

Classificação e Pesquisa de Dados

Aula 27
Compressão de Dados Multimídia:
Compressão de Imagens

UFRGS

INF01124

Compressão de Imagens

- ◆ Reduz a quantidade de dados necessária para representar uma imagem
- ◆ Compressão sem perdas (*lossless*)
 - Explora a redundância presente em imagens
 - A imagem recuperada é idêntica à imagem original
- ◆ Compressão com perdas (*lossy*)
 - Modifica a imagem original de forma sutil (de modo quase imperceptível ao olho humano), forçando a ocorrência de redundâncias que são, então, removidas
 - A imagem recuperada é uma aproximação da imagem original

7/4/2008

INF01124

Tipos de Redundância

- ◆ De Codificação
 - Decorre da representação ineficiente para os valores de pixels
 - Ex.: Uso de representação com 8 bits para imagem com 16 tons
- ◆ Interpixel
 - Valores de pixels vizinhos costumam apresentar forte correlação
- ◆ Psico-visual
 - Diferentes níveis de informações presentes em uma imagem têm importâncias diferentes para o sistema visual humano
- ◆ Interframe
 - De modo geral, existe forte coerência entre quadros vizinhos em um vídeo

7/4/2008

INF01124

Critério de Fidelidade

- ◆ Mede a diferença entre a imagem original (f) e a imagem descomprimida (f')
 - ◆ Exemplo de critério objetivo: root-mean-square (RMS) error
 - Para uma imagem com dimensões $M \times N$
- $$\varepsilon = \left[\frac{1}{MN} \sum_{y=0}^{N-1} \sum_{x=0}^{M-1} [f'(x,y) - f(x,y)]^2 \right]^{0.5}$$
- ◆ O valor de ε não necessariamente correlaciona com a percepção visual humana
 - Critérios subjetivos baseados na percepção humana são preferíveis nestes casos

7/4/2008

INF01124

Compressão sem Perdas

- ◆ Compressão Delta (ou Codificação Diferencial)
- ◆ Run Length Encoding
- ◆ Codificação Estatística
 - Ex.: Codificação de Huffman

7/4/2008

INF01124

Compressão Delta

- ◆ Também conhecida como Codificação Diferencial
- ◆ Representa uma imagem em termos das diferenças de tons entre cada pixel e seu vizinho anterior na mesma linha
- ◆ O 1º pixel de cada linha mantém sua representação original
- ◆ Pequenas diferenças são representadas usando poucos bits

7/4/2008

INF01124

Run Length Encoding

- ◆ Representa grupos de pixels vizinhos (em uma mesma linha) com valores de tons idênticos como uma dupla (n, i)
 - n representa o número de ocorrências consecutivas de pixels com intensidade i
 - Ex.: A sequência {32, 32, 32, 32, 32} é representada por (5,32)
- ◆ Principal aplicação prática
 - Compressão de imagens binárias para transmissão por fax
- ◆ Não deve ser utilizado para imagens ruidosas com muitas texturas
 - Muitas sequências de comprimento 1

7/4/2008

INF01124

Codificação Estatística

- ◆ Elimina a redundância na codificação de imagens
- ◆ Se utiliza de conceitos de Teoria da Informação
- ◆ Substitui os valores de pixels por "codewords" de tamanhos variáveis
- ◆ A quantidade de informação comunicada por um pixel (tom na imagem) é inversamente proporcional à sua frequência
- ◆ Uma codificação ótima utiliza menos bits para os tons mais frequentes e mais bits para os menos frequentes

7/4/2008

INF01124

Compressão com Perdas

- ◆ Compressão JPEG
 - Padrão de compressão definido pelo Joint Photographic Experts Group (JPEG)
 - Define 3 sistemas de codificação:
 - (1) *lossy baseline coding system* - adequado para a maioria das aplicações de compressão e baseado na DCT
 - (2) *extended coding system* - para altas taxas de compressão, alta precisão ou aplicações que utilizem reconstrução incremental
 - (3) *independent coding system* - para compressão sem perdas

7/4/2008

INF01124

Compressão JPEG

lossy baseline coding system

- ◆ Usa codificação por meio de transformada
- ◆ Cria uma representação no domínio frequência e descarta alguns dos coeficientes de altas frequências
- ◆ Baseado na Transformada Discreta do Cosseno
 - Para uma imagem com dimensões NxN

$$T(u, v) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \alpha(u) \alpha(v) \cos \left[\frac{(2x+1)u\pi}{2N} \right] \cos \left[\frac{(2y+1)v\pi}{2N} \right]$$

- ◆ onde

$$\alpha(k) = \begin{cases} \sqrt{1/N} & \text{para } k = 0 \\ \sqrt{2/N} & \text{para } k = 1, 2, \dots, N-1 \end{cases} \quad k = u, v$$

7/4/2008

INF01124

Compressão JPEG

lossy baseline coding system

- ◆ Inversa da Transformada Discreta do Cosseno
 - Para uma imagem com dimensões NxN

$$f(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} T(u, v) \alpha(u) \alpha(v) \cos \left[\frac{(2x+1)u\pi}{2N} \right] \cos \left[\frac{(2y+1)v\pi}{2N} \right]$$

