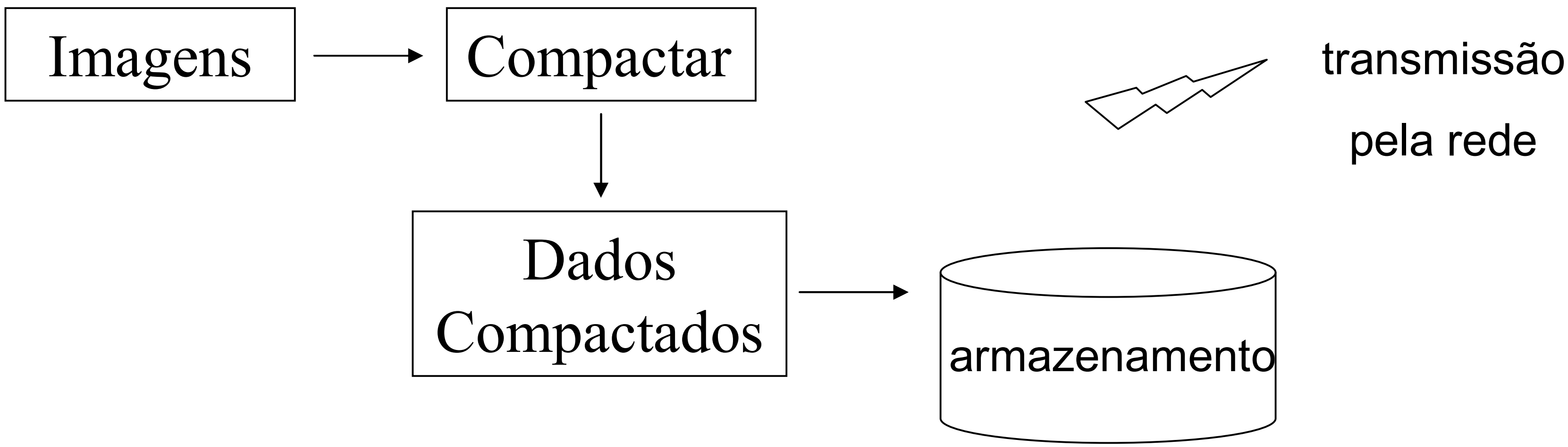
Aula 9.1

Compressão de Imagens

Image Compression

Uma enorme quantidade de dados é produzida quando uma função intensidade de luz bidimensional é amostrada e quantizada para criar uma imagem digital. De fato, a quantidade de dados gerada pode ser tão grande que inviabiliza o armazenamento, o processamento e a comunicação.

A compressão de imagens trata o problema de reduzir a quantidade de dados necessária para representar uma imagem digital



<u>Exemplo:</u>

Seja um vídeo com resolução de 720 x 480 com 24 bits por pixel (R=G=B=8 bits) capturado em uma taxa de 30 fps (frames por segundo), gravado por duas horas

