



Universidade Eduardo Mondlane
Faculdade de Engenharia
Departamento de Engenharia Electrotécnica
Curso de Engenharia Informática

Sistemas Multimédia

Eng. Cristiliano Maculure

Agenda

☐ Codificação e Compressão De Dados Multimédia



Por que Compressão?

- Apesar da expansão das capacidade de comunicação e computação, a demanda das novas aplicações multimédia cresce rapidamente
- Custo de transmissão e armazenagem tem de ser reduzido
- Baixa velocidade dos dispositivos de armazenagem limita as apresentações de médias dependentes do tempo (vídeo, principalmente)
- Definir padrões para facilitar interoperabilidade e implementação em hardware

Por que Compressão?

- Os objetos multimédia podem requerer uma banda passante de rede não disponível
 - Imagem:
 - 18,9Mbits para imagem 1024×768 a 24bpp
 - ~2,5Mbits para imagem VGA a 8bpp
 - Áudio:
 - 64 Kbps (voz, mono, qualidade de telefone)
 - 1,4 Mbps (música, estéreo, qualidade de CD)
 - Vídeo:
 - ~ 332 Mbps para um vídeo YC_RC_B 4:2:2
- Tarifas pela quantidade de dados transmitidos (ou tempo de conexão) – menor quantidade de dados enviados na rede, menor custo

Por que Compressão?

- Armazenamento
 - CD-ROM padrão: 640-700 MB
 - DVD padrão: 4.7 GB ou 8.5 GB (1 ou 2 camadas)
- Taxa de transmissão
 - CD-ROM padrão: 156 Kb/s – atualmente $> 52x = 8 \text{ Mb/s}$
 - DVD padrão: 1.32 MBps (1x) – atualmente $> 26.4 \text{ MBps (20x)}$
 - Ethernet: $> 10 \text{ Mbps}$

Compressão: Princípios

- **Compressão:** codificação de um conjunto de dados de forma a gerar um código final menor que o fonte
- Quando se sabe que um dado possui 50% de informação redundante, por que “pagar” por essa informação?
 - Se a voz humana só produz sons entre 2 e 5.5 KHz, por que transmitir som considerando a largura de 20KHz?
 - Se o uso de vogais é maior em português, por que não codificá-las com menos bits ao invés dos 8 em ASCII?

Compressão : Princípios

- Remoção de redundância (correlação) dos dados
 - Aproveitar-se da forma como o homem “interpreta” os dados
 - Ex: Amostras adjacentes de áudio, remoção de silêncio, similaridade entre linhas “scanneadas” ou amostradas na imagem digital, similaridade entre regiões em um vídeo
- Representação da informação sob diferentes códigos
 - Estatística
 - Codificação

Algoritmos de Compressão

- Duas classes de algoritmos de compressão digital: Algoritmos “sem perda” de informação (*lossless*)
 - Apenas redundância (estatística) dos dados
 - *Codificação por entropia*: dados são uma sequência de bits
 - Entropia = a média mínima de bits necessários para transmitir uma determinada sequência de dados

Algoritmos “com perda” de informação (*lossy*)

- Redundância (estatística) + percepção humana
- *Codificação na fonte* : distinção dos dados relevantes e não relevantes através de processamento na origem

❑ Compressão sem perdas
(entropy coding)





16cm-30cm

Morse code Bracelets

I Love You



Morse code bracelet with a hidden message

Primórdios: Código Morse

A	. _	B	_ . . .	C	_ . _ . . _ . .	D	_ . .
E	.	F	. . _ .	G	_ . . .	H
I	. .	J	. _ . . .	K	_ . _	L	. _ . .
M	_ . .	N	_ .	O	_ . . .	P	. _ . .
Q	_ . . . _	R	. _ .	S	. . .	T	. _ . . .
U	. . _	V	. . . _	W	. _ . .	X	_ . . _
Y	_ . . .	Z	_	1	. _	2	. . _ . . .
3	. . . _ . .	4 _	5	. _	6	_
7	_	8	_	9	_ _	0	_
Ponto	. _ . . . _			Vírgula	_ . . . _		

Repetição de Série

- Conhecida por *Run-Length Encoding* (RLE)
- Técnica que substitui uma sequência “longa” de símbolos por um código “curto” com o número de ocorrências e o valor do símbolo
- Metodo sequencial: Codifica durante a transmissao
- Ex:
 - *Facsimile* : uma linha branca/preta representada por um código seguido do no. de elementos brancos/pretos
 - Regiões de cores semelhantes numa imagem

