

INF01046 - Fundamentos de processamento de imagens

Aula 26 - Compressão de imagens e vídeo

Horacio E. Fortunato

Instituto de Informática
Universidade Federal de Rio Grande do Sul
Porto Alegre - RS

hefortunato@inf.ufrgs.br

Link do curso: <http://www.inf.ufrgs.br/~hefortunato/cursos/INF01046>

2º semestre de 2009



Horacio E. Fortunato (UFRGS)

Processamento Digital de Imagens - Nesta disciplina



Horacio E. Fortunato (UFRGS)

Codificação aritmética

A codificação aritmética pode ser descrita como segue:

1. Cria-se um intervalo corrente iniciado com $[0,1)$
2. Para cada elemento da mensagem:
 1. Particiona-se o intervalo corrente em subintervalos, um para cada letra do alfabeto. O tamanho do subintervalo associado a uma dada letra é proporcional à probabilidade de que esta letra seja o próximo elemento da mensagem, de acordo com o modelo assumido.
 2. O subintervalo correspondente à letra que é o próximo elemento é selecionado como novo intervalo corrente.
3. Codifica-se a mensagem com o menor número de bits necessário para distinguir o intervalo corrente final de todos os outros possíveis intervalos correntes finais.



Horacio E. Fortunato (UFRGS)

Codificação aritmética

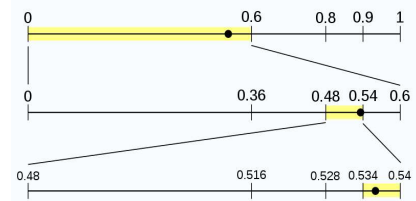


Diagrama representando a subdivisão dos intervalos na codificação aritmética, usando-se quatro símbolos de probabilidades 0.6, 0.2, 0.1, e 0.1. Os círculos negros representam a mensagem codificada, que pode ser decodificada para o primeiro símbolo, depois o terceiro, em seguida o quarto. A primeira linha mostra o intervalo completo $[0,1)$ enquanto as linhas subsequentes mostram as subdivisões proporcionais do intervalo anterior

Origem: Wikipédia, a enciclopédia livre.



Horacio E. Fortunato (UFRGS)

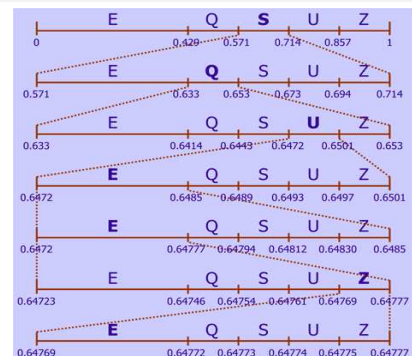
Codificação aritmética

- Os números gerados são todos menores que 1
- Quanto mais pequeno o intervalo final mais dígitos (bits) são necessários para identifica-lo
- Uma sequencia de símbolos mais prováveis resulta em um intervalo maior que uma sequencia de símbolos menos prováveis.



Horacio E. Fortunato (UFRGS)

Codificação aritmética



Horacio E. Fortunato (UFRGS)

Codificação LZ77 (Lempel-Ziv)

- O algoritmo LZ77 se baseia na utilização das partes que já foram lidas de um arquivo como um dicionário, substituindo as próximas ocorrências das mesmas seqüências de caracteres pela posição (absoluta ou relativa) da sua última ocorrência.
- Para limitar o espaço de busca e de endereçamento necessário, as ocorrências anteriores são limitadas por uma "janela deslizante" (do inglês sliding window) que tem tamanho fixo e "desliza" sobre o arquivo, delimitando o início e fim da área onde serão buscadas as ocorrências anteriores
- O tamanho desta janela é um dos fatores primordiais para se ajustar a performance desse algoritmo.

Origem: Wikipédia, a enciclopédia livre



Horacio E. Fortunato (UFRGS)

Codificação LZ77 (Lempel-Ziv) Exemplo

Abaixo ilustramos o algoritmo LZ77 com um exemplo da compressão da cadeia A_ASA_DA_CASA, usando janela de tamanho 8 e buffer de look-ahead de tamanho 4.

Janela	Buffer	Restante do arquivo	Tupla emitida
	A_AS	A_DA_CASA	(0,0,A)
A	_ASA	_DA_CASA	(0,0,_)
A_	ASA_	DA_CASA	(1,1,S)
A_AS	A_DA	_CASA	(3,2,D)
A_ASA_D	A_CA	SA	(2,2,C)
ASA_DA_C	ASA		(7,3,EOF)

Temos então 6 tuplas, cada tupla ocupa 15 bits (4 para a posição dentro da janela, 3 para o tamanho e 8 para o caractere no final), perfazendo 90 bits.