720 x 480 x 3 x 30 x 3.600 x 2 = 223.948.800.000 bytes = 208,56857 Giga Bytes

ou 24,537 DVDs de camada dupla com 8.5 GB

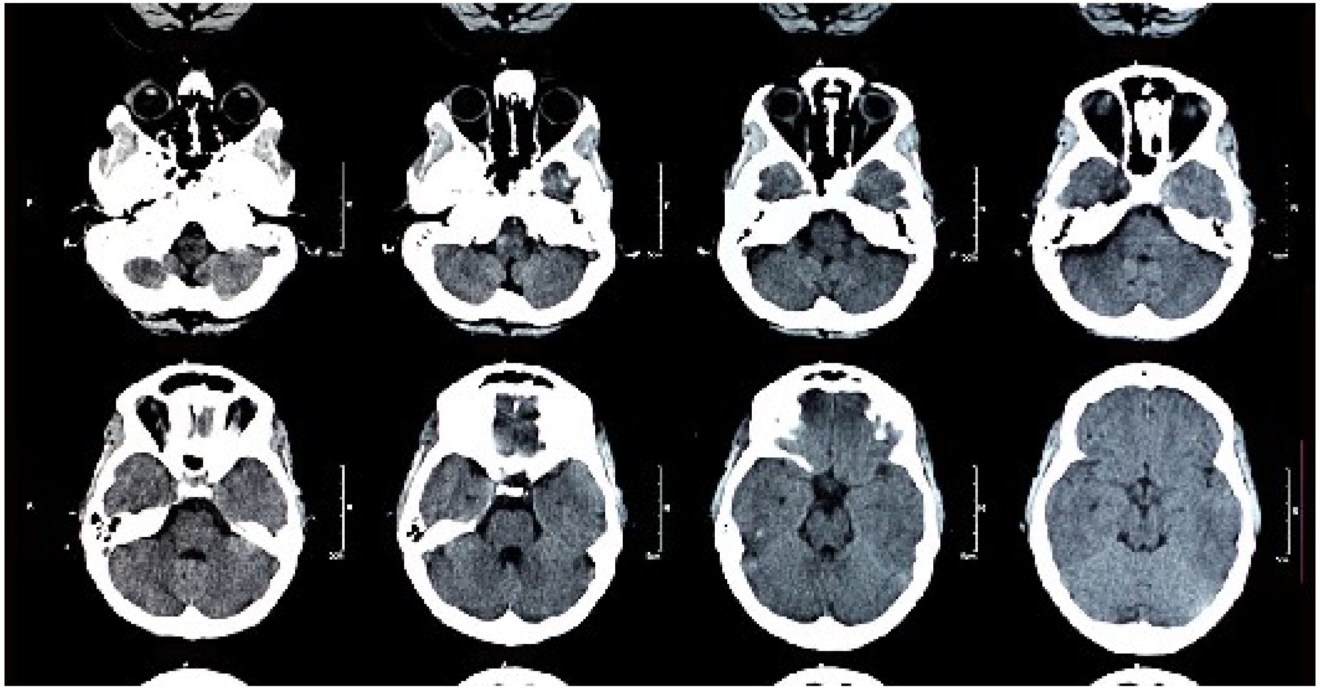
Compressão de Imagens

<u>Armazenamento</u> de imagens se refere ao armazenamento eletrônico de dados de uma imagem, tipicamente em mídias permanentes magnéticas ou outras, em um arquivo.

<u>Transporte</u> de imagem se refere à transferência eletrônica de dados de uma imagem, através de um link de transmissão de dados.

<u>Compressão</u> de imagens procura representar uma imagem, com algum nível de qualidade exigido, numa forma mais compacta, preservando informações essenciais, de forma que a mesma possa ser reconstruída com a precisão desejada

Imagens Médicas – Exames menos invasivos fazem uso intenso de imagens



Imagens de Monitoramento – muitas câmeras espalhadas pelas cidades coletam imagens o tempo todo

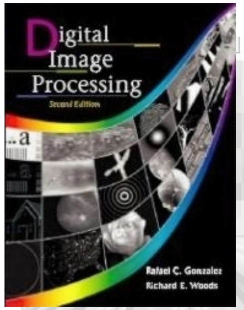
Imagens de Satélite – muitas imagens são coletadas pelos sensores embarcados nos satélites e precisam ser transmitidas e armazenadas

Fatores importantes

- Diminuição da quantidade de armazenamento requerido pela imagem;

- Distorção resultante provocada pela eliminação dos dados não necessários da imagem (redundantes);

- Grau de complexidade computacional que permita a manipulação (salvar e carregar) das imagens sem demora



Compressão x Compactação

Esquemas de compressão de imagem podem ser divididos em dois grupos:

compressão **sem perda** - preserva o conteúdo original
LZW (Lempel Ziv Welch) → **Compressão**
Ex. Arquivos RAR, GIF

compressão **com perda** – gera perdas (aceitáveis)
Ex. Arquivos JPEG → **Compactação**

A base do processo de redução com perdas é a remoção de dados redundantes, conservando as informações básicas, de tal maneira que as informações perdidas não sejam perceptíveis pelo olho humano

Os métodos podem operar no domínio do espaço ou da frequência

No caso dos métodos que usam o domínio da frequência, a transformada que tem se mostrado mais adequada é a **Transformada Discreta do Cosseno**

