da imagem sem que exista uma degradação significativa da qualidade visível da imagem
- Ao contrário dos dois tipos de redundância anteriores, este tipo não é facilmente quantificável, sendo um critério subjectivo
- Exemplo – redução do número de tons (i.é, bits por pixel) utilizados para representar uma imagem

os 8 bits por pixel, utilizados na imagem original, foram reduzidos para 4, utilizando duas técnicas diferentes



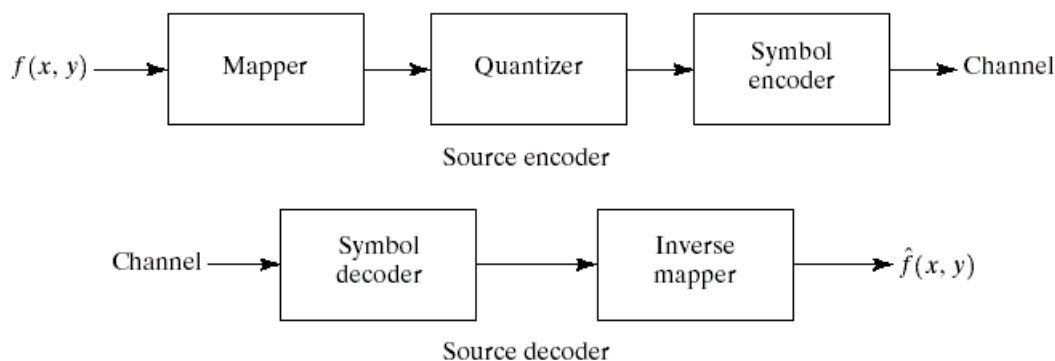
• Critério de fidelidade

- Um critério objectivo por ser obtido calculando o erro total entre as imagens ou a raiz do quadrado dos erros médios:

$$e_{rms} = \left[\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [\hat{f}(x, y) - f(x, y)]^2 \right]^{1/2}$$

Compressão de Imagem

Modelos de Compressão de Imagem



a
b

FIGURE 8.6 (a) Source encoder and (b) source decoder model.

- O **mapper** converte a imagem num formato diferente, com o objectivo de reduzir a redundância inter-pixel
- O **quantizer** reduz a acuidade do resultado produzido pelo **mapper**, de acordo com um critério de fidelidade, com vista à redução da redundância psico-visual
- O **symbol encoder** codifica os símbolos por forma a minimizar a redundância de codificação.
- As três fases de codificação não são obrigatoriamente utilizadas em todos os sistemas de codificação. Por exemplo, na compressão sem perda não é utilizado o **quantizer**.
- Apenas as operações realizadas pelo **mapper** e pelo **symbol encoder** são reversíveis.

Teoria da Informação

- Qual a quantidade mínima de dados que é necessária para representar determinada informação?
- A quantidade de informação (entropia ou incerteza) por elemento pode ser calculada a partir da probabilidade de ocorrência de cada um dos símbolos $P(a_j)$. Se a ocorrência de cada símbolo for independente:

$$H(z) = -\sum_{j=1}^J P(a_j) \log(P(a_j))$$

Compressão de Imagem

• Teoria da informação (continuação)

- Exemplo – Qual a quantidade de informação existente na seguinte imagem:

21 21 21 95 169 243 243 243
21 21 21 95 169 243 243 243
21 21 21 95 169 243 243 243
21 21 21 95 169 243 243 243

- Se não for removida a redundância, serão utilizados 8 bits/pixel
- A estimativa de primeira ordem (i.é., considerando a ocorrência dos símbolos independente) da entropia é:

Tom	Cont.	Probabilidade
21	12	3/8
95	4	1/8
169	4	1/8
243	12	3/8

$$H = - 3/8 \times \log_2(3/8) - 1/8 \times \log_2(1/8) - 1/8 \times \log_2(1/8) - 3/8 \times \log_2(3/8)$$

1,81 bits /pixel ou 58 bits no total

- A estimativa de segunda ordem (i.é., considerando a ocorrência dois símbolos consecutivos interdependente) da entropia é:

