

CAP 3. COMPRESSÃO DE DADOS MULTIMÍDIA

Aula 4: Técnicas de Codificação sem perdas: (A)DPCM e LZ*

INE5431 Sistemas Multimídia
Prof. Roberto Willrich (INE/UFSC)
roberto.willrich@ufsc.br

Cap 3. Compressão de Dados Multimídia

Conteúdo:

- Necessidade de compressão
- Entropia: Teorema da codificação da fonte
- Princípios da compressão
- Classificação das técnicas de compressão
- Medição do desempenho de compressão
- Técnicas de compressão sem perdas
 - RLE, Huffman, (A)DPCM, LZ*
- Técnicas de compressão de áudio, vídeo e imagens
- Padrões de compressão multimídia
 - JPEG, MPEG, MPEG-4, H.261, H.263

Técnicas de compressão sem perdas

Codificação Predictiva

- DPCM (PCM diferencial)
 - Técnica mais simples de codificação preditiva
 - Compara símbolos adjacentes e apenas erros de predição são quantizados e codificados
 - Exemplo ilustrativo:
 - Original (amostras de 8bits)
 - 23, 24, 26, 25, 27 ($8 \times 5 = 40$ bits)
 - Compactado com função de predição $a_i = a_{i-1} + \text{erro}$
 - 23, +1, +2, -1, +2
 - Erro de predição tem uma alta probabilidade de ser menor que o valor sendo codificado
 - Erro pode ser expresso com uma quantidade menor de bits
 - No exemplo, usando 4 bits para codificar o erro, o tamanho será $8 + 4 \times 4 = 24$ bits
 - Na descompactação
 - Função de previsão e erro são usados para restaurar o dado original

Técnicas de compressão sem perdas

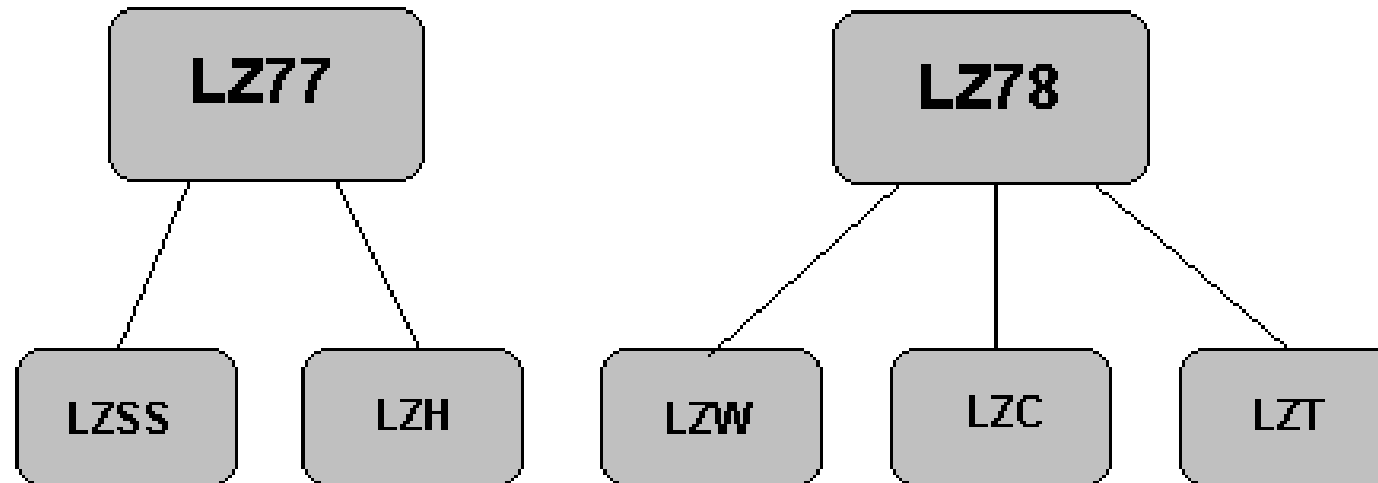
Codificação Predictiva

- ADPCM (DPCM Adaptativo)
 - Existem várias maneiras de implementar ADPCM, a mais comum é variar o tamanho de passo de quantização representado pelos erros
 - Quando o erro é grande, o passo de quantização é maior (gerando perdas de qualidade)
 - Exemplo: se um passo preto-para-branco for detectado, pode-se aumentar o passo de quantificação antes deste passo chegar