O algoritmo mais importante que utiliza esta transformada é o JPEG (Joint Photographic Experts Group) década de 80

Fundamentos

Redundância – é um tema central em compressão de dados
Se n_1 e n_2 denotam o número de unidades de transporte de informação em dois conjuntos de dados que representam a mesma informação, então a redundância de dados relativa R_D do primeiro conjunto pode ser definida como

$$R_D = 1 - (1/C_R)$$

onde $C_R = n_1 / n_2$ é conhecida como a taxa de compressão

Quando $n_1 = n_2$, $C_R = 1$ e $R_D = 0$ (não há redundância)

Quando $n_1 >> n_2$, $C_R \to \infty$ e $R_D = 1$ (alta redundância)

Quando $n_1 << n_2$, $C_R \to 0$ e $R_D \to -\infty$ (o conjunto de dados contém mais informação que o conjunto original)

Redundância de codificação

Utiliza o histograma da imagem para obter a quantidade de informação

Assumindo que a variável aleatória discreta r_k no intervalo [0,1] representa os níveis de cinza da imagem e que cada r_k ocorre com probabilidade $p_r(r_k)$, ou seja,

$$p_r(r_k) = n_k / n \qquad \text{com } k = 0,1,....,L-1$$

onde L é o número de níveis de cinza

Se o número de bits usados para representar r_k for $l(r_k)$, então o número médio de bits necessário para representar cada pixel é:

$$L_{médio} = \sum_{k=0}^{L-1} l(r_k)\, p_r(r_k)$$

ou seja, o comprimento médio das palavras de código atribuídas aos vários valores de cinza é dado pela soma do produto do número de bits usados para representar cada nível de cinza e a probabilidade em que o nível de cinza ocorre

Image Compression

Exemplo - Uma imagem de 8 níveis tem a distribuição de níveis de cinza ilustrada na tabela. Se um código binário natural de 3 bits (código 1) e $l(r_k)$ for usado, $L_{médio}$ será = 3, para todo r_k , pois $l_1(r_k)$=3. Porém, se o código 2 for usado o número médio de bits usado para codificar diminui

Probabilidade do nível de cinza

Níveis de cinza

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

Usando **Code 1** com tamanho fixo de 3 bits

$$L_{médio} = \sum_{k=0}^{L-1} l_2(r_k)p_r(r_k) = 3(0{,}19) + 3(0{,}25) + 3(0{,}21) + 3(0{,}16) + 3(0{,}08) + 3(0{,}06) + 3(0{,}03 + 3(0{,}02) = 3bits$$

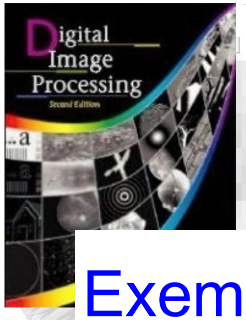


Image Compression

Exemplo - Uma imagem de 8 níveis tem a distribuição de níveis de cinza ilustrada na tabela. Se um código binário natural de 3 bits (código 1) e $l(r_k)$ for usado, $L_{médio}$ será = 3, para todo r_k, pois $l_1(r_k)$=3. Porém, se o código 2 for usado o número médio de bits usado para codificar diminui

Probabilidade do nível de cinza

Níveis de cinza

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

Usando **Code 2** com tamanho variável

$$\left\{ \begin{array}{l} L_{médio} = \sum_{k=0}^{L-1} l_2(r_k) p_r(r_k) = 2(0{,}19) + 2(0{,}25) + 2(0{,}21) + \\ 3(0{,}16) + 4(0{,}08) + 5(0{,}06) + 6(0{,}03 + 6(0{,}02) = 2{,}7 bits \end{array} \right.$$

Image Compression

Exemplo – continuação...