Par de tons	Cont.	Probabilidade
(21,21)	8	1/4
(21,95)	4	1/8
(95,169)	4	1/8
(169,243)	4	1/8
(243,243)	8	1/4
(243,21)	4	1/8

$$H = 1,24 \text{ bits /pixel}$$

- As estimativas de ordem superior são extremamente complexas. Utilizando blocos de 5 pixels pode originar $(2^8)^5 \approx 10^{12}$ combinações
- A estimativa de primeira ordem da entropia indica o limite que se consegue obter apenas com codificação de comprimento variável
- A diferença entre a primeira e segunda estimativas da entropia indica a existência de redundância inter-pixels

Compressão de Imagem

• Técnicas de compressão de imagem sem perda

- **Huffman** – Esquema de codificação com *códigos de comprimento variável* que obtém o menor número médio de bits por símbolo *quando não existe redundância inter-pixels*

Fase 1 – Os vários símbolos são ordenados por probabilidade decrescente, sendo sucessivamente somados os símbolos com menor probabilidade, até restarem apenas duas somas:

Original source		Source reduction			
Symbol	Probability	1	2	3	4
a_2	0.4	0.4	0.4	0.4	→ 0.6
a_6	0.3	0.3	0.3	0.3	→ 0.4
a_1	0.1	0.1	→ 0.2	→ 0.3	
a_4	0.1	0.1	0.1	→ 0.2	
a_3	0.06	→ 0.1			
a_5	0.04				

Fase 2 – São codificados os vários símbolos, adicionando um bit ao código de cada símbolo, para cada soma efectuada, por ordem inversa:

Original source		Source reduction								
Sym.	Prob.	Code	1	2	3	4				
a_2	0.4	1	0.4	1	0.4	1	0.4	1	→ 0.6	0
a_6	0.3	00	0.3	00	0.3	00	0.3	00	→ 0.4	1
a_1	0.1	011	0.1	011	→ 0.2	010	→ 0.3	01		
a_4	0.1	0100	0.1	0100	→ 0.1	011				
a_3	0.06	01010	→ 0.1	0101						
a_5	0.04	01011								

Neste exemplo, a codificação de Huffman produziu uma média de 2,2 bits/pixel o que é um valor próximo da entropia (2,14)

A codificação de Huffman atribui um código único a cada símbolo, o que permite posteriormente efectuar a descodificação, sem perda de informação

A codificação de Huffman é pouco adequada quando existem muitos símbolos, uma vez que é necessário somar as probabilidades. Existem um esquema modificado que permite truncar os códigos com mais do que determinado número de bits

Compressão de Imagem

• Técnicas de compressão de imagem sem perda (cont.)

- **LZW** (Lempel-ZivWelch) – Associa códigos de comprimento fixo a palavras de comprimento variável, permitindo também reduzir dependências inter-pixels
 - Esquema patenteado, actualmente utilizado em GIF, TIFF e PDF
 - Pode ser utilizado para a compressão de informação a qual não se conhece *a priori* (i.é., não é possível determinar a probabilidade de cada símbolo)
 - As palavras reconhecidas são mantidas num dicionário, construído dinamicamente
 - Exemplo – codificar a seguinte imagem (9 bits por símbolo)

```

39 39 126 126
39 39 126 126
39 39 126 126
39 39 126 126

```

Resultado da codificação

Currently Recognized Sequence	Pixel Being Processed	Encoded Output	Dictionary Location (Code Word)	Dictionary Entry
	39			
39	39	39	256	39-39
39	126	39	257	39-126
126	126	126	258	126-126
126	39	126	259	126-39
39	39			
39-39	126	256	260	39-39-126
126	126			
126-126	39	258	261	126-126-39
39	39			
39-39	126			
39-39-126	126	260	262	39-39-126-126
126	39			
126-39	39	259	263	126-39-39
39	126			
39-126	126	257	264	39-126-126
126		126		

- Um aspecto delicado do algoritmo é a escolha das palavras mantidas no dicionário e a própria dimensão do dicionário.

