

# Capítulo 3

# Compressão de Dados

# Multimídia

Roberto Willrich  
INE - CTC-UFSC

# Cap 3. Compressão de Dados Multimídia

- Conteúdo:

- Necessidade de compressão
- Princípios da compressão
- Classificação das técnicas de compressão
- Medição do desempenho de compressão
- Técnicas de compressão sem perdas
  - RLE, Huffman, LZW (GIF)
- Técnicas de compressão de áudio, vídeo e imagens
  - Técnicas de compressão de voz
  - Técnicas de compressão de som
- Padrões de compressão multimídia
  - JPEG, MPEG, MPEG-4, H.261, H.263

# Necessidade da compressão

- Técnicas de compressão são essenciais para as aplicações multimídia, devido
  - Ao grandes requisito de espaço para armazenamento de dados multimídia
  - A largura de banda da rede que não permite a transmissão de mídia em tempo-real
  - A pequena largura de banda dos dispositivos de armazenamento que não podem apresentar dados multimídia em tempo-real

# Requisitos das Informações multimídia

- Requisitos de espaço para armazenamento

Aplicações	Requisitos de Armazenamento (MBytes)
Livro de 500 páginas	1
100 imagens monocr.	7
100 imagens coloridas	100
1h de áudio qual. telefone	28,8
1h de Áudio-CD	635
1h Vídeo qual. VHS	24300
1h TV	97000
1h HDTV	389000

- Aplicação multimídia típica contém
  - 30 minutos de vídeo, 2000 imagens e 40 minutos de som estéreo
  - aplicação necessitaria de 50 GBytes para armazenar o vídeo, 15 Gbytes para armazenar as imagens e 0,4 GBytes para armazenar o áudio (total de 65,4 Gbytes)
- ☞ É necessária a utilização de técnicas de compressão de dados multimídia para viabilizar o armazenamento

# Requisitos das Informações multimídia

- Requisitos de largura de banda

Aplicações	Taxa de bits (Kbps)
CD-Audio	1.411
DAT	1.536
Telefone Digital	64
Radio digital, long play DAT	1.024
Vídeo de qualidade televisão	216.000
Vídeo de qualidade VHS	54.000
HDTV	864.000



- Transmissão de som de qualidade CD não compactado
  - é possível em redes locais
    - 10 Mbits/s (Ethernet) e 100 Mbits/s (fast Ethernet)
  - redes de media e longa distância este taxa torna-se atualmente inviável
- Transmissão de vídeo de qualidade televisão
  - incompatível com qualquer rede local e transmissão em WAN

# Necessidade da compressão

- Pequena largura de banda dos dispositivos de armazenamento
  - Não é possível apresentar vídeo não compactado em tempo-real devido a taxa de bits insuficiente de alguns dispositivos de armazenamento
  - Dispositivo de armazenamento deveria ter uma taxa de 30 MBytes/s para apresentar um vídeo em tempo real com um quadro de 620x560 pixeis a 24 bits por pixel a 30 fps
    - 1x no CD = 150 kBps (velocidade para cd áudio)
    - 1x no DVD = 1,32 MBps
    - tecnologia de CD-ROM de hoje fornece uma taxa de transferência de 7,62 MBps (x52) a 10,8 MBps (x70)
- ☞ Única solução é compactar o dado antes de armazenar e descompactar ele antes da apresentação



# Necessidade da compressão

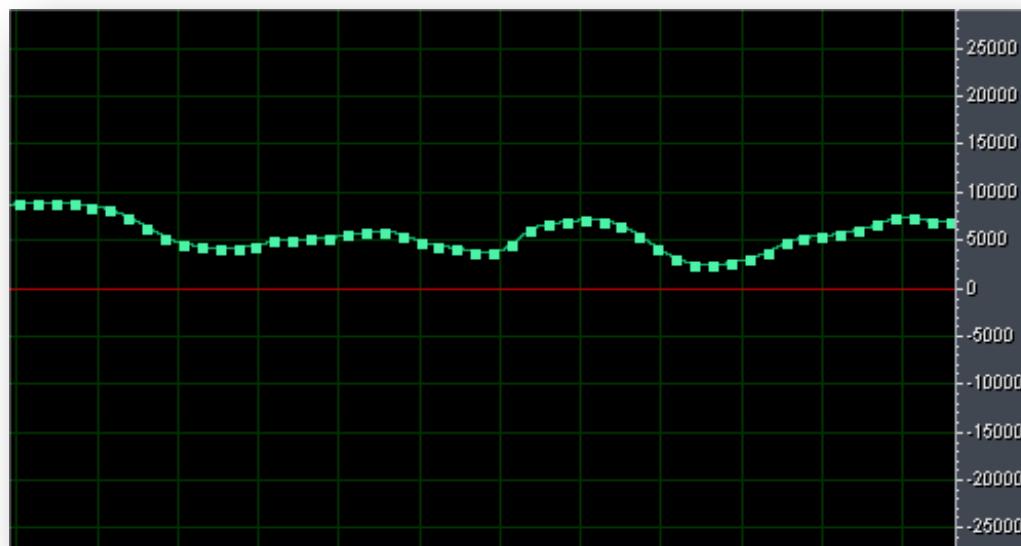
- Conclusão
  - É necessário compactação afim de armazenar, apresentar e transmitir informações multimídia
    - técnicas de compressão modernas reduzem os requisitos de armazenamento e portanto os requisitos de largura de banda da rede e do dispositivo de armazenamento

# Princípios da Compressão de Dados

- Fatores explorados pelas técnicas de compressão
  - Redundância de dados
  - Propriedades da percepção humana
- Redundância de Dados
  - Representação de dados multimídia
    - áudio digital é uma série de valores amostrados
    - imagem é uma matriz de valores amostrados (píxeis)
    - vídeo é uma seqüência de imagens apresentadas numa certa taxa
  - Amostras vizinhas não são inteiramente diferentes
    - valores vizinhos são de algum modo relacionados (redundância)
  - Remoção da redundância não altera o significado do dado

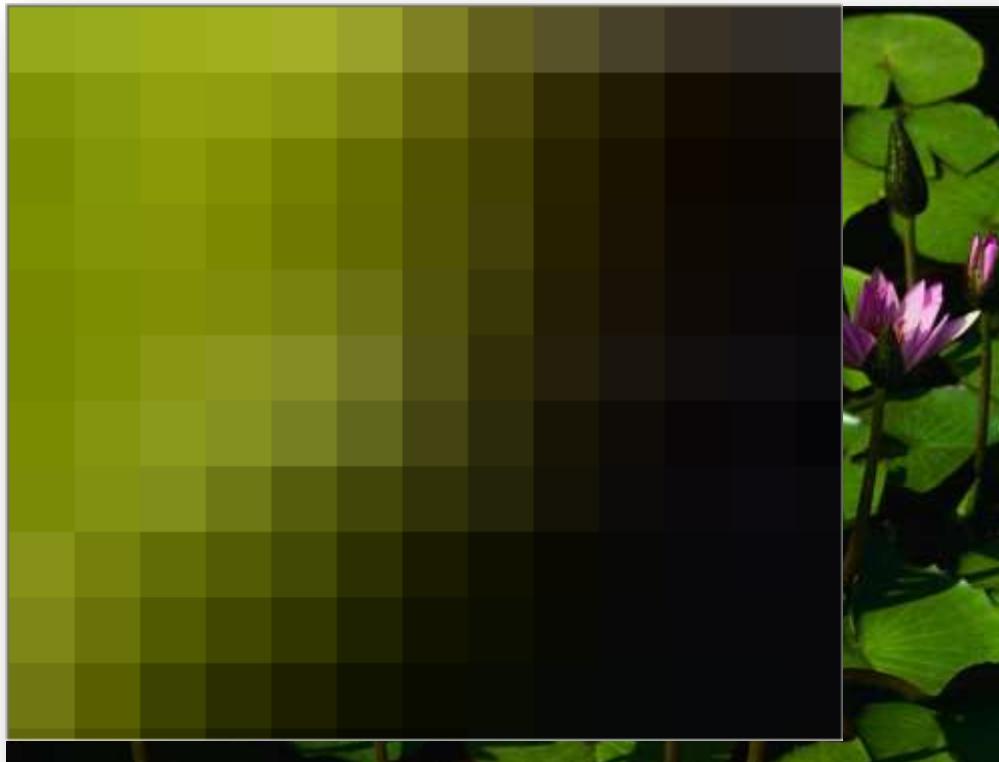
# Princípios da Compressão de Dados

- Redundância em áudio digital
  - Amostragens adjacentes são similares:
    - próximo valor pode ser previsto baseado no valor atual
      - técnicas de compressão: **Codificação preditiva**



# Princípios de Compressão: Redundância

- Redundância em imagem digital
  - Amostras vizinhas são similares
    - chamada de redundância espacial
      - removida utilizando técnicas de codificação predictiva ou outras



# Princípios de Compressão: Redundância

- Redundância em vídeo digital
  - Vídeo é uma seqüência de imagens
    - imagens tem redundância espacial
  - Imagens vizinhas são normalmente similares
    - **redundância temporal**
      - removida utilizando técnicas de codificação predictiva ou outras



# Princípios de Compressão: Propriedades Percepção Humana

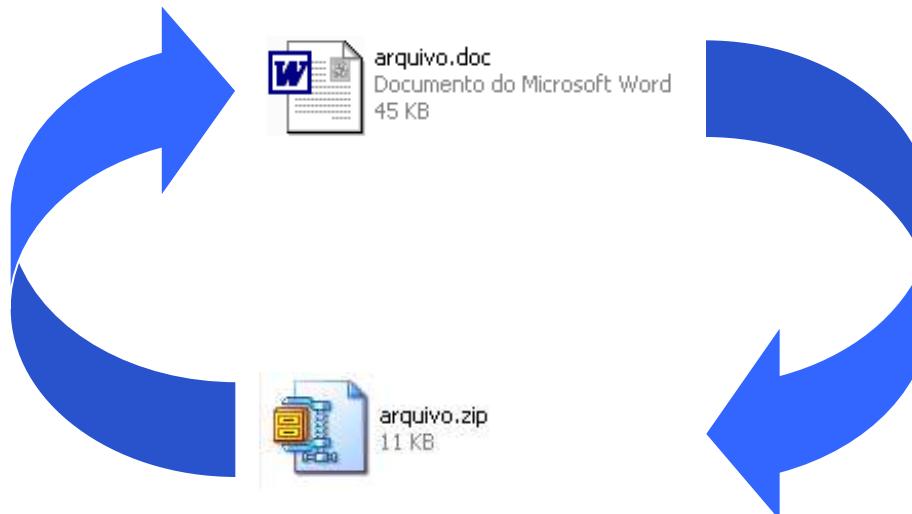
- Humanos não são perfeitos
  - Podem tolerar alguns erros de informação ou perdas sem afetar a efetividade da comunicação
    - versão compactada não necessita representar exatamente a informação original
  - Em dados alfanuméricos não se tolera qualquer erro ou perda
- Algumas informações são mais importantes para a percepção humana que outras
  - Técnicas de compressão podem remover informações desnecessárias
    - áudios mascarados, intensidade luminosas/cor



# Classificação das Técnicas de Compressão

- **Sem perda (Codificação por Entropia):**

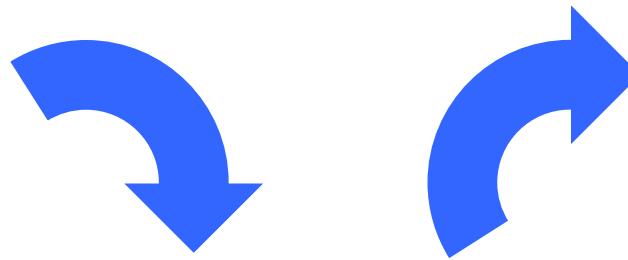
- Dado original pode ser exatamente reconstruído (reversível)
  - Técnica genérica: trata cadeias de bytes sem levar em conta seu significado
  - técnicas de compressão sem perda exploram apenas estatísticas de dados (redundância de dados)
    - baixas taxa de compressão



# Classificação das Técnicas de Compressão

- **Com perda (codificação na origem)**

- utilizado em dados multimídia onde erros e perdas são toleráveis
  - leva em consideração a semântica dos dados
    - removendo dados irrelevantes compactando o dado original
  - utiliza propriedades da percepção humana
    - altas taxa de compressão



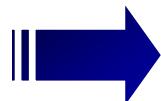
Ninféias.jpg  
800 x 600  
Imagem no formato JPEG



# Classificação das Técnicas de Compressão

- Codificações Híbridas

- Combinam técnicas com perda e sem perdas
  - várias técnicas são agrupadas para formar uma nova técnica de codagem
- Taxa de compressão mais altas



Descarta dados irrelevantes para a percepção (com perdas)



Elimina Redundância (sem perdas)



Ninféias.jpg  
800 x 600  
Imagem no formato JPEG

# Parâmetros de Desempenho

- **Taxa de compressão**
  - tamanho do dado original/tamanho do dado após a compressão
    - para sem perdas: quanto maior esta taxa melhor é a técnica
- **Qualidade da mídia reconstituída** (para técnicas com perda)
  - medida em SNR (Razão Sinal/Ruído)
  - maior SNR melhor é a qualidade
- **Complexidade de implementação e velocidade de compressão**
  - importante para aplicações tempo-real (como videoconferência)
    - compressão e descompressão devem ser realizadas em tempo-real
  - para aplicações de obtenção e apresentação de informação:
    - velocidade de compressão não é muito importante
    - velocidade de descompressão é importante

# Técnicas de compressão sem perdas

- Codificação RLE (Codificação por entropia)
  - Técnica simples de compressão de dados
  - Aplicação: formatos padrões como PCX, BMP (RLE) e Photoshop
    - BMP RLE suporta 256 cores
  - Dados podem ser compactados através da supressão de seqüências de mesmos símbolos
  - Um exemplo simples
    - Original:  
WWWWWWWWWWWWWWBWWWWWWWWWWWWWWBWWWWWWWWWWWW  
WWWWWWWWWWWWWWWWWWWWWWWWWWBWWWWWWWWWWWWWWWW  
WWWW
    - Compactado: 12W1B12W3B24W1B14W

# Técnicas de compressão sem perdas

- Codificação RLE (Codificação por entropia)
  - Tem diversas variações
    - Seqüências idênticas são substituídas por um símbolo especial, número de ocorrências e o símbolo repetido
      - Original: UHHHHHHIMMG1223
      - Compactado: U!6HIMMG1223
      - Se o símbolo especial ocorrer no dado de entrada, ele deve ser substituído por dois símbolos
        - » entrada: U!HIIID
        - » saída: U!!H!5ID
      - Técnica não é utilizada para seqüências menores que 4
        - » exemplo: U!6HI!2MG1223 (não a compactação)
    - Algoritmo pode ser facilmente otimizado
      - pode-se substituir sequências maiores que um
      - requer que o tamanho da seqüência seja codificado ou pode-se usar um caractere especial de fim
        - » entrada: UFYUGDUFHUFHUFHUFHUFHBFD
        - » saída: UFYUGD!5UFH\$BFD

# Técnicas de compressão sem perdas

- Codificação RLE (Codificação por entropia)
  - Mais uma variação (Modified Huffman coding)
    - Considere a codificação Run-Length de uma imagem binária de FAX cujas primeiras linhas de varredura são mostradas a seguir

```
ImagenFAX={  
    11111111110000000000000000000011111111111111  
    0000000000000111111111111110000000000000000  
}
```

- Código RLC=[....11,22,17,EOL,0,14,20,16,EOL,...]

# Técnicas de compressão sem perdas

- Codificação RLE (Codificação por entropia)
  - Fator de compressão depende do dado de entrada
    - Demonstração usando BMP RLE



Nome	Tamanho	Tipo
Ninféias24bits.bmp	1.407 KB	Imagen de bitmap
Ninféias8bits.bmp	470 KB	Imagen de bitmap
Ninféias8bitsRLE.bmp	389 KB	Imagen de bitmap
Jazz Man24bits.bmp	1.650 KB	Imagen de bitmap
Jazz Man8bits.bmp	552 KB	Imagen de bitmap
Jazz Man8bitsRLE.bmp	210 KB	Imagen de bitmap

# Técnicas de compressão sem perdas

- Codificação Run-Length

- Só traz ganhos relevantes se houver grandes agrupamentos de símbolos iguais
- As principais aplicações são imagens bitmap
  - em imagens com grandes espaços envolvendo uma só cor
  - em imagens geradas por computador
    - onde os dados estão agrupados de forma mais geometricamente definida

# Técnicas de compressão sem perdas

- Codificação de Huffman (Codificação Estatística)
  - Método que atribui menos bits a símbolos que aparecem mais freqüentemente e mais bits para símbolos que aparecem menos
  - Idéia usada no código de Morse

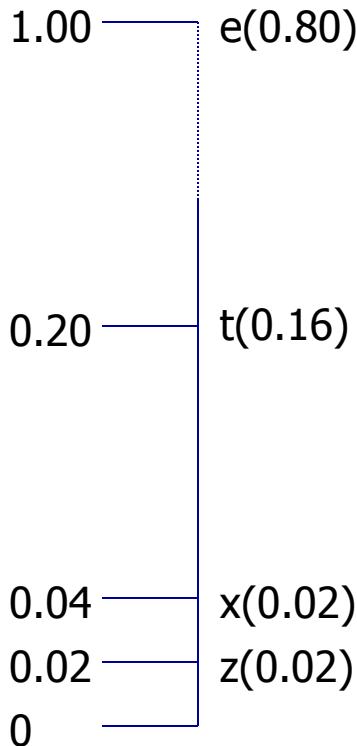
A	..-	M	--	Y	...--	6	-.....
B	-...	N	-.	Z	---..	7	--....
C	-.-.	O	---	Ä	.---	8	---... —
D	-..	P	.--.	Ö	---.	9	-----.
E	.	Q	---.-	Ü	...--	.	...---
F	...-.	R	.-.	Ch	----	,	-----
G	--.	S	...	0	-----	?	...---
H	.....	T	-	1	.----	!	...--.
I	..	U	..-	2	...--	:	---...
J	...--	V	.....	3	....--	“	...---*
K	-.-	W	.---	4	....-	‘	-----.
L	-...-	X	-...-	5	.....	=	-....-

# Técnicas de compressão sem perdas

- Codificação de Huffman (Codificação Estatística)
  - Exemplo de funcionamento:
    - suponha um arquivo de 1000 símbolos: e, t, x, z.
      - frequência de ocorrência: e = 0.8, t = 0.16, x = 0.02 e z = 0,02
    - necessitamos de 2 bits para representar cada um dos 4 símbolos
      - e = 00, t = 01, x = 10 e z = 11
      - arquivo terá um tamanho de  $2 \times 1000 = 2000$  bits
    - usando Codificação de Huffman podemos usar quantidades diferentes para representar estes símbolos (de acordo com a frequência de ocorrência)
      - e = 0, t = 10, x = 110 e z = 111
    - tamanho do arquivo seria  $1000 \times (1 \times 0.8 + 2 \times 0.16 + 3 \times 0.02 + 3 \times 0.02) = 1240$ 
      - apesar de x e z terem sido representados com um maior número de bits, o arquivo será menor pois eles ocorrem menos

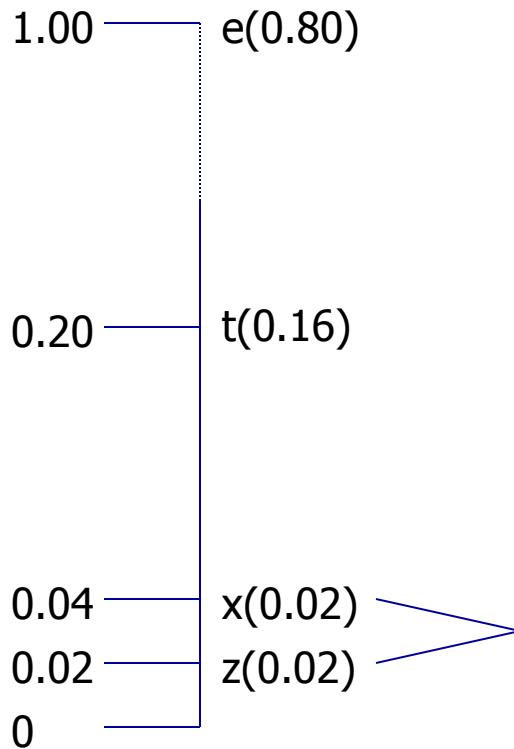
# Técnicas de compressão sem perdas

- Codificação de Huffman (Original)
  - Geração dos códigos Huffman
    - a) colocação dos símbolos ao longo de uma linha de probabilidade acumulada (probabilidade aumenta de baixo para cima)
      - símbolos de mesma frequência: colocar em qualquer ordem



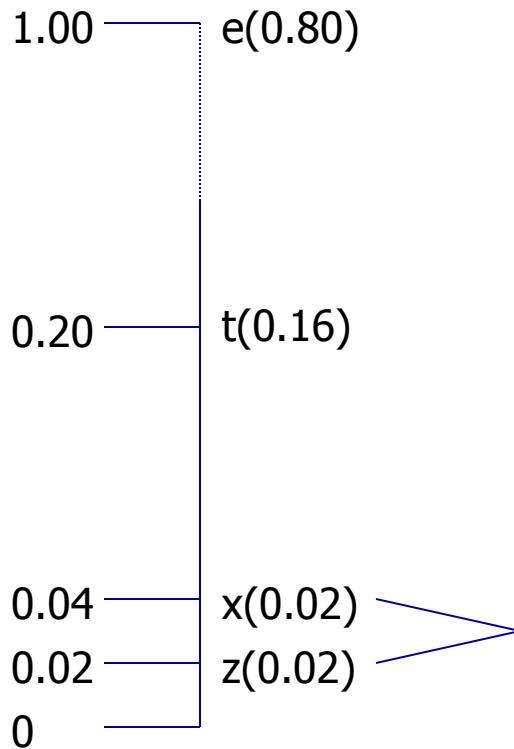
# Técnicas de compressão sem perdas

- Codificação de Huffman
  - Geração dos códigos Huffman
    - b) Junta-se os dois símbolos de menor probabilidade a um nó para formar dois ramos na árvore



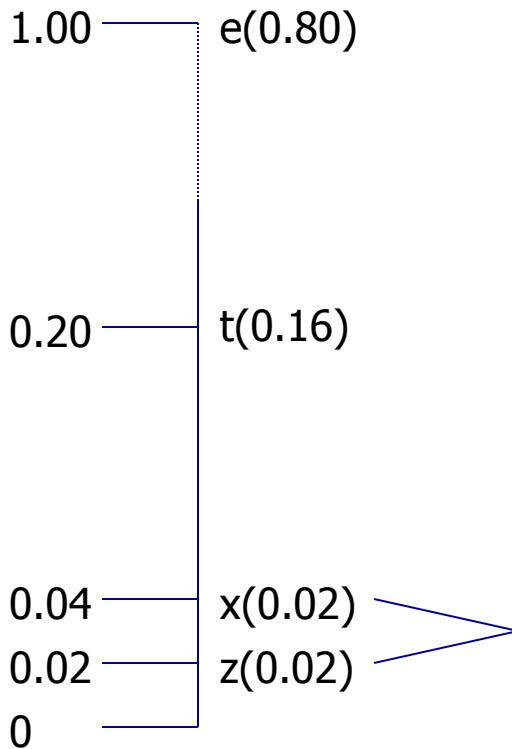
# Técnicas de compressão sem perdas

- Codificação de Huffman
  - Geração dos códigos Huffman
    - b) Junta-se os dois símbolos de menor probabilidade a um nó para formar dois ramos na árvore



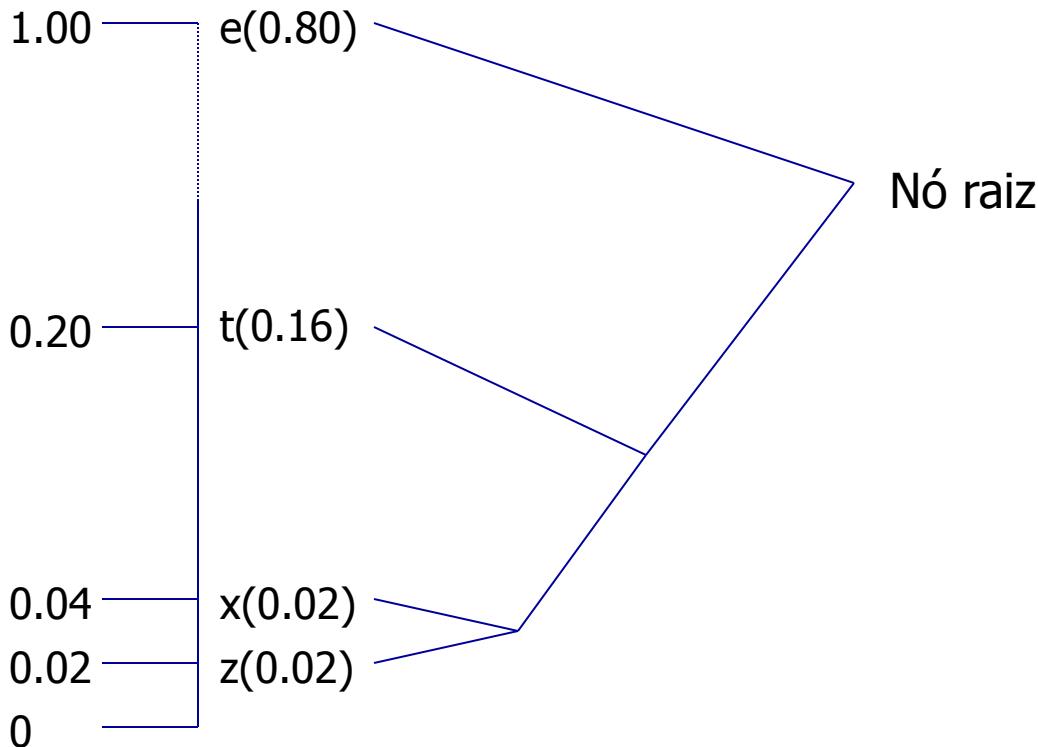
# Técnicas de compressão sem perdas

- Codificação de Huffman
  - Geração dos códigos Huffman
    - c) Nova árvore formada é tratada como um símbolo único com a probabilidade igual a soma dos símbolos ramos



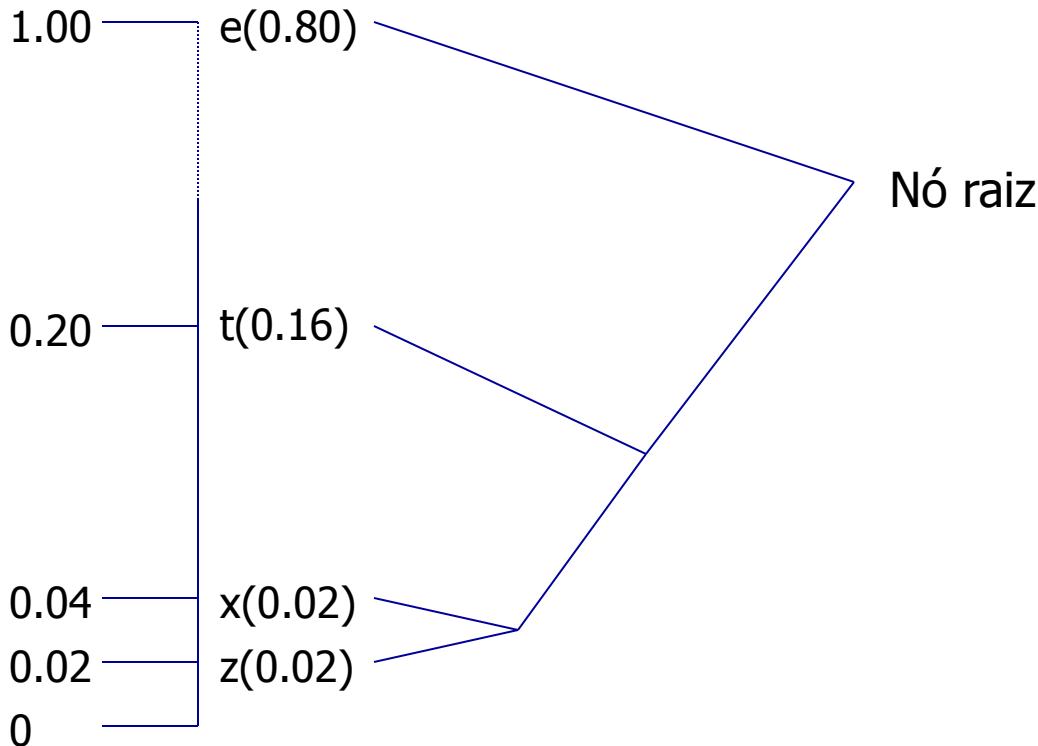
# Técnicas de compressão sem perdas

- Codificação de Huffman
  - Geração dos códigos Huffman
    - d) Repita b) e c) até que todos os símbolos sejam inseridos na árvore
      - último nó é chamado de raiz