A taxa de compressão resultante fica:

$CR = n_1 / n_2$

$C_R = 3 / 2,7 = 1,11$

Logo $RD = 1 – (1/CR)$

$R_D = 1 – (1 / 1,11) = 0,099$, ou seja, aproximadamente 10%

Ou seja:

Aproximadamente, 10% dos dados resultantes do uso do código 1 é redundante

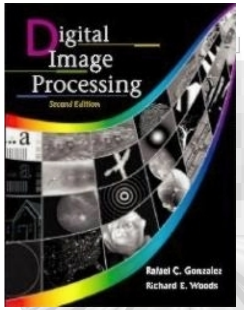


Image Compression

Codificação de comprimento variável

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

Usa códigos mais curtos para as palavras que mais ocorrem (>probabilidade)
Usa códigos mais longos para as palavras que menos ocorrem (<probabilidade)

verde : tamanho do código
vermelho : probabilidade dele ocorrer

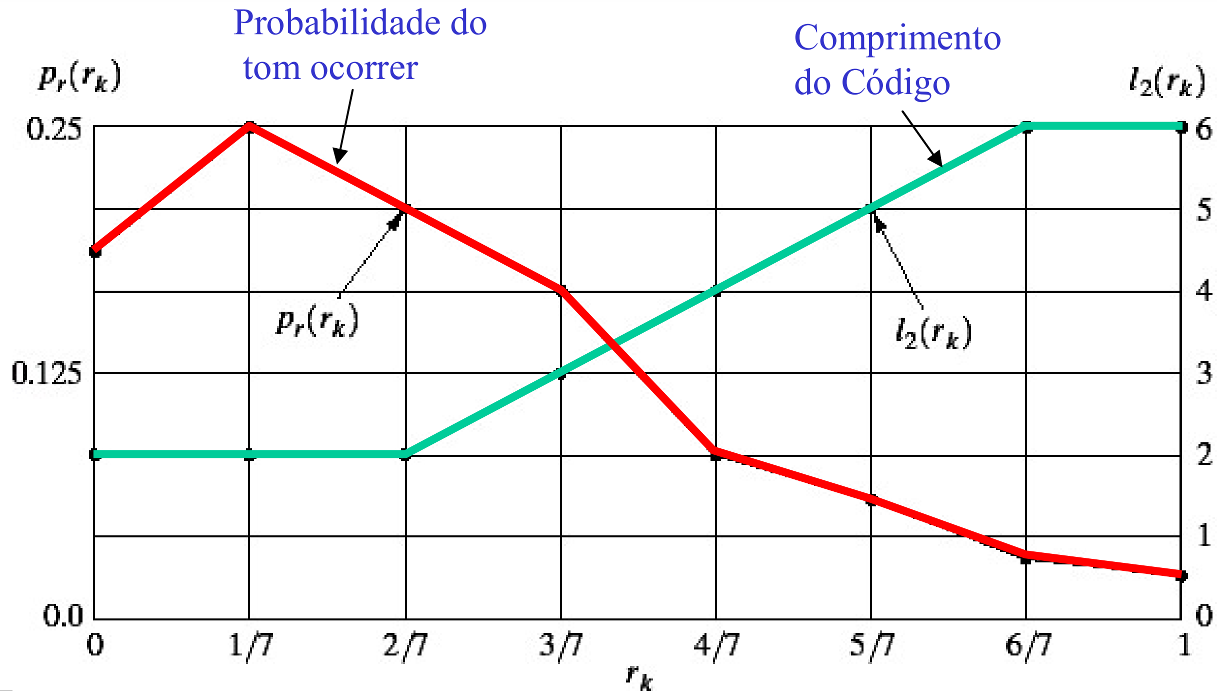
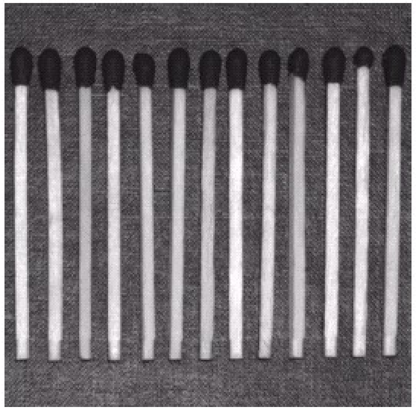


