



C209 – Computação Gráfica e Multimídia  
EC215 – Multimídia

## Compressão de Imagens – Parte 1: Tipos

Marcelo Vinícius Cysneiros Aragão  
marcelovca90@inatel.br

# Visão geral

- Imagens de alta resolução e cor verdadeira podem ocupar até vários megabytes de espaço.
- Na maioria dos casos, pode-se conseguir grande redução do tamanho do arquivo por meio das técnicas de compressão de imagens estáticas:
  - Compressão sem perdas
  - Compressão com perdas
  - Outras técnicas de compressão

# Compressão sem perdas

- As técnicas de compressão sem perdas mantêm toda a informação contida na imagem original.
- É feita por meio da técnica de **codificação**; varia-se a quantidade de bits usada para representar determinados padrões de informação conforme a frequência desse padrão na imagem a comprimir.
- Técnicas genéricas de compressão sem perdas são utilizadas em arquivos e também em dispositivos de comunicação de dados.
- A compressão de imagens combina técnicas genéricas e técnicas que se aproveitam e características específicas das imagens.

# Compressão sem perdas

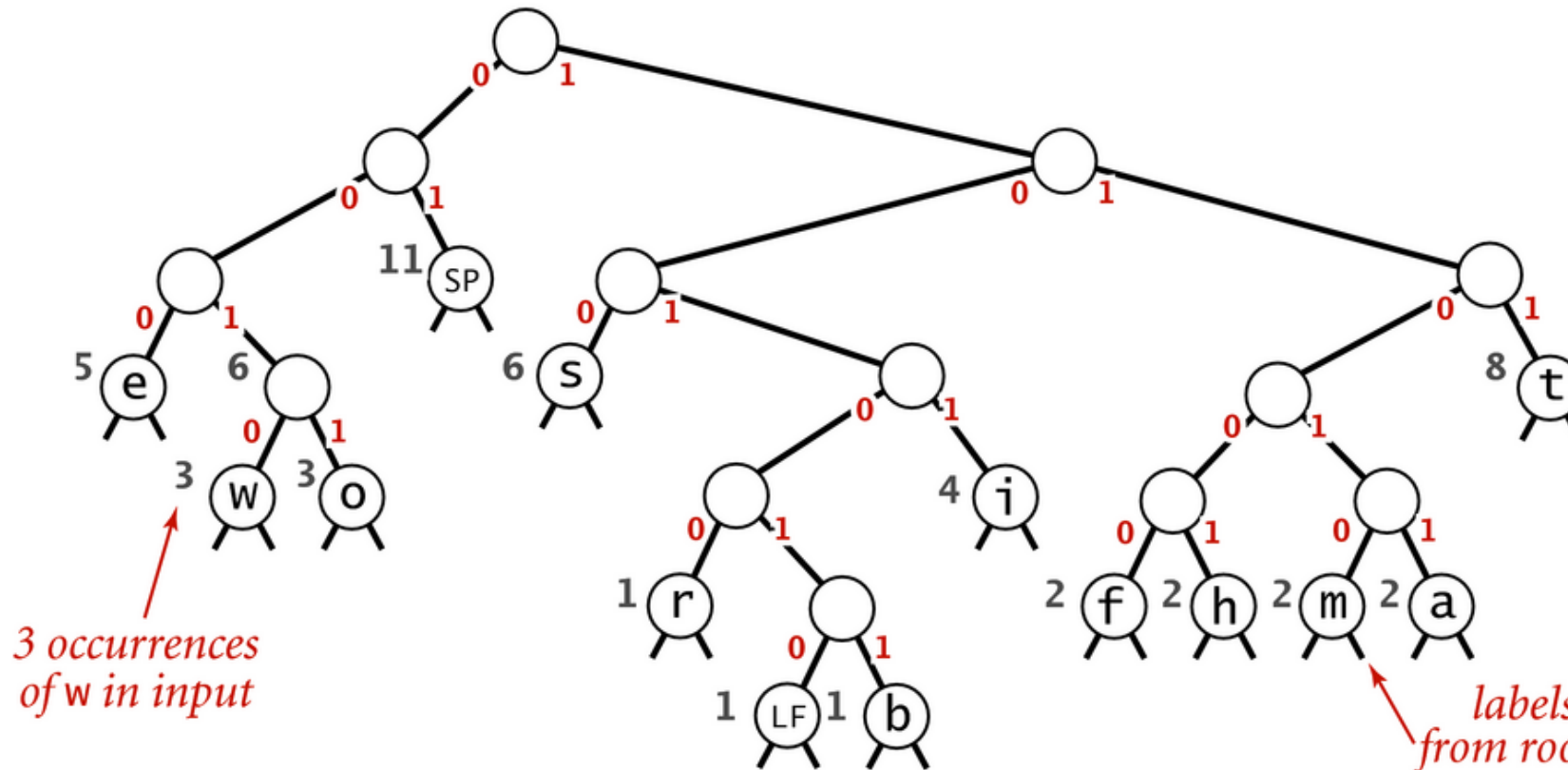
- Exemplos técnicas genéricas de compressão sem perdas:
  - ZIP: um dos mais comuns na atualidade (7-Zip)
  - ARC: predecessor do formato zip (lZArc)
  - TAR e Gzip (GZ): usados geralmente em sistemas Unix
  - ISO: usado para imagens de CDs e DVDs
  - ARJ: similar ao zip
  - CAB: usado em instaladores de programas
  - RAR: um dos concorrentes do zip