# Técnicas de compressão sem perdas

- Codificação de Huffman
  - Geração dos códigos Huffman
    - d) Repita b) e c) até que todos os símbolos sejam inseridos na árvore
      - último nó é chamado de raiz

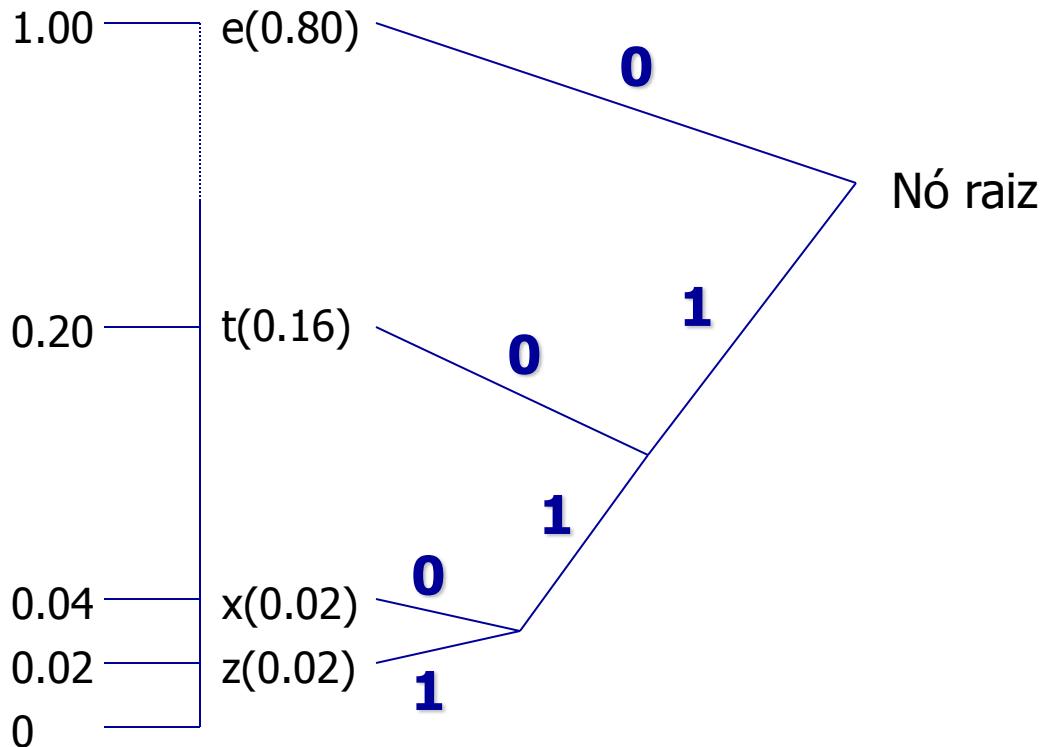


# Técnicas de compressão sem perdas

- Codificação de Huffman

- Geração dos códigos Huffman

- e) Partindo do nó raiz, atribua bit 0 ao ramo de maior prioridade e bit 1 ao ramo de menor prioridade de cada nó



# Técnicas de compressão sem perdas

- Codificação de Huffman

- Geração dos códigos Huffman

- f) Código para cada símbolo é obtido montando códigos ao longo do caminho entre nó raiz ao símbolo

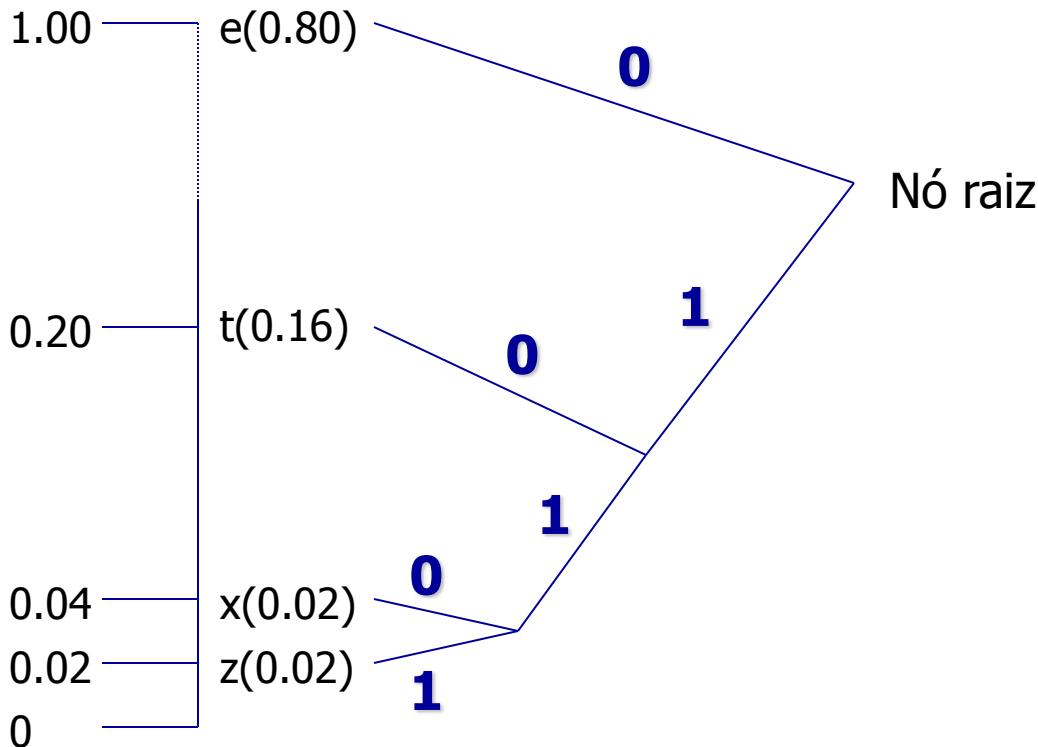


Tabela chamada  
de **codebook**

Símbolo	Código
e	0
t	10
x	110
z	111

# Exemplo de Codificação de Huffman

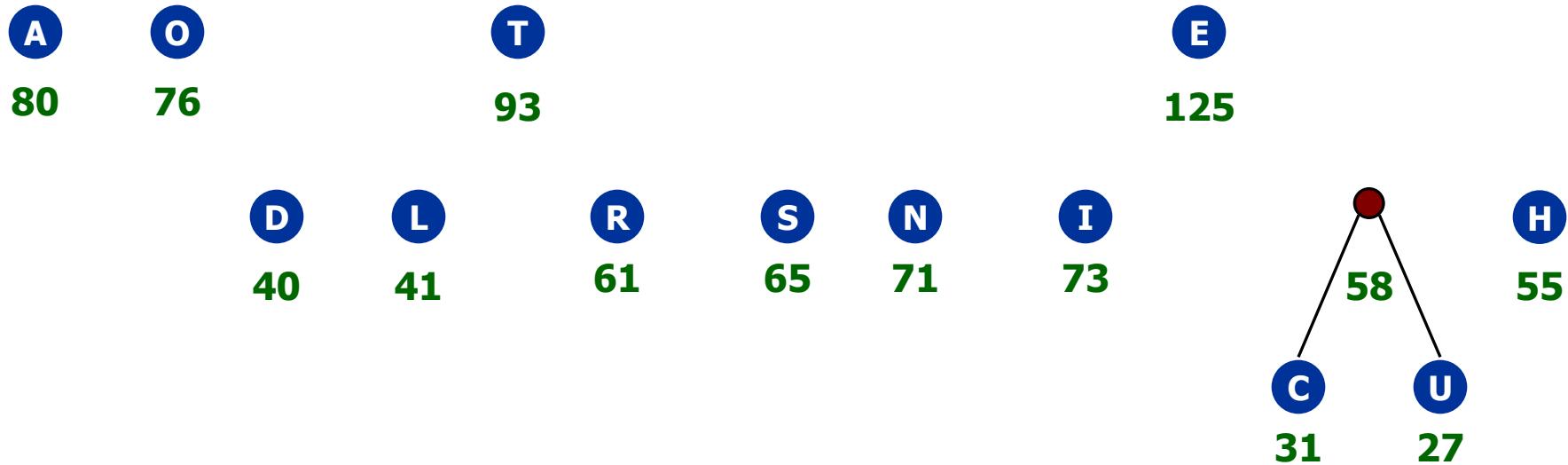
- Exercício mais completo
  - Freqüências dos caracteres
  - Gere a tabela de Huffman para o arquivo.

Char	Freq	Fixo
E	125	0000
T	93	0001
A	80	0010
O	76	0011
I	73	0100
N	71	0101
S	65	0110
R	61	0111
H	55	1000
L	41	1001
D	40	1010
C	31	1011
U	27	1100
Total	838	4.00