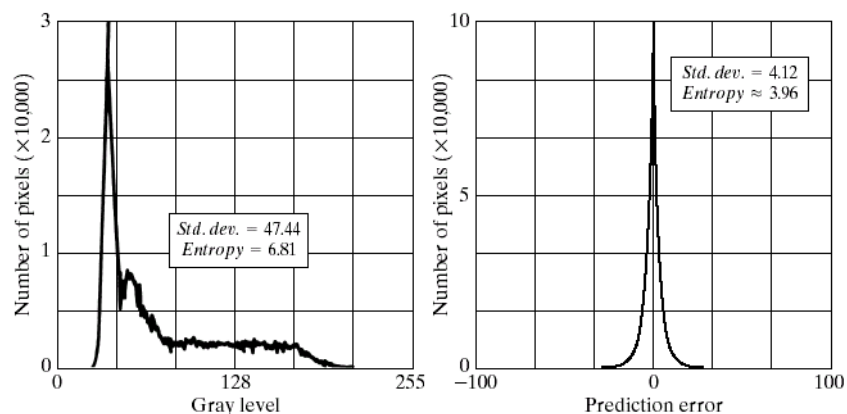
Compressão de Imagem

• Técnicas de compressão de imagem sem perda (cont.)

- **Codificação de planos de bits** – A imagem é decomposta em planos de bits, sendo cada plano comprimido individualmente
 - Uma decomposição alternativa utiliza *gray codes*, onde dois tons com valores adjacentes apenas diferem num bit.
 - Os planos de bits podem ser codificados através de tom-duração ou através da codificação de regiões de tom constante (divisão da imagem em blocos).
- **Codificação com previsão** (sem perda) – É utilizada uma função para prever o tom dos pixels seguintes, em função dos anteriores, sendo apenas armazenada a diferença entre o valor previsto e o tom efectivo do pixel: $e_n = f_n - \hat{f}_n$
 - A função de previsão é na maior parte dos casos uma função linear dos pixels existentes nessa linha da imagem:

$$\hat{f}_n(x, y) = \text{round} \left[\sum_{i=1}^m \alpha_i f(x, y - i) \right]$$

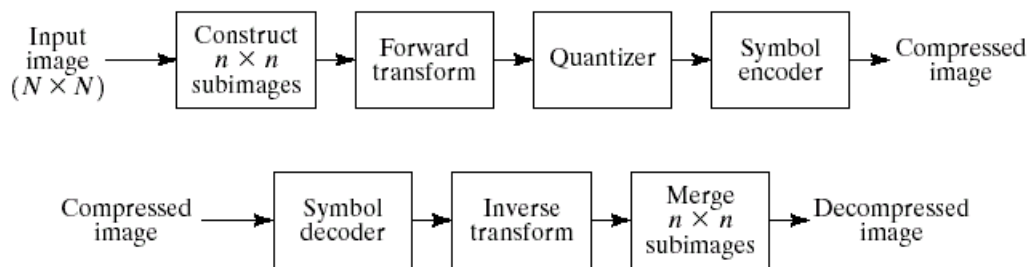
- Exemplo: $\hat{f}_n(x, y) = \text{round}[f(x, y - 1)]$



Compressão de Imagem

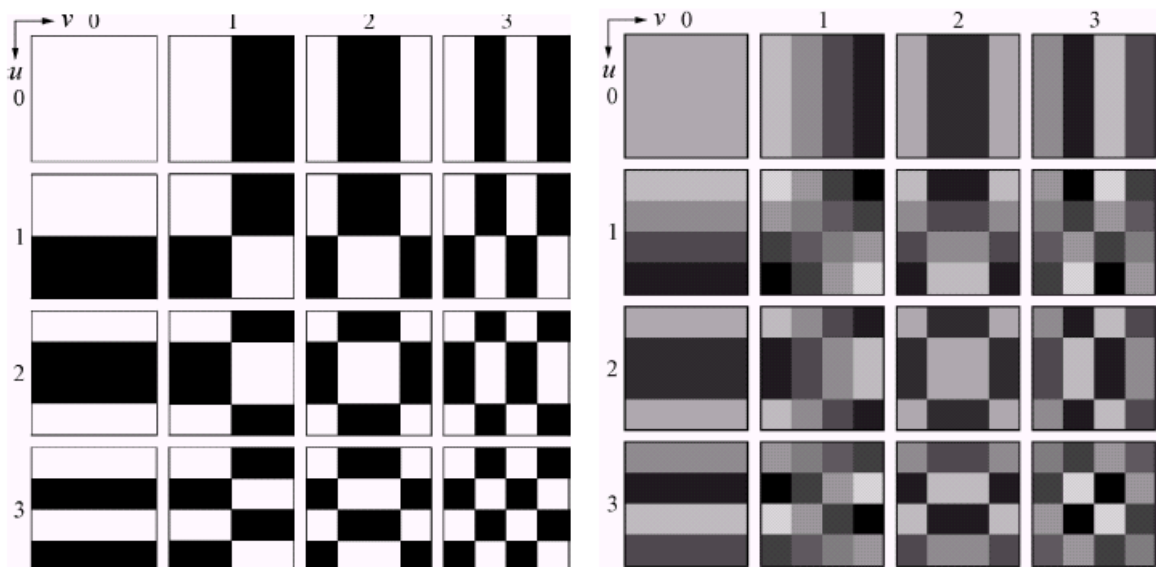
• Técnicas de compressão de imagem com perda