# Compressão sem perdas: Huffman

- Baseiam-se em **códigos de Huffman**, que usam sequências de bits mais curtas para os símbolos mais frequentes (codificação entrópica).
- Por exemplo, para um texto em português, usamos sequências curtas para representar o “a” ou o “e” e sequências longas para representar o “z” ou o “x”.
- Com isso, obtemos uma **codificação mais eficiente** do que se usarmos sempre 8 bits por símbolo, como acontece com o código ASCII.

# Compressão sem perdas: Huffman

## trie representation



## codeword table

<i>key</i>	<i>value</i>
LF	101010
SP	01
a	11011
b	101011
e	000
f	11000
h	11001
i	1011
m	11010
o	0011
r	10100
s	100
t	111
w	0010

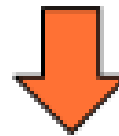
Huffman code for the character stream “it was the best of times it was the worst of times LF”

# Compressão sem perdas: RLE

- Uma técnica eficiente de compressão de dados com muita redundância é o algoritmo RLE (*run-length encoding*).
- No caso de imagens, são consideradas “tiras” de cor constante.
- Para cada tira, só é necessário guardar sua cor e comprimento.
- Essa técnica é adequada para imagens artificiais, produzidas por editores bidimensionais, em que se tende a colorir grandes áreas com a mesma tinta (exemplo: formatos PCX e TGA).

# Compressão sem perdas: RLE (imagem)

## Lossless pixel compression





# Compressão sem perdas: RLE (texto)

[https://www.youtube.com/watch?v=ypdNscvym\\_E](https://www.youtube.com/watch?v=ypdNscvym_E)



# Compressão sem perdas: LZW

- São mais eficientes as técnicas que tomam partido da repetição de padrões de bits, e não simplesmente de valores de cor.
- Uma técnica é adaptativa quando os códigos usados se adaptam ao material que está sendo comprimido, durante a própria compressão.
- Uma técnica adaptativa particularmente importante é a de **Lempel-Ziv-Welsh (LZW)**, que toma partido da **coerência entre linhas** para identificar padrões repetidos em imagens.
- Essa técnica é a base do formato **GIF**.