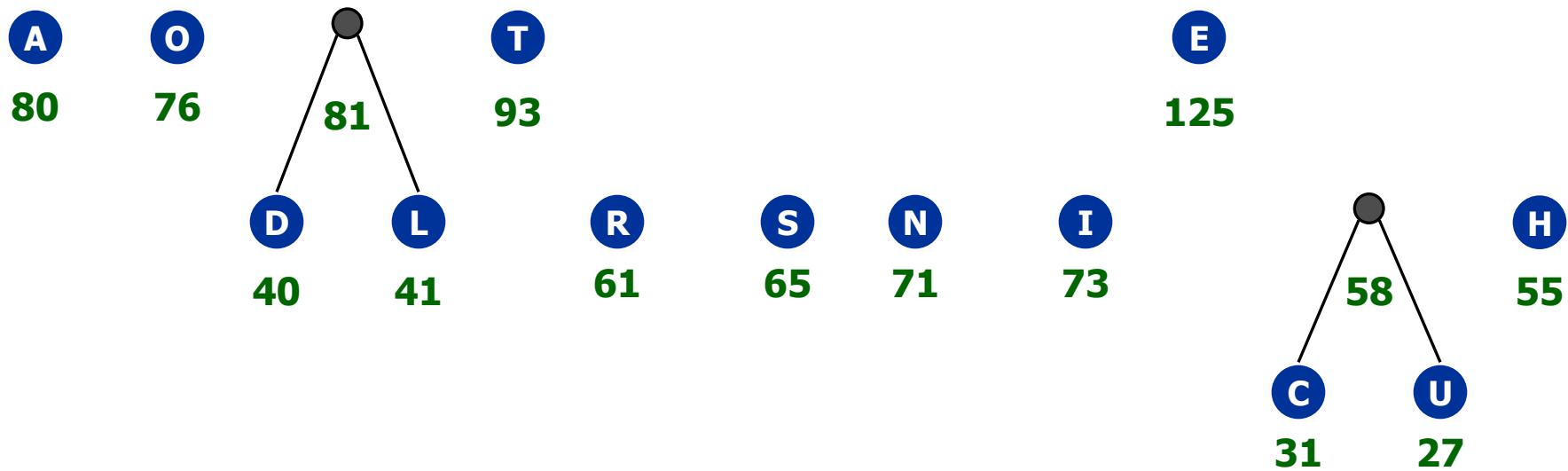
# Exemplo de Codificação de Huffman

A	O	T	E
80	76	93	125
D	L	R	I
40	41	61	73
S	N	H	
65	71	55	
C	U		
31	27		

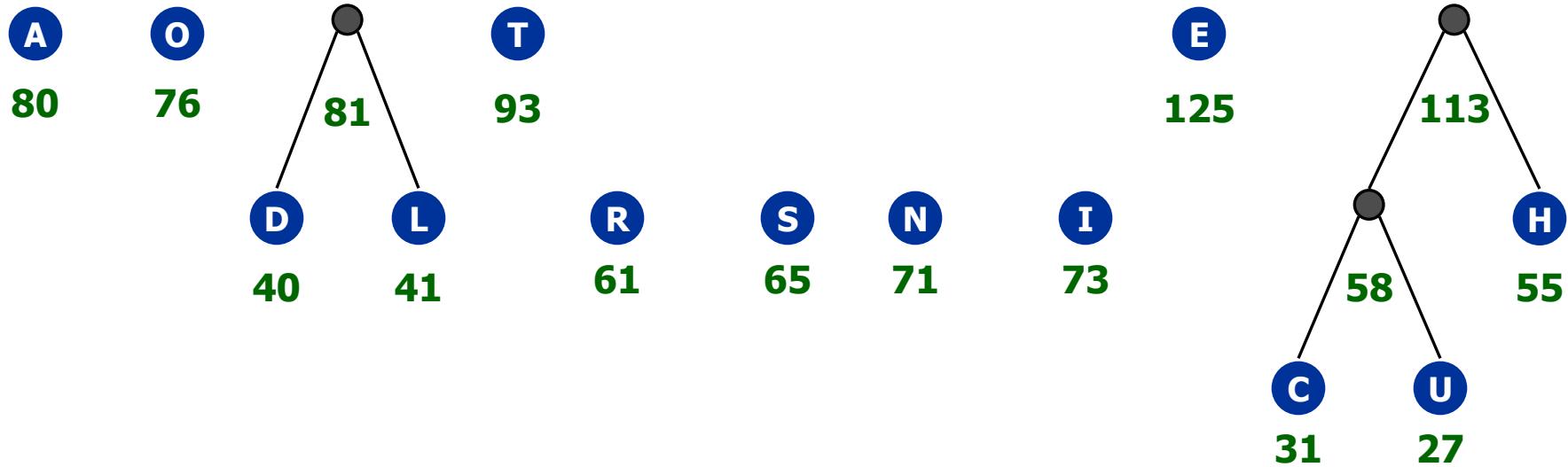
# Exemplo de Codificação de Huffman



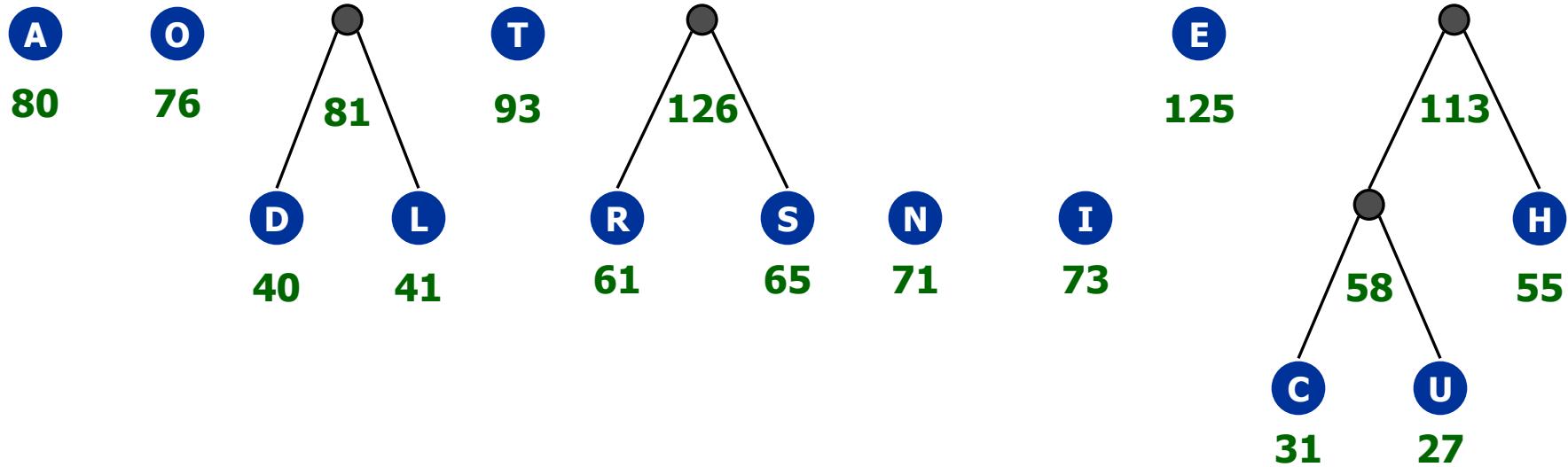
# Exemplo de Codificação de Huffman



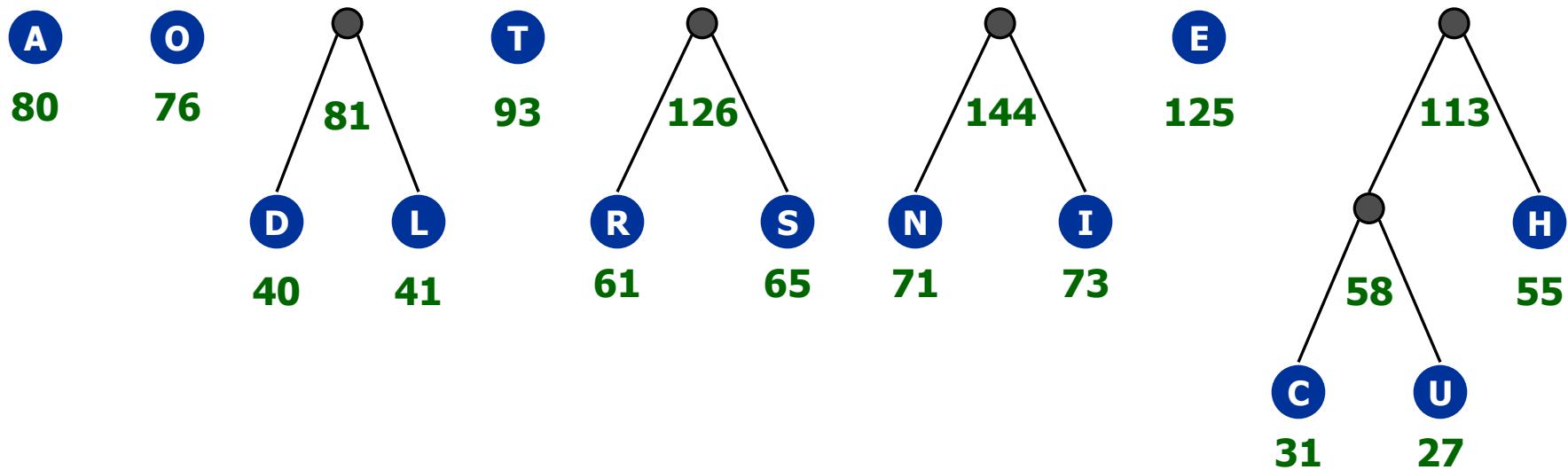
# Exemplo de Codificação de Huffman



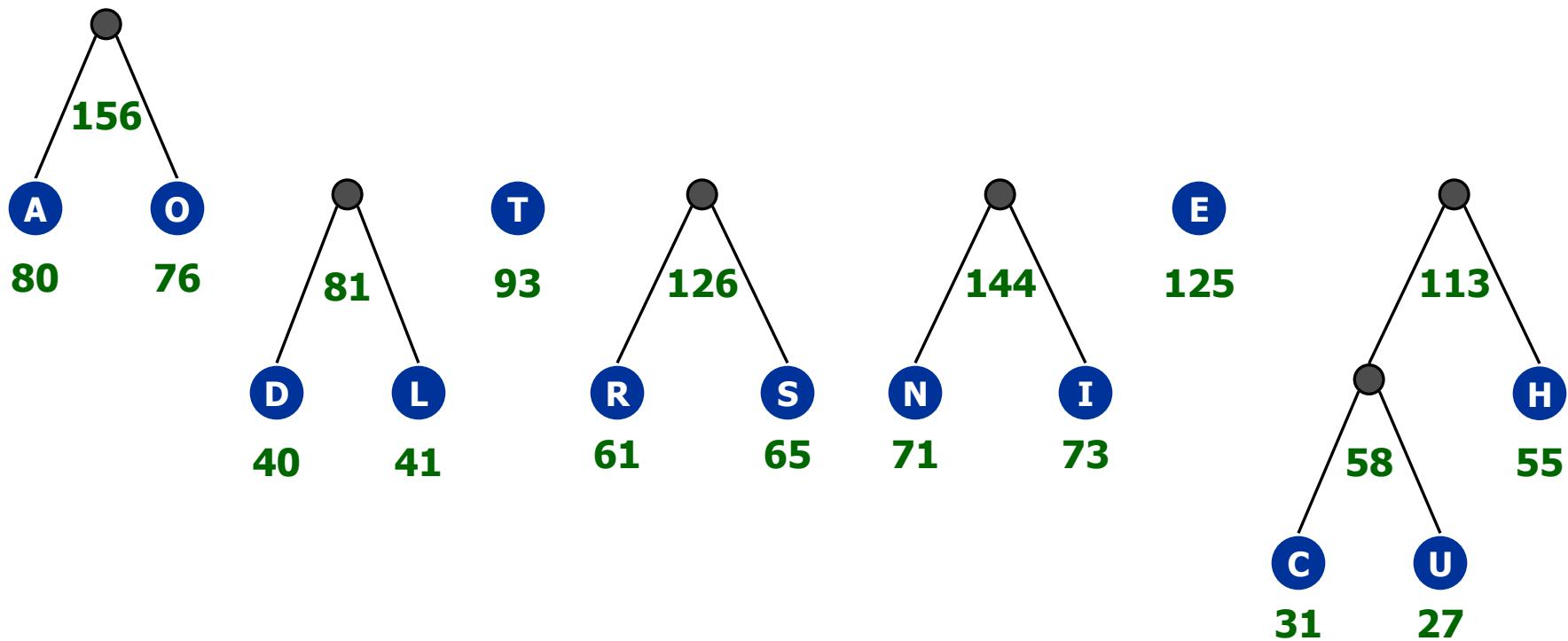
# Exemplo de Codificação de Huffman



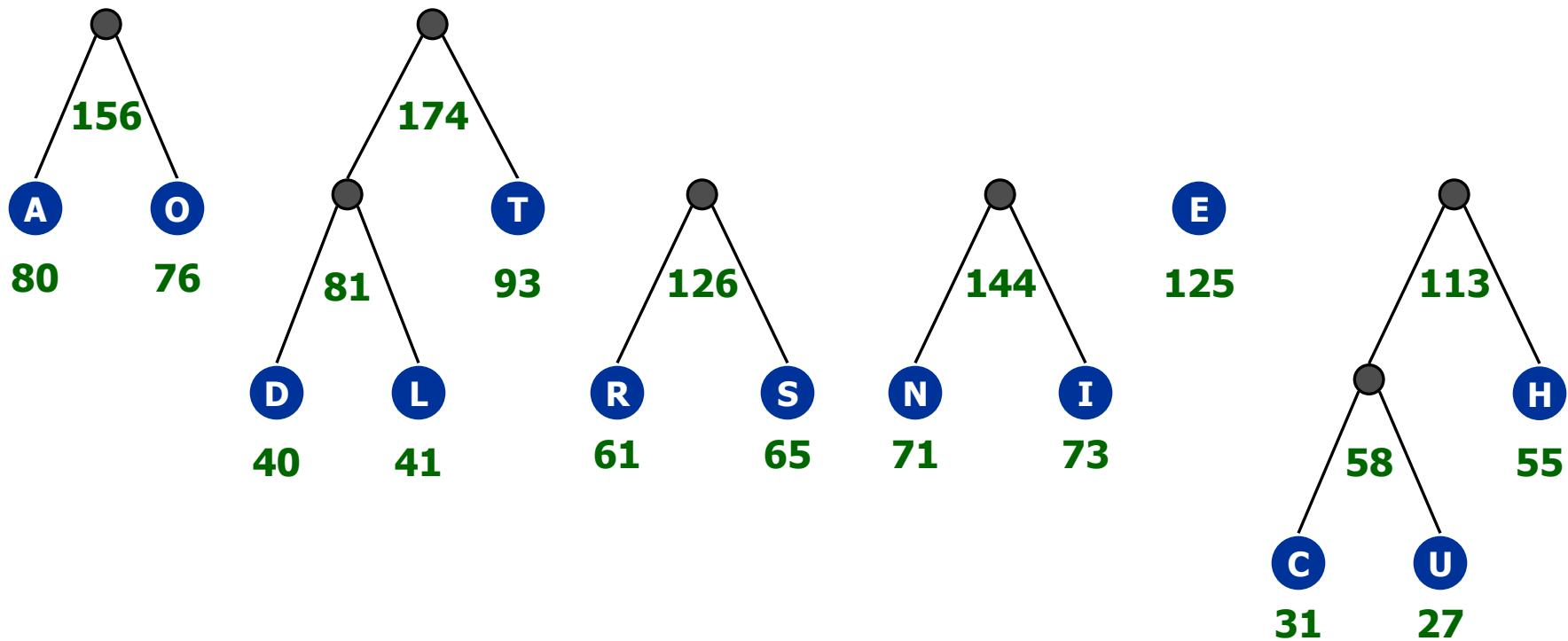
# Exemplo de Codificação de Huffman



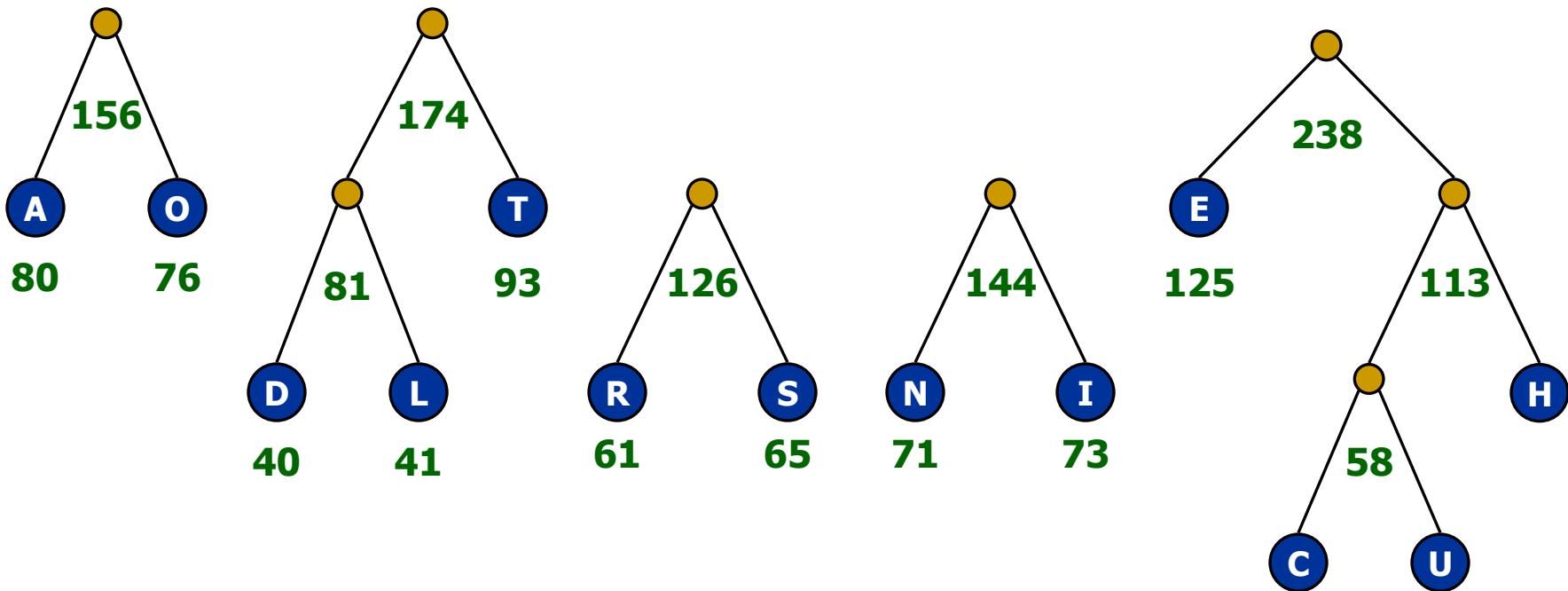
# Exemplo de Codificação de Huffman



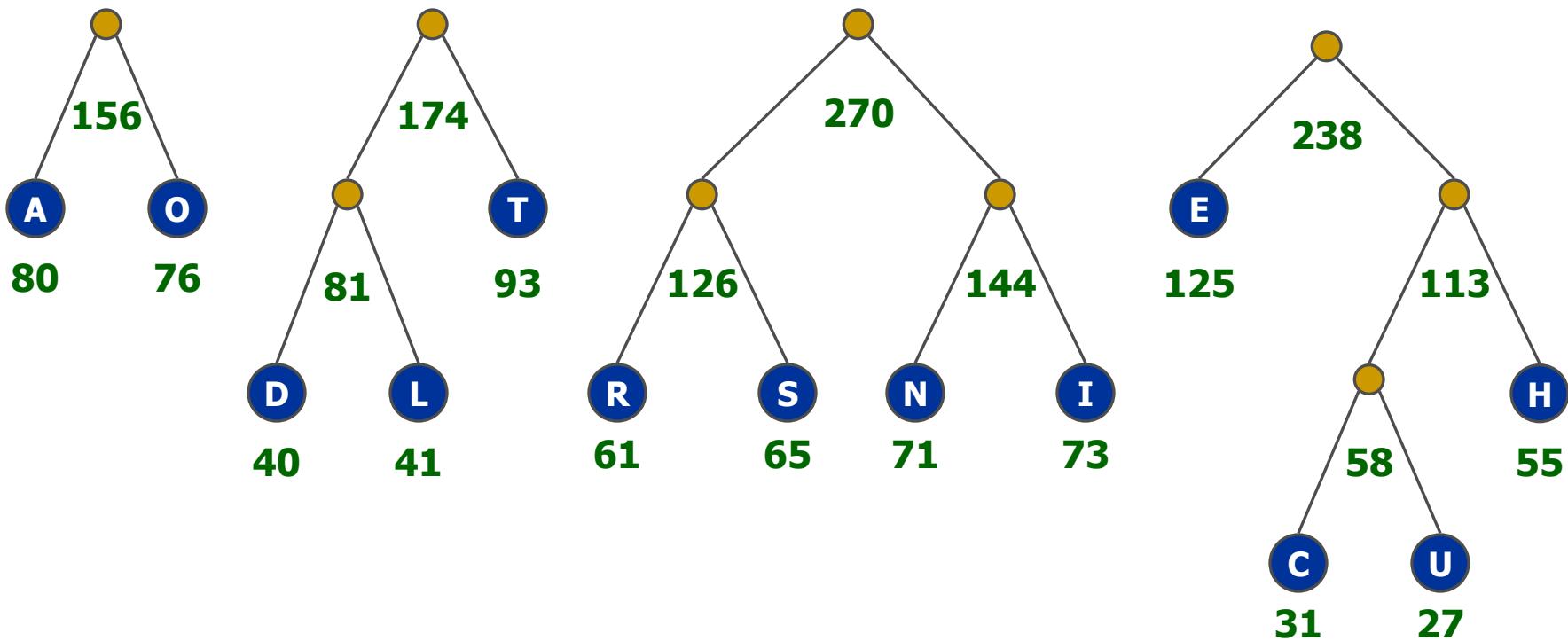
# Exemplo de Codificação de Huffman



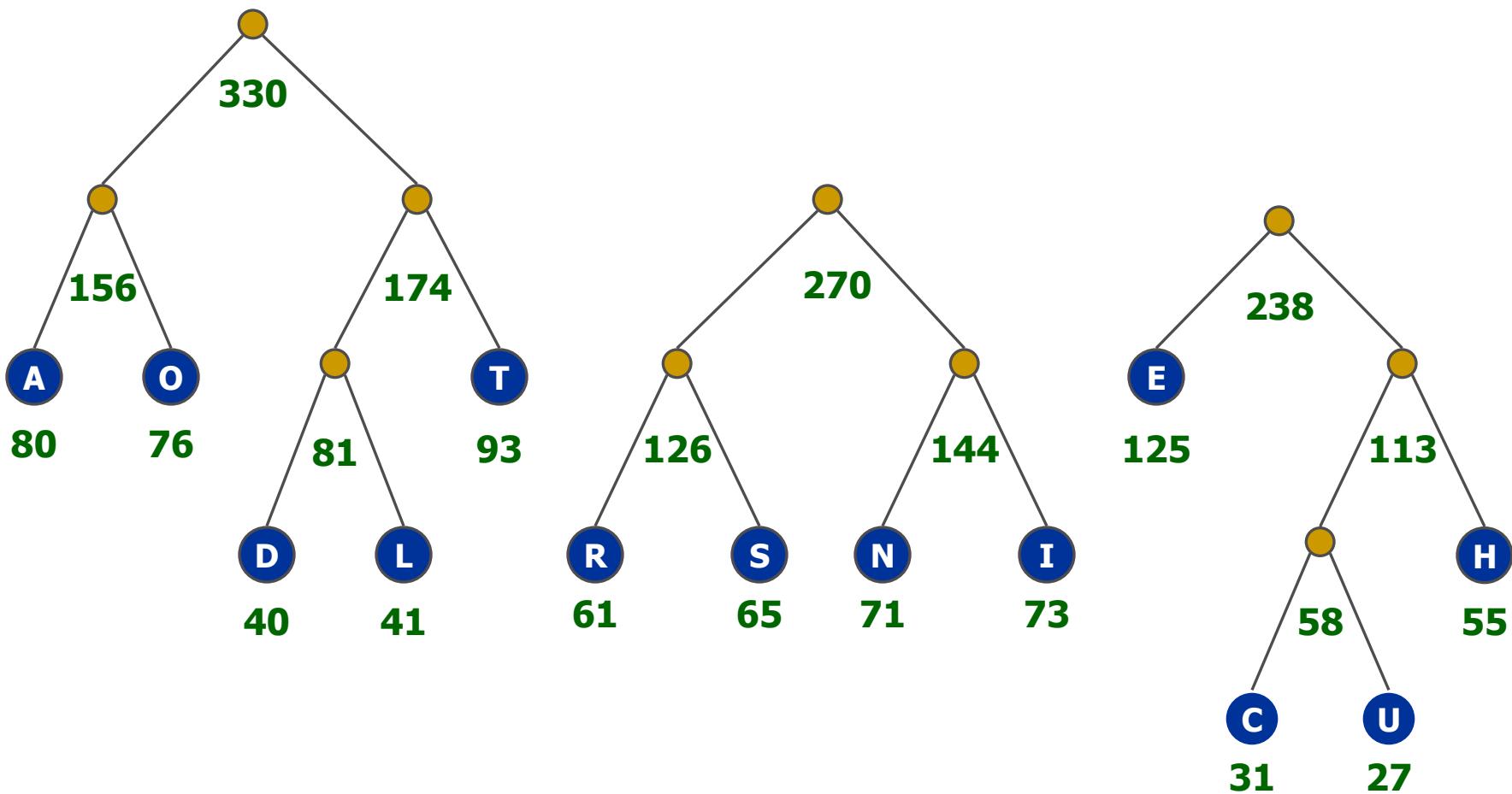
# Exemplo de Codificação de Huffman



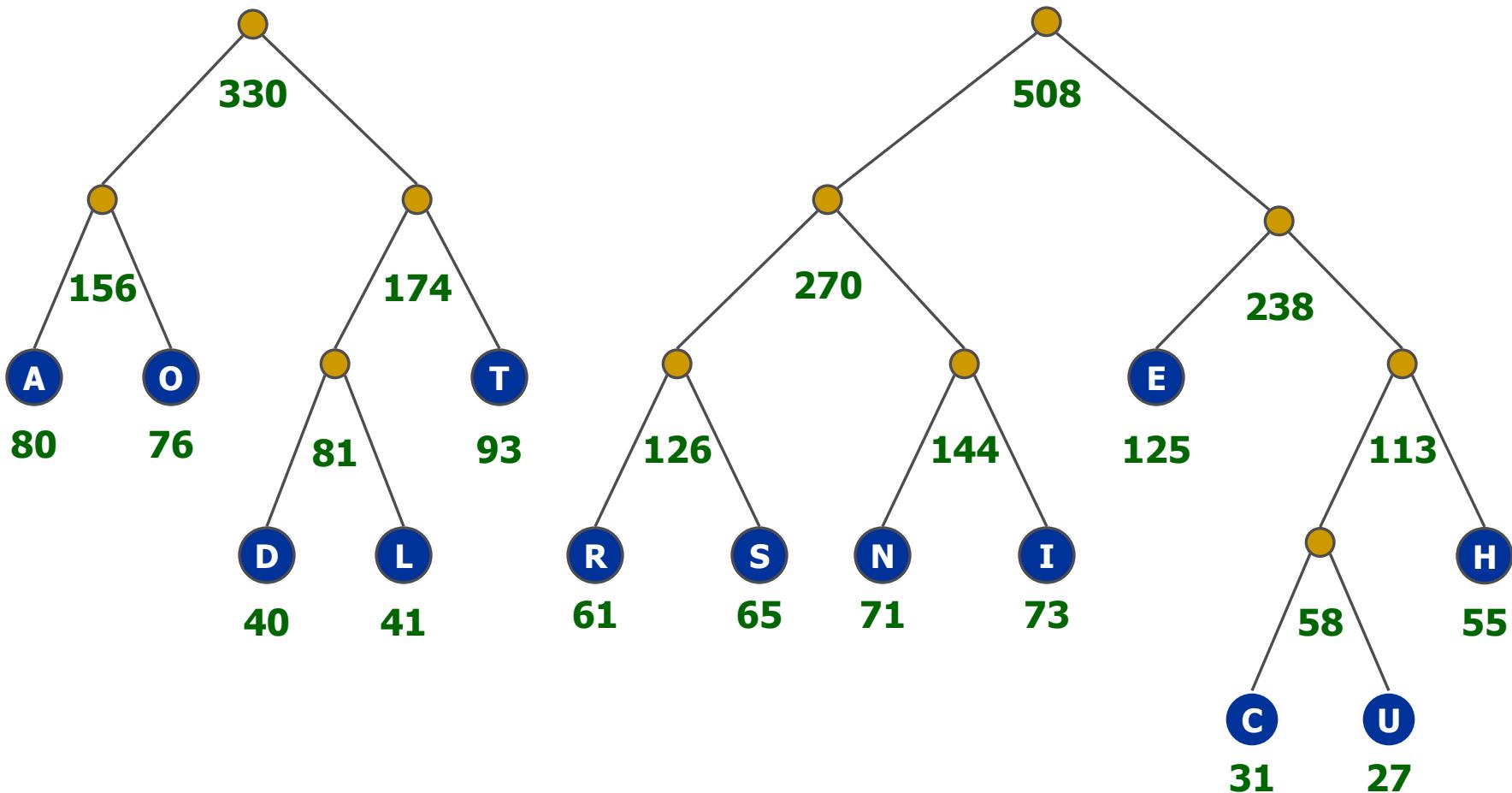
# Exemplo de Codificação de Huffman



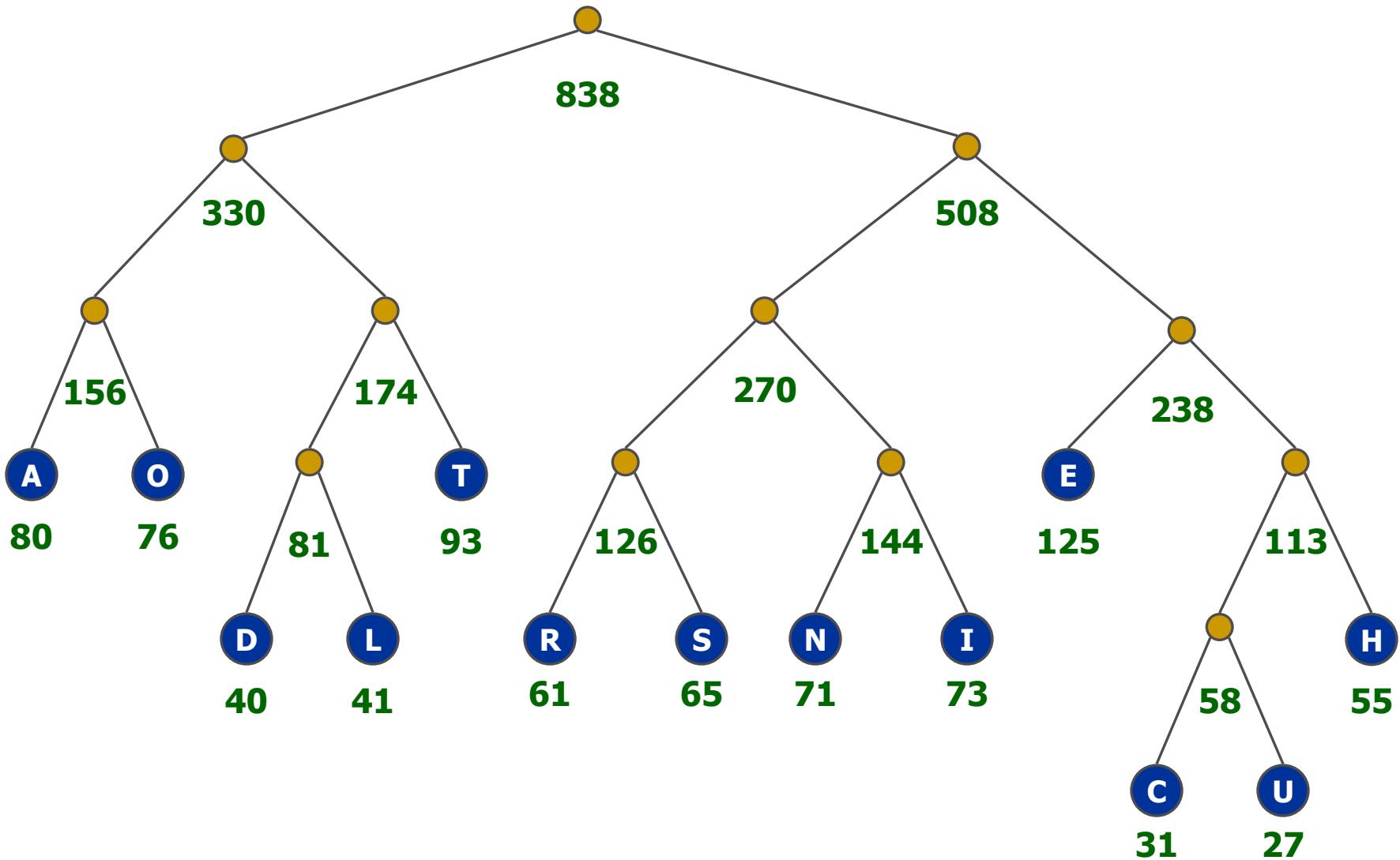
# Exemplo de Codificação de Huffman



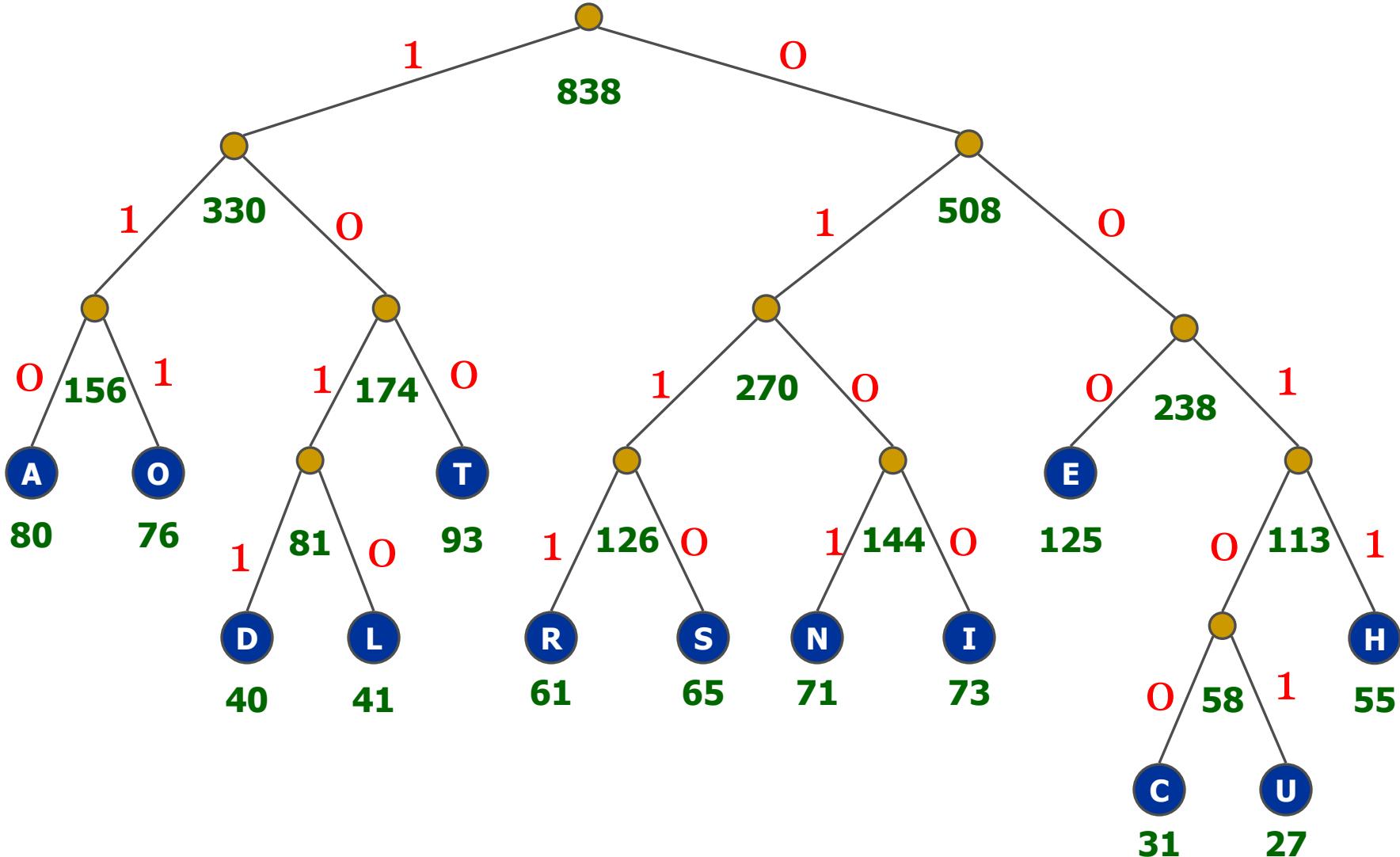
# Exemplo de Codificação de Huffman



# Exemplo de Codificação de Huffman



# Exemplo de Codificação de Huffman



# Exemplo de Codificação de Huffman

Char	Freq	Fixo	Huff
E	125	0000	000
T	93	0001	100
A	80	0010	110
O	76	0011	111
I	73	0100	0100
N	71	0101	0101
S	65	0110	0110
R	61	0111	0111
H	55	1000	0011
L	41	1001	1010
D	40	1010	1011
C	31	1011	00100
U	27	1100	00101
Total	838	4.00	3.62

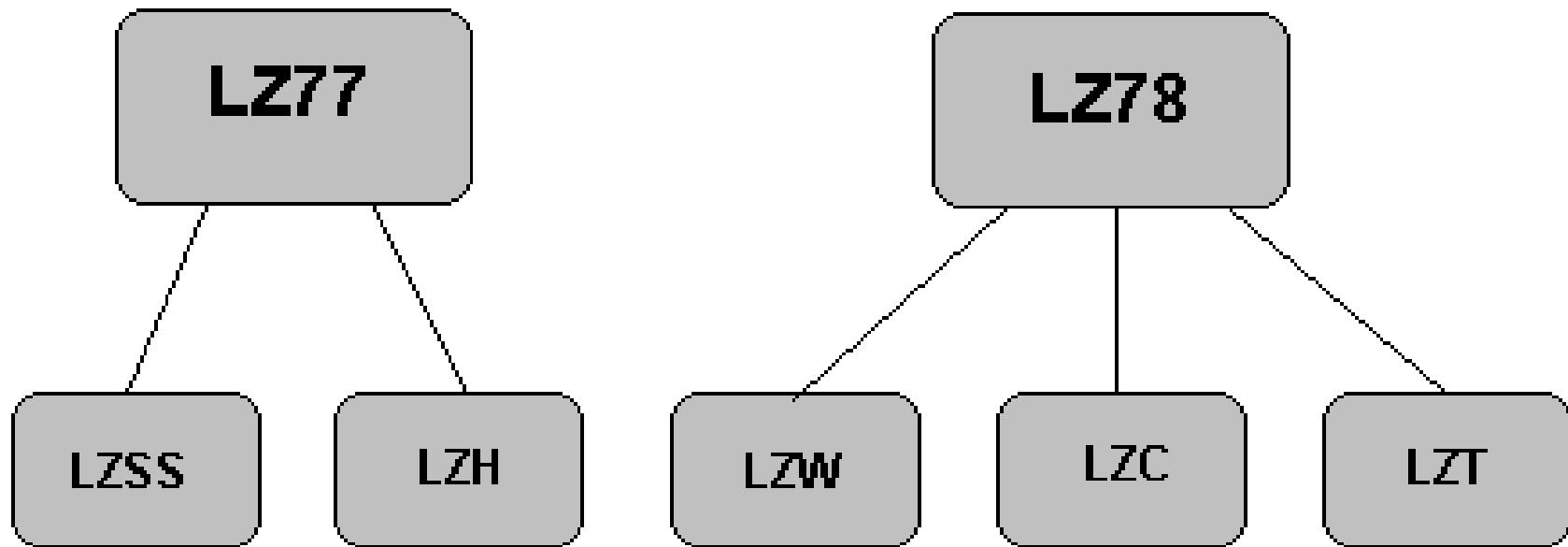
# Técnicas de compressão sem perdas

- Codificação de Huffman

- Operação computacional mais custosa
  - adição de floats (probabilidades)
  - ocorre no lado do codificador
- No decodificador
  - realiza uma simples verificação na tabela de Huffman
    - tabela de Huffman é parte do fluxo de dados ou é conhecida pelo decodificador
- Tabelas de Huffman padrões são muito usadas
  - usada para vídeo em tempo-real
  - tabelas são conhecidas pelo codificador e decodificador
    - codificação e decodificação são mais rápidas
  - desvantagem: tabelas padrões obtém fator de compressão um pouco menores
    - elas não são necessariamente ótimas

# Técnicas de compressão sem perdas

- Lempel-Ziv (LZ)
  - Algoritmos de codificação baseada em dicionário
  - Finais dos anos 70, Jacob Ziv e Abraham Lempel
    - Muitas variantes com objetivo de solucionar limitações das versões originais



# Técnicas de compressão sem perdas

- Codificações derivadas do Lempel-Ziv (LZ)
  - Explora a redundância de dados -> a repetição de padrões (partes) de símbolos no arquivo.
    - É 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” dicionário
      - um token 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)
  - Exemplo do poder da codagem LZW
    - suponha que temos um arquivo de 10000 caracteres
    - se nós representarmos o arquivo usando 8 bits por caractere, o arquivo requer 80000 bits para representá-lo
    - assumindo que o arquivo tenha 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
      - na prática, o dicionário armazenando todas as frases únicas deve ser armazenado também
        - » baixando a taxa de compressão obtida

# Técnicas de compressão sem perdas

- Exemplo de compactação LZW

- Compressão da cadeia de caracteres  
TOBEORNOTTOBEORTOBEORNOT#  

- Inicialização do Dicionário

Símbolo	Binário	Índice
#	00000	0
A	00001	1
B	00010	2
C	00011	3
D	00100	4
E	00101	5
F	00110	6
G	00111	7
H	01000	8
I	01001	9
J	01010	10
K	01011	11
L	01100	12
M	01101	13
N	01110	14

Símbolo	Binário	Índice
O	01111	15
P	10000	16
Q	10001	17
R	10010	18
S	10011	19
T	10100	20
U	10101	21
V	10110	22
W	10111	23
X	11000	24
Y	11001	25
Z	11010	26

# Técnicas de compressão sem perdas

- Exemplo de compactação LZW

- Compressão da cadeia de caracteres

TOBEORNOTTOBEORTOBEORNOT#

Símbolo	Binário	Índice
#	00000	0
A	00001	1
B	00010	2
C	00011	3
D	00100	4
E	00101	5
F	00110	6
G	00111	7
H	01000	8
I	01001	9
J	01010	10
K	01011	11
L	01100	12
M	01101	13
N	01110	14
O	01111	15
P	10000	16
Q	10001	17
R	10010	18
S	10011	19
T	10100	20
U	10101	21
V	10110	22
W	10111	23
X	11000	24
Y	11001	25
Z	11010	26

Sequencia atual	Próx. caractere	Saída		Dicionário extendido	Comentários	
		Código	Bits			
NULL	T					
T	O	20	10100	27:	TO	27 = próximo índice disponível
O	B	15	01111	28:	OB	
B	E	2	00010	29:	BE	
E	O	5	00101	30:	EO	
O	R	15	01111	31:	OR	
R	N	18	10010	32:	RN	32 requer 6 bits, assim as próximas saídas usam 6 bits
N	O	14	001110	33:	NO	
O	T	15	001111	34:	OT	

# Técnicas de compressão sem perdas

- Exemplo de compactação LZW

- Compressão da cadeia de caracteres

TOBEORNOTTOBEORTTOBEORNOT#

Símbolo	Binário	Índice	Símbolo	Binário	Índice
#	00000	0	O	01111	15
A	00001	1	P	10000	16
B	00010	2	Q	10001	17
C	00011	3	R	10010	18
D	00100	4	S	10011	19
E	00101	5	T	10100	20
F	00110	6	U	10101	21
G	00111	7	V	10110	22
H	01000	8	W	10111	23
I	01001	9	X	11000	24
J	01010	10	Y	11001	25
K	01011	11	Z	11010	26
L	01100	12			
M	01101	13			
N	01110	14			

Sequencia atual	Próx. caractere	Saída		Dicionário extendido	Comentários	
		Código	Bits			
T	T	20	010100	35:	TT	
TO	B	27	011011	36:	TOB	
BE	O	29	011101	37:	BEO	
OR	T	31	011111	38:	ORT	
TOB	E	36	100100	39:	TOBE	
EO	R	30	011110	40:	EOR	
RN	O	32	100000	41:	RNO	
OT	#	34	100010			# para o algoritmo; envia o caractere; envia sequencia atual
		0	000000			E o código de parada

# Codificação derivadas do Lempel-Ziv (LZ)

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

# 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 sem perdas que oferece as melhores taxas de compressão.
    - taxas de compressão não são grandes, em geral 4:1
  - GIF apenas admite o tratamento de imagens com uma profundidade de cor até 8 bits/pixel
    - 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
  - Extensão GIF89a permite
    - definir uma cor transparente
    - entrelaçamento
    - animação



# Técnicas de compressão sem perdas

- LZW e o formato de imagem GIF
  - Algoritmo LZW do GIF é propriedade da Unisys
    - era do domínio público
    - Unisys resolveu passar a cobrar uma taxa pela sua utilização
  - 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

# Compressão de Dados Multimídia

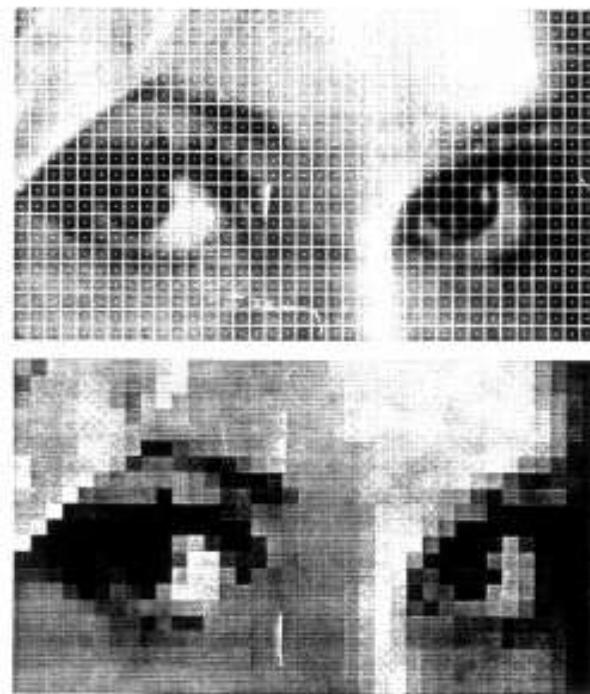
- Conteúdo:
  - Necessidade de compressão
  - Princípios da compressão
  - Classificação das técnicas de compressão
  - Medição do desempenho de compressão
  - Técnicas de compressão sem perdas
    - RLE, Huffman, LZW (GIF)
  - **Técnicas de compressão de áudio, vídeo e imagens**
    - **Técnicas de compressão de voz**
    - **Técnicas de compressão de som**
  - **Padrões de compressão multimídia**
    - **JPEG, MPEG, MPEG-4, H.261, H.263**

# Técnicas de Compressão de Vídeo e Imagem

- **Imagens e vídeos digitais puros são codificadas em PCM**
  - Representados por vetores bi-dimensionais de píxeis
- **Técnicas de compressão de vídeo e imagens**
  - Baseiam-se na alta redundância das imagens e vídeos
  - Certas áreas de figuras são uniformemente coloridas ou altamente correlatas (podendo formar padrões)
    - redundância espacial ou correlação espacial
    - removida tanto quanto possível para uma certa qualidade de apresentação
  - Não existem grandes diferenças entre quadros de um vídeo
    - redundância temporal ou correlação temporal
    - alta taxa de compressão

# Técnicas de Compressão de Vídeo e Imagem

- Técnica de Redução da Resolução Geométrica
  - Redução da resolução das imagens
    - Redução de linhas e colunas do bitmap



# Técnicas de Compressão de Vídeo e Imagem

- **Técnica de Truncagem**

- Consiste em truncar dados arbitrariamente baixando o número de bits por píxel (imagem) ou taxa de quadros (vídeo)
  - feito pela eliminação dos bits menos significativos de cada píxel (imagem) e imagens por segundo (vídeo)
- Técnica é atrativa pois ela é simples



- Exemplo: imagens coloridas com 24 bits por píxel poderiam ser reduzidas para 8 bits

# Técnicas de Compressão de Vídeo e Imagem

- Codificação Predictiva

- DPCM (PCM diferencial)

- Técnica mais simples de codificação preditiva
    - Compara píxeis adjacentes e apenas erros de predição são quantificados e codificados
      - Erro de predição tem uma alta probabilidade de ser menor que o valor do novo pixel
        - » Erro pode ser expresso com uma quantidade menor de bits
      - na descompressão
        - » informação de erro é usado para modificar o(s) píxel(eis) anterior(es)

- ADPCM (DPCM adaptativo)

- Existem várias maneiras de implementar ADPCM, a mais comum é variar o tamanho de passo representado pelos bits diferenças
      - Exemplo: se um passo preto-para-branco for detectado, pode-se aumentar o passo de quantificação antes deste passo chegar

# Técnicas de Compressão de Vídeo e Imagem

- Codificação Preditiva
  - Preditores típicos

^

$$\hat{s}_n = 0.97s_{n-1} \text{ Predictor de 1ª ordem, 1D}$$

^

$$\hat{s}_{m,n} = 0.48s_{m,n-1} + 0.48s_{m-1,n} \text{ Predictor de 2ª ordem, 2D}$$

^

$$\hat{s}_{m,n} = 0.8s_{m,n-1} - 0.62s_{m-1,n-1} + 0.8s_{m-1,n} \text{ Predictor de 3ª ordem, 2D}$$

$s_{m-1,n-1}$	$s_{m,n-1}$	
$s_{m-1,n}$	$s_{m,n}$	

# Técnicas de Compressão de Vídeo e Imagem

- Codificação Preditiva:

- Usar para a primeira fila e primeira coluna o preditor de 1<sup>a</sup> ordem

^

$$\hat{s}_n = 0.97s_{n-1} \text{ Predictor de 1ª ordem, 1D}$$

- Para as outras filas e colunas o de 3<sup>a</sup> ordem.