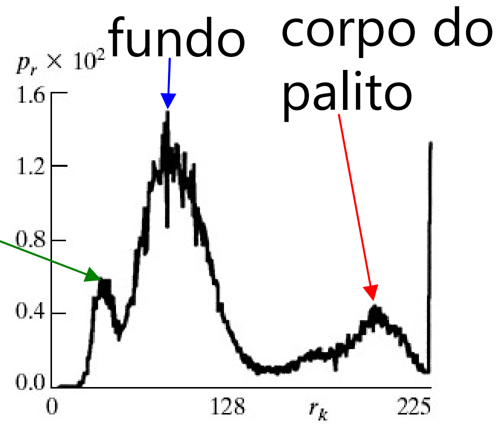
Image Compression

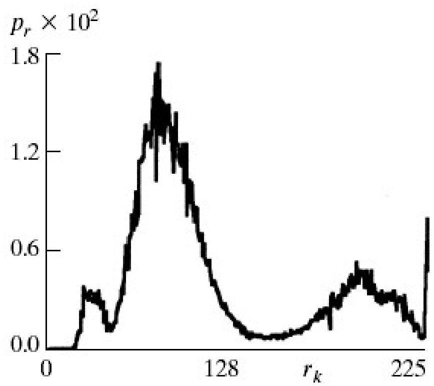
Redundância interpixel

Se refere às diferenças entre um pixel x,y e o próximo x+1, y

As duas imagens possuem histogramas muito parecidos, entretanto, a imagem da direita possui uma redundância interpixel maior, ou seja, pois os padrões se repetem mais regularmente







Redundância interpixel

Uma maneira de reduzir esta redundância interpixel consiste em calcular as diferenças entre os pixels adjacentes e, representar a imagem com esta informação

Estas transformações são chamadas mapeamentos, que são chamados reversíveis se os elementos da imagem original puderem ser reconstruídos a partir dos dados transformados

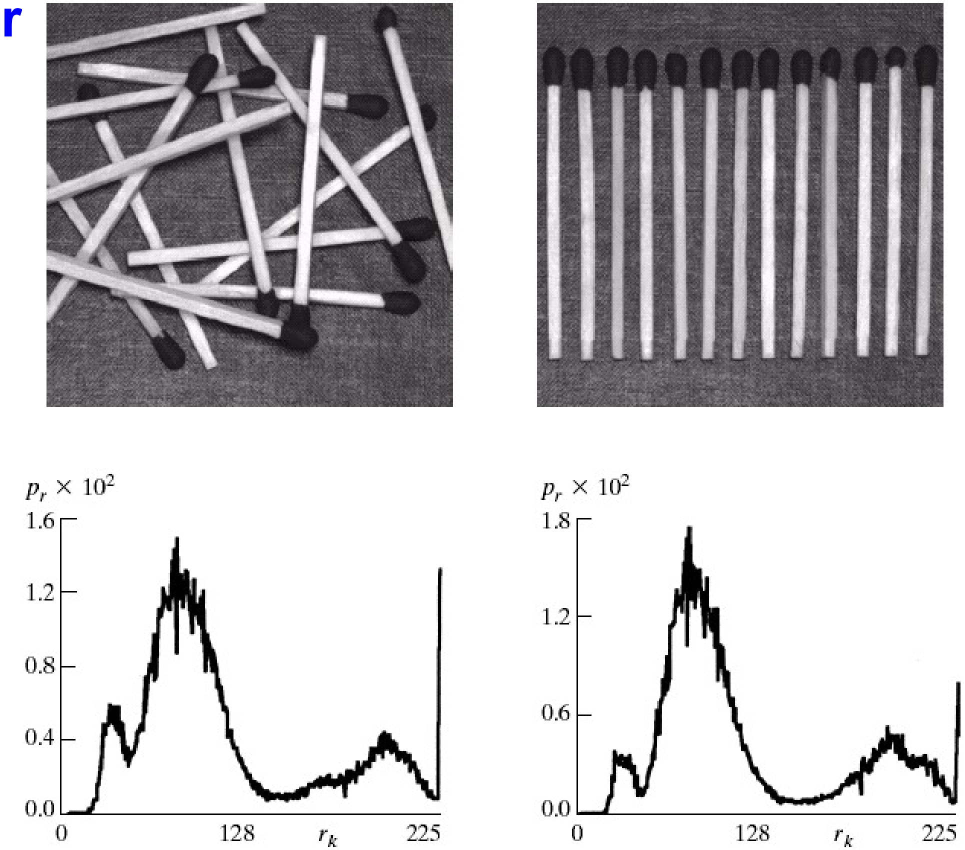


Image Compression

Quando a imagem é "bem comportada", as diferenças entre os pixels vizinhos é pequena e, assim, fica mais fácil guardar estes valores (usa-se menos bits para números menores)