Comparado com a cadeia original de 104 bits (13 bytes) a compressão não é muito boa, mas para arquivos maiores o tamanho da janela pode ser ajustado, assim como o tamanho do buffer, conseguindo taxas de compressão bem melhores.

Origem: Wikipédia, a enciclopédia livre



Horacio E. Fortunato (UFRGS)

Codificação LZ77 (Lempel-Ziv) Aplicações

Alguns programas e formatos de compressão de dados largamente utilizados usam este algoritmo, pois ao contrário do LZ78 e do LZW, seus parentes mais próximos, ele não estava coberto por patentes.

A variante mais comum do LZ77 é conhecida como **DEFLATE** e combina o uso de **LZ77** com o uso de **Código Huffman**.

- Entre os programas e formatos que usam LZ77 e DEFLATE temos:
 - O programa PKZIP e o formato de arquivos **ZIP** (além de todos os outros programas baseados nesse formato).
 - O programa gzip.
 - O formato de imagens **PNG**.

Origem: Wikipédia, a enciclopédia livre



Horacio E. Fortunato (UFRGS)

Codificação LZW (Lempel-Ziv-Welch)

- O método de codificação LZW baseia-se na construção progressiva de uma tabela (dicionário) contendo as strings de símbolos de vários comprimentos, encontradas na sequência de símbolos da fonte.

- Explora a redundância inter-pixel

- Na codificação, a tabela (dicionário) de strings é inicializada como todos os símbolos individuais

- A sequência de entrada é então examinada, símbolo a símbolo, e a mais longa string para a qual exista uma entrada correspondente na tabela é extraída, a palavra-código para esta string é transmitida

- Utilizado em:
 - Unix 'compress'
 - Formato de imagens **GIF**



Horacio E. Fortunato (UFRGS)

Codificação LZW (Lempel-Ziv-Welch)

- Vamos supor que queremos transmitir uma string de dados, formada pelos símbolos a, b, c:

abacabcbacbacbcbccca

- A codificação LZW trabalha com uma tabela de símbolos (dicionário).
- O dicionário inicia com os símbolos a, b, c
- Vai adicionando ao dicionário as seqüências que aparecem na string, na ordem em que aparecem
- Vai transmitindo os códigos das seqüências conhecidas (índice no dicionário)
- O receptor constrói a mesma tabela na medida que recebe os dados



Horacio E. Fortunato (UFRGS)

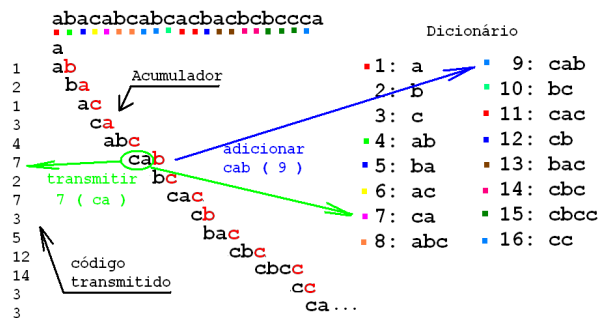
Codificação LZW (Lempel-Ziv-Welch)

```
1. ACUM = get input character
2. WHILE there are still input characters DO
3.   CHARACTER = get input character
4.   IF ACUM + CHARACTER is in the dictionary then
5.     ACUM = ACUM + character
6.   ELSE
7.     output the code for ACUM
8.     add ACUM + CHARACTER to the dictionary
9.     ACUM = CHARACTER
10.  END of IF
11. END of WHILE
12. output the code for ACUM
```



Horacio E. Fortunato (UFRGS)

Codificação LZW (Lempel-Ziv-Welch)



Formato BMP Microsoft Windows Device Independent Bitmap

- Device Independent Bitmap (DIB) ou Windows Bitmap (BMP) é usado pelo subsistema de gráficos (GDI) do Microsoft Windows
- Formato mais popular não utiliza compressão
- Estrutura de arquivo muito simples
 - Arquivo composto por um 'file header', 'bitmap info header', 'paleta de cores' (se existir) e os pixels da imagem a continuação

Formato TIFF Tagged Image File Format

- Criado pela Aldus para uso no processo de impressão PostScript
- Agora é controlado pela Adobe.
- Formato complexo, flexível e extensível
- Podem armazenar-se varias imagens e outro tipo de informação num único arquivo
- Suporta varios esquemas de Compressão:
 - None
 - LZW
 - PackBits (run lenght)
 - Deflate (LZ77 + Huffman)
 - JPEG