Técnicas de compressão sem perdas

Lempel-Ziv (LZ)

- Algoritmos de codificação baseada em dicionário
- Proposta no final dos anos 70, Jacob Ziv e Abraham Lempel
 - Muitas variantes com objetivo de solucionar limitações das versões originais



Técnicas de compressão sem perdas

Aplicações do Lempel-Ziv (LZ):

- UNIX Compression
 - O algoritmo LZC é usado pelo utilitário “compress” do sistema operativo UNIX.
- GIF (Graphics Interchange Format)
 - Muito similar ao “compress” do UNIX, usa o algoritmo LZW.
- Protocolo V.42bis (compressão de dados em Modem)
 - Utiliza o LZT.
- Zip e o gzip
 - usam uma variante do LZ77 combinada com Huffman estático.
- ARJ
 - usa a codificação de Huffman e o algoritmo LZSS.
- WINRAR
 - usa o LZ77 e Huffman.
- WINZIP
 - entre outros algoritmos usa o LZW.

Técnicas de compressão sem perdas

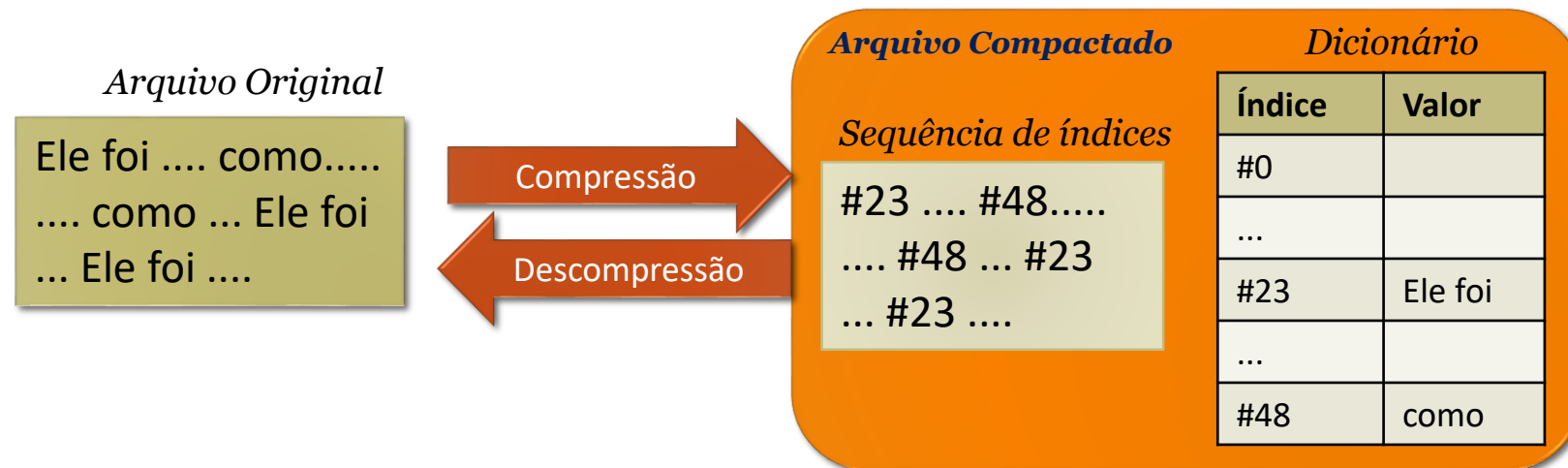
Codificações derivadas do Lempel-Ziv (LZ)