^

$$\hat{s}_{m,n} = 0.8s_{m,n-1} - 0.62s_{m-1,n-1} + 0.8s_{m-1,n} \text{ Predictor de 3ª ordem, 2D}$$

- Saída DPCM calculada subtraindo a saída prevista com os valores originais

$$\begin{bmatrix} 20 & 21 & 22 & 21 \\ 18 & 19 & 20 & 19 \\ 19 & 15 & 14 & 16 \\ 17 & 16 & 15 & 13 \end{bmatrix}$$

Original

$$\begin{bmatrix} 20 & 19.4 & 20.37 & 21.34 \\ 19.4 & 18.8 & 19.78 & 19.16 \\ 17.46 & 19.24 & 16.22 & 14.00 \\ 18.43 & 13.82 & 14.70 & 16.2 \end{bmatrix}$$

Saída prevista

$$\begin{bmatrix} X & 1.6 & 1.63 & -0.34 \\ -1.4 & 0.20 & 0.22 & -0.16 \\ 1.54 & -4.24 & -2.22 & 2.00 \\ -1.43 & 2.18 & 0.30 & -3.12 \end{bmatrix}$$

Saída DPCM

# Técnicas de Compressão de Vídeo e Imagem

- Codificação Preditiva

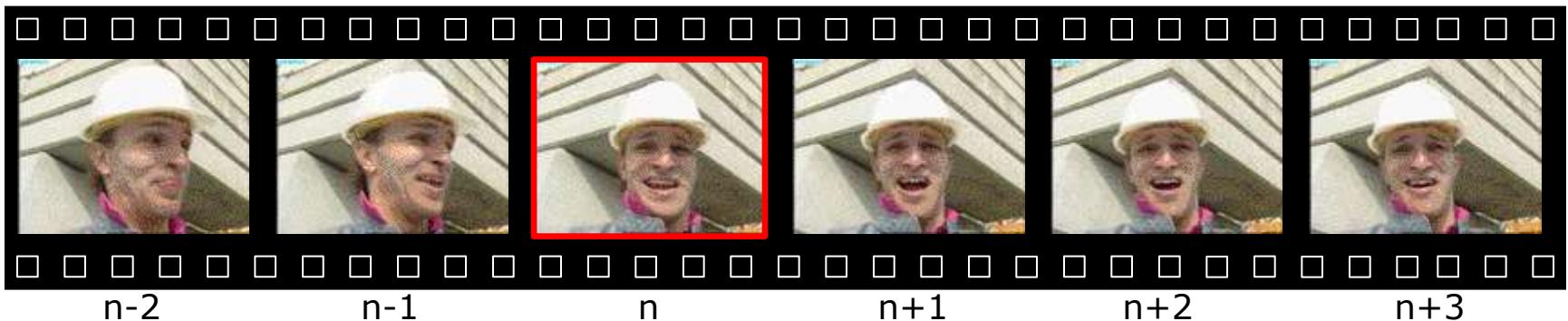
- Imagem original e imagem com apenas o erro de predição
  - Se os pixels tiverem valores muito próximos, pode-se usar um número menor de bits para armazenar o erro de predição do que aquele usado para codificar o valor absoluto



# Técnicas de Compressão de Vídeo

- Revisão de Vídeo

- Um vídeo é uma sequência de imagens amostradas rapidamente;
- A velocidade da amostragem engana o cérebro, criando a ilusão de movimento.



Foreman 30fps  
(30 imagens exibidas a cada segundo)



# Técnicas de Compressão de Vídeo

- Preenchimento Condisional
  - Explora redundância temporal em vídeos
    - animação de imagens implica que píxeis na imagem anterior estão em diferentes posições que na imagem atual
  - Imagem é segmentada em áreas estacionárias e com movimento
    - são transmitidos apenas os dados de áreas com movimento
    - detector de movimento localiza diferenças inter-quadros significantes
  - Uma forma particular de DPCM onde se envia o erro de predição se este for superior a um dado limite



# Técnicas de Compressão de Vídeo

- Preenchimento Condisional

Quadro Preditor



Quadro Atual

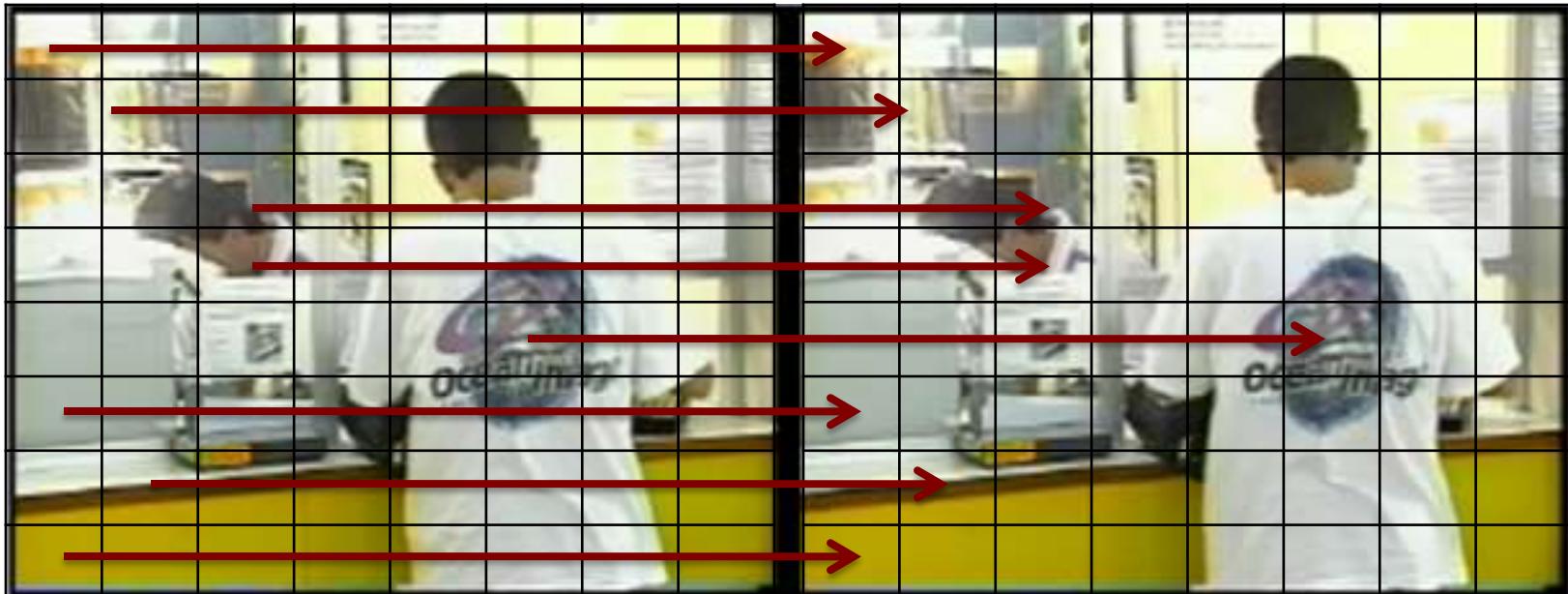


Diferença



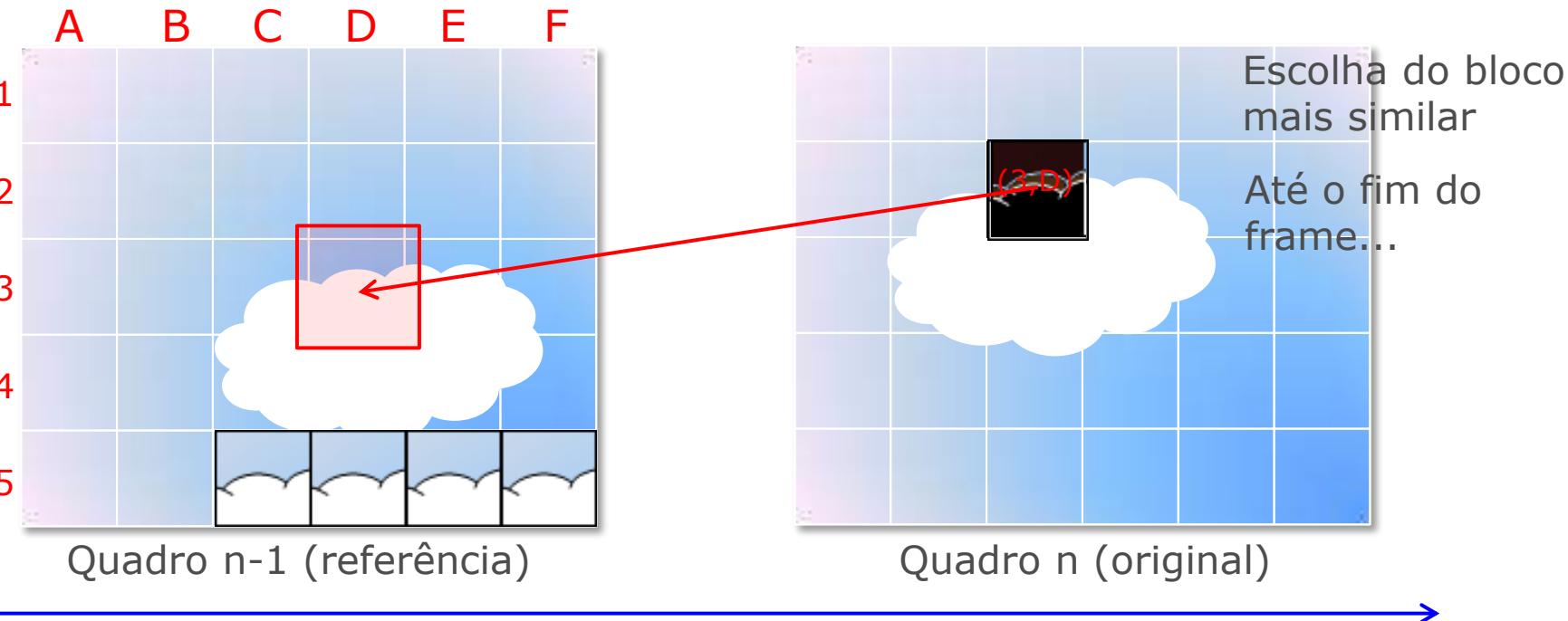
# Técnicas de Compressão de Vídeo e Imagem

- **Estimativa e Compensação de Movimento**
  - Imagem é dividida em blocos de tamanho fixos
    - um casamento para cada bloco é procurado na imagem anterior
      - deslocamento entre estes dois blocos é chamado vetor de movimento
    - uma diferença de blocos é obtida calculando diferenças píxel a píxel
  - Vetor de movimento e a diferença de bloco é codificado e transmitido



# Técnicas de Compressão de Vídeo e Imagem

- Exemplo simples: Compara a similaridade entre blocos

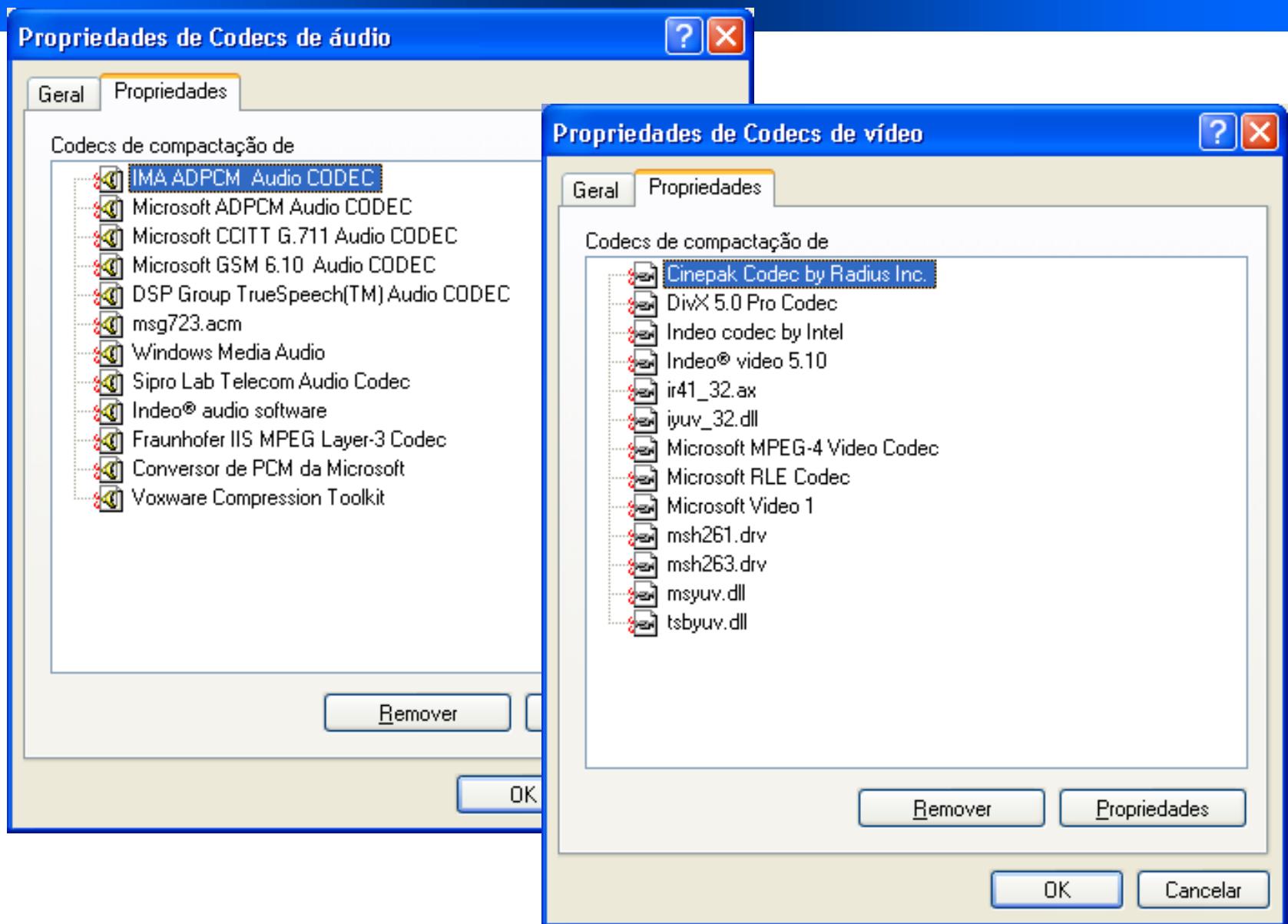


- Mantém a diferença entre os blocos (resíduo);
- Cria o vetor de movimento, referenciando o bloco do quadro anterior;

# Padrões de Compressão Multimídia

- Várias técnicas e produtos para compressão são disponíveis
  - utilização de padrões promove a compatibilidade entre diferentes equipamentos/aplicações (interoperabilidade)
- Exemplos de formatos de Imagens
  - PCX: usado por aplicativos gráficos mais antigos (256 cores);
  - GIF: usado para distribuição comercial de imagens sem perda (256 cores)
  - BMP: padrão fundamental do windows
  - TGA: usado pelos adaptadores gráficos Targa e por programas de animação e processamento de vídeo
- Formatos de vídeo
  - RealVideo, Microsoft RLE, Microsoft Video 1, Microsoft MPEG 4, Cinepak, Indeo, DivX, VDOWave/VDOLive, ClearVideo
- Formato de áudio
  - Wav: padrão do windows
  - voxware

# Padrões de Compressão Multimídia



# Padrões de Compressão Multimídia

- Várias técnicas e produtos para compressão são disponíveis
  - utilização de padrões promove a compatibilidade entre diferentes equipamentos/aplicações (interoperabilidade)
- Exemplos de padrões
  - TIFF: padrão independente de fabricante para imagens
  - PNG: padrão de imagens alternativo ao GIF
  - ISO JPEG para compressão de imagens;
  - ISO JBIG para compressão sem perda de imagens bi-níveis (1 bit/píxel) para transmissão fac-símile
  - ITU-TS H.261 para videofonia e aplicações de teleconferências na taxa de bits múltiplos de 64 Kbps;
  - ITU-TS H.263 para aplicações de videofonia na taxa abaixo de 64 Kbps;
  - ISO MPEG para compressão de vídeo e áudio associado;

# Padrão de Compressão de Imagens JPEG

- **JPEG colaboração entre a ISO/IEC e a ITU-TS (1992)**
  - Uma das melhores tecnologia de compressão de imagem
  - Implementado em software e hardware
  - Codificação/decodificação JPEG tempo-real tem sido implementada para vídeo (Motion JPEG - MJPEG)
- **Compressão parametrizável**
  - JPEG cobre grande faixa de qualidades de imagens e permite especificar o comportamento do codificador a partir de parâmetros
  - Quatro modos de operação:
    - Codificação seqüencial (baseline)
    - Codificação progressiva
    - Codificação sem perda
    - Codificação hierárquica

# Padrão de Compressão de Imagens JPEG

- Codificação Seqüencial (baseline)
  - Suportado por toda implementação JPEG
  - Modo com perdas baseada em DCT
  - Componentes de imagem são codificados em uma única varredura da esquerda para direita e de cima para baixo
- Codificação progressiva
  - Com perdas baseada em DCT expandido
    - Fornece avanços ao modo baseline
  - Varreduras sucessivas
    - imagem é compactada em um processo de múltiplas linhas de varredura
  - Geralmente utilizada em arquivos que são transmitidos pela Internet
    - pois possibilita a visualização da imagem inteira, em menor resolução, enquanto o restante da imagem esta sendo enviada

# Padrão de Compressão de Imagens JPEG

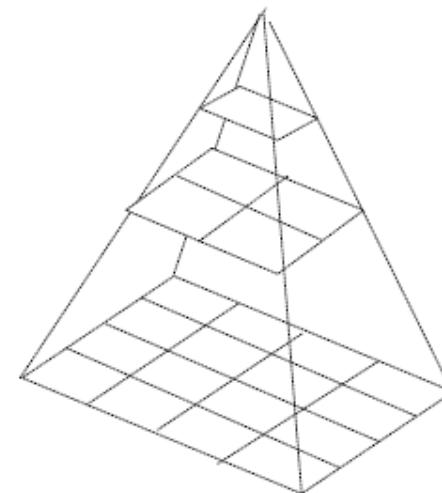
- Codificação sem perda
  - Reprodução é exata
  - Necessária em aplicações que não toleram perdas (médicas e legais)
  - Existem duas variações:
    - o original, que foi normalizado em 1992, e o novo método JPEG-LS, que deverá deixar obsoleto o formato JPEG “lossless” original.
  - A codificação de Huffman é aplicada após a codificação preditiva



# Padrão de Compressão de Imagens JPEG

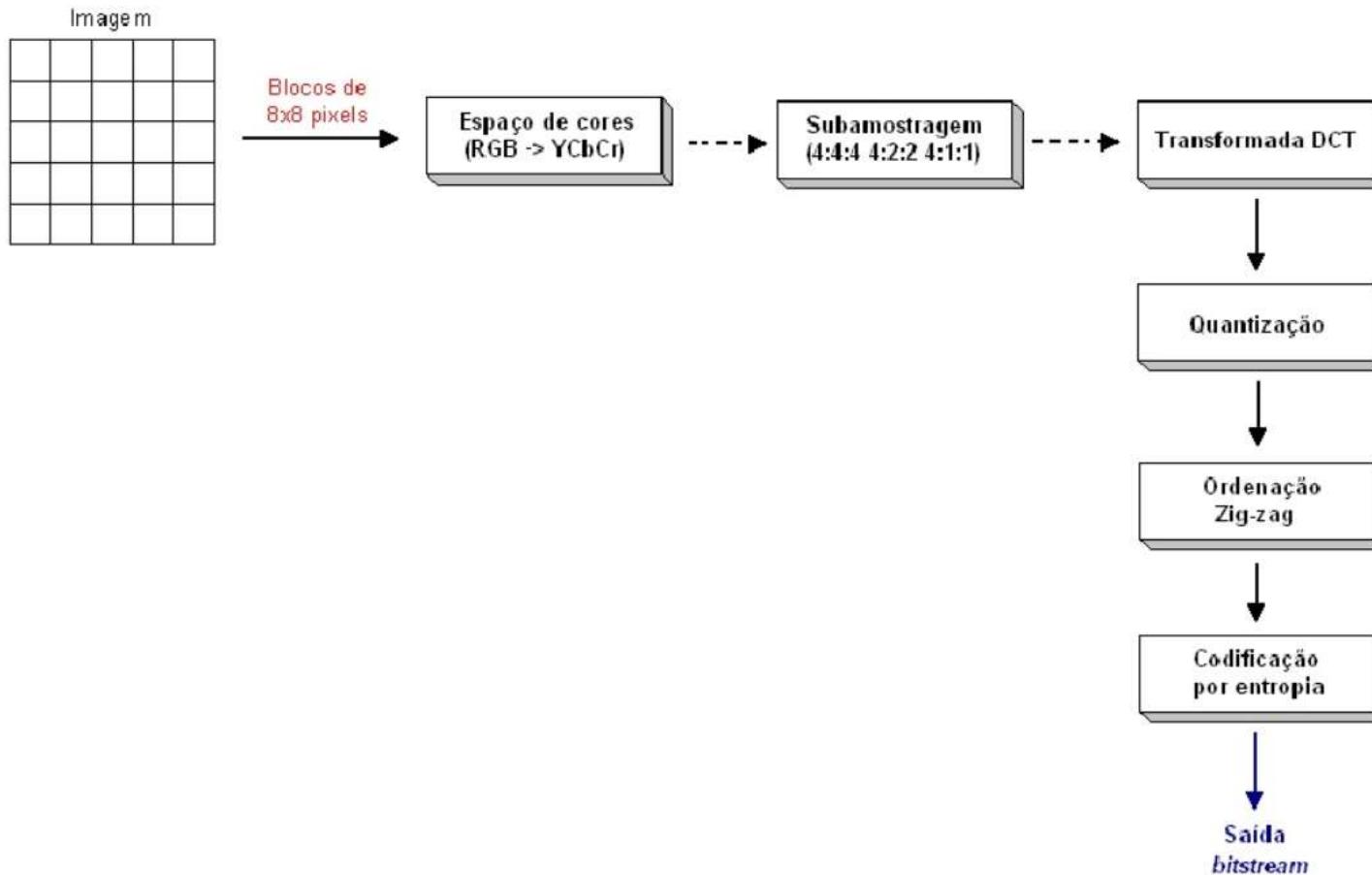
- Codificação hierárquica

- Oferece uma codificação progressiva que aumenta de resolução espacial entre estágios progressivos
- Versões podem ser acessadas sem a necessidade de primeiro descompactar a imagem na resolução completa
- Os elementos de imagem das resoluções já recebidas são utilizados na próxima resolução, diminuindo desta forma o tamanho do arquivo
- Taxa de compressão é mais baixa que ter uma resolução única



# Algoritmo de compressão JPEG

- Operações a compressão JPEG (Sequencial)



# Algoritmo de compressão JPEG

- Transformação do espaço de cores para YCrCb
  - Componentes “RGB” da imagem são convertidos para componentes de luminância (“Y”) e crominância (“Cr” e ”Cb”)
    - Y: Luminância é uma escala de representação numérica do cinza,
    - CrCb: Crominância são duas escalas numéricas, que juntas representam as cores.

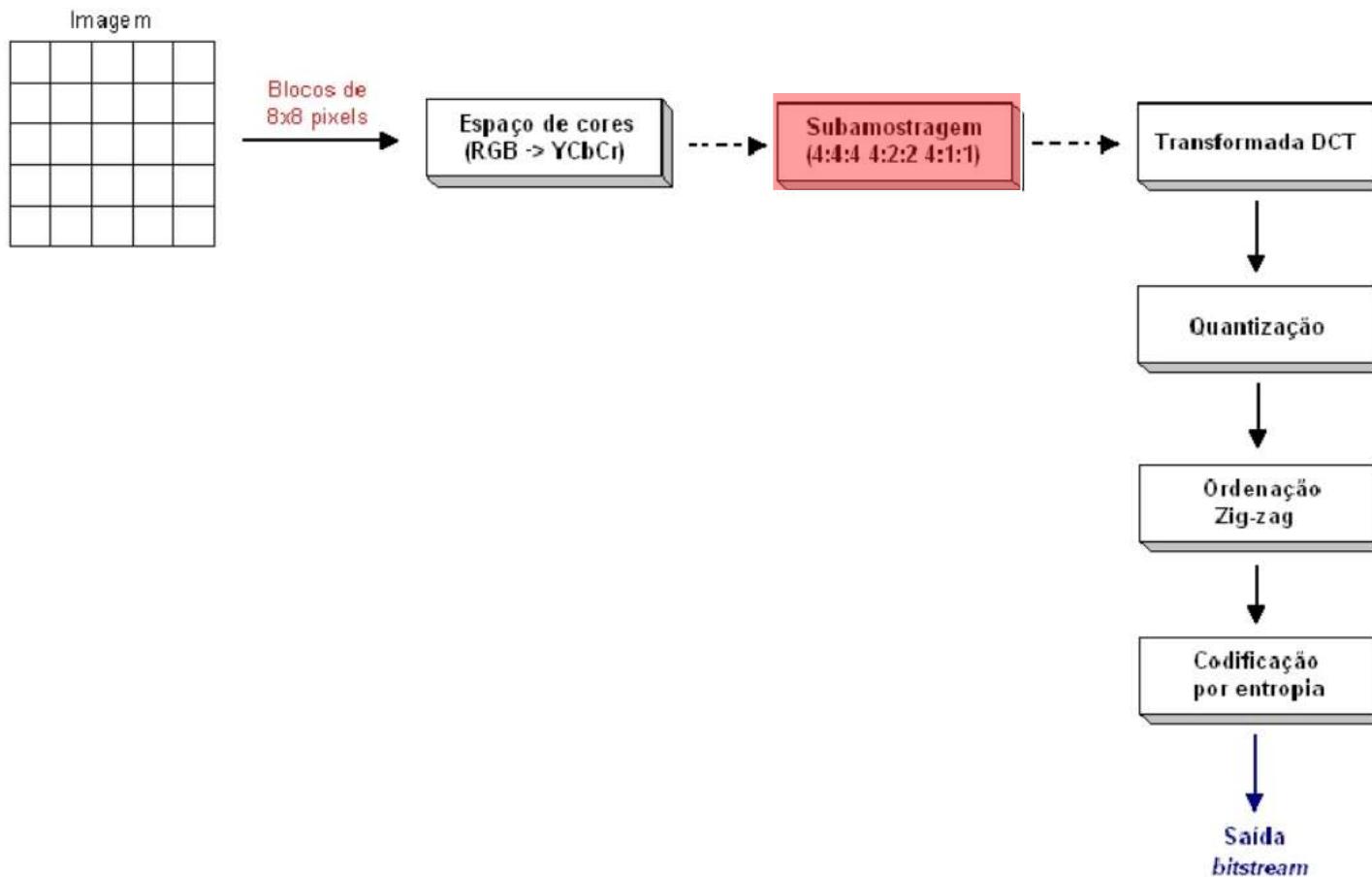
$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.2990 & 0.5870 & 0.1140 \\ -0.1687 & -0.3313 & 0.5000 \\ 0.5000 & -0.4187 & -0.0813 \end{bmatrix} * \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

- YCbCr permite uma maior compressão sem um efeito significante na qualidade da imagem percebida.