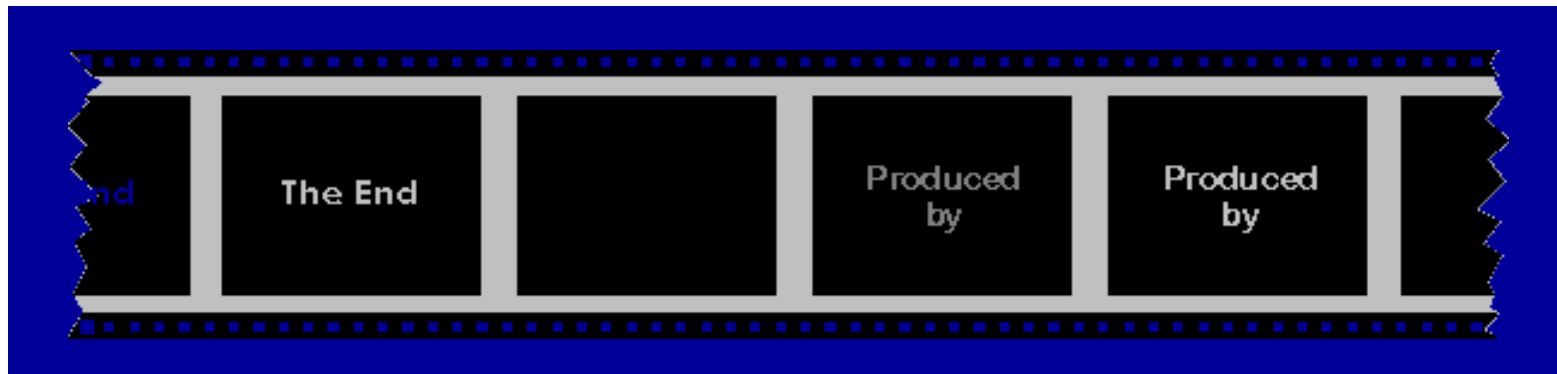
Exemplo RLE



2/3 dos bits originais

Mapeamento de Diferença

- Identificar as diferenças entre símbolos consecutivos e armazenar (transmitir) o valor
- Usar símbolos de referência para calcular os valores reais a partir das diferenças (MPEG faz isso...)



Codificação por Transformada

- Transforma a informação original de um formato para outro mais apropriado para compressão
- Não traz perdas, mas os valores abaixo de certo limiar, podem ser desprezados na codificação
- Exemplo: DCT transforma uma matriz de pixels (imagem) em uma matriz equivalente de componentes de frequências espaciais
 - Frequência espacial: taxa de mudança na magnitude dos valores dos pixels na matriz da imagem
 - Componentes de alta frequência podem ser desprezadas, pois não são captadas pelo sistema visual humano
 - JPEG e MPEG se utilizam desta observação