Muitos pixels iguais, vão ter a mesma diferença de um para o próximo (igual a zero), basta anotar a quantidade de níveis de cinza no lugar de guardar cada um deles

Ex.

supondo a imagem |88|88|88|88|88|88|88|88|88| → 9 bytes

toma-se as diferenças |88|0|0|0|0|0|0|0|0|

guarda-se as quantidades |88|9| → 2 bytes

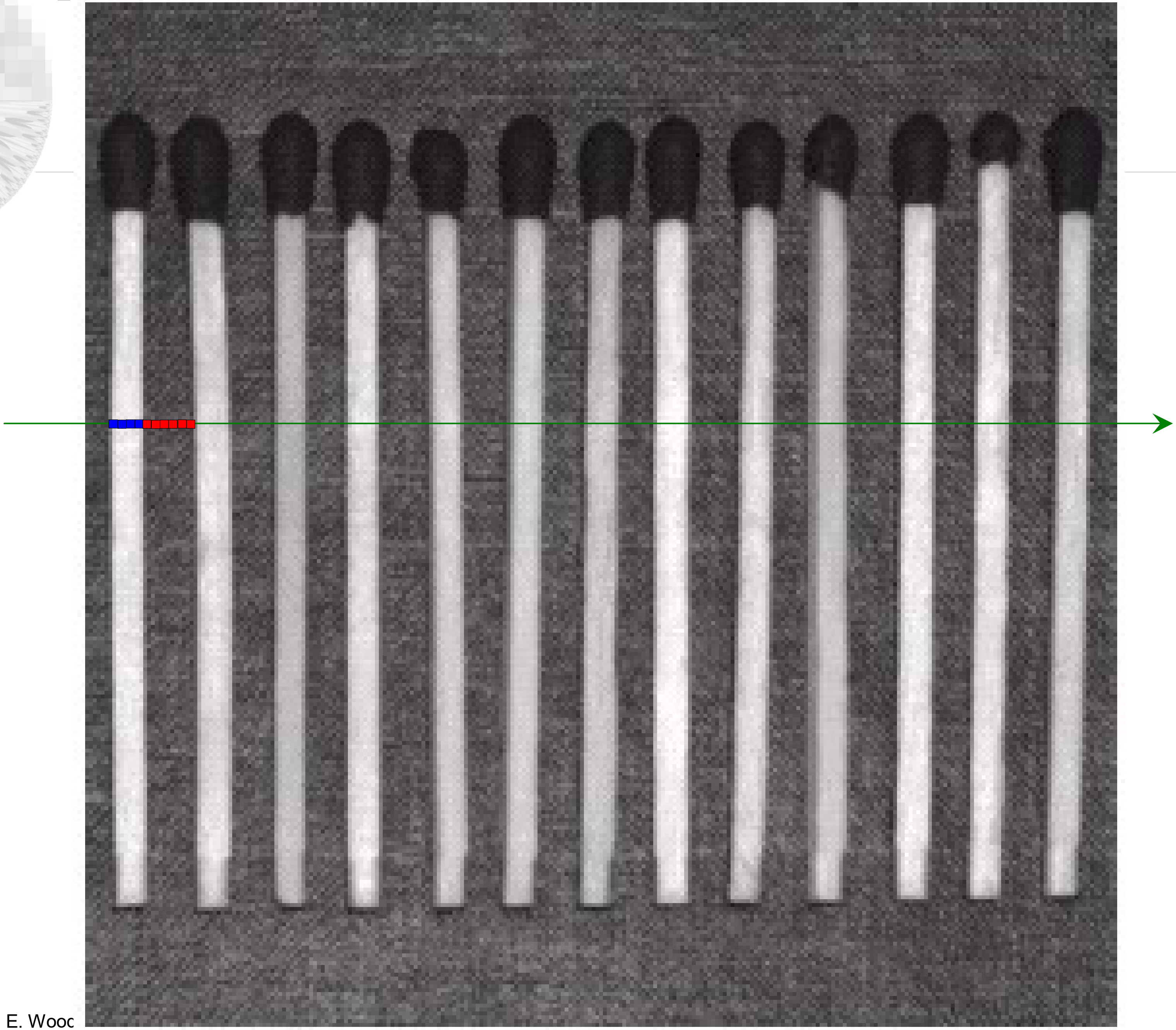
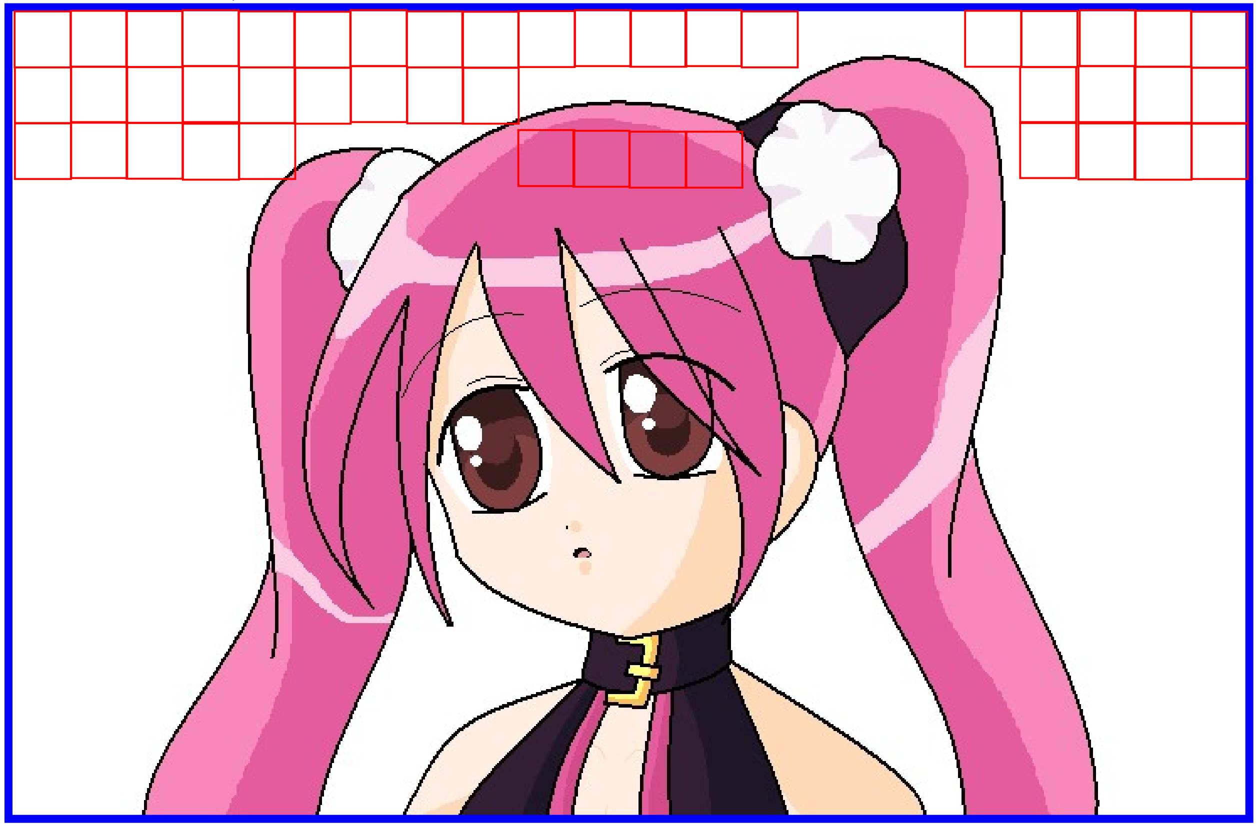


Image Compression

Desenhos são exemplos de imagens que apresentam ganho alto, pois possuem alta redundancia interpixel. Imagens capturadas possuem menor redundância interpixel