# Algoritmo de compressão JPEG

- Operações a compressão JPEG



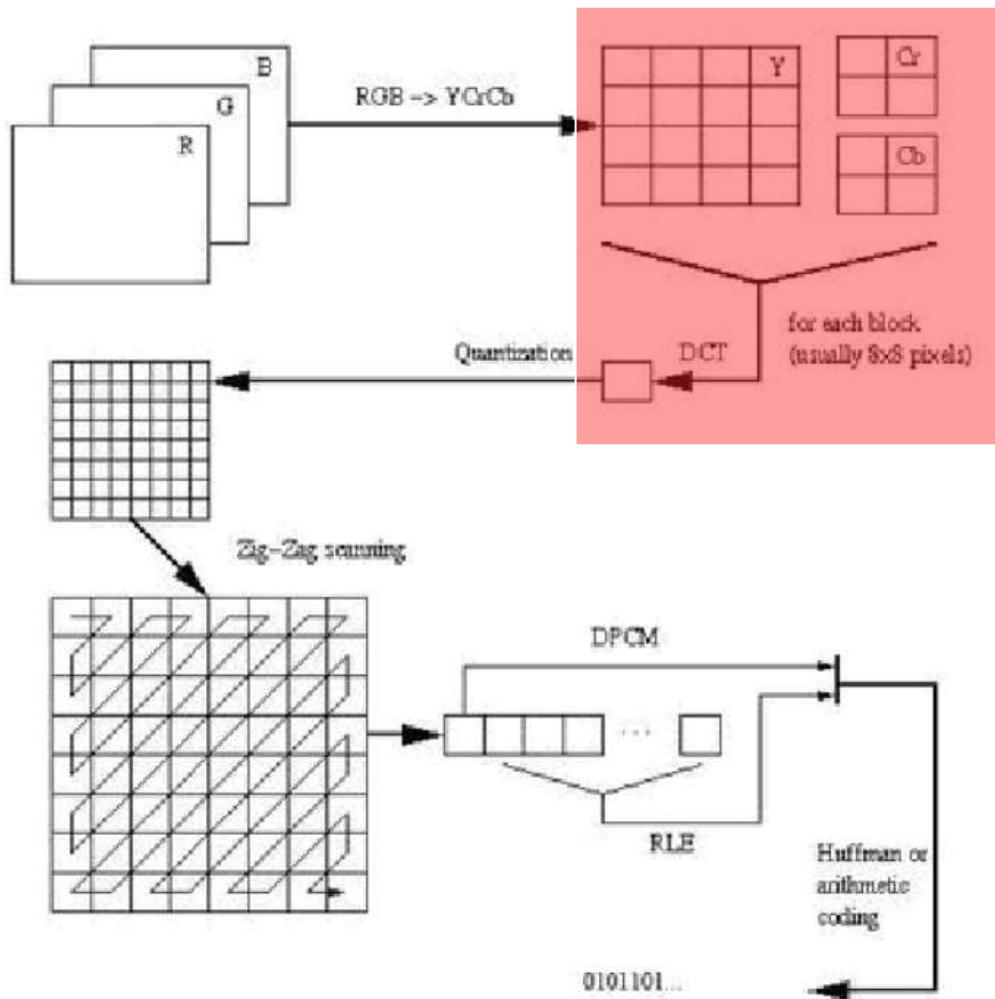
# Algoritmo de compressão JPEG

- **Subamostragem**

- Onde é feita uma redução da resolução das matrizes YCbCr.
- Taxas de subamostragem que são normalmente aplicados no JPEG
  - 4:4:4 (sem subamostragem)
  - 4:2:2 onde as matrizes de crominância são reduzidas na taxa de 2:1 horizontalmente (cada duas linhas é convertida em uma),
  - 4:2:0 mais comumente adotada, onde a uma redução do fator 2 nas direções horizontais e verticais.
- A matriz de luminância não é reduzida
  - pois o olho humano é mais sensível à luminância (tonalidade de cinza) do que à crominância (tonalidades das cores), o que permite maior taxa de perda de crominância sem que esta perda seja percebida
- No resto do processo de compressão, Y, Cb e Cr são processadas separadamente de maneira muito similar.

# Algoritmo de compressão JPEG

- Operações a compressão JPEG



# Algoritmo de compressão JPEG

- **Decomposição da imagem em blocos**
  - Imagem é decomposta em blocos de 8x8 píxeis
- **Transformação discreta de co-seno (DCT) dos blocos**
  - Blocos 8x8 são transformado para o domínio da frequência espacial usando a transformada DCT
    - efetua uma separação das componentes de baixa e de alta freqüência presentes numa imagem
    - permitindo que se faça uma seleção destas últimas, de acordo com a qualidade pretendida para a imagem compactada
      - maior é a taxa de compressão escolhida, maior é o número de componentes de alta freqüência desprezados
      - levando ao aparecimento do efeito de bloco, ou seja, perda de definição nos contornos das imagens

# Algoritmo de compressão JPEG

- Transformada Discreta do Coseno

No espaço bidimensional de uma imagem de 8x8 pixels, a transformada discreta de co-senos (*FDCT: Forward Discrete Cosine Transform*) é dada por:

$$F(u,v) = \frac{1}{4} C(u)C(v) \sum_{x=0}^7 \sum_{y=0}^7 f(x,y) \cos\left[\frac{(2x+1)u\pi}{16}\right] \cos\left[\frac{(2y+1)v\pi}{16}\right]$$

$$C(w) = \frac{1}{\sqrt{2}} \text{ para } w = 0$$

$$C(w) = 1 \text{ para } w = 1, 2, \dots, 7$$

E a transformada inversa (*IDCT: Inverse Discrete Cosine Transform*) por:

$$f(x,y) = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 C(u)C(v) F(u,v) \cos\left[\frac{(2x+1)u\pi}{16}\right] \cos\left[\frac{(2y+1)v\pi}{16}\right]$$

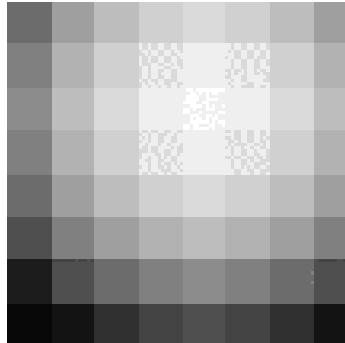
# Algoritmo de compressão JPEG

- Transformação discreta de co-seno (DCT) dos blocos
  - Sinal discreto de 64 pontos (um para cada bloco) transformado é uma função de duas dimensões espaciais, x e y
    - estas componentes são chamadas de freqüências espaciais ou coeficientes DCT
    - maior parte dos coeficientes DCT são zero ou muito próximo de zero
      - não necessitam ser codificados

Valores dos elementos

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10

Bloco de imagem

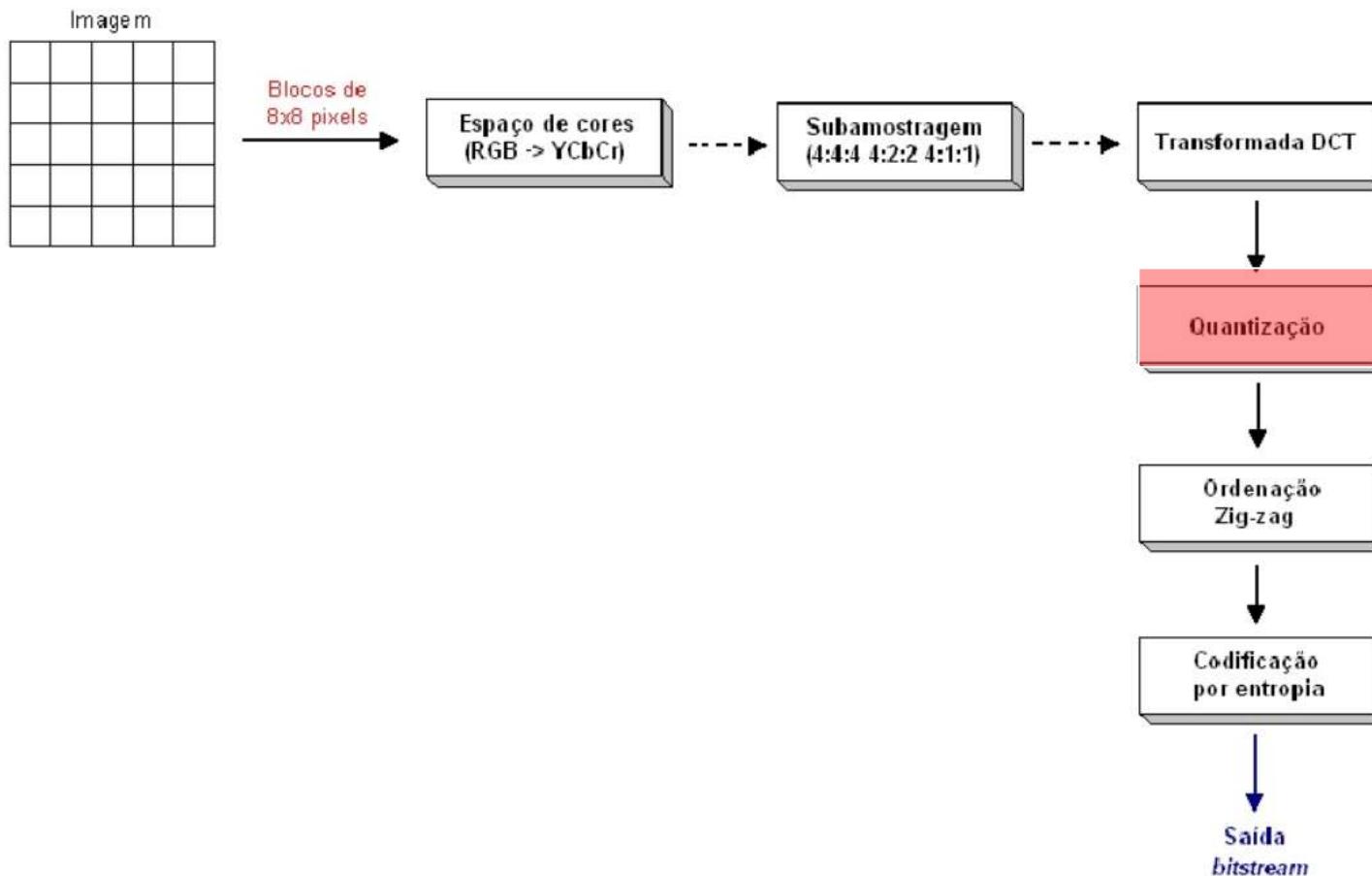


Coeficientes da DCT

159	-10	-20	-1	-4	-1	-2	0
31	-1	-1	-1	0	0	0	0
-11	1	1	0	0	0	0	0
3	-1	0	0	0	0	0	0
-5	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

# Algoritmo de compressão JPEG

- Operações a compressão JPEG



# Algoritmo de compressão JPEG

- Quantificação

- No domínio da frequência

- as mudanças abruptas que acontecem nos contornos de uma figura estão concentradas nas frequências mais altas.
      - uma imagem com poucos contornos deve concentrar seus coeficientes nas frequências baixas.
    - coeficientes das frequências altas são menos importantes e perdas nesses coeficientes podem diminuir um pouco a nitidez da imagem, mas para muitas aplicações isto pode ser aceitável.

- Quantificação prioriza a baixa frequência

- os coeficientes gerados são quantizados de forma diferenciada, usando uma maior precisão para as frequências mais baixas.
    - Fator de quantificação (Qc) definido pela qualidade selecionada pelo usuário

Coeficientes DCT

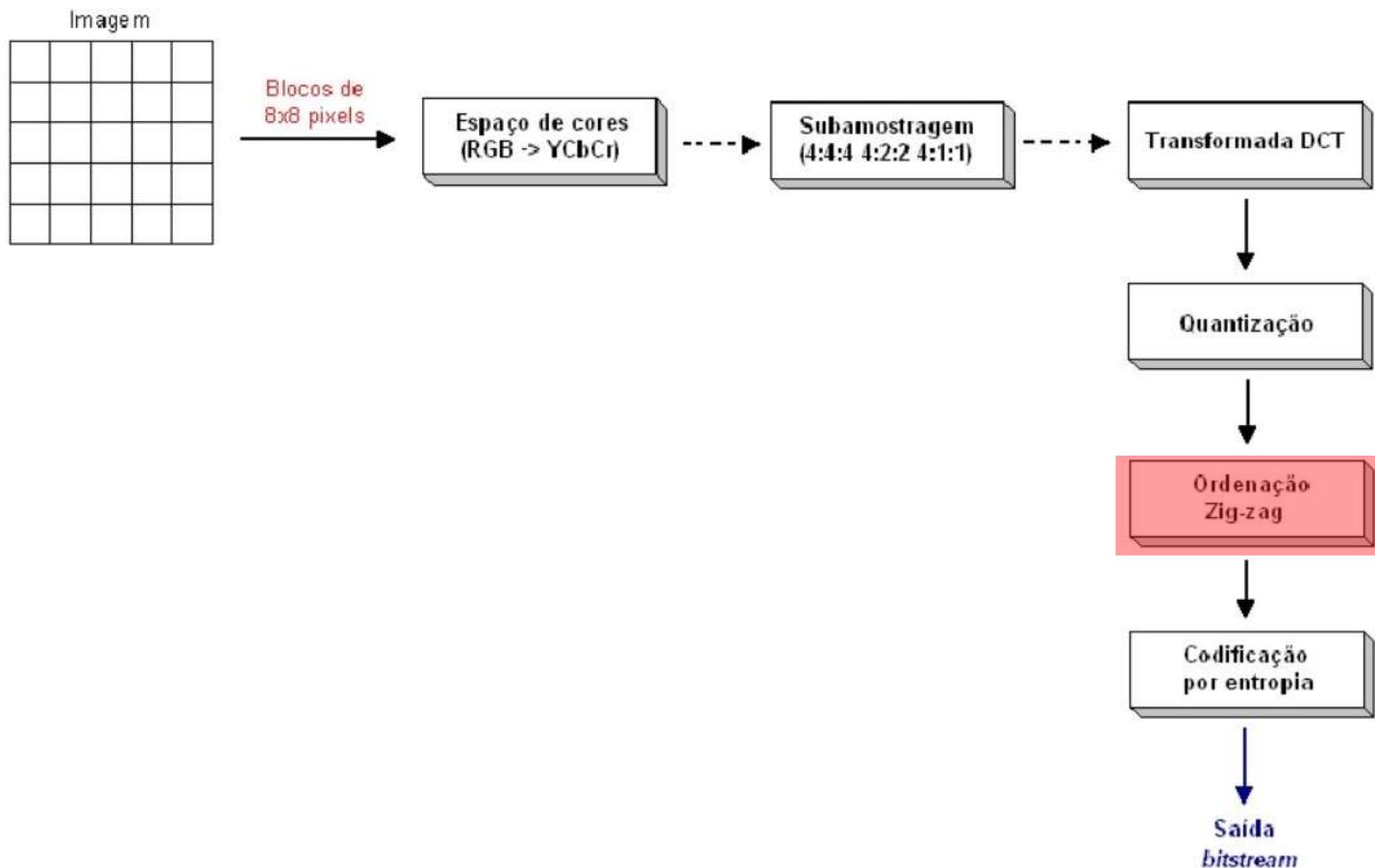
51	49	30	21	18	8	3	1
48	40	32	15	9	5	1	0
39	36	24	13	9	6	3	0
30	28	21	20	8	5	2	1
23	19	18	13	7	0	3	1
15	9	7	7	6	4	2	1
9	4	3	4	3	1	1	0
3	0	1	2	3	1	0	0

Coeficientes DCT quantizados

51	19	5	1	0	0	0	0
13	10	6	1	0	0	0	0
4	6	3	1	0	0	0	0
2	2	1	1	0	0	0	0
1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

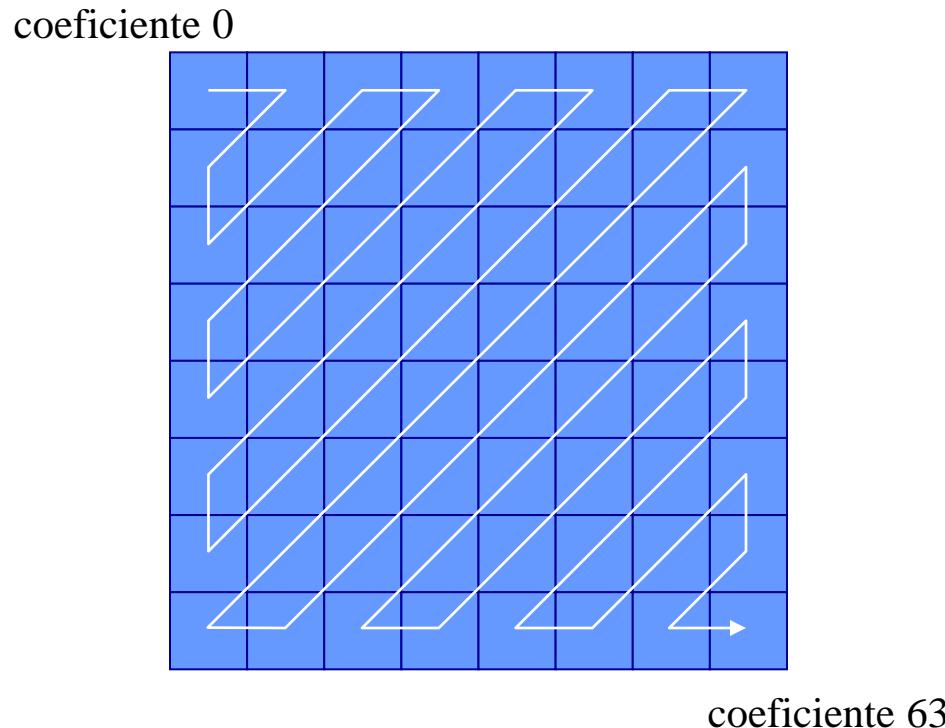
# Algoritmo de compressão JPEG

- Operações a compressão JPEG



# Algoritmo de compressão JPEG

- Ordenação dos coeficientes DCT
  - Coeficientes DCT são ordenados em uma seqüência zig-zag
    - para obter uma seqüência unidimensional de dados para ser usado na codificação por entropia



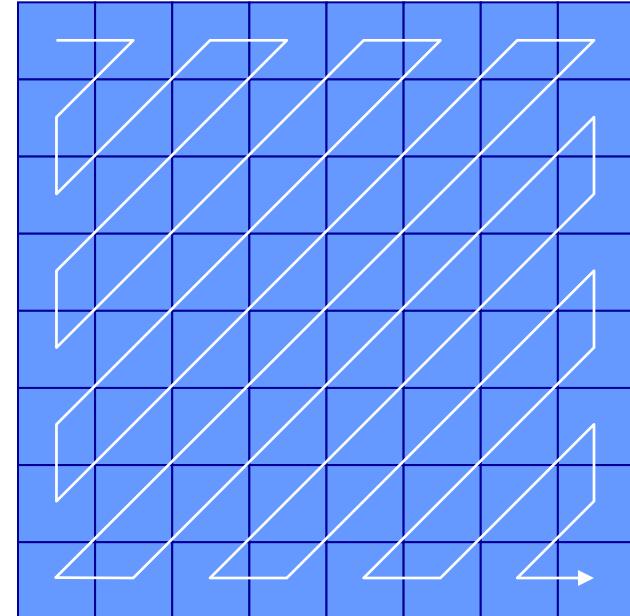
# Algoritmo de compressão JPEG

- Ordenação dos coeficientes DCT

- Propósito do escaneamento zig-zag é ordenar os coeficientes em uma ordem decrescente de freqüências espectral
  - coeficientes de alta freqüências (no canto direito inferior) tem valores mais próximos a zero
    - isto leva a uma maior eficiência da codificação por entropia

1055	86	40	22	15	10	7	5
53	37	25	17	11	8	6	4
21	21	19	13	9	7	5	4
12	12	11	9	7	5	4	3
7	7	7	7	5	4	3	3
5	5	5	4	4	3	3	3
3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	2

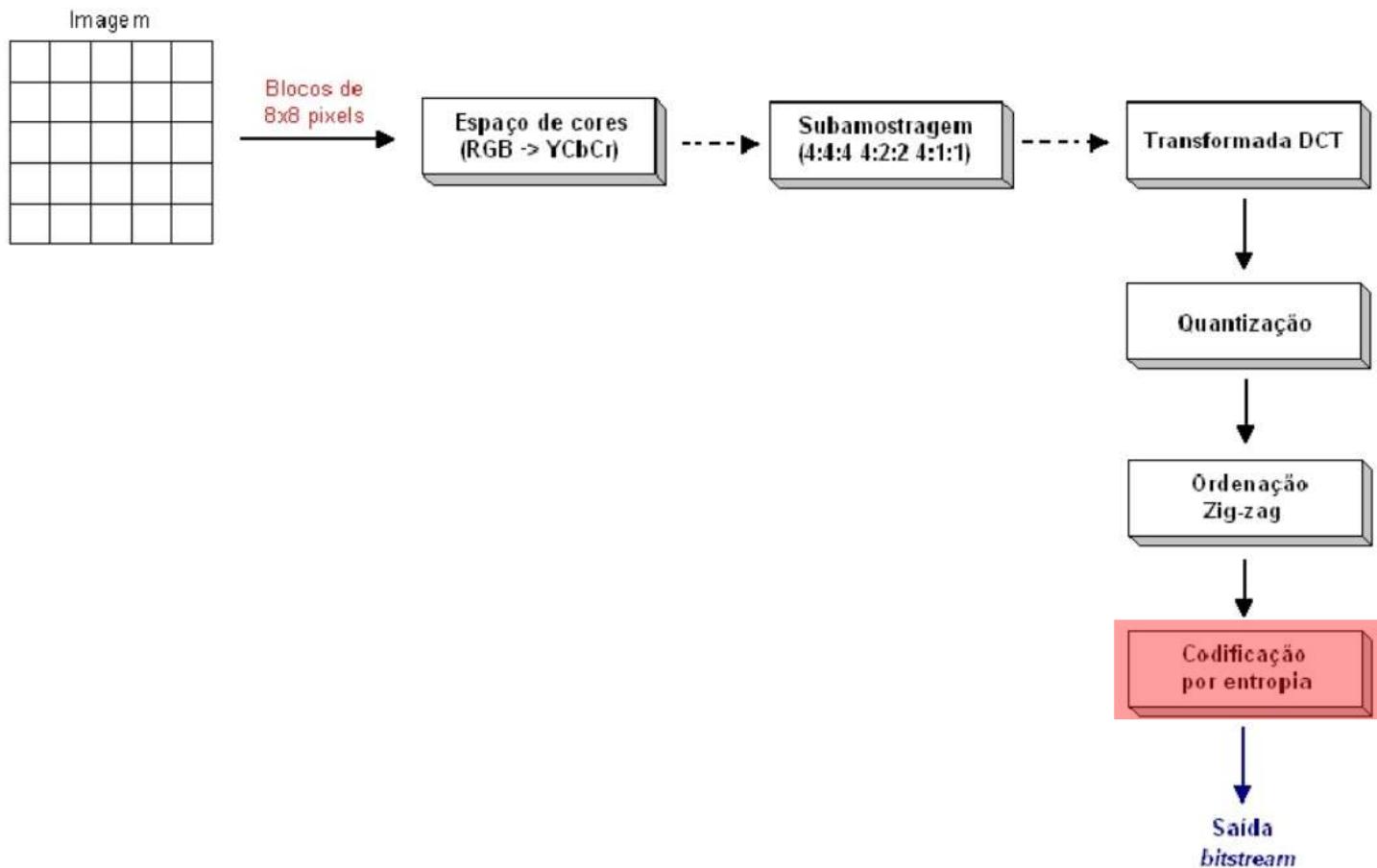
coeficiente 0



coeficiente 63

# Algoritmo de compressão JPEG

- Operações a compressão JPEG



# Algoritmo de compressão JPEG

- Codificação por entropia
  - Esta etapa fornece uma compressão adicional
  - JPEG define dois métodos de codificação por entropia
    - Codificação de Huffman
      - única especificado no modo baseline
    - Codificação aritmética
      - normalmente 10% mais eficiente que a codificação de Huffman

# Padrão de Compressão de Imagens JPEG

- **Taxas de compressão obtidas**
  - Quanto maior for a taxa de compressão maior será o número de componentes de alta freqüência desprezados
    - para obter taxas de compressão muito elevadas é descartado um número significativo de componentes de alta freqüência
      - levando ao aparecimento do efeito de bloco (perda de definição nos contornos das imagens).
  - Valores médios
    - Taxas de compressão de 10:1 a 20:1 – Alta qualidade de imagem
    - Taxas de compressão de 30:1 a 50:1 – Média qualidade de imagem
    - Taxas de compressão de 60:1 a 100:1 – Fraca qualidade de imagem

# Algoritmo de compressão JPEG

- Taxas de compressão obtidas e qualidades

- Alta qualidade
    - Taxa de 2.6:1



- Boa qualidade
    - Taxa de 15:1



# Algoritmo de compressão JPEG

- Taxas de compressão obtidas e qualidades
  - Qualidade média
    - Taxa de 23:1
  - Baixa qualidade
    - Taxa de 46:1



# Algoritmo de compressão JPEG

- **Taxas de compressão obtidas e qualidades**
  - Mais baixa qualidade
    - Taxa de 144:1



- **Demonstração:**
  - <http://cgjennings.ca/toybox/hjpeg/>

# Padrão de Compressão de Imagens JPEG

- **JPEG é para imagens fotográficas**
  - JPEG apresenta ótimas taxas de compressão para imagens fotográficas naturais multi-tonais
  - Qualidade diminui consideravelmente quando aplicado a
    - imagens gráficas com contornos e áreas bem definidas de cor, ou
    - imagens com texto, como é o caso dos logotipos
- **Para imagens gráficas e com texto**
  - JPEG introduz ruído nas zonas de imagem compostas por cores sólidas
    - pode distorcer o aspecto geral da imagem
  - Imagem PNG ou GIF compactam mais eficazmente que JPEG e apresenta uma melhor definição dos contornos do texto

# Algoritmo de compressão JPEG

- JPEG é para imagens fotográficas

Imagen GIF (9,4 Kbytes)



Imagen JPEG (11,7 Kbytes)



- Conclusão
  - imagem GIF comproiva mais eficazmente que a JPEG
  - imagem GIF apresenta uma melhor definição dos contornos do texto