# Compressão sem perdas: LZW

Input	Current String	Seen this Before?	Encoded Output	New Dictionary Entry/Index
<i>b</i>	<i>b</i>	yes	nothing	none
<i>ba</i>	<i>ba</i>	no	<i>1</i>	ba / 5
<i>ban</i>	<i>an</i>	no	<i>1,0</i>	an / 6
<i>banana</i>	<i>na</i>	no	<i>1,0,3</i>	na / 7
<i>banana</i>	<i>an</i>	yes	no change	none
<i>banana</i>	<i>ana</i>	no	<i>1,0,3,6</i>	ana / 8
<i>banana_</i>	<i>a_</i>	no	<i>1,0,3,6,0</i>	a_ / 9
<i>banana_b</i>	<i>_b</i>	no	<i>1,0,3,6,0,4</i>	_b / 10
<i>banana_ba</i>	<i>ba</i>	yes	no change	none
<i>banana_ban</i>	<i>ban</i>	no	<i>1,0,3,6,0,4,5</i>	ban / 11
<i>banana_ban</i>	<i>nd</i>	no	<i>1,0,3,6,0,4,5,3</i>	nd / 12
<i>banana_ban</i>	<i>da</i>	no	<i>1,0,3,6,0,4,5,3,2</i>	da / 13
<i>banana_ban</i>	<i>an</i>	yes	no change	none
<i>banana_ban</i>	<i>ana</i>	yes	<i>1,0,3,6,0,4,5,3,2,8</i>	none

- Extras:

- <https://en.wikipedia.org/wiki/Lempel%E2%80%93Ziv%E2%80%93Welch>
- <https://www.dcode.fr/lzw-compression>
- <https://www2.cs.duke.edu/csed/curious/compression/lzw.html>

# Compressão sem perdas: LZW

- O GIF tem grande difusão na publicação de imagens na WWW.
- É **limitado a 256 cores**, mas foi estendido no formato GIF89a, que suporta recursos adicionais importantes:
  - Possibilidade de definir uma **cor transparente**;
  - **Entrelaçamento**, que permite o envio das tiras da imagem em ordem intercalada, possibilitando que o usuário perceba uma versão mais grosseira da imagem antes que esta seja completamente carregada;
  - **Animação**, por meio de arquivos com múltiplas imagens parciais.

# Compressão sem perdas: LZW

- Nos anos 1990, a Unisys, que havia adquirido patente do método LZW, começou a cobrar *royalties* dos produtores de software que processam o formato GIF.
- A reação negativa do público levou ao desenvolvimento do formato PNG.
- Atualmente, as patentes da Unisys expiraram, e o uso do GIF voltou a ser livre.