- **Codificação através de transformadas** – A imagem é transformada numa nova representação (ex. FFT) sendo a quantificação efectuada na nova representação



- Geralmente a imagem é dividida em blocos, sendo a codificação efectuada isoladamente a cada bloco da imagem. A divisão em blocos pode ser um processo irregular (i.é., a divisão é adaptativa)
- O *quantizer* geralmente elimina valores da transformada (coeficientes) com um impacto visual reduzido (i.é. com menor amplitude) o que reduz o espaço necessário para representar a imagem.
- O *symbol encoder* codifica os coeficientes que não foram eliminados
- As transformadas de imagem podem ser interpretadas como um processo em cada imagem é representada por uma soma de imagens base, em que cada coeficiente da transformada determina a contribuição de cada uma das imagens base para a imagem original

- o Exemplos: Walsh-Hadamard e transformada discreta do coseno (4x4)

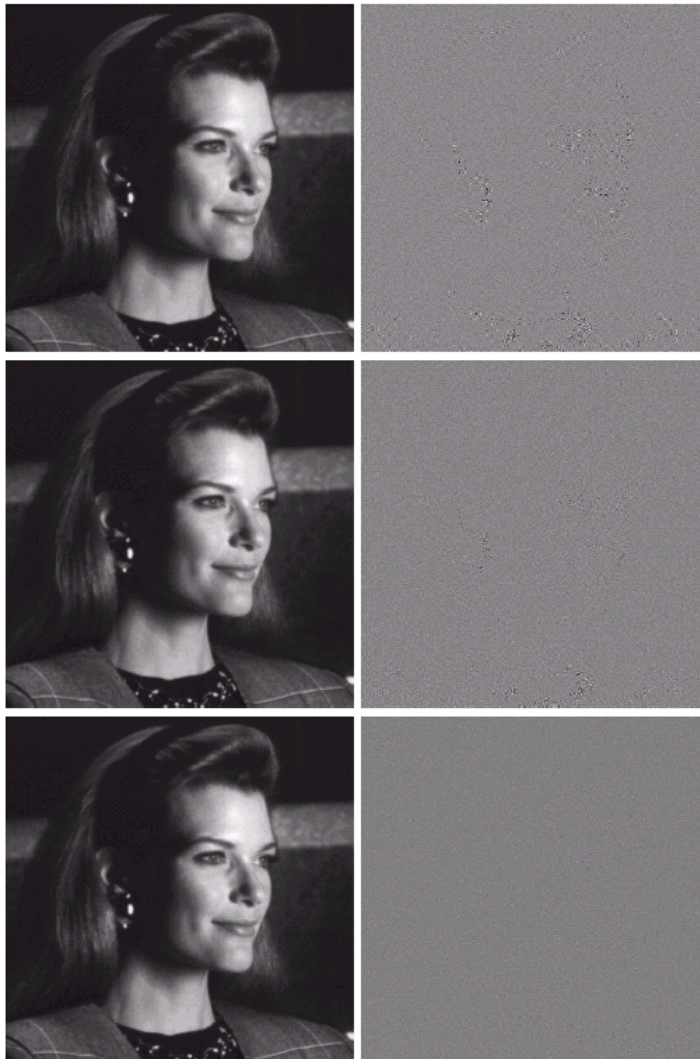


Compressão de Imagem

• Técnicas de compressão de imagem com perda (cont)

• Codificação através de transformadas (cont)

- Exemplo: comparação de três transformadas de imagem (FFT, WHT e DCT), desprezando 50% dos coeficientes (mantendo os de maior amplitude) e subdividindo a imagem em blocos de 8x8 pixels