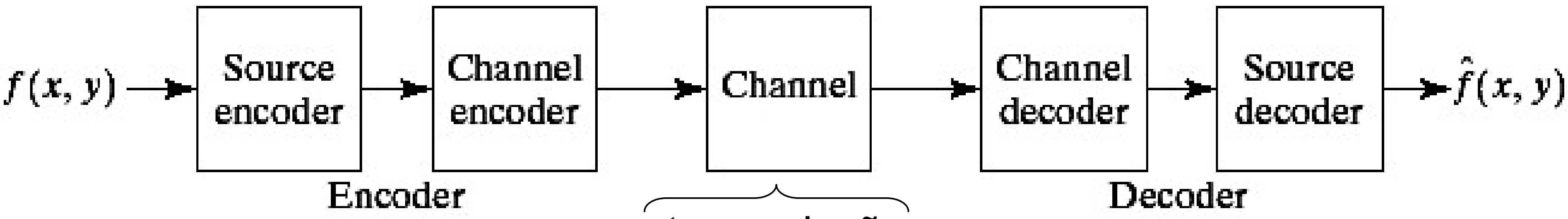


Modelos de compressão de imagens

Consiste de dois blocos estruturais distintos:

- codificador – pega a imagem e a transforma em um código reduzido

- decodificador – pega o código reduzido e contrói a imagem



transmissão
ou
armazenamento

Usando a Teoria da Informação

A teoria da informação fornece ferramentas básicas para o tratamento de representação e manipulação de informação direta e quantitativamente

Elementos da teoria da informação

Esta teoria vai ajudar a responder a questão:

Qual é a quantidade mínima de dados, realmente necessária, para representar uma imagem?

Medidas de informação

A premissa fundamental da teoria da informação é que a geração da informação pode ser modelada como um processo probabilístico, que pode ser medido de maneira que concorde com nossa intuição.

Assim, um evento aleatório E que ocorre com probabilidade P(E) contém:

$$I(E) = \log \frac{1}{P(E)} = -\log P(E)$$

Usa-se o log na base 2 para trabalhar com bit

unidades de informação. I(E) é chamada auto-informação de E

Image Compression

Genericamente, a quantidade de informação atribuída a um evento E é inversamente relacionada á probabilidade de E

Se P(E) = 1 (o evento sempre ocorre), I(E) = 0

A base do logarítimo determina a unidade para medir a informação

Quando se usa a base 2, a unidade é o bit

Assim, se $P(E) = ½$, $I(E) = -\log_2 ½ = 1$ bit
(significa que um bit é a quantidade de informação transferida quando um dos dois eventos possíveis e igualmene prováveis ocorre)

Canal de informação

Quando auto-informação é transferida entre uma fonte de informação e um usuário de informação, diz-se que a fonte de informação está conectada ao usuário de informação por um canal de informação (por exemplo, uma linha telefônica)

Canal de Informação e a Entropia

Assumindo que a fonte de informações gera uma sequência aleatória de símbolos $\{a_1, a_2, \ldots, a_j\}$ denominado alfabeto fonte. A probabilidade da fonte produzir o símbolo a_j é $P(a_j)$ e

$$\sum_{j=1}^{J} P(a_j) = 1$$

A informação média por saída da fonte, chamada incerteza ou **entropia** da fonte é:

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

define a quantidade média de informação

Image Compression

$$\log_2 x = \log_{10} x / \log_{10} 2$$

Usando a teoria da informação

Exemplo 1: Considere o problema de estimar o conteúdo de informação da imagem de 8 bits abaixo:

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32

Fazendo

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

Nível de cinza	Contagem	probabilidade
i	1	1/32

Tem-se

$$H(z) = -\sum_{j=1}^{32} 1/32.\log(1/32) = \mathbf{5}$$

usando o logaritmo natural ln(x) / ln(2)

Assim, uma estimativa, **chamada de primeira ordem**, da entropia da fonte é calculada, sendo igual a 5 bit/pixel

Como são 32 pixels, a entropia da fonte (imagem) é (5) * (32) = 160 bits

Image Compression

Exercício 1: Calcule a entropia da fonte (imagem) de 8 bits abaixo:

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

Resposta na aula 9a

entregar agora