# Compressão sem perdas: LZW



## Rocketboom – The History of the GIF

<https://www.youtube.com/watch?v=y0MSN7MRUcU>



## Vox – The business of GIFs: Then and now

<https://www.youtube.com/watch?v=pzC40p-6Tc8>

# Compressão sem perdas: DEFLATE

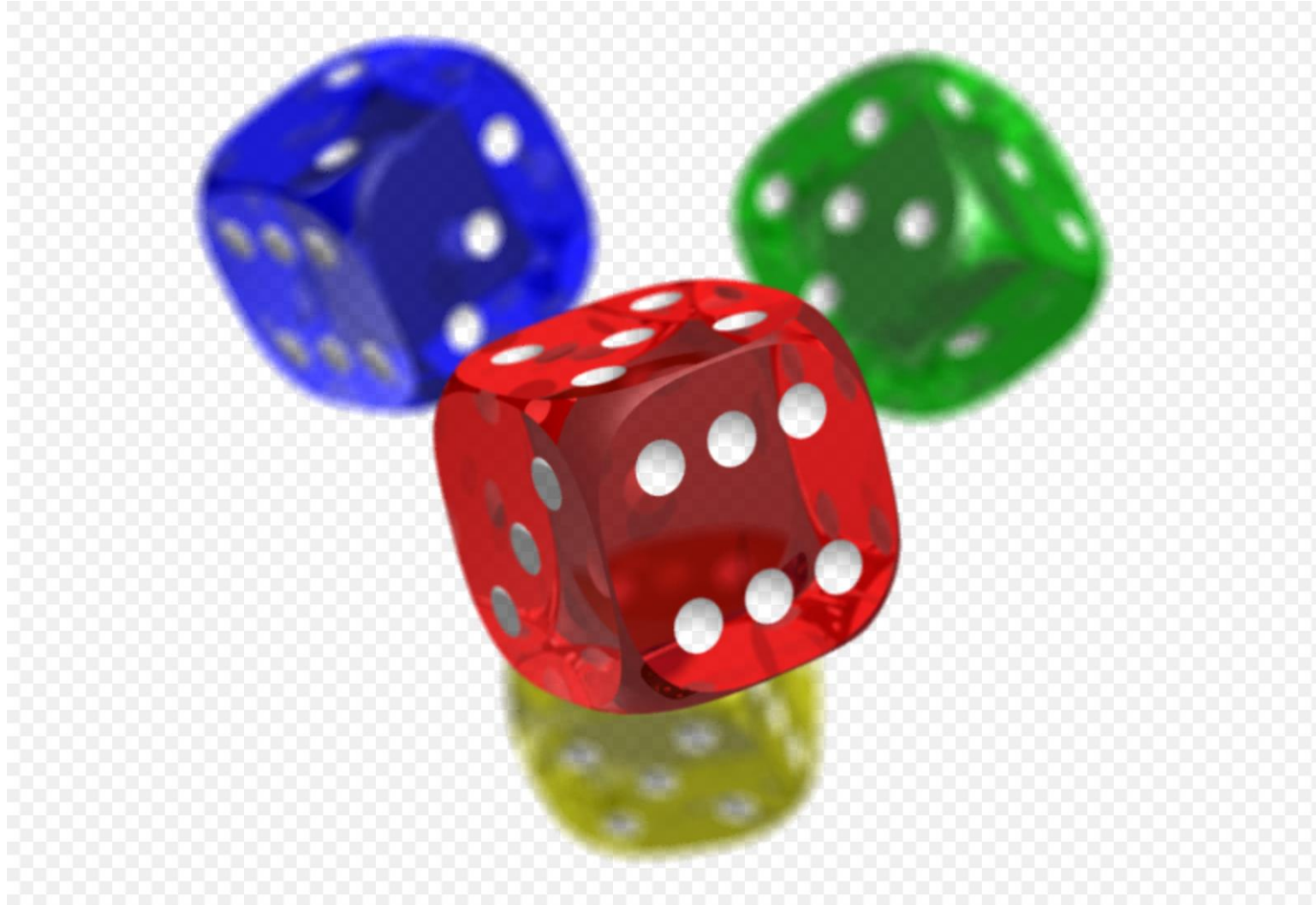
- O formato PNG utiliza o método de compressão **DEFLATE**, que combina a codificação de Huffman com o algoritmo LZ77, derivado (assim como o LZW) dos artigos originais de Lempel e Ziv, mas não coberto pelas patentes do LZW.
- O método DEFLATE também é usado nos formatos ZIP e GZIP.

# Compressão sem perdas: DEFLATE

- O formato PNG apresenta os seguintes recursos:
  - Suporte a **múltiplos sistemas de cor** (verdadeira ou de paleta);
  - Múltiplos níveis de **transparência** (canal alfa);
  - Correção do **gama** (suporte para ajuste da exibição da imagem às características do monitor);
  - **Entrelaçamento** mais avançado que o GIF.
- Por outro lado, o formato PNG **não suporta imagens animadas**.
- Para isto, foram desenvolvidos o MNG e APNG (concorrentes).



# Compressão sem perdas: DEFLATE



# Compressão com perdas

- São usadas em casos em que a perda de alguma informação é tolerável, por corresponder a detalhes que a visão humana não percebe, ou percebe apenas com dificuldade.
- Ao comprimir a imagem, pode-se perder alguma informação existente na imagem original.
- A taxa de perda é um parâmetro fixado durante a compressão. Quanto maior a perda admitida, maior a compressão que se consegue.