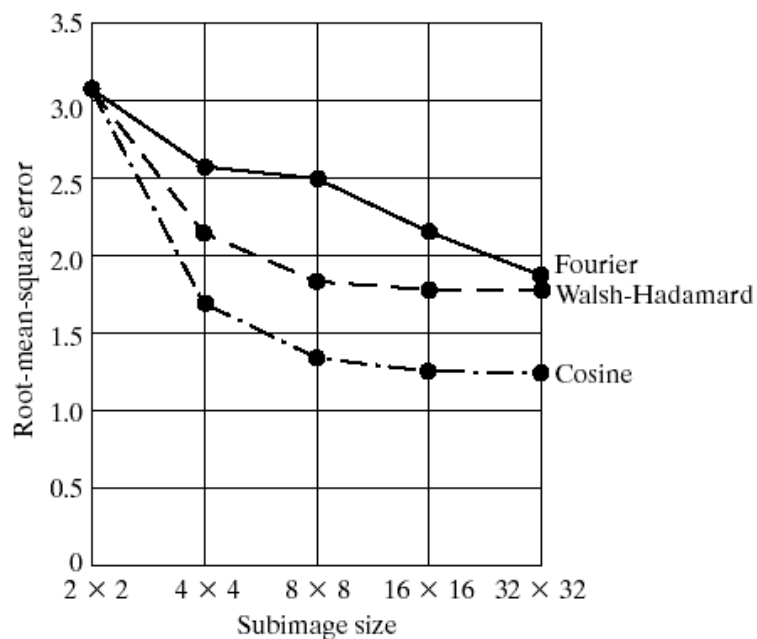
- Os erros rms, para cada uma das transformadas são, respectivamente, 1,28, 0,86 e 0,68 níveis de cinzento
- As transformadas que concentram mais informação num menor número de coeficientes são as mais indicadas para efectuar compressão de imagem (DCT é geralmente superior às outras transformadas)
- A DCT tem a vantagem adicional de minimizar o efeito de blocos que aparece nas imagens após a compressão

Compressão de Imagem

• Técnicas de compressão de imagem com perda (cont)

• Codificação através de transformadas (cont)

- As imagens são subdivididas em blocos por forma a reduzir a redundância entre blocos adjacentes. Os blocos geralmente possuem uma dimensão que seja potência de 2 para reduzir a quantidade de cálculos. Dimensões frequentemente utilizadas são 8x8 e 16x16.
- O grau de compressão e a complexidade dos cálculos aumentam com a dimensão dos blocos.
- Variação do erro em função da dimensão dos blocos (truncando 75% dos coeficientes):



- **Wavelet** – transformada semelhante às anteriores, mas que é computacionalmente mais eficiente e inerentemente local o que torna desnecessária a divisão da imagem em blocos.

Compressão de Imagem

• Standards de compressão de imagem

• CCITT Group 3

- Originalmente utilizado para compressão de FAX transmitidos pela linha telefónica
- Utilizado para compressão sem perda de imagens binárias, baseado num esquema de compressão tom-duração com códigos Huffman modificados

• JPEG

- Standard mais popular de compressão de imagens com perda
- Baseado na transformada discreta do coseno, aplicada a blocos de 8x8 pixels

• JPEG 2000

- Extensão ao JPEG para permitir maior flexibilidade na compressão e no acesso a imagens comprimidas
- Baseado nas técnicas de codificação com *wavelets*

• Standards de compressão de vídeo

• MPEG 1, MPEG 2 e MPEG 4

- Abrangem várias qualidade de vídeo: MPEG 1 – qualidade de entretenimento (até 1.5 Mbit/2), MPEG-2 – qualidade PAL (2 a 10 Mbit/s) e MPEG-4 – flexibilidade acrescida e maior eficiência na compressão
- Baseado num esquema híbrido com previsores e DCT
- Explora as redundâncias intra-imagem e inter-imagens, possuindo vários tipos de blocos para codificação:
 1. *Intra-frame* (I-Frame) – *frame* comprimida de forma independente de todas as outras *frames* . É a mais próxima de JPEG
 2. *Predictive frame* (P-Frame) – diferença entre a *frame* actual e a sua previsão baseada na *frame* anterior
 3. *Bidirectional frame* (B-Frame) – diferença entre a *frame* actual e a sua previsão baseada na *frame* anterior e na próxima *P-frame*