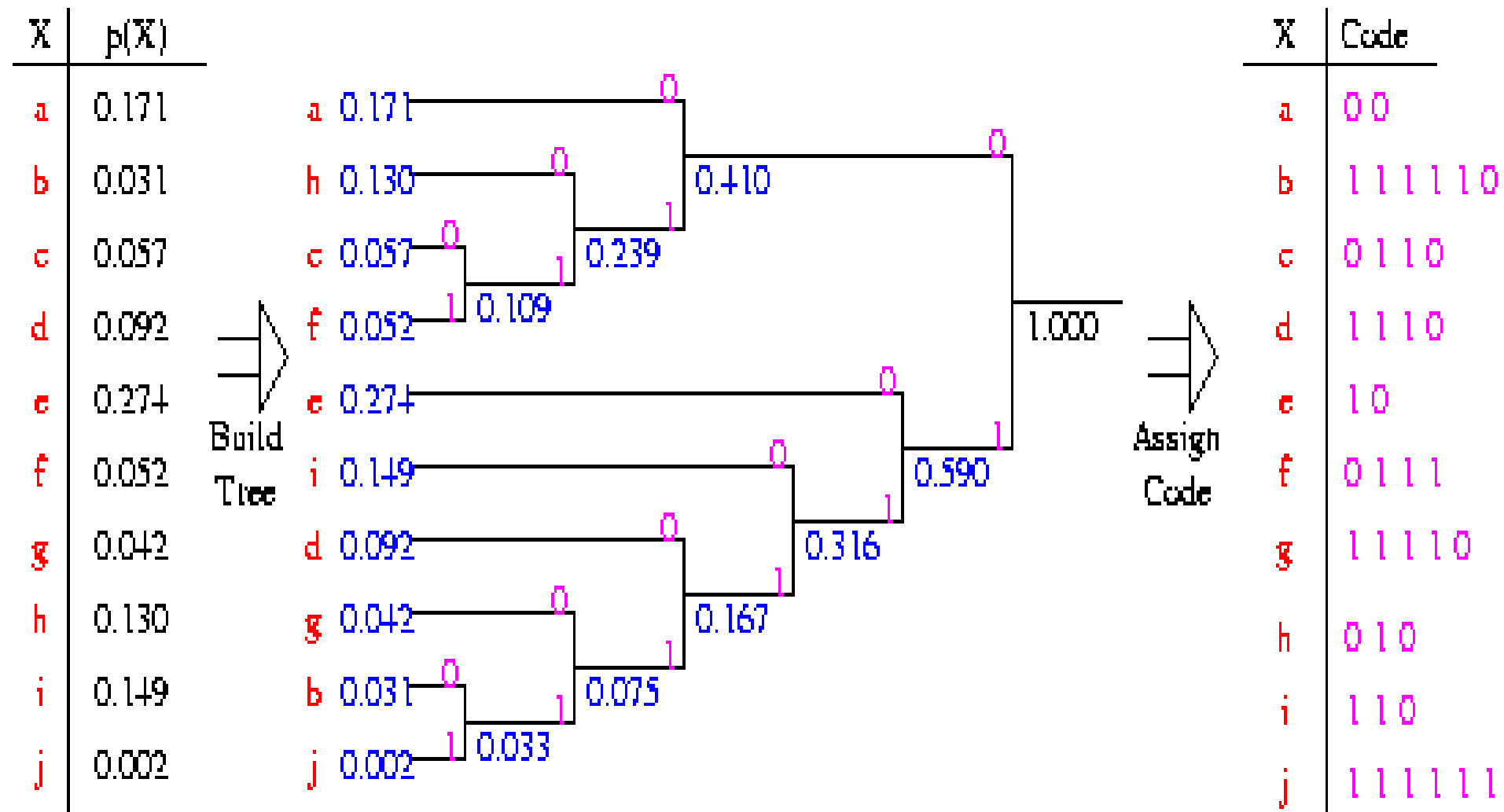
Codificação Huffman

- David Huffman (1952)
- Usado “pesadamente” na compressão de textos
- Códigos de tamanho variável associados à frequência de ocorrência do símbolo nos dados
- Uma árvore binária de construção *bottom-up* é usada para a codificação (decodificação), evitando ambiguidades

Codificação de Huffman

- A informação a ser comprimida é analisada e uma árvore desbalanceada é construída baseada na frequência relativa de cada caractere (*Árvore do Código de Huffman*).
- A *Árvore do Código de Huffman* é uma árvore digital binária (0 e 1 para cada sub-árvore).
- Símbolos codificados nas folhas e o caminho da raiz até uma folha indica a sequência de bits que representa o símbolo.

Codificação Huffman



Codificação Huffman

- Apenas os símbolos usados são codificados...
- Tipicamente: 30 a 55% de compressão
- Variação das tabelas de código para melhor desempenho :
Língua, aplicações, etc.
- Problemas: Exige o envio de uma tabela de código e um tempo de codificação alto!

Estudar a codificação de Huffman

Faculdade

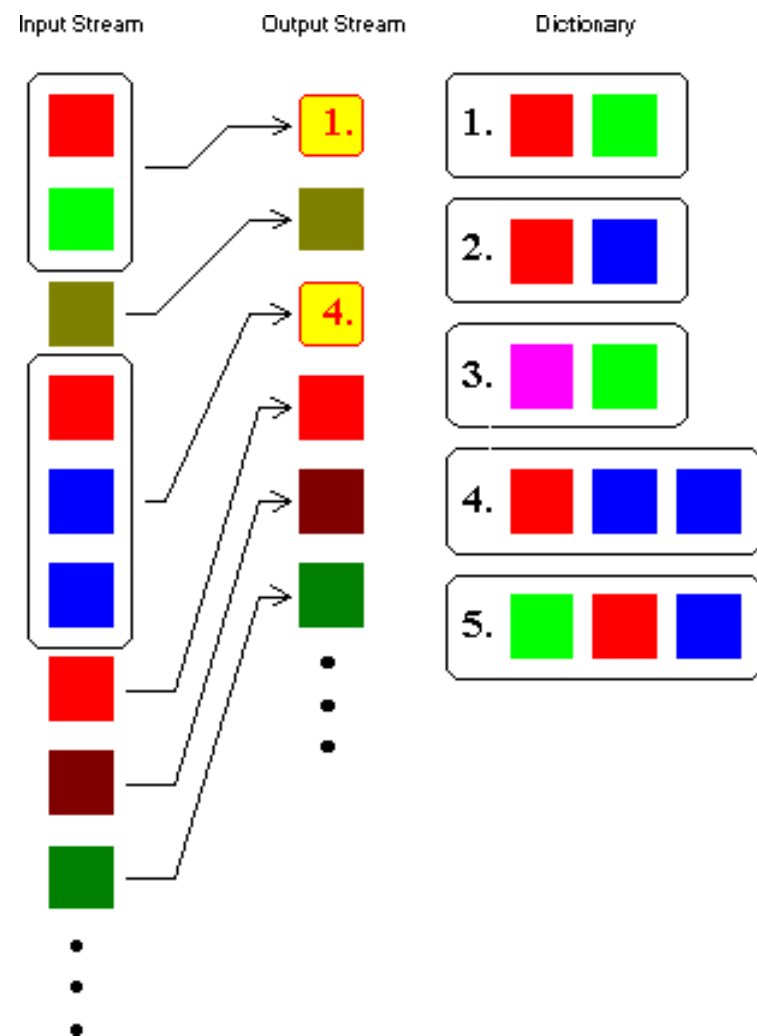
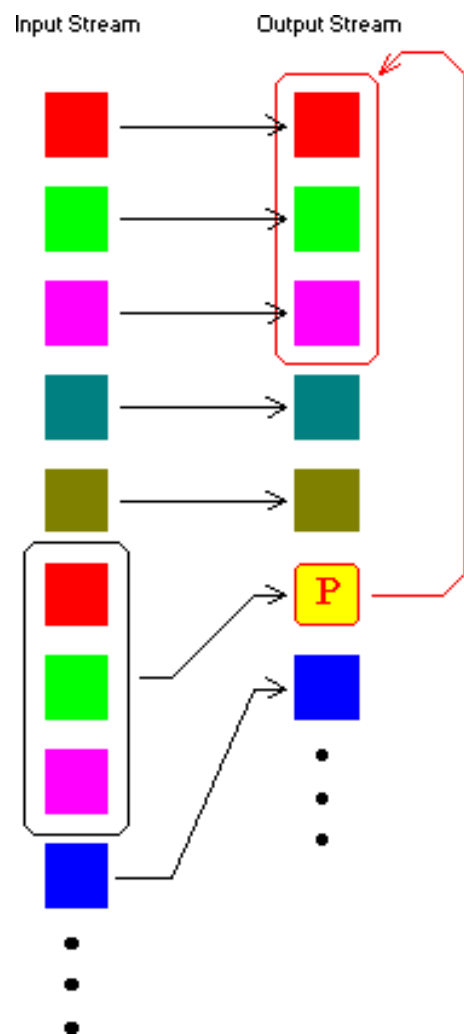
Codificação por Dicionário