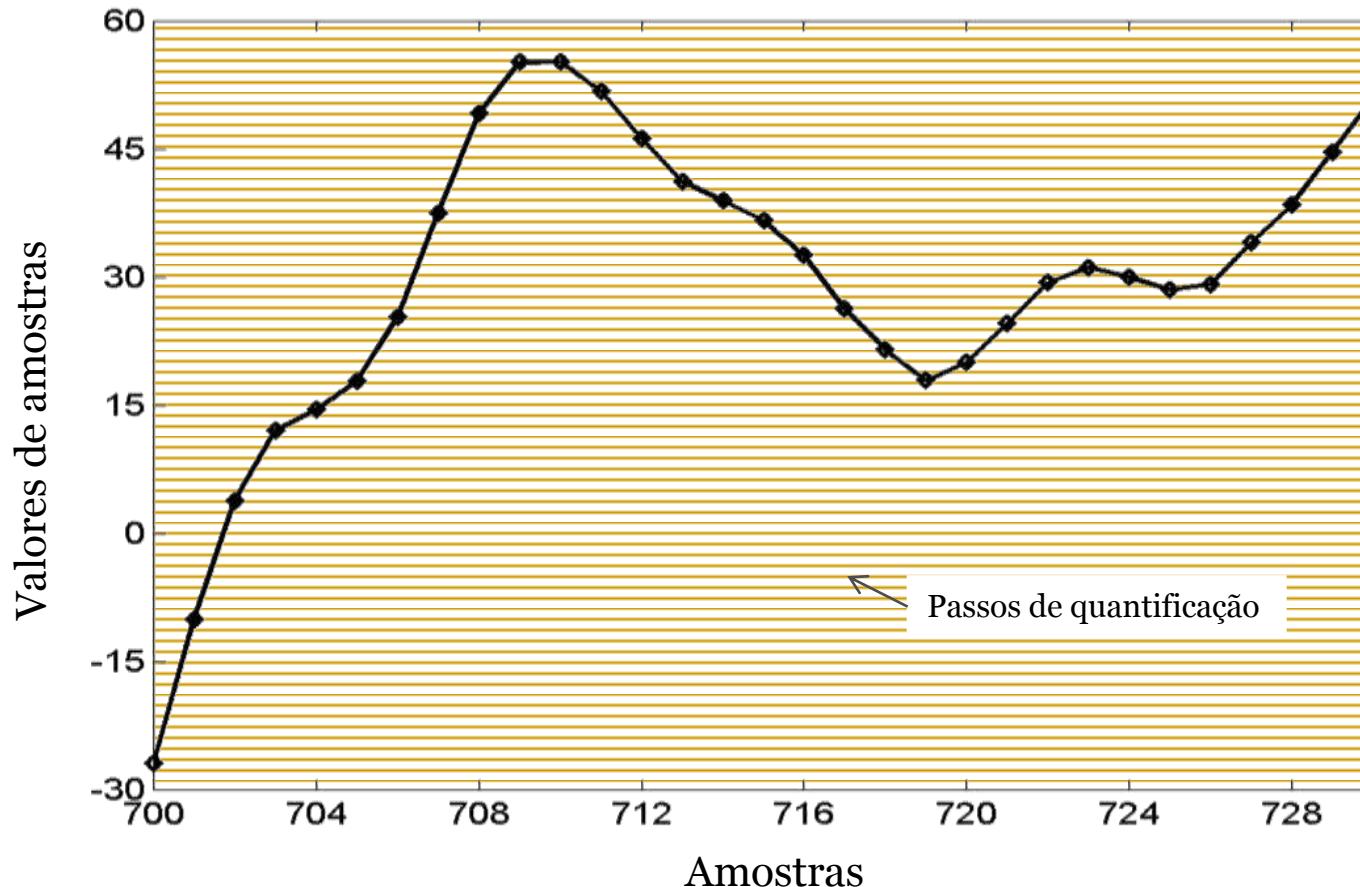
# Cap 3. Compressão de Dados Multimídia

- Conteúdo:

- Necessidade de compressão
- Princípios da compressão
- Classificação das técnicas de compressão
- Medição do desempenho de compressão
- Técnicas de compressão sem perdas
  - RLE, Huffman, LZW (GIF)
- Técnicas de compressão de áudio, vídeo e imagens
  - Técnicas de compressão de voz
  - Técnicas de compressão de som
- Padrões de compressão multimídia
  - JPEG, MPEG, MPEG-4, H.261, H.263

# Técnicas de Compressão de Áudio Digital

- Características do áudio
  - Amostras vizinhas são relacionadas



# Técnicas de Compressão de Áudio Digital

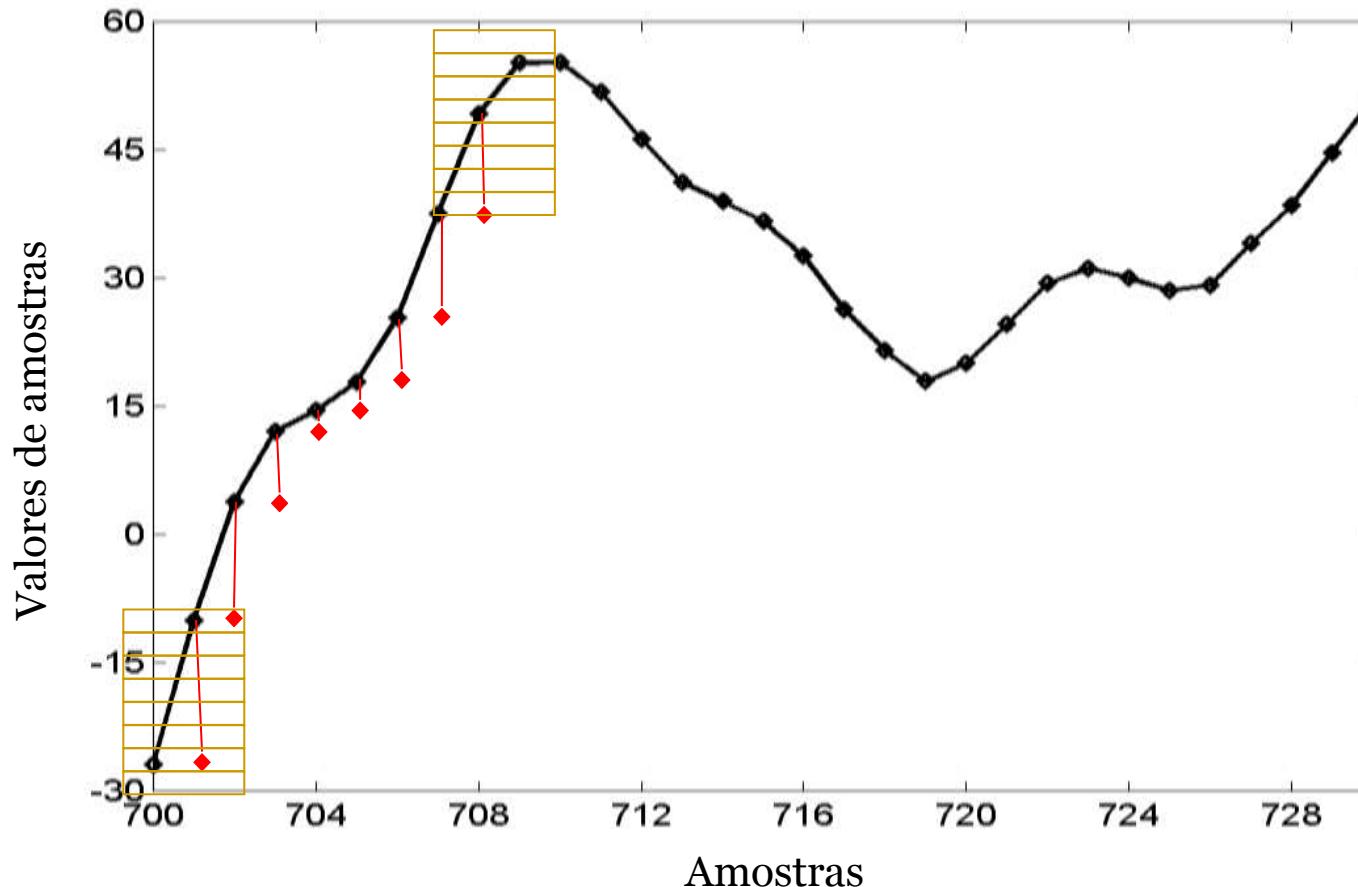
## • Codificação Preditiva

- Quantifica e codifica a diferença entre uma previsão do valor da amostra e do valor real (erro de previsão) em vez da amostra
  - Modulação por pulso codificado diferencial (DPCM)
    - erro de previsão é quantificado e codificado com base nas amostras anteriores
    - na decodificação: erro é adicionado ao valor previsto da amostra (3 ou 4 bits)
- Reduz a representação da informação pois:
  - valores de amostras vizinhas são correlacionados
    - erro de previsão será menor que o valor original da amostra

# Técnicas de Compressão de Áudio Digital

- Características do áudio

- Previsão  $a_i = a_{i-1}$



# Técnicas de Compressão de Áudio Digital

- **No LPC (Linear Predictive Coding)**
  - Uma amostra de áudio é prevista com base nas amostras anteriores

$$x[n] = \sum_{k=1}^P a_k x[n-k] + e[n]$$

- $x[n-k]$ : amostras anteriores
- $p$ : ordem do modelo
- $a_k$ : coeficiente de previsão
- $e[n]$ : erro de previsão

# Técnicas de Compressão de Áudio Digital

- Codificação Predictiva

- DPCM adaptativo (ADPCM) pode ser utilizado para aumentar a qualidade mantendo a taxa de bits
  - tamanho passo de quantificação aumenta com o aumento da variação do sinal
    - Se o sinal passa bruscamente de uma tensão elevada a uma tensão baixa, o valor do passo será grande; ao contrário, se o sinal de entrada apresenta variações de tensão baixas, o tamanho do passo será pequeno
    - Menos bits são usados para diferenças maiores
- Usado para mapear uma série de amostras de 8 bits  $\mu$ -law (ou a-law) em uma série de amostras ADPCM de 4 bits
  - A capacidade da linha é dobrada

# Recomendações ITU-TS para Voz

Recomend.	Técnica de Compressão	Largura de banda da voz (kHz)	Taxa de amostragem (kHz)	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	PCM não linear	3,4	8	64	20	160
G.721	ADPCM	3,4	8	32		
G.722	ADPCM sub-banda	7	16	48, 56, 64		
G.723.1m	MP-MLQ	3,4	8	6.4	30	24
G.723.1a	ACELP	3,4	8	5.3	30	20
G.726	ADPCM	3,4	8	16, 24, 32, 40	15	60
G.728	LD-CELP	3,4	8	16	20	40
G.729A	CS-CELP	3,4	8	8	20	20

# Codecs, Quadros e Pacotes de Voz

- **Quadro de voz**
  - A maioria dos codificadores de voz se baseia em quadros
    - Quadros de voz ou pacotes de voz
    - Cada quadro de voz tem uma duração de 1 a 30 ms
  - Codecs compactam quadros de voz
    - contendo um número fixo de amostras
    - Número de amostras depende do codec utilizado

# Codecs, Quadros e Pacotes de Voz

CAD

Quadro

- Montagem do quadro
  - Fluxo de dados de áudio precisa ser acumulado
    - até que ele atinja o tamanho do bloco antes de ser processado pelo codificador
  - Acumulação de amostra leva tempo
    - soma-se ao atraso fim-a-fim.

# Codecs, Quadros e Pacotes de Voz

CAD

Quadro

Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra

# Codecs, Quadros e Pacotes de Voz

CAD

Quadro



- Quadro é compactado

- Usando um algoritmo particular

# Codecs, Quadros e Pacotes de Voz

CAD

- Quadro é compactado

- Usando um algoritmo particular

Quadro

Quadro Compactado



# Codecs, Quadros e Pacotes de Voz

CAD

Quadro

Quadro  
Compactado

# Codecs, Quadros e Pacotes de Voz

CAD

Quadro

Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra  
Amostra

Quadro  
Compactado

# Codecs, Quadros e Pacotes de Voz

CAD

Quadro



Quadro  
Compactado

# Codecs, Quadros e Pacotes de Voz

CAD

Quadro

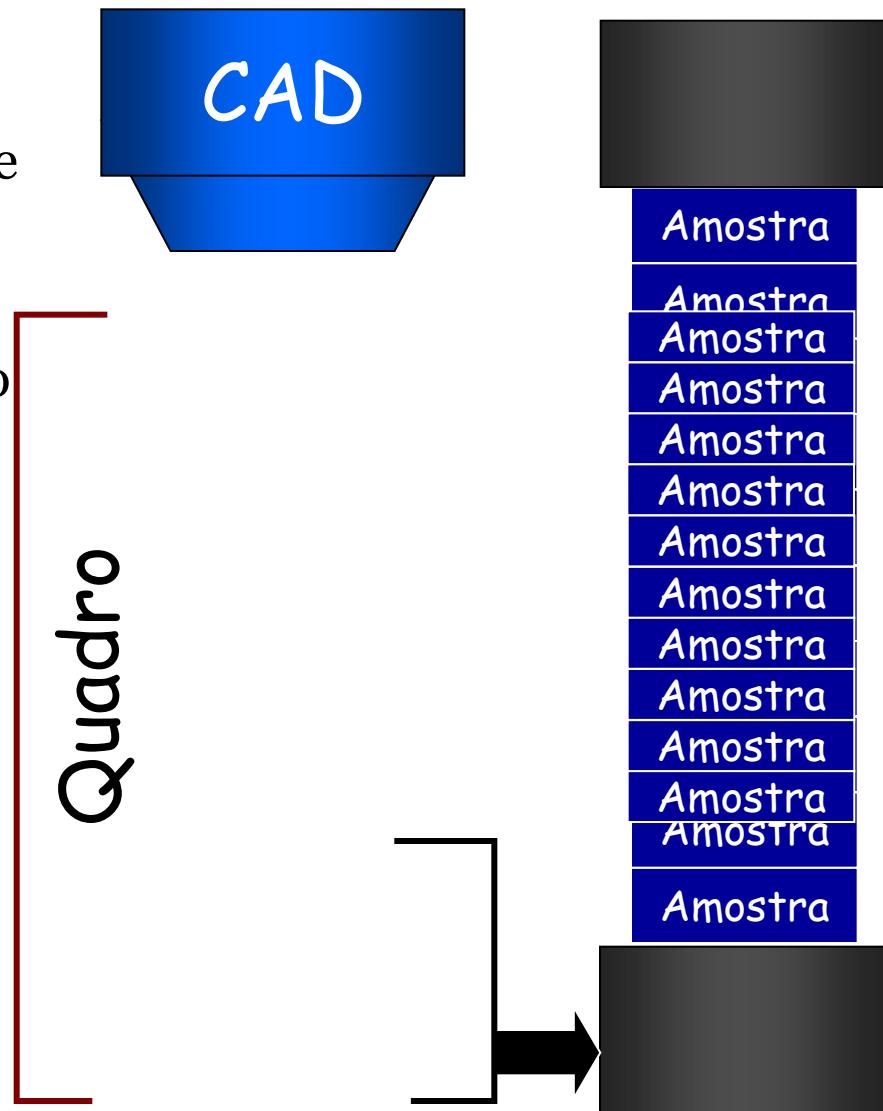
Quadro  
Compactado

Quadro  
Compactado

# Codecs, Quadros e Pacotes de Voz

- Look-ahead

- Alguns codificadores precisam conhecer mais amostras do que aquelas contidas no quadro que eles vão codificar.
- Aumenta o atraso do algoritmo do codec



# Codecs, Quadros e Pacotes de Voz

- Tamanho do pacote de voz

- Tamanho do payload (em bytes)

Taxa do Codec (em bits/sec) x tempo do pacote de voz (ms)

$$8 \text{ (bits/byte)} \times 1000 \text{ (ms/sec)}$$

- Exemplo 1:

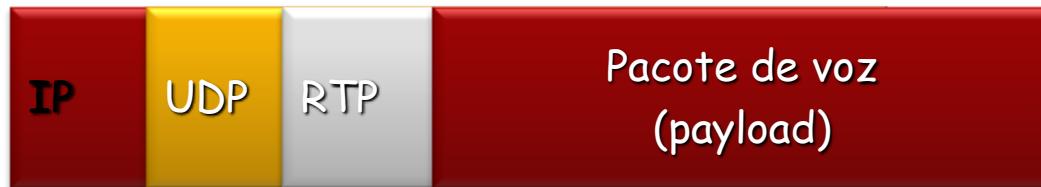
- Codec G.711 => 64 kbps
    - Tamanho do pacotes = 20ms
    - Tamanho do payload =  $(64000 \times 20)/8000 = 160$  bytes
    - Teríamos  $(1000/20) = 50$  pacotes de 160 B de dados a cada segundo

- Exemplo 2:

- Codec G.711 => 64 kbps
    - Tamanho do pacotes = 30ms
    - Tamanho do payload =  $(64000 \times 30)/8000 = 240$  bytes
    - Teríamos 33 pacotes de 240 B de dados a cada segundo

# Codecs, Quadros e Pacotes de Voz

- Problema da sobrecarga de protocolos
  - Para ser transmitido na rede, o pacote de voz deve ser encapsulado em diversos protocolos
    - Até chegar à camada de enlace, aos pacotes de voz vão ser adicionados 40 bytes: RTP (12 bytes) + UDP (8 bytes) + IP (20 bytes) = 40 bytes.
  - Exemplo
    - Tamanho típico do payload quando usado o codec G.729 é 20 bytes
    - Taxa do codec sobre de 8 kbps para 24 kbps na camada de rede



# Codecs, Quadros e Pacotes de Voz

- **Redução de atraso x Taxa de compressão**
  - Para redução do atraso, o codec escolhido deveria ter um tamanho de quadro pequeno
    - Mas caso o quadro for pequeno a taxa de compressão tende a ser menor
  - Exemplo 1:
    - Codec G.711 => 64 kbps
    - Tamanho do pacotes = 20ms
    - Tempo de empacotamento será 20ms
  - Exemplo 2:
    - Codec G.711 => 64 kbps
    - Tamanho do pacotes = 30ms
    - Tempo de empacotamento será de 30ms

# Codecs, Quadros e Pacotes de Voz

- Redução do atraso x Taxa de bits
  - Quadro voz são transportados por pacotes encapsulando diversos protocolos (por exemplo, RTP, UDP, IP)
    - Se o quadro é pequeno, a sobrecarga do cabeçalho dos protocolos será maior
  - Exemplo 1: Codec G.711 => 64 kbps
    - Tamanho do pacote de voz = 20ms
    - Tamanho do Pacote IP  $40 + 160 = 200$  B
    - Teríamos 50 pacotes IP de 200 B de dados a cada segundo
      - Taxa de bits é de  $50 * 200 * 8 = 80$  kbps
  - Exemplo 2: Codec G.711 => 64 kbps
    - Tamanho do pacote de voz = 30ms
    - Tamanho do pacote IP =  $40 + 240 = 280$  B
    - Teríamos 33 pacotes de 280 B de dados a cada segundo
      - Taxa de bits é de  $33 * 280 * 8 = 73,9$  kbps

# Recomendações ITU-TS para Voz

Recomend.	Técnica de Compressão	Largura de banda da voz (kHz)	Taxa de amostragem (kHz)	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	PCM não linear	3,4	8	64	20	160

- **G.711**
  - Usa PCM compandido (escala semilogarítmica)
    - Serve para aumentar a resolução de sinais de baixa amplitude
      - Mais importante para os humanos
    - Operando de forma análoga ao ouvido humano
  - Dois tipos de escala
    - A-law (Europa e Brasil)
    - $\mu$ -law (EUA e Japão)
  - Usado em redes ISDN e na maioria dos backbones telefônicos digitais
  - Fluxo de bits de 64 kbps
    - 8 bits por amostra, 8000 amostras/s (uma amostra a cada  $125\mu\text{s}$ )
  - Supressão de silêncio é opcional
    - Reduz a taxa de bits gerada

# Recomendações ITU-TS para Voz

Recomend.	Técnica de Compressão	Largura de banda da voz (kHz)	Taxa de amostragem (kHz)	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	PCM não linear	3,4	8	64	20	160

- **G.711**
  - Quanto utilizado em aplicações conversacionais (voz sobre IP, etc.)
    - amostras são agrupadas em blocos para envio na forma de pacotes IP: payload
    - Pacotes de voz são enviados em intervalos constantes
  - Valor típico do payload no G.711 é de 160 amostras
    - constituindo um tempo de voz de 20ms
  - Tamanho do payload em bytes pode ser calculado por
    - $\text{Payload} = (\text{taxa} * \text{duração}) / 8$

# Recomendações ITU-TS para Voz

Recomendação	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	64	20	160
G.723.1m	6.4	30	24
G.723.1a	5.3	30	20
G.726	16, 24, 32, 40	15	60
G.728	16	20	40
G.729A	8	20	20

- **Tamanho do Payload**
  - Quanto menor o payload de voz maior é a banda do canal necessária
    - isto devido ao aumento da sobrecarga dos diversos protocolos de transmissão da voz
  - Quanto maior o payload maior é o atraso na aplicação
    - para aguardar a montagem do payload

# Recomendações ITU-TS para Voz

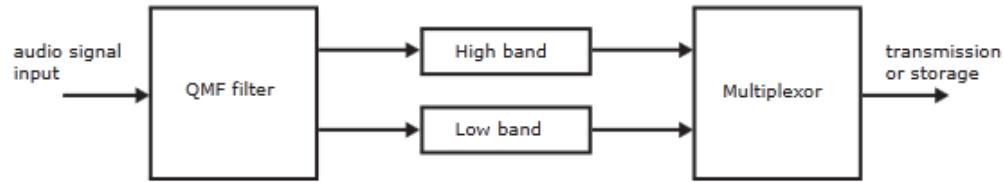
Recomend.	Técnica de Compressão	Largura de banda da voz (kHz)	Taxa de amostragem (kHz)	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	PCM não linear	3,4	8	64	20	160
G.721	ADPCM	3,4	8	32		
G.722	ADPCM sub-banda	7	16	48, 56, 64		

- **G.721**
  - Converte um fluxo de 64 kbps em um fluxo de 32 kbps aplicando uma compressão ADPCM
  - A previsão e o tamanho do passo altera com o histórico do sinal

# Recomendações ITU-TS para Voz

Recomend.	Técnica de Compressão	Largura de banda da voz (kHz)	Taxa de amostragem (kHz)	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	PCM não linear	3,4	8	64	20	160
G.721	ADPCM	3,4	8	32		
G.722	ADPCM sub-banda	7	16	48, 56, 64		

- **G.722**
  - fornece uma melhor qualidade que o G.711 e G.721
    - Utiliza 14 bits por amostra
  - ADPCM Sub-banda: sinal de voz é dividido em duas sub-bandas: alta (4-8kHz) e baixa (0-4kHz)
    - 2 bits/amostra para banda alta (16 kbps)
    - 6 bits/amostra para banda baixa (48 kbps)
  - Próprio para aplicações de videoconferência uma vez que telefones comuns não respondem na faixa de 7kHz



# Recomendações ITU-TS para Voz

Recomend.	Técnica de Compressão	Largura de banda da voz (kHz)	Taxa de amostragem (kHz)	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	PCM não linear	3,4	8	64	20	160
G.723.1m	MP-MLQ	3,4	8	6.4	30	24
G.723.1a	ACELP	3,4	8	5.3	30	20

- **G.723.1**
  - Codec mínimo de referência para o padrão H.323
  - Opera a 6,4 kbps (*Multipulse-Maximum Likelihood Quantification*) e a 5,3 kbps (*Algebraic-Code-Excited Linear Prediction*)
  - Em cada janela de 30 ms do sinal de voz
    - são analisadas 240 amostras de 16 bits do sinal de voz (tomadas a 8kHz) para identificação de padrões repetitivos (pitches) e são gerados 12 ou 10 códigos de 16 bits, conforme o algoritmo esteja configurado para uma taxa de 6,3 ou 5,3 kbps
  - Valor típico de tamanho do pacote de voz (payload) é de 30ms (20 ou 24 bytes)

# Recomendações ITU-TS para Voz

Recomend.	Técnica de Compressão	Largura de banda da voz (kHz)	Taxa de amostragem (kHz)	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	PCM não linear	3,4	8	64	20	160
G.723.1m	MP-MLQ	3,4	8	6.4	30	24
G.723.1a	ACELP	3,4	8	5.3	30	20
G.726	ADPCM	3,4	8	16, 24, 32, 40	15	60
G.728	LD-CELP	3,4	8	16	20	40
G.729A	CS-CELP	3,4	8	8	20	20

- G.726
  - O G.726 utiliza o ADPCM a 40, 32, 24 e 16 kbps
    - codificação da diferença entre amostras consecutivas é feita de forma não linear já que o valor do passo de quantificação é variável
  - Sinal de voz é amostrado a 8kHz, codificado em 8 bits (leis A ou  $\mu$ ) e são transmitidas diferenças entre amostras com 5, 4, 3 ou 2 bit em quantificação adaptativa
  - valor típico de tamanho do pacote de voz (payload) é de 15ms (60 bytes)

# Recomendações ITU-TS para Voz

Recomend.	Técnica de Compressão	Largura de banda da voz (kHz)	Taxa de amostragem (kHz)	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	PCM não linear	3,4	8	64	20	160
G.723.1m	MP-MLQ	3,4	8	6.4	30	24
G.723.1a	ACELP	3,4	8	5.3	30	20
G.726	ADPCM	3,4	8	16, 24, 32, 40	15	60
G.728	LD-CELP	3,4	8	16	20	40
G.729A	CS-CELP	3,4	8	8	20	20

- **G.728**
  - Técnica de codificação LD-CELP (Low-Delay, Code-Excited Linear Prediction), gerando uma taxa de bits de 16 kbps
  - Em cada janela de 0,625ms do sinal de voz são analisadas 5 amostras de 8 bits e é gerado 1 código de 10 bits
  - Tabela (codebook) utilizada é formada por 1024 ( $2^{10}$ ) valores
    - contém os valores de códigos (vetores) que representam as possíveis amostras do sinal de voz
  - Valor típico de tamanho do pacote de voz é de 20ms (40 bytes)

# Recomendações ITU-TS para Voz

Recomend.	Técnica de Compressão	Largura de banda da voz (kHz)	Taxa de amostragem (kHz)	Taxa de bit compactado (kbps)	Duração do payload (ms)	Tamanho do payload (bytes)
G.711	PCM não linear	3,4	8	64	20	160
G.723.1m	MP-MLQ	3,4	8	6.4	30	24
G.723.1a	ACELP	3,4	8	5.3	30	20
G.726	ADPCM	3,4	8	16, 24, 32, 40	15	60
G.728	LD-CELP	3,4	8	16	20	40
G.729A	CS-CELP	3,4	8	8	20	20

- **G.729**

- Bastante popular em aplicações de voz sobre frame relay e em modems V.70 para voz e dados
- G.729 Técnica de codificação LD-CELP gerando uma taxa de bits de 8 kbps e G.729A a codificação CS-ACELP (Algebraic-ACELP)
- Em cada janela de 10ms do sinal de voz são analisadas 80 amostras de 8 bits para geração de 10 códigos de 8 bits
- Valor típico de tamanho do pacote de voz é de 20ms (20 bytes)

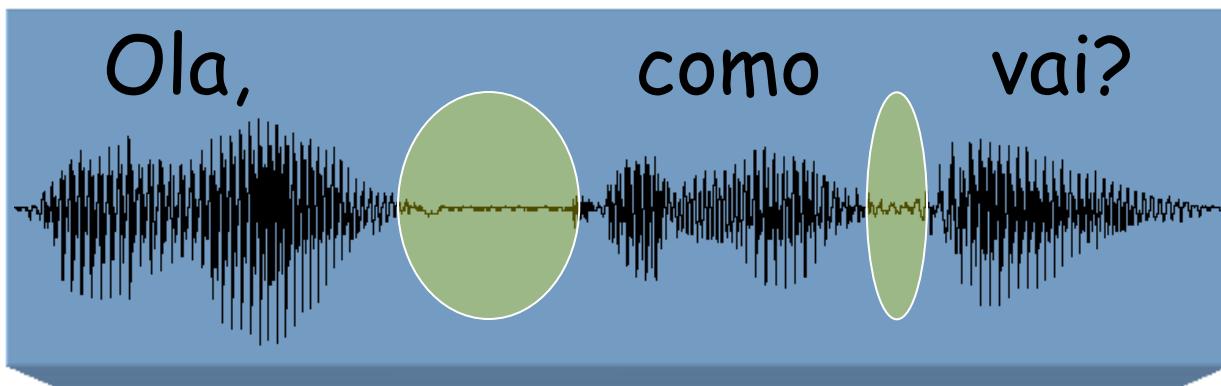
# Supressão de Silêncio e remoção de sons repetitivos

- Compressão da voz via remoção dos períodos de silêncio e de informações redundantes encontradas na fala humana
  - Existem informações na fala humana que não são necessárias para que uma comunicação efetiva exista através de uma rede
- Sons repetitivos, inerentes à voz, são causados pela vibração das cordas vocais
  - transmissão destes sons idênticos não é necessária para efetivação da comunicação e a sua remoção resulta em um aumento de eficiência na utilização da banda de rede
- Composição da fala
  - 22% do que se fala são componentes essenciais da comunicação
    - devem ser transmitidos para o entendimento do diálogo
  - 22% são padrões repetitivos
  - 56% representa as pausas entre falas

# Supressão do silêncio: Componentes

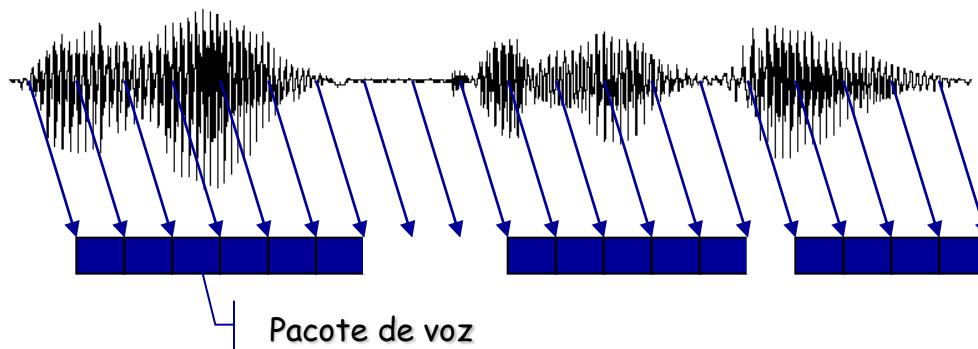
- **VAD (Detector de Presença de Voz)**

- Responsável por determinar quando o usuário está conversando e quando ele está em silêncio
- É útil para economizar energia no caso de dispositivos que funcionam a bateria
- Deve ser bastante sensível
  - Caso contrário, o início das palavras podem ser perdidas e um silêncio inútil pode ser incluído no final das sentenças
  - Mas ao mesmo tempo não pode ser disparado por ruído de fundo.