7/4/2008

INF01124

O Algoritmo JPEG

- ◆ Força a ocorrência de redundância para então eliminá-la
- ◆ Quebra a imagem original em pequenos blocos para manter os efeitos das perdas localmente

7/4/2008

INF01124

Compressão JPEG

imagem em tons de cinza com 8 bits

Compressão_JPEG

```
{
  Divida a imagem em blocos de 8 x 8 pixels
  para cada bloco de pixels faça
    subtraia 128 do valor do tom de cada pixel;
    calcule a DCT do bloco;
    quantize os coeficientes da DCT;
    coloque os coeficientes quantizados em um array 1D;
    aplique compressão delta ao primeiro coeficiente (frequência zero);
    comprima as sequências de zeros utilizando codificação run length;
    aplique codificação de Huffman ao resultado;
    retorne os coeficientes codificados para o bloco
  fim faça
}
```

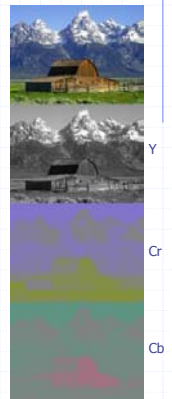
7/4/2008

INF01124

Compressão JPEG

imagem colorida

- ◆ Converta a imagem de RGB para YCbCr
 - ◆ **Y** (luminância) representa o brilho do pixel
 - ◆ **Cb** e **Cr** juntos representam a crominância
- ◆ O olho humano é mais sensível à variações de luminância do que de crominância
 - ◆ Chroma subsampling
 - ◆ 4:4:4 (sem redução); 4:2:2 (redução de 2x na horizontal); 4:2:0 (redução de 2x na horizontal e na vertical)
- ◆ Os canais **Y**, **Cb** e **Cr** são processados de modo análogo



7/4/2008

INF01124

Quantização de Coeficientes

- ◆ A DCT produz coeficientes (números) reais
 - Ocuparia mais espaço que armazenar os próprios pixels
- ◆ A quantização dos coeficientes é obtida por meio de

$$T'(u,v) = \text{round}\left(\frac{T(u,v)}{Q(u,v)}\right)$$

- u e v são os parâmetros da frequência espacial e variam de 0 a 7

$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

Exemplo de uma típica tabela de quantização de luminância

Uma tabela diferente é usada para quantização da crominância

7/4/2008

INF01124

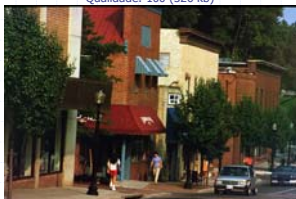
Tabelas de Quantização

- ◆ Os valores contidos na tabela Q podem ser escalados
- ◆ Aumentando-se os valores de Q, aumenta-se o número de coeficientes $T'(u,v)$ iguais a zero
- ◆ Reduzindo-se os valores de Q, tem-se o efeito oposto
- ◆ A taxa de compressão em arquivos JPEG é controlada pelo escalamento da tabela de quantização
- ◆ Parâmetro de qualidade usado em implementações JPEG
 - Maior qualidade → menor número de coeficientes $T'(u,v)$ iguais a zero

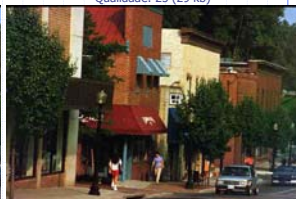
7/4/2008

INF01124

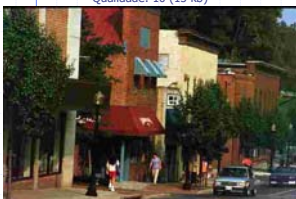
Qualidade: 100 (326 kb)



Qualidade: 25 (29 kb)



Qualidade: 10 (15 kb)



Qualidade: 5 (9 kb)



7/4/2008

INF01124

O Algoritmo JPEG

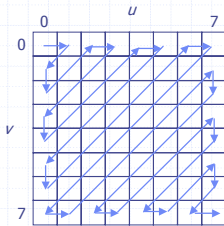
- ◆ Em áreas relativamente homogêneas da imagem (bloco)
 - Vários dos coeficientes da DCT serão suficientemente pequenos para serem zerados pelo processo de quantização
- ◆ Em regiões de transições bruscas (altas frequências)
 - Poucos coeficientes da DCT serão zerados pela quantização
- ◆ O algoritmo JPEG introduz borramento apenas nas áreas da imagem em que é menos perceptível

7/4/2008

INF01124

Ordenamento dos Coeficientes

- ◆ Após a quantização, os coeficientes são colocados em um array 1D por ordem de frequência



- Uso de Delta Encoding para o primeiro elemento de cada bloco (entre blocos)
- Run length encoding para comprimir sequências de zeros
- Huffman para aumentar a compressão
- Descompressão é obtida invertendo todo o processo