- Lempel e Ziv (1970) : LZ77 e LZ78 (patente)
- Símbolos são lidos e comparados com entradas de um dicionário (tabela hash) : se o símbolo existe, a posição da tabela é gerada como saída
- O resultado da compressão é o próprio dicionário combinado com os dados
- Os dados podem ser comprimidos “*on-the-fly*” sem a necessidade de geração de tabelas e de maneira dinâmica

Métodos de Dicionário: Dois grupos

- 1o. grupo: *MMCAVELA*
 1. Verifica se a sequência que deve ser comprimida já apareceu anteriormente;
 2. Ao invés de repeti-la, define um ponteiro para essa ocorrência na própria sequência
- 2o. grupo: *Sqmo*
 1. Cria um dicionário de “frases” que ocorrem na sequência de entrada;
 2. Quando a frase é novamente encontrada, simplesmente o índice da frase no dicionário é codificada.

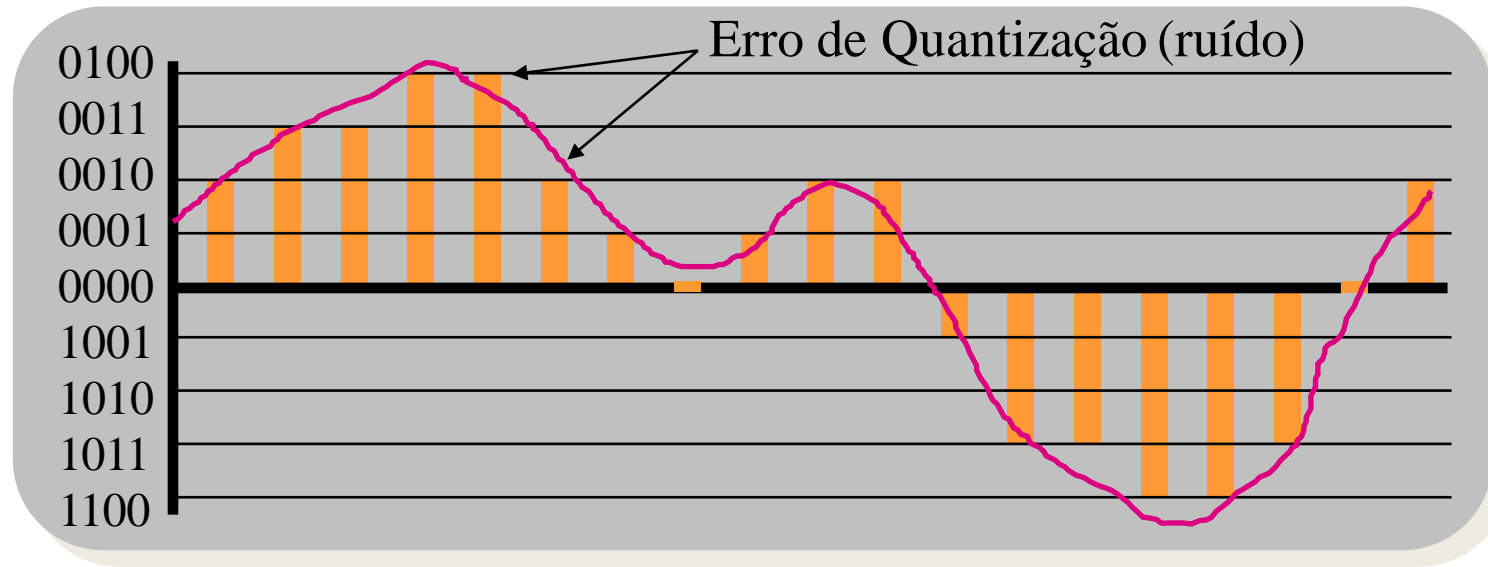