# Supressão do silêncio: Componentes

- **DTX (Discontinuous Transmission)**
  - Capacidade de um codec de parar de transmitir quadros quando o VAD tiver detectado um período de silêncio
  - VAD + DTX: modo eficiente de liberar dinamicamente a banda
    - proporcionando uma economia de até 50% da banda
  - Alguns codecs avançados não vão interromper a transmissão completamente
    - Em vez disso, vão para um modo de silêncio no qual usam muito menos largura de banda e enviam apenas os parâmetros mínimos para que o receptor possa restituir o ruído de fundo (intensidade, etc.)



# Supressão de Silêncio e remoção de sons repetitivos

- Alguns pontos devem ser considerados na supressão do silêncio
  - Quando a fala é muito frequente, contínua, os ganhos com a supressão do silêncio não são alcançados;
  - Os algoritmos de compressão avançados já possuem integradas as funções de VAD;
  - Como a detecção da presença de voz na transmissão não é imediata
    - Pode ocorrer o corte das primeiras sílabas da locução
    - Fenômeno é denominado de clipping;
  - Quando o ruído de fundo é muito alto
    - Torna-se difícil distinguir entre o que é ruído e o que realmente é fala
    - Corre-se o perigo de empacotamento de ruído.

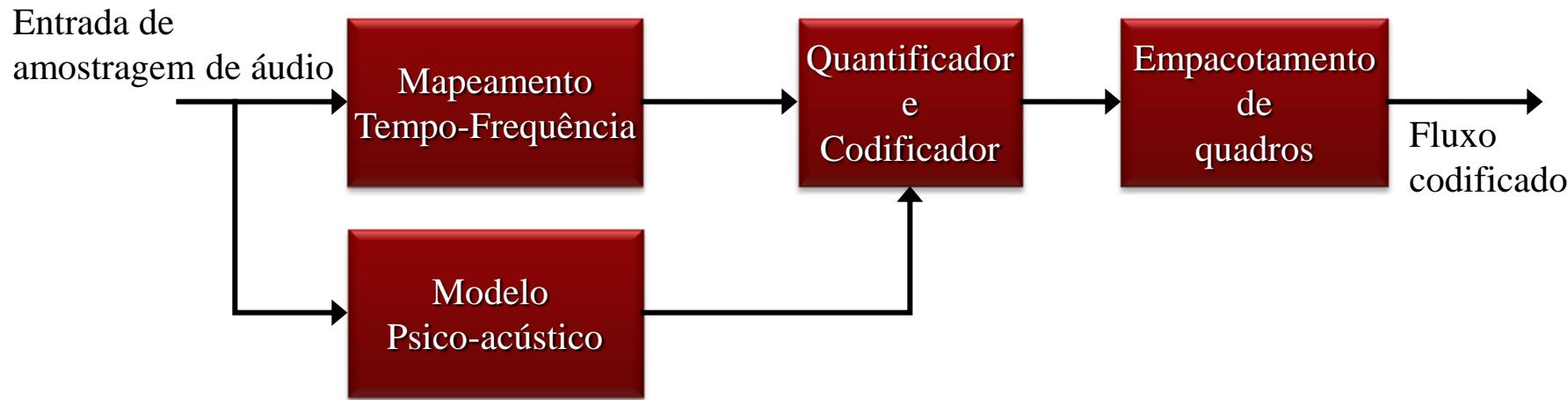
# MPEG-Audio

- Padrão de compressão de áudio genérico (até 20 kHz)
  - e não apenas para voz (de 3,4 a 7 kHz)
  - explora a percepção humana e não as características da fonte do áudio
- Princípio de Compressão
  - Faixa de freqüência audível humana
    - Filtra sons acima de 20 KHz
  - Limiar de audição na faixa de freqüência audível
    - Explora a curva de percepção da audição humana dentro da faixa de freqüências audíveis (limiar de audição)
      - Sensibilidade para sons dentro desta faixa não é uniforme (depende da freqüência)
      - O que se faz é descartar amostras que se encontrem abaixo deste limiar.
  - Mascaragem: um som pode tornar outro impossível de ser ouvido, ou pode tornar o outro sem peso
    - tipos de mascaração: total ou parcial
    - sons mascarados podem ser descartados (não são audíveis)
    - característica explorada pelo padrão internacional MPEG-Áudio
      - explora as limitações perceptivas do sistema auditivo humano

# MPEG-Audio

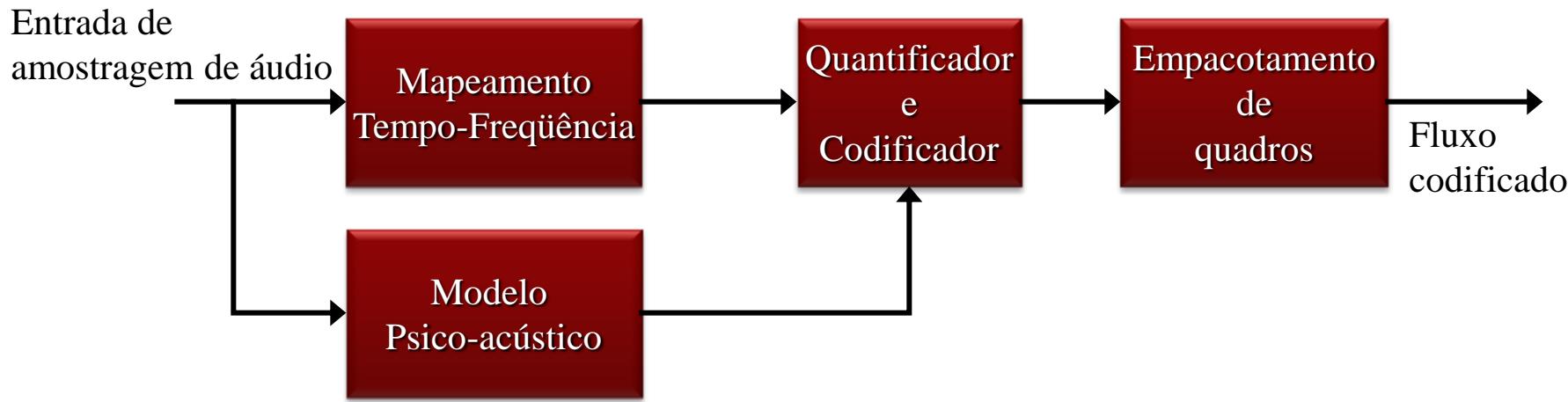
- **Principais características do MPEG-1 Audio:**
  - Seqüência de bits compactada pode suportar um ou dois canais
    - um canal único
    - dois canais independentes
    - um sinal estéreo
  - Três taxas de amostragens
    - 32, 44.1 ou 48 kHz
    - MPEG-2.5 (não oficial) 8, 11.025, 12, 16, 22.05 e 24 kHz.
  - Fluxo compactado pode ter uma das várias taxas de bits fixas e predefinidas variando de 32 a 320 kbps
  - Padrão MPEG-2.5 (não oficial)
    - Taxa de bits de 8, 16, 24, e 144 kbps
  - Razão de compressão: 2,7 a 24 (depende da taxa de amostragem)
    - 6:1 ouvintes experientes não detectam diferenças

# Um codificador básico MPEG-Áudio



- **Bloco mapeamento tempo-frequência**
  - Divide a entrada em sub-bandas de frequências múltiplas
- **Bloco modelo psico-acústico**
  - Cria um conjunto de dados para controlar a operação do bloco quantificador e codificador

# Um codificador básico MPEG-Áudio



- **Bloco quantificador e codificador**
  - Cria um conjunto de símbolos de código
    - sub-bandas menos importantes e áudios inaudíveis são removidos
- **Bloco Empacotamento de quadros**
  - Monta e formata os símbolos de código e adiciona outras informações

# MPEG Audio

- MPEG Audio especifica uma família de 3 esquemas de codificação de áudio
  - Chamadas de Layer-1, Layer-2 e Layer-3
    - de Layer-1 a Layer-3, a complexidade e desempenho (qualidade de som e taxa de bits) aumentam
  - Os três codificadores são compatíveis no modo hierárquico
    - decodificador Layer-N é capaz de decodificar um fluxo de bits fluxo codificado com codificador Layer-N e abaixo de N
  - MP3 é MPEG-1 Layer-3
- Padrão especifica o formato do fluxo de bits e o decodificador para cada esquema de codificação
  - não especifica o codificador para avanços futuros

# MPEG-2 Áudio

- Estende as funcionalidades do MPEG-1 Áudio
  - cinco canais (esquerdo, direito, centro, e dois canais surround)
  - mais um canal de baixa frequência
  - ou sete canais multilíngues/comentários
  - tem taxas de amostragens adicionais

# Padrão ISO/IEC MPEG

- Motion Picture Expert Group
  - Grupo de padrões de representação codificada de vídeos, áudios e suas combinações
    - armazenados e recuperados em Digital Storage Media (DSM)
      - dispositivos de armazenamento convencionais, CD-ROMs, drivers de fita, HDs, drivers ótico escrevíveis e canais de telecomunicação (redes de longa distância, locais, etc.)
- Características
  - MPEG usa a compressão intra e inter-quadros de vídeos
    - obtém altas taxas de compressão devido a alta redundância dos vídeos
  - Especificações MPEG também incluem um algoritmo para compressão de áudio
    - Compressão do áudio associado e a sincronização áudio-visual não podem ser independente da compressão do vídeo

# Padrão ISO/IEC MPEG

- Vários itens de trabalho
  - MPEG-1 (1993)
    - vídeo pode codificar imagens de até 4096x4096 pixels e 60 fps.
    - A maioria das aplicações usam o formato SIF, com 240x352 pixels e 30fps , e sub-amostragem de crominância 4:2:0.
  - MPEG-2 (1994)
    - Pode codificar imagens de até 16.383 x 16.383 pixels
    - Padrão organizado em perfis e níveis. Exemplos:
      - nível baixo (240 x 352 pixels x 30 fps – idêntico ao SIF MPEG 1),
      - nível principal, visando a codificação com qualidade de TV (720 x 480 e 30 fps), e
      - os níveis alto, visando a TV de alta resolução - HDTV, e a produção de filmes (em geral 1280 x 720 e 30fps; 1920 x 1080 e 30 fps ou 1440 x 1152 e 30 fps).
    - O padrão permite subamostragem de crominância 4:2:0, 4:2:2 e 4:4:4.

# Padrão ISO/IEC MPEG

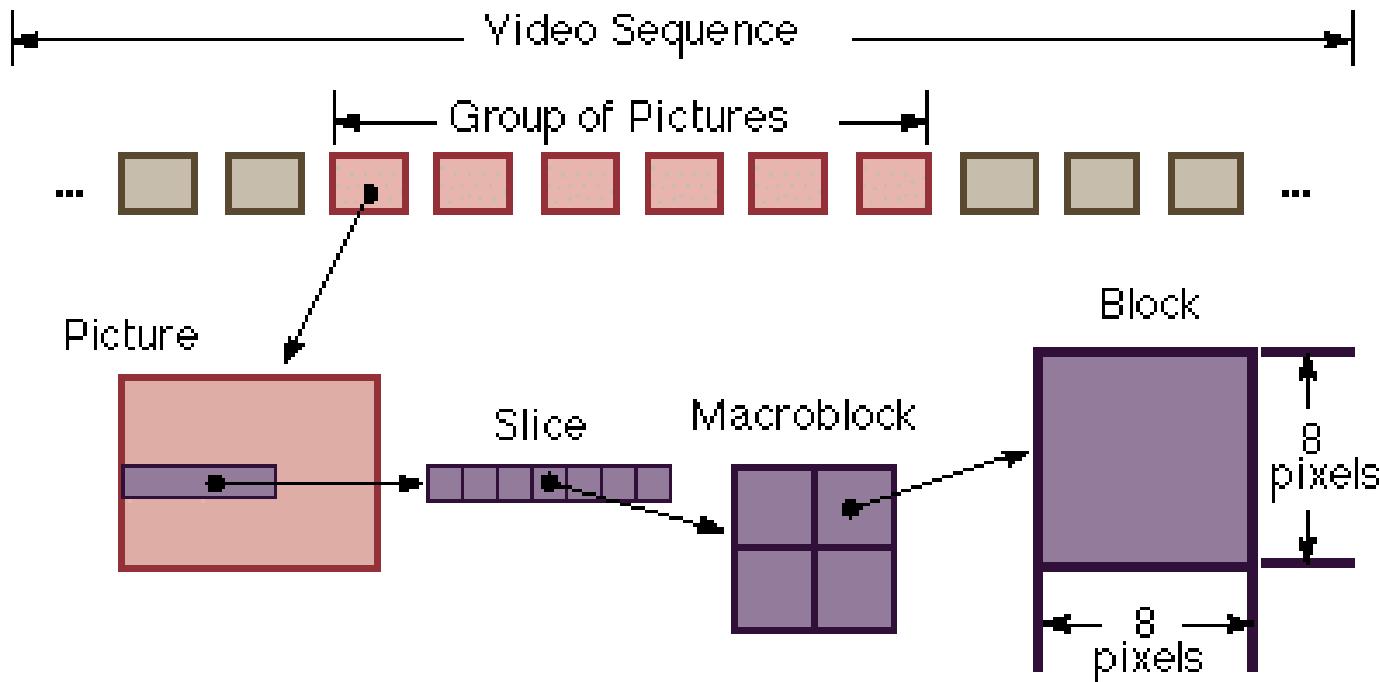
- Vários itens de trabalho
  - MPEG-3
    - para vídeo com qualidade HDTV na taxa de 40 Mbps
      - interrompido em julho 1992
  - MPEG-4 (1998)
    - Objetivo inicia: codificação para audiovisual com muito baixa taxa de transmissão (variando de 4,8 a 64 Kbps)
    - Hoje: oferece soluções para vários tipos de aplicações com qualidades diferentes
  - MPEG-7 (2001)
    - Interface de Descrição de Conteúdo Multimídia: um padrão de descrição de dados multimídia (informações audiovisuais)
      - Permitindo a busca e filtragem

# Padrão ISO/IEC MPEG

- **MPEG 1/2 tem quatro partes principais**
  - **MPEG-Vídeo**
    - compressão de sinais de vídeo;
  - **MPEG-Áudio**
    - compressão de um sinal de áudio digital
  - **MPEG-Sistemas**
    - sincronização e multiplexação de fluxos de áudio e vídeo compactados
  - **Conformidade**
    - especifica procedimento para determinar as características dos fluxos codificados e para testar a conformância com os requisitos identificados no Áudio, Vídeo e Sistemas

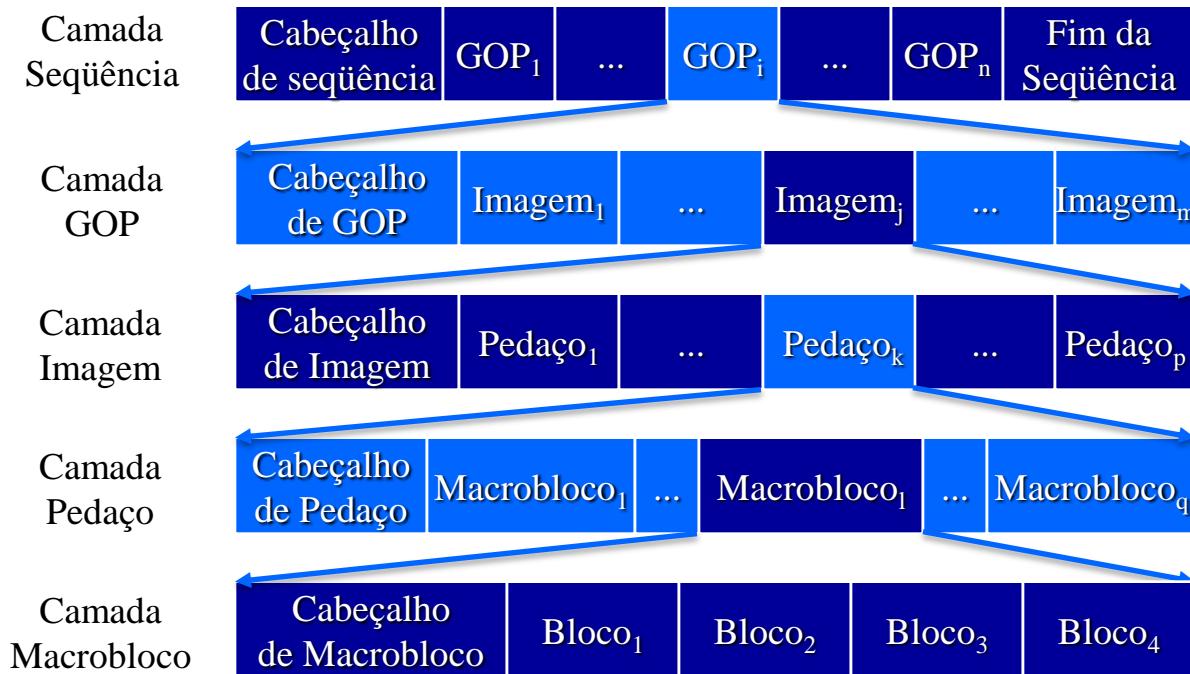
# Padrão ISO/IEC MPEG

- Hierarquia do fluxo de dados MPEG-1



# Especificação do Fluxo de Vídeo MPEG-1

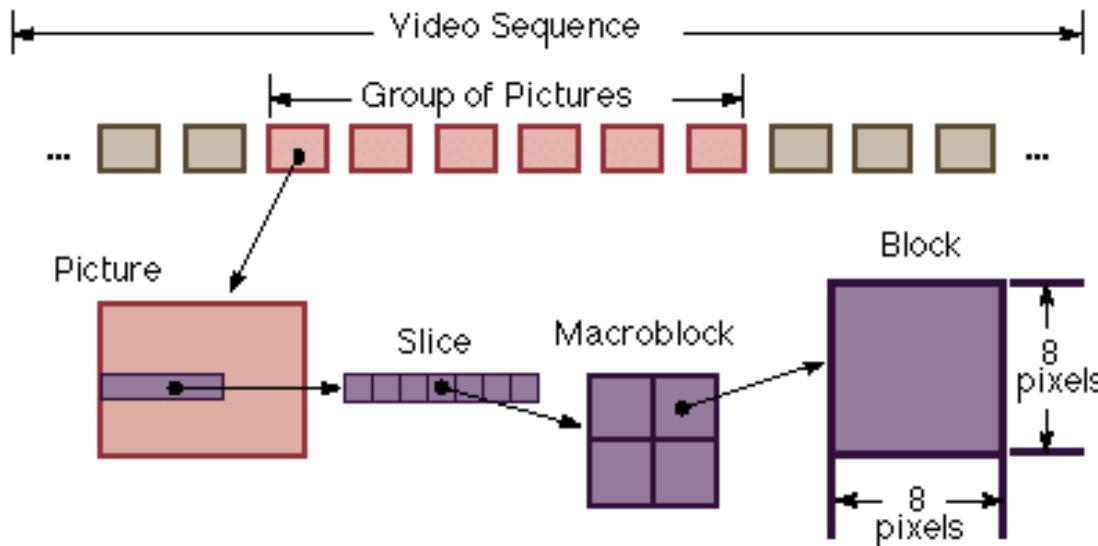
- Fluxo de vídeo:



- GOP (Grupo de imagens): fornece um ponto de acesso aleatório
- Camada de imagem contém todas as informações codificadas de uma imagem
  - cabeçalho contem a referência temporal de uma imagem, o tipo de codificação, etc..

# Especificação do Fluxo de Vídeo MPEG-1

- Fluxo de vídeo



- Imagens são divididas em pedaços (slices):
  - cada pedaço consiste de um número de macroblocos (16x16 pixéis)
  - Importante para o controle de erro
    - Se existe um erro no fluxo de dados, o decodificador pode saltar um pedaço
    - Maior o número de pedaços, melhor é o tratamento de erro
- Macrobloco
  - Usado na estimativa e compensação do movimento
- Bloco de Imagem
  - um bloco é uma matriz 8x8 pixéis tratados como unidades e entrada para o DCT

# Padrão ISO/IEC MPEG

- MPEG especifica a sintaxe dos fluxos codificados para que decodificadores possam decodificar
  - como gerar o bitstream não é padronizado
    - permite inovações no projeto e implementação de codificadores

# Especificação do Fluxo de Vídeo MPEG-1

- **Imagen**

- É a unidade elementar para a codificação do vídeo
- Imagem MPEG
  - Grupo de três matrizes retangulares que representam a luminância (Y) e a crominância (Cr e Cb)
    - Um elemento da matriz é 1 pixel
  - Representação YCrCb é equivalente ao RGB
    - É preferível YCrCb pois o olho é mais sensível a luminosidade que a crominância
      - armazenando menos informação nas matrizes Cr e Cb que na matriz Y
      - Na codificação 4:2:2 (mais corrente), as matrizes Cr e Cb são de dimensão duas vezes menor que a matriz Y



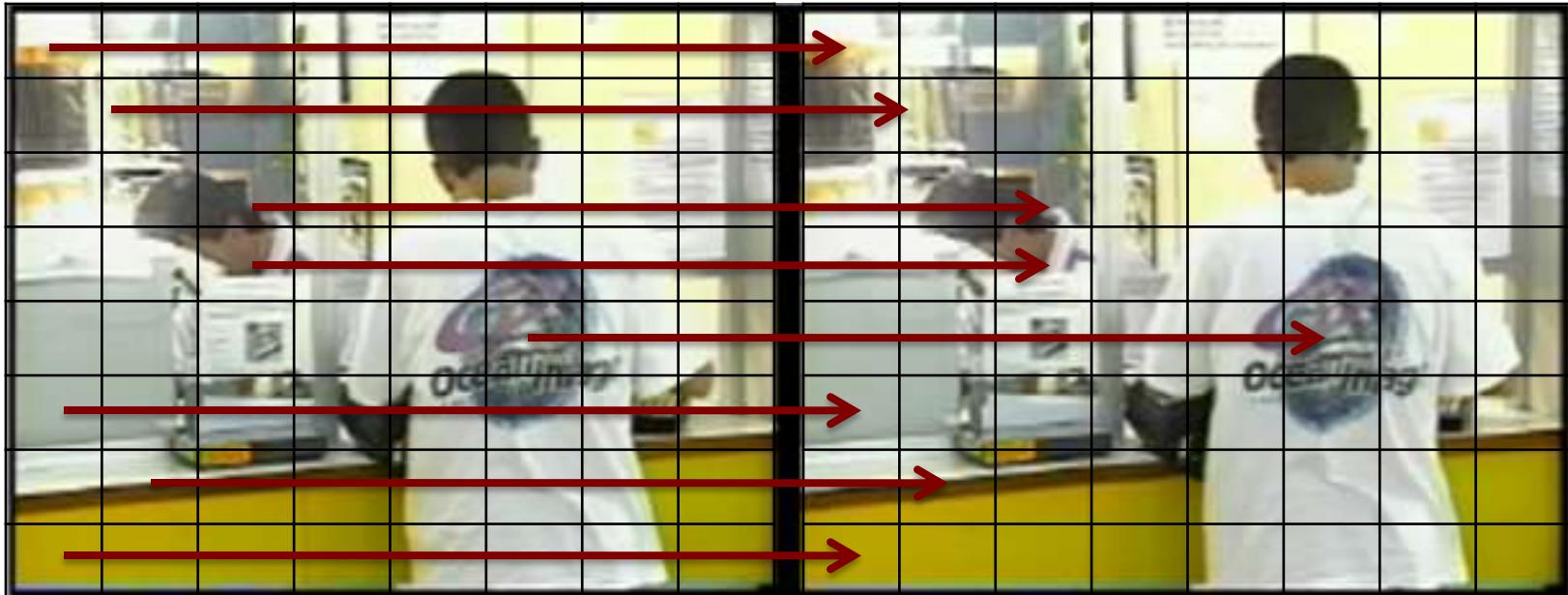
# Padrão ISO/IEC MPEG

- Compactação MPEG-1 Video
  - Em vídeo existem dois tipos de redundância: espacial e temporal
    - MPEG-1 explora estes duas redundâncias
  - Redundância espacial
    - pode ser explorada pela simples codificação em separado de cada quadro com o JPEG
    - estratégia utiliza quando há a necessidade de se acessar aleatoriamente cada quadro, como na edição de produções de vídeo
  - Redundância temporal
    - compactação adicional pode ser obtida explorando o fato de que dois quadros consecutivos são, com frequência, quase idêntico
    - MPEG faz compensação de movimento
      - Calcula o vetor de movimento dos macroblocos e a diferença macrobloco a macrobloco



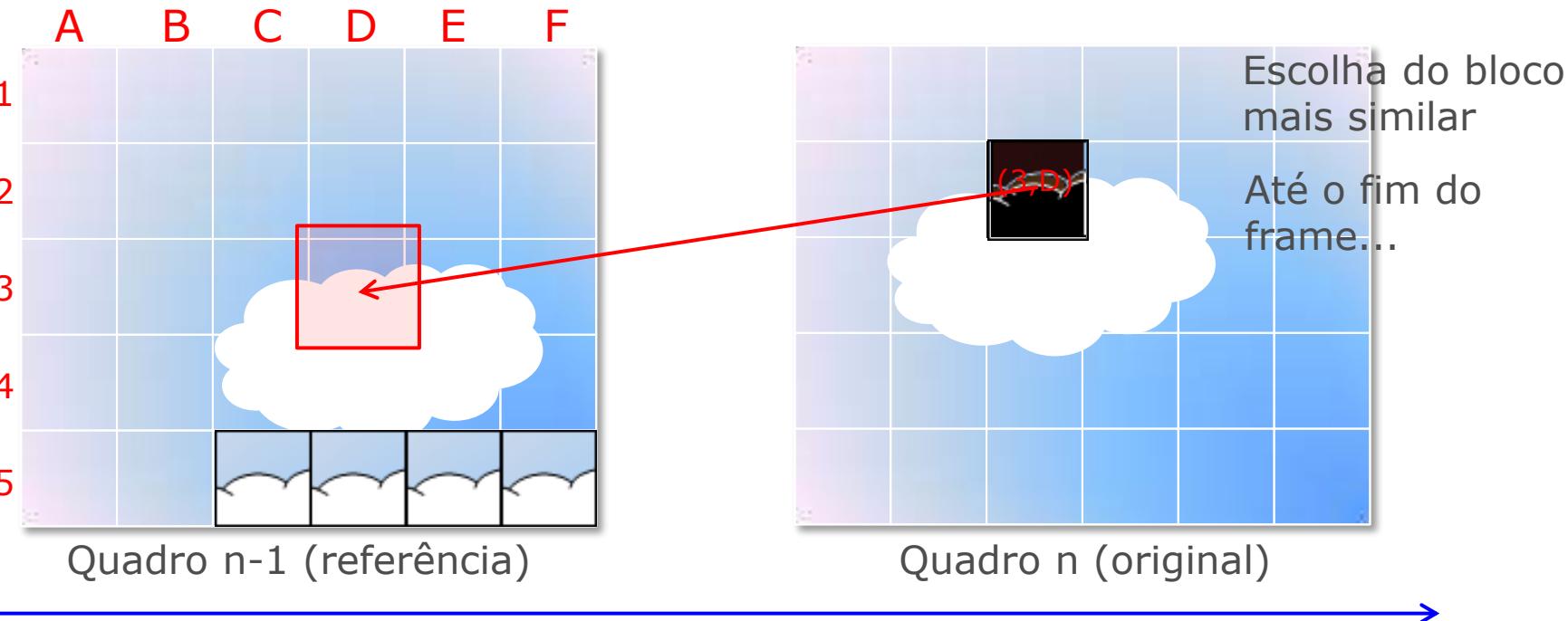
# Técnicas de Compressão de Vídeo e Imagem

- **Estimativa e Compensação de Movimento**
  - Imagem é dividida em blocos de tamanho fixos
    - um casamento para cada bloco é procurado na imagem anterior
      - deslocamento entre estes dois blocos é chamado vetor de movimento
    - uma diferença de blocos é obtida calculando diferenças píxel a píxel
  - Vetor de movimento e a diferença de bloco é codificado e transmitido



# Técnicas de Compressão de Vídeo e Imagem

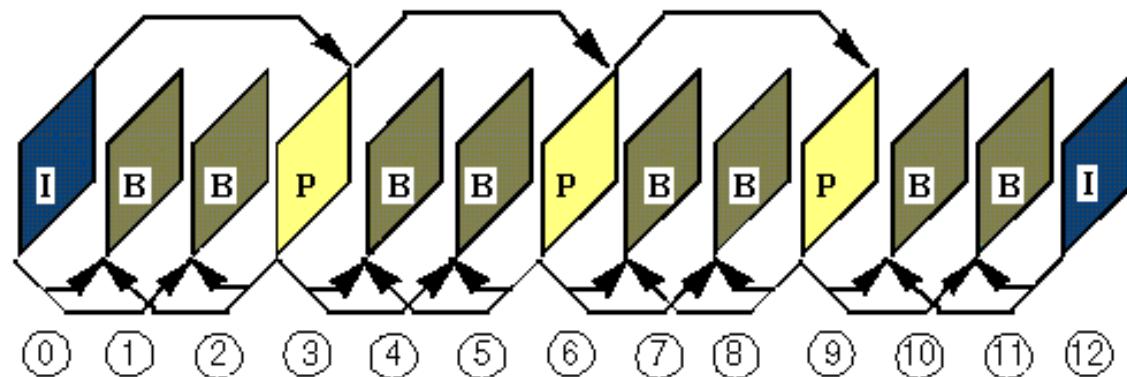
- Exemplo simples: Compara a similaridade entre blocos



- Mantém a diferença entre os blocos (resíduo);
- Cria o vetor de movimento, referenciando o bloco do quadro anterior;

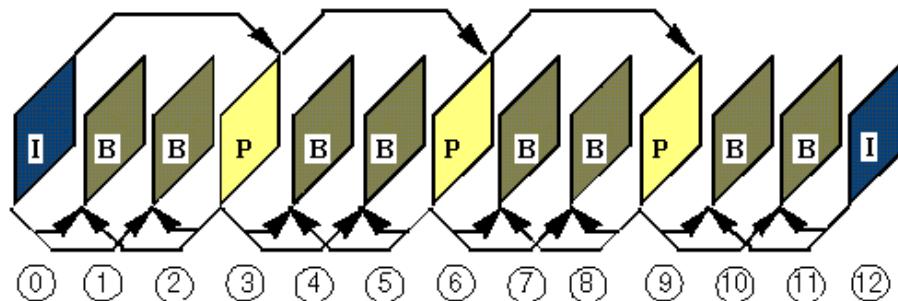
# Compactação MPEG-1 Video

- Grupo de Imagem consiste de quatro tipos de quadros:
  - **Quadros I** (Intracoded)
    - imagens estáticas, independentes e codificadas com o JPEG.
  - **Quadro P** (Predictive)
    - diferença bloco a bloco com o quadro I ou P anterior
  - **Quadro B (Birectional)**
    - diferença com o último quadro e com o quadro seguinte
  - **Quadro D (DC-coded)**
    - Médias de bloco usadas para o avanço rápido (fast forward).