Formato PNG Portable Network Graphics

- Surgiu em 1996 como substituto para o formato GIF, devido ao facto de este último incluir algoritmos patenteados.
- Recomendado pela W3C
- **Compressão sem perdas** (zlib - DEFLATE)
- Livre de patentes
- Suporta Canal alfa: Define o nível de opacidade de cada pixel
- Suporta imagens em tons de cinza, RGB, RGBA e indexadas (com paleta de cores)
- Tem uma maior gama de profundidade de cores (até 16 bits por canal)

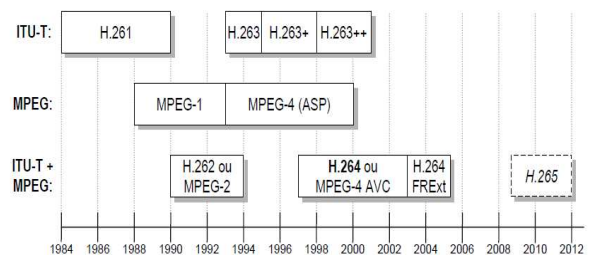
Codificação de vídeo Padronização

Essencial para permitir que equipamentos de vídeo digital de diferentes fabricantes se comuniquem.

Principais órgãos de padronização:

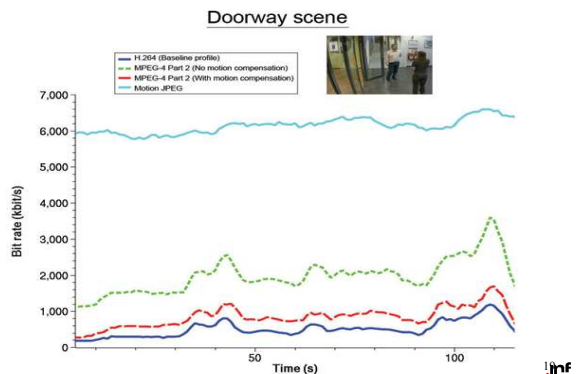
- ITU-T Video Coding Experts Group (VCEG)
- ISO/IEC Moving Picture Experts Group (MPEG).

Codificação de vídeo Padronização



Cronologia do desenvolvimento dos diversos padrões de compressão de vídeo, apresentados isoladamente pelos órgãos ITU-T e MPEG, ou em um esforço conjunto de ambos

Codificação de vídeo Comparação



Horacio E. Fortunato (UFRGS)

Codificação de vídeo H.261

- Estabelecido em 1991, faz parte de um conjunto de padrões do ITU-T para serviços audiovisuais em telecomunicações (videofone e vídeo conferencia)
- É a especificação de um conjunto de protocolos, que todo fluxo de dados codificado tem que seguir e também um conjunto de operações que todo decodificador compatível tem que ser capaz de executar
- Optimizado para taxas baixas de transmissão de dados
- Utiliza codificação inter-frame: o quadro anterior é utilizado para prever o quadro atual e somente a diferença é transmitida
- Destinado a comunicação em tempo real, somente utiliza o ultimo quadro para prever o atual
- Alcança taxas de compressão de 100:1 a 2000:1

Horacio E. Fortunato (UFRGS)

Codificação de vídeo H.263

- Estabelecido pela ITU-T para serviços audiovisuais em telecomunicações (videofone e vídeo conferencia)
- Posterior à H.261
- Optimizado para taxas baixas de transmissão de dados muito baixas (64 kbps)
- Quatro métodos de codificação avançados: vetor de movimento irrestrito, predição avançada, quadros P-B e codificação aritmética baseada em sintaxe

2inf

Horacio E. Fortunato (UFRGS)

Codificação de vídeo H.264

- H.264 é um padrão para compressão de vídeo, baseado no MPEG-4 Part 10 ou AVC (Advanced Video Coding).
- O padrão foi desenvolvido pela ITU-T Video Coding Experts Group (VCEG) em conjunto com a ISO/IEC MPEG que formaram uma parceria conhecida por Joint Video Team (JVT).
- A versão final, formalmente chamada por ISO/IEC 14496-10, foi lançada em Maio de 2003
- O H.264 possui uma série de vantagens em relação a seus antecessores, (principalmente em comparação ao MPEG-1, MPEG-2 e H.263) Com respeito às etapas de predição, transformação e codificação.
- Muitas dessas novas características apareceram primeiro nos anexos do H.263 (resultando no H.263+ e H.263++), bem como no padrão MPEG-4 ASP.