Exemplos



COMPRESSÃO DE ÁUDIO DIGITAL



PCM

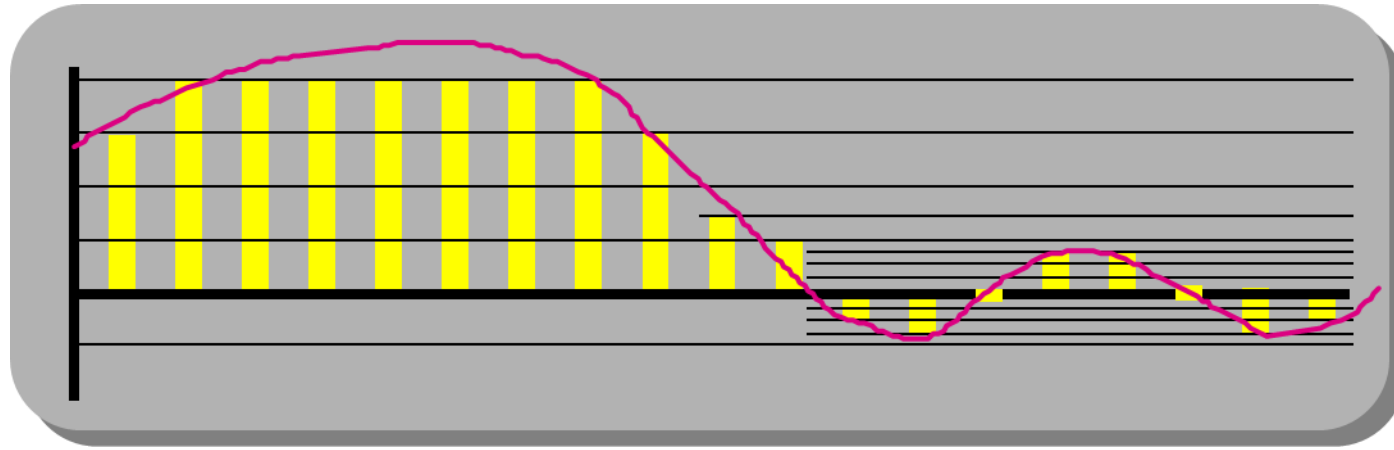


- constrói um sinal digital a partir de uma série de impulsos
- formato obtido após amostragem e quantificação do sinal áudio
- é simplesmente uma sequência de amostras descomprimidas
- **vantagens** boa utilização da largura de banda; menos sensível a ruído
- **desvantagens** débito binário muito elevado

Amostragem Não-Linear

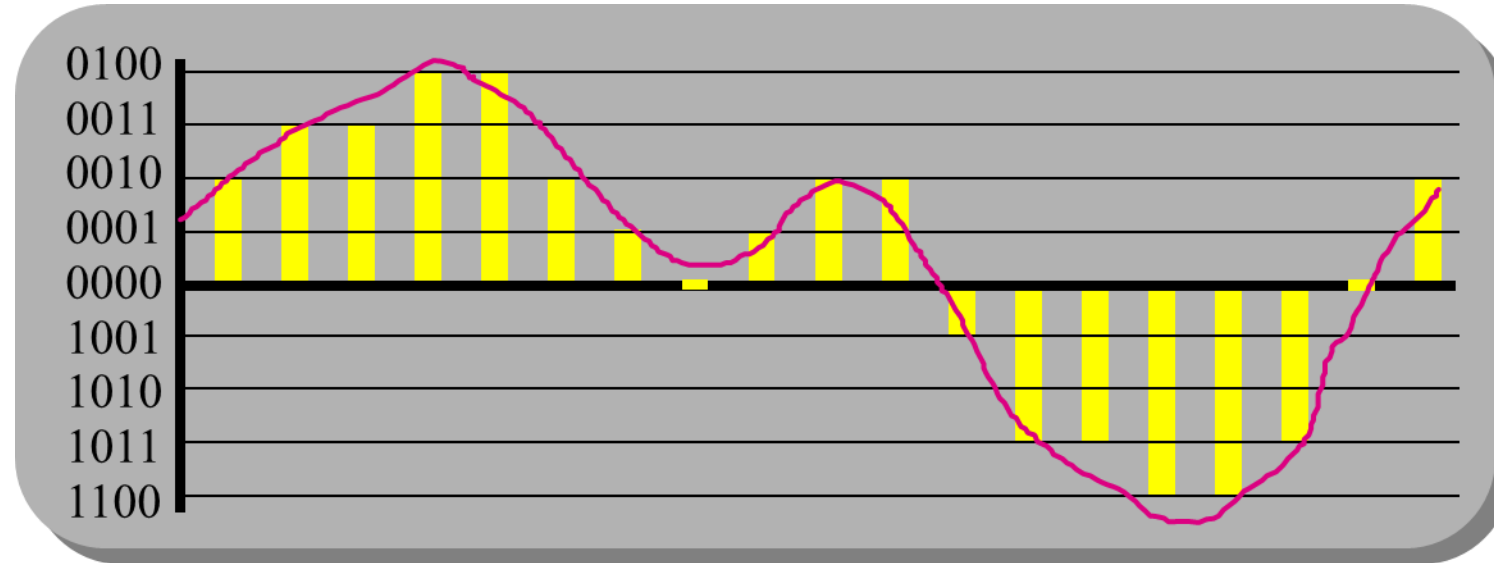
- Com 8 bits por amostra, um sinal de ruído poderia ser identificado...
- Em particular, não estariam sendo “bem codificados” os valores para baixas amplitudes
- Uma solução é amostrar mais “densamente” as baixas amplitudes e menos “densamente” as amplitudes mais altas → menos bits no total!
- Este é exatamente o comportamento da escala logarítmica

