

RESUMO DOS TÓPICOS TEÓRICOS

PRIORITÁRIOS

1. Compensação de Movimento em Vídeo

- **O que é:** Técnica que explora redundância temporal entre frames consecutivos
 - **Como funciona:**
 - Divide o frame atual em blocos (ex: 16×16 pixels - macroblocos)
 - Procura cada bloco no frame de referência (anterior ou futuro)
 - Encontra a melhor correspondência através de busca numa janela
 - Guarda apenas o **vetor de movimento** (deslocamento) e o **erro de predição** (diferença entre bloco original e predito)
 - Muito mais eficiente que guardar o frame inteiro
 - **Vantagem:** Reduz drasticamente a quantidade de dados a transmitir
-

2. Quadros I: MPEG-1 vs H.261

- **MPEG-1:** Tem **frames tipo I completos** (intra-frames)
 - Frame inteiro codificado independentemente (como JPEG)
 - Permite random access, recuperação de erros
 - Usado em aplicações de armazenamento (CD-ROMs, vídeo)
 - **H.261:** Tem apenas **macroblocos tipo I**, mas **NÃO** frames I completos
 - Cada macrobloco pode ser I ou P, mas nunca o frame inteiro
 - Porquê? H.261 é para **videoconferência em tempo real**
 - Precisa de latência baixa e stream contínuo
 - Não precisa de random access (não vai rebobinar uma chamada)
 - Frame I completo causaria picos de bitrate indesejáveis
-

3. YUV/YCbCr vs RGB

Razões para usar YUV:

1. **Separação luminância/crominância:**
 - Y = luminância (brilho)
 - U/V (Cb/Cr) = crominância (cor)
 2. **Exploração da visão humana:**
 - Olho humano é **MUITO** mais sensível a variações de brilho que de cor
 - Permite **subsampling de crominância** (4:2:0, 4:2:2)
 - Exemplo: guardar Y em resolução completa, mas U/V em metade ou um quarto
 - Redução de ~50% de dados com perda de qualidade imperceptível
 3. **Compatibilidade:** Histórico com TV preto e branco (bastava ignorar U/V)
 4. **RGB:** Não permite essa separação - as 3 componentes são igualmente importantes
-

4. Débito Variável (VBR) vs Fixo (CBR)

Vantagens VBR:

- Melhor qualidade global: cenas complexas recebem mais bits, cenas simples menos
- Mais eficiente: taxa média menor para mesma qualidade
- Melhor para armazenamento (MP3 files, vídeos arquivados)

Desvantagens VBR:

- Impossível garantir taxa instantânea (problema para streaming em tempo real)
- Precisa de buffers maiores
- Difícil prever tamanho final do ficheiro
- Problemas em canais com largura de banda fixa
- Não adequado para transmissão ao vivo

CBR: Contrário - previsível mas menos eficiente

5. JPEG Sequencial

Funcionamento:

1. Divide imagem em blocos 8×8 pixels
2. Aplica DCT (Discrete Cosine Transform) em cada bloco
3. Quantização dos coeficientes DCT (perde informação aqui - compressão lossy)
4. Organiza coeficientes em ordem zig-zag (DC primeiro, depois AC por frequências)
5. Codifica DC por diferença (DPCM)
6. Codifica AC por run-length + Huffman
7. **Sequencial:** Codifica bloco por bloco, da esquerda → direita, cima → baixo
8. Imagem aparece gradualmente de cima para baixo durante carregamento

vs Progressivo: Progressivo envia primeiro versão de baixa qualidade da imagem inteira, depois refina

● IMPORTANTES

6. Impossibilidade de Comprimir Tudo

Prova por contagem (Pigeonhole Principle):

- Suponha ficheiros de n bits: existem 2^n ficheiros possíveis
- Se compressor reduz TODOS os ficheiros: ficheiros comprimidos têm no máximo $n-1$ bits
- Com $n-1$ bits só há $2^{(n-1)}$ possibilidades
- $2^{(n-1)} < 2^n \rightarrow$ **não há representações suficientes!**
- Pelo menos metade dos ficheiros tem que expandir ou ficar igual

Conclusão: Compressão universal é matematicamente impossível. Compressores só funcionam para dados com padrões/redundância.