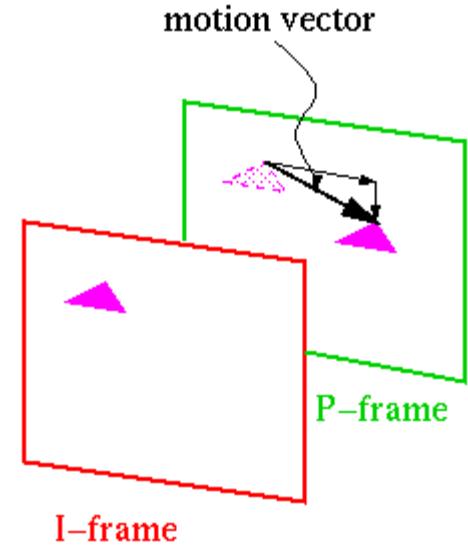
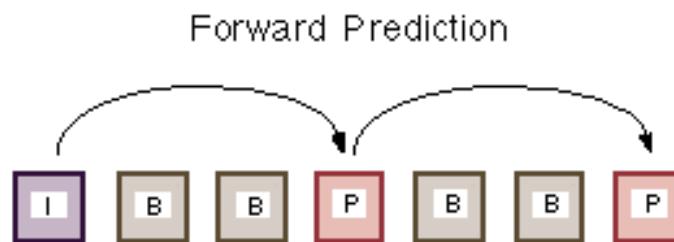
# Compactação MPEG-1 Video

- **Quadro I**
  - Imagens estáticas, independentes e codificadas com o JPEG
  - É necessário que quadros I apareçam periodicamente no fluxo de saída
    - no caso de transmissão multicast
      - receptores podem entrar no grupo em tempos distintos, requerendo um quadro I para começar a decodificação MPEG-1
    - se um quadro for recebido com erro
      - decodificação não será mais possível
- ☞ Quadros I são inseridos na saída uma ou duas vezes por segundo

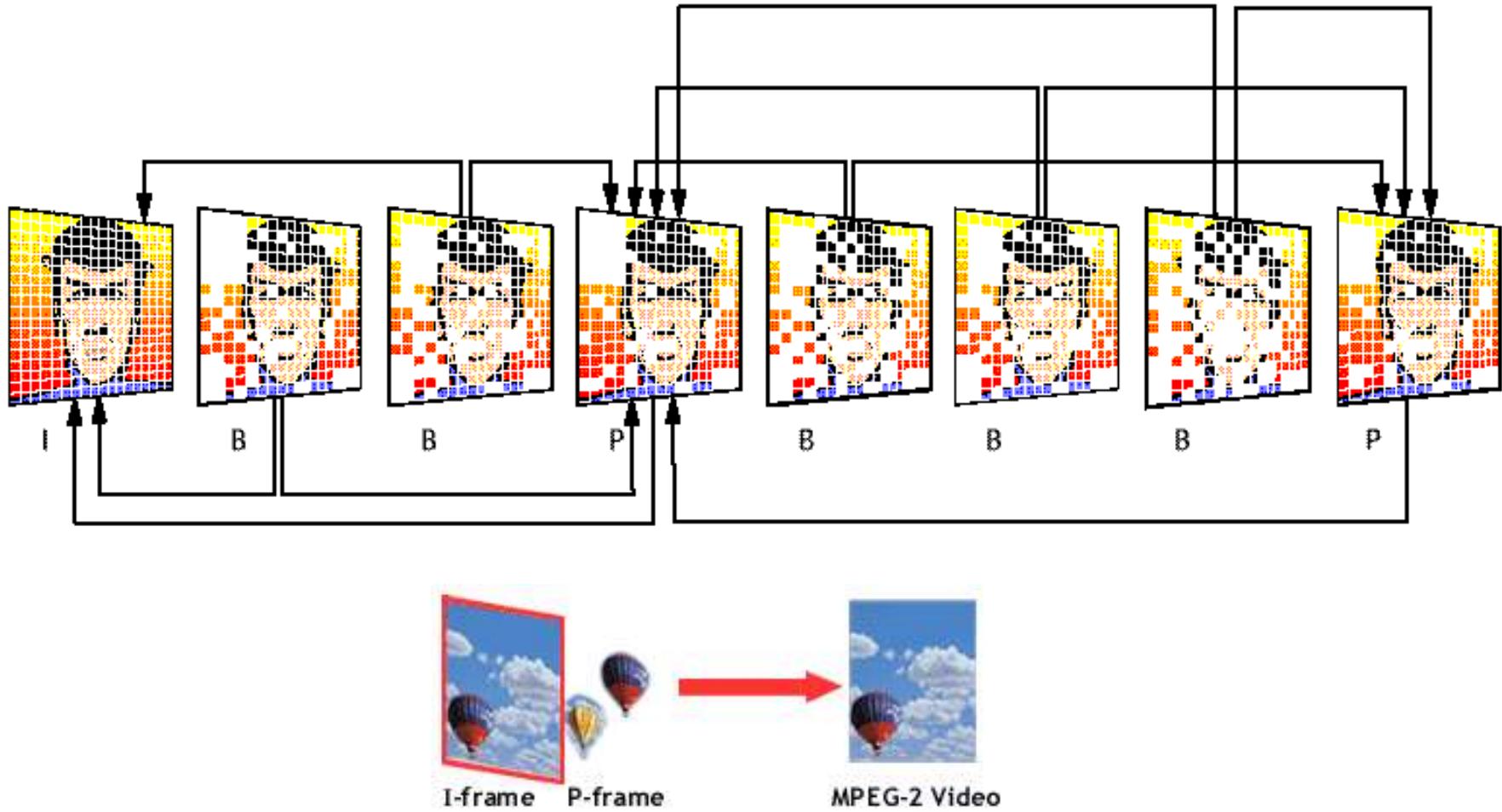


# Compactação MPEG-1 Video

- **Quadro P**
  - Codificam as diferenças entre os quadros
    - 50% do tamanho de um quadro I
  - Se baseiam na idéia dos macroblocos, de 16x16 pixels
    - macrobloco é codificado da seguinte forma:
      - tentando-se localizá-lo, ou algo parecido com ele, no quadro anterior
  - Decodificar quadros P requer que o decodificador armazene o quadro I ou P anterior em um buffer
    - a partir do qual o novo quadro é construído baseado em macroblocos completamente codificados e macroblocos contendo diferenças com o quadro anterior

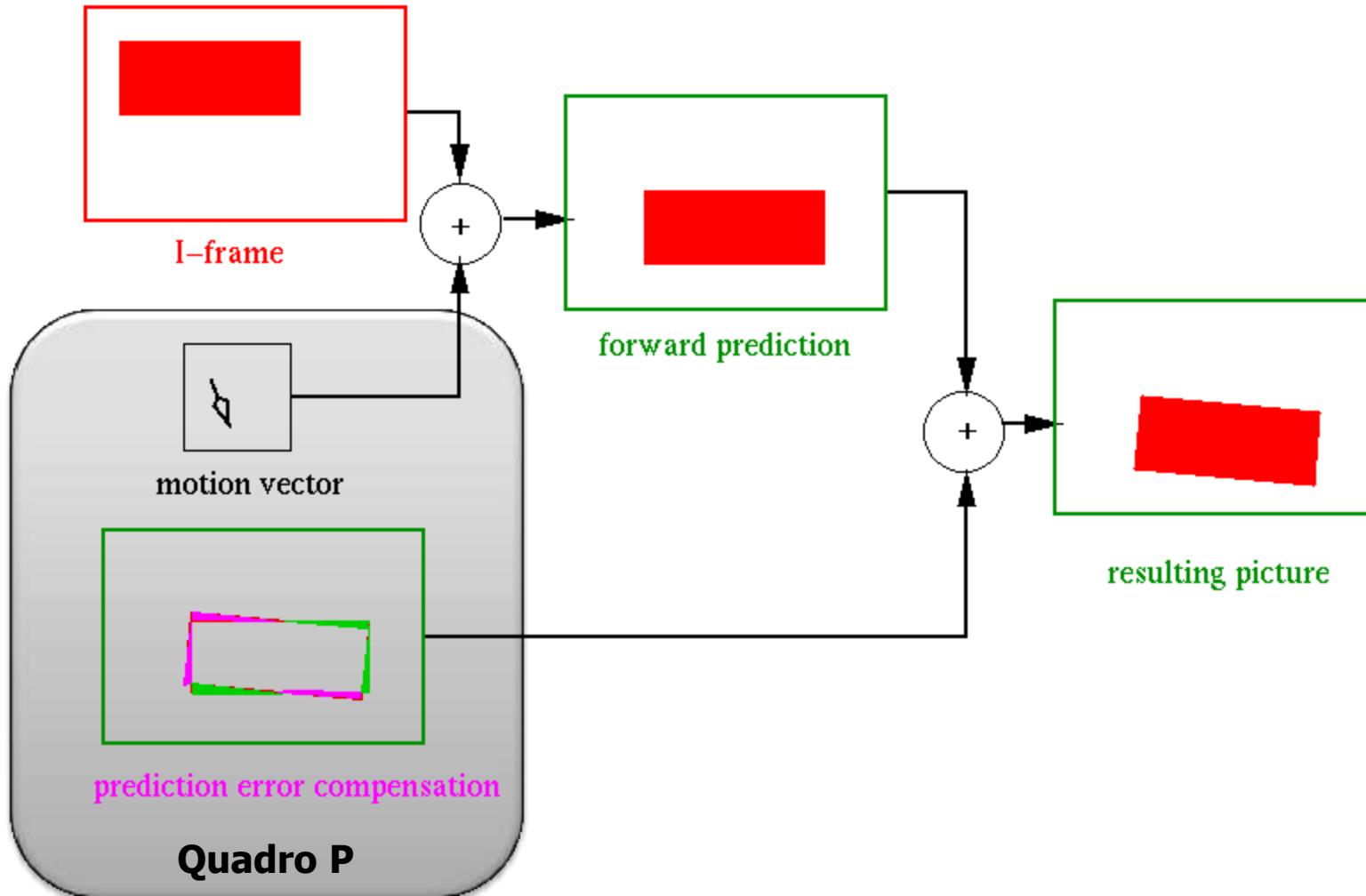


# Compactação MPEG-1 Video



# Compactação MPEG-1 Video

- Quadro P
  - Estimativa e compensação do movimento



# Compactação MPEG-1 Video

- Quadro P
  - Estimativa e compensação do movimento



I-frame  
1



B-frame  
2



B-frame  
3



B-frame  
4



B-frame  
5



B-frame  
6

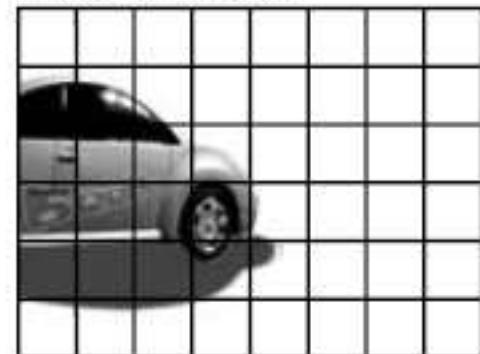


P-frame  
7

I-frame



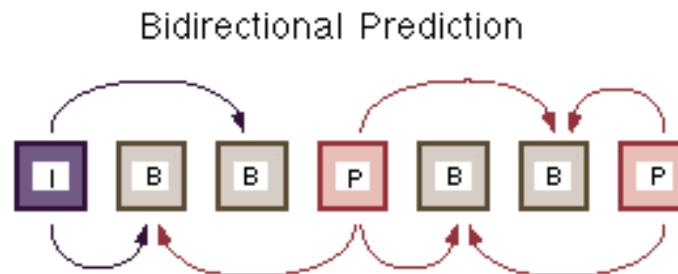
P-frame differences



# Compactação MPEG-1 Video

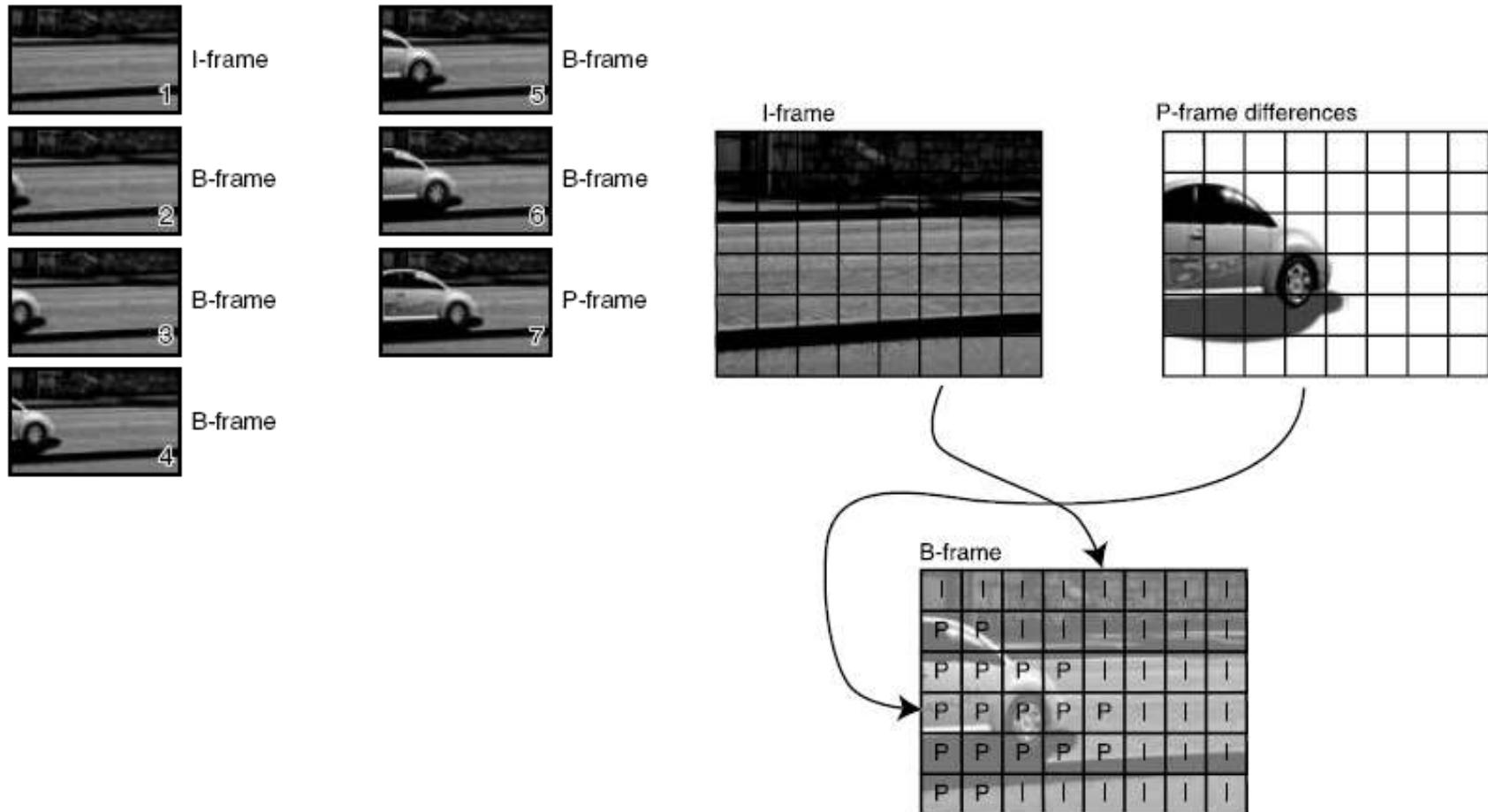
- **Quadro B**

- Codificam as diferenças com o último quadro I ou P e com o quadro seguinte
  - 15% do tamanho de um quadro I
  - permitem que o macrobloco de referência esteja tanto no quadro anterior quanto no quadro seguinte
  - acarreta uma melhoria na compensação do movimento
- Para decodificar quadros B
  - decodificador precisa manter três quadros decodificados na memória ao mesmo tempo: o quadro anterior, o atual e o próximo



# Compactação MPEG-1 Video

- Quadro P
  - Estimativa e compensação do movimento



# Compactação MPEG-1 Video

- **Quadro D**
  - Só são usado para possibilitar a apresentação de uma imagem de baixa resolução quando um avanço rápido ou um retrocesso
- **Um fluxo MPEG-1**
  - Uma seqüência de quadros codificados teria a seguinte forma:
    - IBBPBBPBBPBBIBBPBBPBBPB.....
- **Codificação MPEG-2**
  - É fundamentalmente semelhante à codificação MPEG-1
    - com quadros I, P e B
    - quadros D não são aceitos
  - Transformação discreta de co-seno é de 10x10 em vez de 8x8
    - para proporcionar mais 50 por cento de coeficientes
      - melhor qualidade

# MPEG-1 Sistemas

- Define uma estrutura para:
  - Combinar fluxos elementares, incluindo áudio, vídeo e outros fluxos de dados
    - chamado de Fluxo MPEG
    - até 32 fluxos de áudio MPEG e 16 fluxos de vídeo MPEG podem ser multiplexados juntamente com 2 fluxos de dados de diferentes tipos
- Especifica o modo de representar as informações temporais necessárias para reprodução de seqüências sincronizadas em tempo real
  - sincronização de fluxos elementares
  - gerenciamento de buffer nos decodificadores
  - acesso aleatório
  - identificação do tempo absoluto do programa codificado

# MPEG-4

- Padrão MPEG-4
  - começou a ser concebido em julho de 1993, tendo sido aprovado como padrão internacional em 2000.
- Uso
  - Vários vídeos transmitidos pela Internet fazem uso deste padrão, assim como telefones celulares que utilizam imagens
  - Também é utilizado em diversos padrões de transmissão de TV digital, especialmente os de alta definição (HDTV)

# MPEG-4

- MPEG-4 absorve muita das características do MPEG-1 e MPEG-2 e outros padrões relacionados, adicionando novas características
  - suporte VRML (Virtual Reality Metadata Language) para apresentações 3D,
  - arquivos compostos orientados a objetos (incluindo objetos de áudio, vídeo e VRML),
  - suporte para Gerenciamento de Direitos Autorais externamente especificados
  - e vários tipos de interatividade.

# MPEG-4

- **Partes do MPEG-4**
  - 23 Partes abordando diferentes aspectos do padrão.
  - Exemplos
    - Parte 1 descreve a sincronização de áudio e vídeo;
    - Parte 2 é uma tecnologia de compressão de vídeo;
    - Parte 3 o processo de compressão do áudio;
    - Parte 10 do padrão foi incluída quando uma versão mais otimizada da parte 2 (compressão de vídeo) foi desenvolvida

# MPEG-4

- **MPEG-4 Parte 2**

- Tecnologia de compressão de vídeo desenvolvida pela MPEG.
- Um padrão de compressão DCT similar aos padrões MPEG-1 e MPEG-2
- 21 Perfis (Profiles)
  - Agrupam características em perfis (profiles) e níveis.
    - Para permitir seu uso em várias aplicações, variando de câmeras de segurança de baixa qualidade, baixa resolução a HDTVs e DVDs,
  - Perfil Simple Profile (SP): usado em situações onde a baixa taxa de bits e baixa resolução são mandatórios devido a largura de banda da rede, tamanho do dispositivo, etc
    - telefones celulares, sistemas de segurança, etc.
  - Perfil Advanced Simple Profile (ASP): muito similar ao H.263, incluindo suporte para a quantificação do estilo MPEG, suporte a vídeo entrelaçado, suporte a imagens do tipo B, compensação de movimento QPel (Quarter Pixel) e Global (GMC).

# MPEG-4

- **MPEG-4 Parte 10**

- Também conhecidos como H.264 ou AVC (Advanced Video Coding)
- Um padrão de codec de vídeo digital que tem a característica de alta taxa de compressão.
- O padrão define 7 perfis, voltadas a classes de aplicações específicas. Por exemplo:
  - Baseline Profile (BP) é voltado para aplicações de custo mais baixo com limitado recursos computacionais, usado em aplicações de videoconferência e móveis.
  - Extended Profile (XP) é voltado para streaming de vídeo, com alta taxa de compressão e robustez para perda de dados.
  - High Profile (HiP) é o principal perfil para aplicações de armazenamento em disco e broadcast, particularmente para aplicações de HDTV e adotado pelos discos HD-DVD e Blu-ray.

# CCITT H.261

- **Origem**

- Necessidade de fornecer serviços de vídeo onipresentes na Rede Digital de Serviços Integrados (ISDN)
- Um dos padrões da família H.320 para videofonia e teleconferência na taxa de 64 Kbps a 2 Mbps

- **Características**

- Para aplicações de videofonia e teleconferência
- Algoritmo de compressão de vídeo opera em tempo-real com atraso mínimo
  - Algoritmo combina codificação intraquadro e interquadro para fornecer um rápido processamento para compressão/descompressão tempo-real de vídeo
- Fornece uma resolução cerca de oito vezes mais baixa que a qualidade TV PAL/SECAM
- É para aplicações usualmente sem movimentos intensos
  - algoritmo usa uma limitada estratégia de busca e estimativa de movimento para obter taxas de compressão mais altas
- H.261 pode obter taxas de compressão de 100:1 a mais de 2000:1

# CCITT H.261

- Padrão de compressão de vídeo para transmissão em taxas de  $p^*64\text{Kbps}$ 
  - Cobre as capacidades do canal ISDN
    - $p^*64\text{ Kbps}$  ( $p=1$  a 30)
  - $p = 1$  ou  $2$  é apropriado para comunicação visual face-a-face e baixo movimento (videofonia)
  - $p > 5$  melhor qualidade (videoconferência)
  - Máxima taxa de bits disponível é  $1,92\text{ Mbps}$  ( $p=30$ )
    - suficiente para obter imagens de qualidade VHS

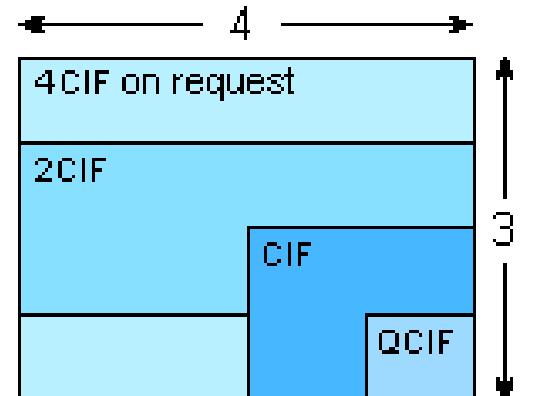
# CCITT H.261

- **Formatos de imagens**

- H.261 opera com dois formatos de imagem
  - CIF (Common Intermediate Format) - 320x288
    - permite usar um formato único dentro e entre regiões usando padrões de TV de 625 e 525 linhas
  - QCIF (quarter-CIF) - 160x144
    - mais útil em taxas de bit menores ( $p < 6$ ).

- **Algoritmo de compressão**

- Algoritmo de codificação é híbrido
  - Eliminação da redundância espacial
    - transform coding (DCT): similar ao JPEG
  - Predição inter-quadro
    - remove redundância temporal
    - compensação de movimento
  - codificação por entropia (codificação de Huffman)



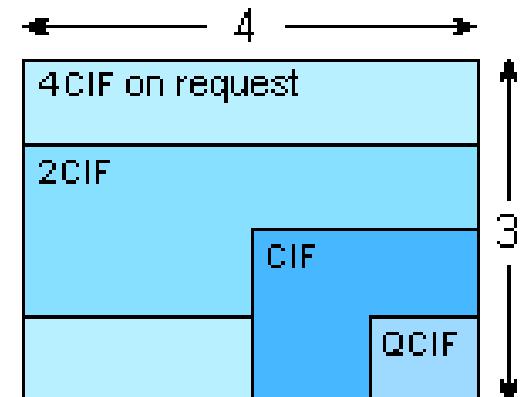
CIF = Common Intermediate Format

# CCITT H.263

- **H.263**
  - Padrão de vídeo a baixa taxa de bits para aplicações de teleconferência que opera a taxas abaixo de 64 Kbps
- **Algoritmo de codificação**
  - É uma extensão do H.261 e descreve um método de codificação DPCM/DCT
  - Uma idéia interessante do H.263 é o quadro PB
    - consiste de duas imagens codificadas em uma unidade
    - nome PB é derivado da terminologia MPEG dos quadros P e B
      - quadro PB consiste de um quadro P que é produzido a partir do último quadro P decodificado e um quadro B que é produzido a partir do último quadro P decodificado e do quadro P sendo decodificado

# CCITT H.263

- **H.263 suporta cinco resoluções**
  - QCIF e CIF que é suportado pelo H.261
  - SQCIF - aproximadamente a metade da resolução do QCIF
  - 4CIF e 16CIF - 4 e 16 vezes a resolução do CIF
    - codec poderia competir com outras codificações de mais altas taxas de bits como os padrões MPEG
- **H.263 x H.261**
  - Testes atuais mostram que o H.263 tem um desempenho 1 a 2,5 melhor que o H.261
    - taxa de bits H.261 é aproximadamente 2,4 vezes a gerada pelo H.263 (dada uma qualidade de imagem)



CIF = Common Intermediate Format