μ -Law companding (ITU Rec. G.711)



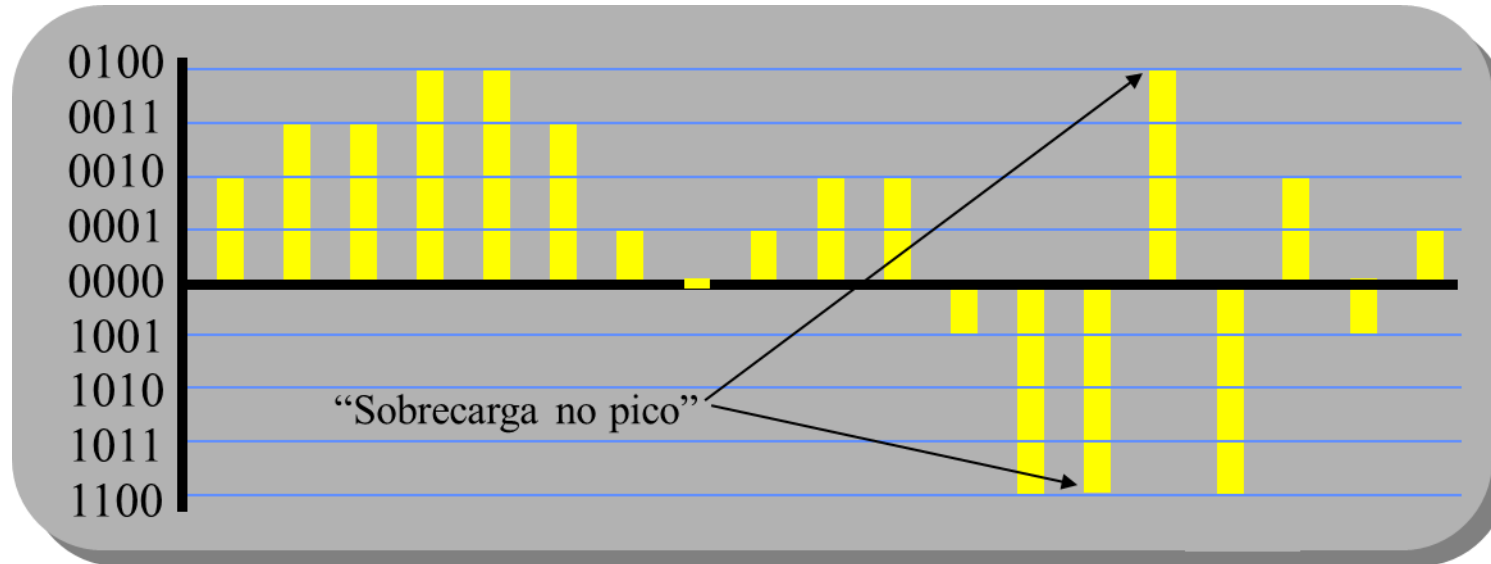
- Quantização não-linear da amplitude do sinal
 - O número de bits diminui logaritmicamente com o nível do sinal amostrado
 - Amostras de baixa amplitude são amostradas com maior precisão que as de alta amplitude
 - Isso porque os humanos são menos sensíveis a mudanças de intensidade (*loudness*) do som que em relação a períodos de “silêncio”