7. Mascaramento em Áudio

Mascaramento Simultâneo (Frequency Masking):

- Som forte numa frequência "mascara" sons fracos em frequências próximas
- Exemplo: 1kHz a 80dB mascara 1.1kHz a 40dB - o segundo é inaudível
- **Uso:** Não precisa codificar/transmitir componentes mascaradas \rightarrow economia de bits

Mascaramento Temporal:

- Som forte mascara sons fracos imediatamente antes ($\sim 5\text{ms}$) e depois ($\sim 100\text{ms}$)
- **Pré-mascaramento:** curto, difícil de explorar
- **Pós-mascaramento:** mais longo, mais explorado
- **Uso:** Permite quantização mais agressiva em momentos de transientes

Como codificadores usam:

- Modelo psico-acústico calcula limiar de mascaramento
 - Ruído de quantização mantido abaixo desse limiar
 - Aloca mais bits onde mascaramento é menor
-

8. Redundância Estatística vs Perceptual

Redundância Estatística:

- Dados têm padrões, repetições, correlações
- **Exemplo:** "aaaaabbbbb" \rightarrow "6a5b"
- **Métodos:** Huffman, LZ77/78, aritmética, DPCM
- **Compressão:** **LOSSLESS** (sem perdas)
- Aplica-se quando: dados têm distribuição não-uniforme

Redundância Perceptual:

- Informação que humanos não percebem ou não valorizam
- **Exemplo:** Frequências de áudio acima de 20kHz, detalhes visuais em movimento rápido
- **Métodos:** Quantização, subsampling de crominância, remoção de componentes mascaradas
- **Compressão:** **LOSSY** (com perdas)
- Aplica-se quando: perda de qualidade aceitável

● ÚTEIS

9. Pré-ecos em Áudio

O que são:

- Artefactos audíveis ANTES de um transiente forte (ataque súbito)
- Ruído que "antecipa" o som que ainda não deveria existir

Causa:

- Transforms (MDCT) usam janelas temporais longas
- Ruído de quantização de um ataque forte "vaza" para trás no tempo
- Em silêncio antes do ataque, esse ruído fica audível

Prevenção:

- **Window switching:** Usar janelas curtas quando detecta transientes
 - MP3/AAC alternam entre janelas longas (melhor freq. resolution) e curtas (melhor time resolution)
 - Trade-off: janelas curtas têm pior resolução em frequência
-

10. Modelos Psico-acústicos

Em codificadores COM perdas:

- **Uso:** Determinam QUANTO ruído de quantização é aceitável em cada banda de frequência
- Calculam limiar de mascaramento baseado no sinal de áudio
- Controlam alocação de bits: mais bits onde ouvido é sensível, menos onde ruído é mascarado
- Exemplos: MP3, AAC, Vorbis

Em codificadores SEM perdas:

- **NÃO são usados** - não há liberdade para introduzir ruído
 - Compressão lossless preserva TUDO
 - Apenas exploram redundância estatística
 - Exemplos: FLAC, ALAC
-

11. Frames Intra (tipo I)

Vantagens:

- **Random access:** Pode começar a decodificar de qualquer frame I
- **Recuperação de erros:** Se frame P/B corrompe, próximo I "limpa" o erro
- **Edição:** Pontos de corte naturais
- **Robustez:** Não propaga erros

Desvantagens:

- **Muito maiores:** $\sim 10\times$ maior que frames P
- **Aumentam bitrate médio:** Quanto mais frames I, maior o ficheiro
- **Picos de bitrate:** Causam variações súbitas na taxa
- **Menor eficiência:** Não exploram redundância temporal


Solução típica: GOP (Group of Pictures) - ex: 1 frame I a cada 15 frames

12. Modelos Preditivos ($x_n = x_{n-1} + r_n$)

Condições para vantagem:

1. **Alta correlação temporal:** Amostras consecutivas devem ser semelhantes
 - $x_n \approx x_{n-1}$ na maior parte do tempo
2. **Resíduo r_n pequeno:** Diferença entre valor real e predito deve ser pequena
 - Se r_n é pequeno \rightarrow precisa menos bits para codificar
3. **Distribuição de r_n concentrada:**
 - Muitos valores perto de zero
 - Permite codificação eficiente (Golomb, Huffman)
4. **Sinais suaves/lentos:**
 - Áudio de fala, temperatura, sinais biomédicos
 - NÃO funciona bem com: ruído branco, sinais aleatórios

Exemplo: DPCM (Differential Pulse Code Modulation)

 **DICA DE ESTUDO:** Os 5 primeiros tópicos (compensação movimento, quadros I, YUV, débito variável, JPEG) são praticamente garantidos. Domina esses primeiro!