2inf

Horacio E. Fortunato (UFRGS)

Codificação de vídeo M – JPEG e MPEG

- Apesar de JPEG ter sido concebido para compressão de imagens estáticas, muitos fabricantes aplicaram o JPEG para sequências de imagens de vídeo (M-JPEG)
- Falta de padrão ocasiona diferentes implementações
- O Padrão reconhecido é o MPEG (Moving pictures Experts Group) comprime quadros isolados em JPEG e utiliza redundância inter frame
- MPEG tem taxa de compressão cerca de 3 vezes superior e M-JPEG
- A compressão inter-frame dificulta a edição de vídeo o que favorece a popularidade de M-JPEG nos equipamentos de edição de vídeo

2inf

Horacio E. Fortunato (UFRGS)

Codificação de vídeo MPEG

- O comitê MPEG (Motion Pictures Experts Group) foi estabelecido pelo isso em 1988 para desenvolver padrões de codificação de vídeo e áudio associado para [armazenamento](#) em mídia digital.
- O objetivo inicial foi estabelecer padrões para codificação de vídeo (e áudio) a três taxas de dados: 1,5 , 10 e 40 Mbps (MPEG 1, 2 e 3)
- MPEG-1 : Orientado a vídeo em CDs (qualidade similar a VHS) 1.5 Mbps
- MPEG-2 : Maior qualidade e resolução (televisão broadcast) 4-100 Mbps
- MPEG-3 : (HDTV) Abandonado em favor do MPEG-2
- MPEG-4: Padronizar codificação de conteúdo multimídia
- MPEG-7: Descrição de conteúdo multimídia

2inf

Horacio E. Fortunato (UFRGS)

Codificação de vídeo MPEG

- O MPEG especifica a sintaxe que um fluxo de dados compatível deve seguir.
- Também especifica um processo de decodificação típico
- Não restringe inovações na codificação e decodificação do bitstream MPEG
- Aplicações:
 - Simétricas: codificação e decodificação na hora (vídeo conferência)
 - Assimétricas: codifica uma vez decodifica depois.

Na elaboração do MPEG houve preocupação em se prever suporte para:

- Acesso aleatório
- Busca rápida em avanço (forward search) e retrocesso (reverse search)
- Playback reverso
- Sincronismo entre áudio e vídeo
- Tratamento de erros
- Editabilidade
- Flexibilidade de formatos



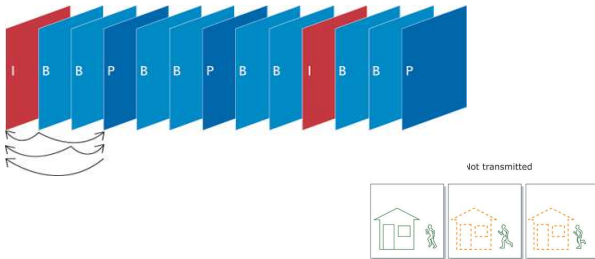
Codificação de vídeo Estrutura dos quadros MPEG

- Três tipos de quadros:
 - I: Intraframe
 - P: Preditivo
 - B: Bidirecional
- I: Codificado similar ao JPEG (uma imagem isolada), utilizados como pontos de acesso aleatório
- P: Usa predição baseado em um quadro prévio de tipo I ou P (não necessita ser o quadro anterior imediato)
- B: Codificado usando-se dois quadros como referência um anterior (passado) e outro posterior (futuro)



Codificação de vídeo Estrutura dos quadros MPEG

- Grupos de quadros (GOP): Sequência de quadros que inicia com um quadro I e termina no quadro anterior do I seguinte



Codificação de vídeo

http://dvd-hq.info/data_compression_3.php



Processamento Digital de Imagens - Tarefas

Tarefas Novas:

- Leia o Capítulo 8 (aula 25/26) do livro Gonzalez, R. & Woods 2da Ed. (em Inglês)
- Faça os exercícios do Capítulo 8 (aula 25/26) do livro Gonzalez, R. & Woods 2da Ed. (em Inglês)

Nota Importante: No livro Gonzalez, R. & Woods em português os capítulos possuem número diferente

Livro Gonzalez, R. & Woods 2ª Ed. (em Inglês):
Gonzalez, R. & Woods, R. Digital Image Processing 2ª Ed. Prentice Hall, 2002.
Link do curso: <http://www.inf.ufrgs.br/~hefortunato/cursos/INF01046>