Differential PCM – DPCM



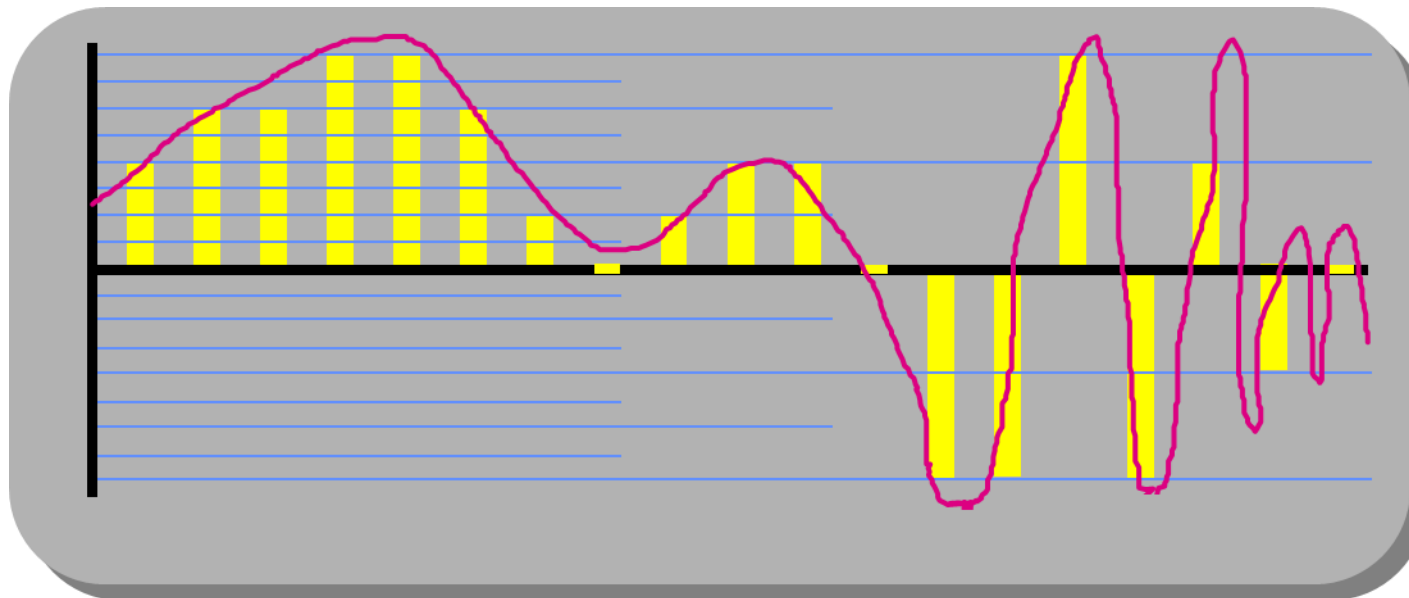
- Explora a redundância temporal das amostras
- A diferença entre 2 amostras consecutivas de n-bits podem ser representadas com menos de n-bits
- Transmissão da diferença ao invés da amostra

Problemas do DPCM



- As diferenças em sinais de alta-frequência não podem (próximas a Nyquist) não podem ser representadas em poucos bits!
 - O erro introduzido leva a sérias distorções do sinal em frequências mais altas

Adaptive DPCM – ADPCM



- Usa menos bits para codificar as diferenças entre altas-frequências e mais bits para diferenças entre amostras de baixas-frequências
- Usa os valores das amostras anteriores para estimar as mudanças seguintes no sinal
- utilizado: CD-i, AIFF, WAV

Compressão de Voz

- *APC (Adaptive Predictive Coding)*
 - Sinal de áudio dividido em pequenos segmentos de comprimento fixo, cada um analisado separadamente. Um conjunto de coeficientes otimizado para cada segmento é selecionado (8Kbps com qualidade aceitável)
- *LPC (Linear Predictive Coding)*
 - Analisar a onda do sinal de áudio para identificar um conjunto de características de percepção do sinal, as quais são quantizadas e utilizadas para gerar um sinal sonoro comparável ao sinal original (som “sintético”, mas inteligível e taxas de 1.2 a 2.4 Kbps)
- *CELP (Code Excited LPC)*
 - Evolução do LPC, utilizando um conjunto limitado de segmentos (*waveform templates*). Cada amostra digitalizada que compõe uma template é codificada por diferença. A template que codifica um segmento é escolhida de modo que a diferença tenha um erro de quantização mínimo (recom. ITU-T G.728, G.729, G.729(A) e G.723.1)

Geralmente usados em telefonia!

MP3: MPEG Layer 3

- Portável, independente de plataforma, padrão “*de facto*” para áudio de qualidade na Internet
- Mantém arquivos de som de 16 bits 44.1 KHz (sem ruídos, perda de faixa dinâmica ou da resposta em frequência)
- Modelo “de codificação perceptual” que “simula” os dados perdidos na compressão:
 - Até 12:1 de compressão sem perdas significantes

Codificação Perceptual

- O efeito varia de acordo com a frequência em questão.
- A largura das curvas de frequência (largura de banda crítica - LBC) define a faixa de frequências afetadas pelo mascaramento resultante de um dado sinal.
 - Frequências abaixo de 500Hz a LBC é de 100Hz.
 - Frequências acima de 500Hz a LBC aumenta de forma quase linear em múltiplos de 100Hz.
 - Um sinal de 1kHz (2x500) teria uma LBC de 200Hz (2x100) enquanto que um sinal de 5kHz (10x500) teria uma LBC de 1kHz (10x100).
- As frequências afetadas podem ser calculadas e eliminadas do código a ser enviado.

Codificação Perceptual

Modelo Psicoacústico - Mascaramento Temporal

- Após escutar um som alto, o ouvido humano fica surdo a sons mais baixos por um intervalo de tempo → mascaramento temporal.
- Depois que um som alto termina são necessários alguns décimos de segundo para o ouvido humano “se recuperar” e a amplitude do sinal ouvido reduza. Neste intervalo de tempo o ser humano é incapaz de escutar sons com amplitude abaixo de um certo limite. Tais sons não precisam, portanto, ser transmitidos.