- Explora a redundância de dados -> a repetição de padrões de símbolos no arquivo
 - Baseada na construção de um dicionário de símbolos (grupos de um ou mais símbolos) a partir do fluxo de entrada
- Ilustração em um arquivo de texto
 - Quando uma nova “frase” é encontrada
 - a máquina de compressão adicionada a “frase” ao dicionário
 - um token (índice) que identifica a posição da “frase” no dicionário substitui a frase no documento
 - Se a “frase” já foi registrada
 - ela é substituída pelo token de sua posição no dicionário

Técnicas de compressão sem perdas

Codificações derivadas do Lempel-Ziv (LZ)

- Explora a redundância de dados -> a repetição de padrões de símbolos no arquivo
- Baseada na construção de um dicionário de símbolos (grupos de um ou mais símbolos) a partir do fluxo de entrada
- Exemplo ilustrativo



Técnicas de compressão sem perdas

Codificações derivadas do Lempel-Ziv (LZ)

- Exemplo do poder da codificação
 - Arquivo original de 10000 caracteres (8 bits/caractere)
 - arquivo requer 80000 bits para representá-lo
 - Assumindo que arquivo tem 2000 palavras ou frases das quais 500 são diferentes
 - necessitamos 9 bits como token para identificar cada palavra ou frase
 - precisamos de 9×2000 bits para codificar o arquivo
 - obtemos uma taxa de compressão de 4,4
 - Dicionário armazenando todas as frases únicas deve ser armazenado também



Técnicas de compressão sem perdas

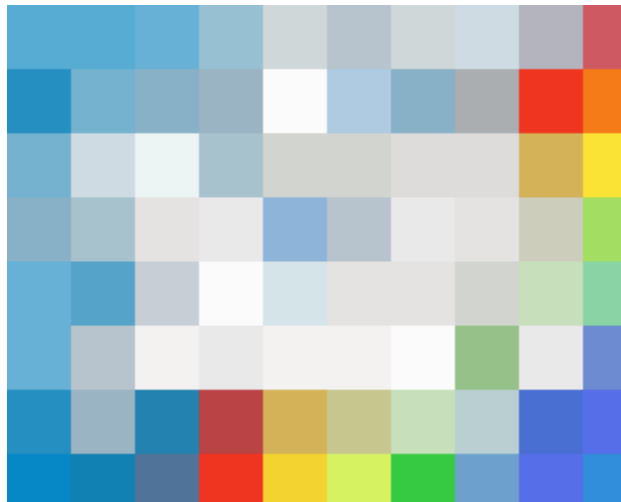
LZW e o formato de imagem GIF

- GIF utiliza a técnica LZW
- GIF é um dos formatos de armazenamento de imagens 256 cores sem perdas
 - imagens com um máximo de 256 cores
 - ao converter imagem true color, com 24 bits/pixel, para o formato GIF, estamos perdendo grande parte da informação de cor
- Taxas de compressão não são grandes
 - em geral 4:1
- Extensão GIF89a permite
 - definir uma cor transparente
 - entrelaçamento
 - animação

Técnicas de compressão sem perdas

LZW e o formato de imagem GIF

- Extensão GIF89a permite
 - definir uma cor transparente
 - entrelaçamento
 - animação



Entrelaçado

Não Entrelaçado

Técnicas de compressão sem perdas

LZW e o formato de imagem GIF

- Algoritmo LZW do GIF era propriedade da Unisys
 - Era do domínio público e a Unisys resolveu passar a cobrar uma taxa pela sua utilização
 - Patentes estão espiadas desde 2006 (pode ser usado livremente)
- Este motivo provocou a definição de uma alternativa válida ao formato GIF
 - formato PNG (Portable Network Graphics)
 - Suporta múltiplos níveis de transparência
 - Correção gama para ajuste da exibição da imagem às características do monitor
 - Entrelaçamento mais avançado que o GIF
 - suporta 48-bit truecolor ou 16-bit escalas de cinza
 - não suporta animação
 - usa os algoritmos LZ77 e de Huffman (DEFLATE)
 - Formatos MNG (Multiple-Image Network Graphics) e APNG
 - Extensões do PNG que suportam animações

Pontos Importantes

(A)DPCM e LZ*

- Entender o princípio geral, vantagens e desvantagens

GIF e PNG

- Saber comparar esses formatos