MPEG Layer 3

Qualidade	banda	modo	<i>Bitrate</i> (Kbps)	compressão
Telefone	2.5 kHz	mono	8	96:1
Melhor que OC	4.5 kHz	mono	16	48:1
Melhor que AM	7.5 kHz	mono	32	24:1
Similar a FM	11 kHz	stereo	56...64	26...24:1
Similar a CD	15 kHz	stereo	96	16:1
Melhor que CD	15 kHz	stereo	112..128k	14..12:1

TPC

Pesquisar essas ferramentas para striming

☐ *OBS Studio* Shelton

☐ *Adobe Flash Media Live Encoder* Falisherto

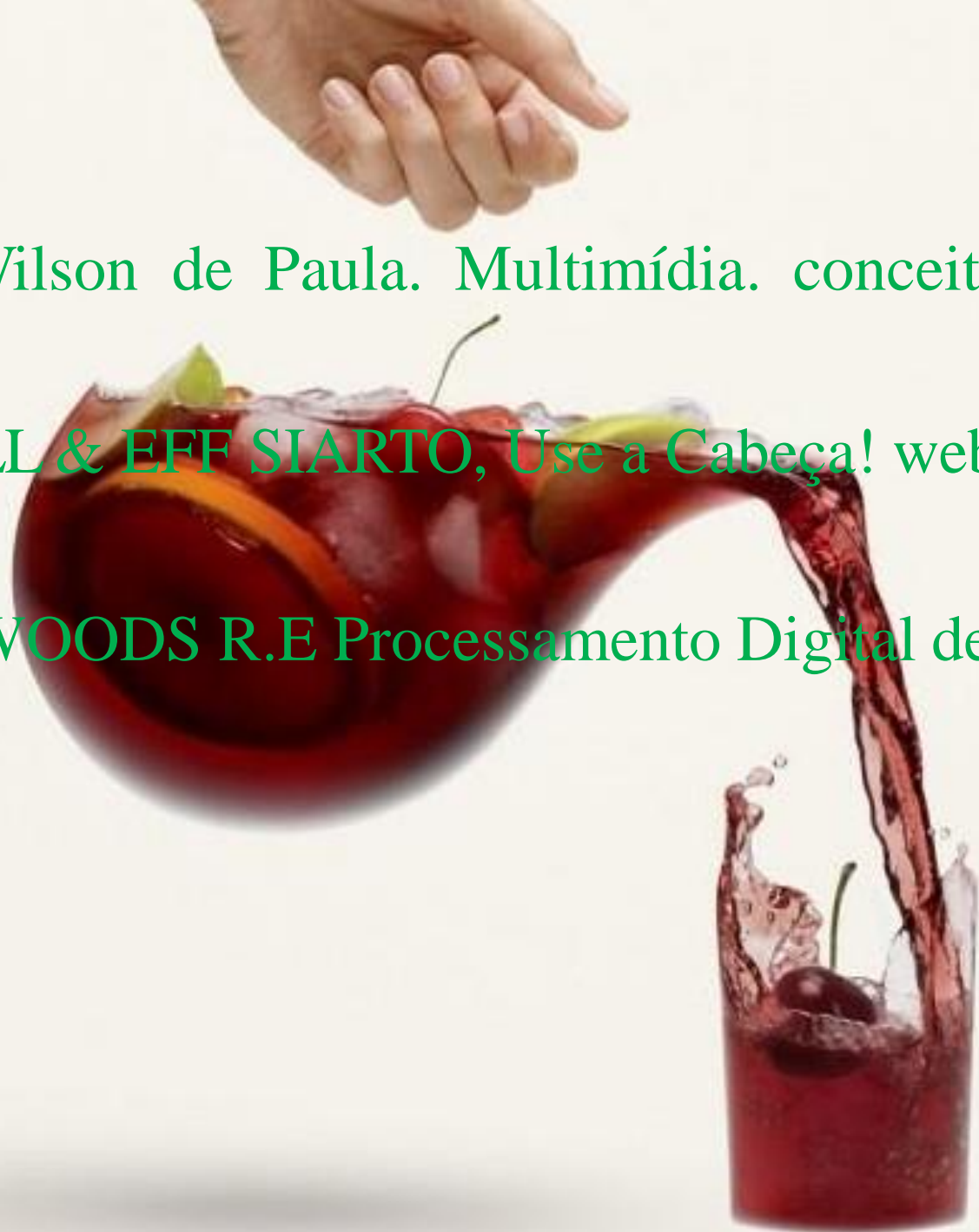


Referências

PAULA FILHO, Wilson de Paula. Multimídia. conceitos e aplicações. Rio de Janeiro: LTC, 2009.

J ETHAN WATRALL & EFF SIARTO, Use a Cabeça! web Design, 1ª Edição, Ed Alta Books, 2009

GONZALEZ R.C. WOODS R.E Processamento Digital de Imagens, 3ª Edição, Ed. Pearson, 2010.



OBRIGADO!