# Compressão com perdas

## *Example of Lossy Compression*



**Original Lena Image  
(12KB size)**



**Lena Image,  
Compressed (85%  
less information,  
1.8KB)**



**Lena Image, Highly  
Compressed (96%  
less information,  
0.56KB)**

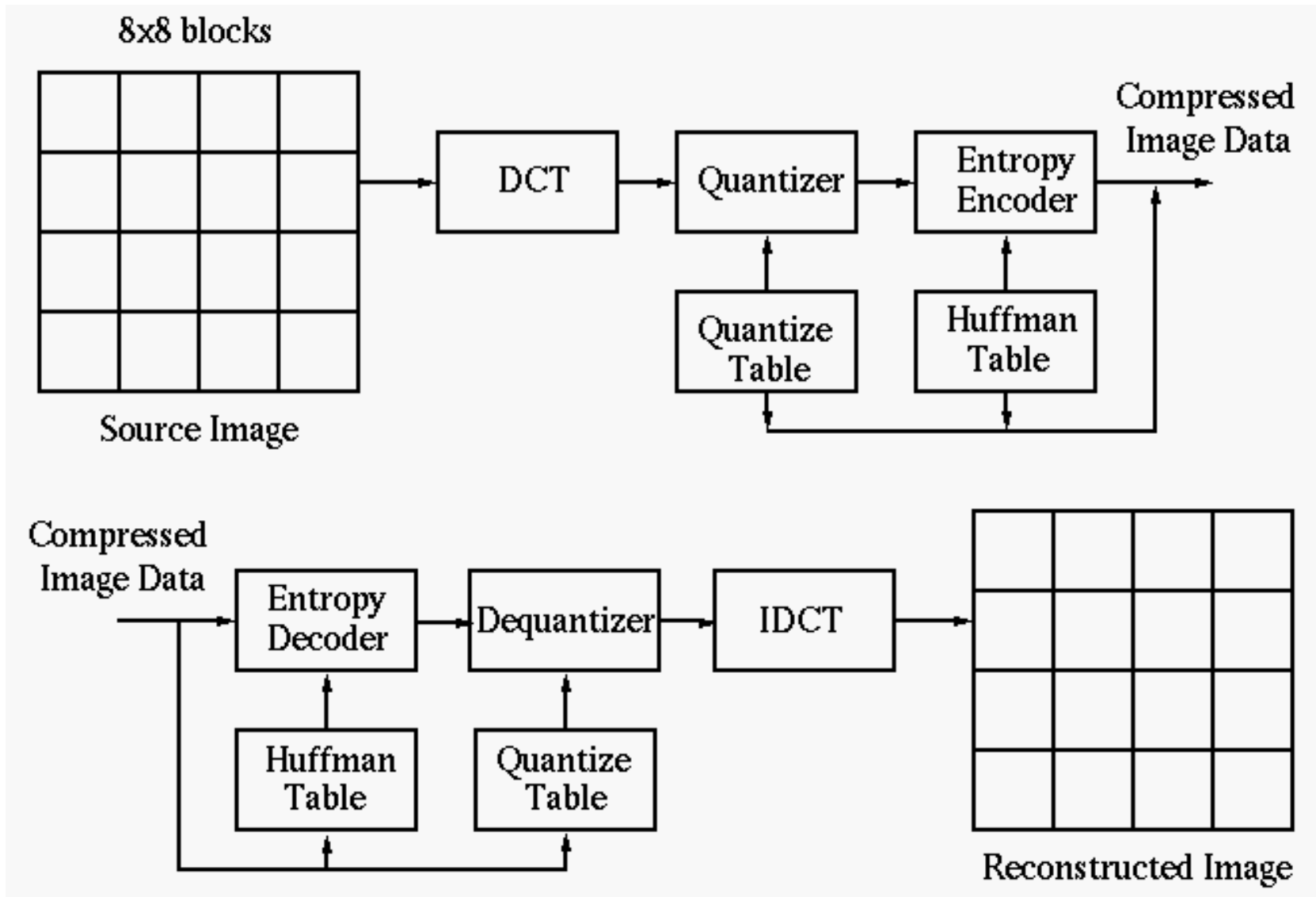
# Compressão com perdas

- Um algoritmo central para muitas técnicas de compressão com perdas é algum tipo de **transformação da imagem para uma forma de espectro**.
- Cada quadro a ser comprimido é dividido em blocos, cujos valores de pixels são submetidos a uma transformação matemática que traduz o bloco na forma de um espectro bidimensional, ou seja, uma **matriz de distribuição de energia**.
- Os coeficientes dessa matriz são truncados e em seguida codificados através de um algoritmo de compressão de dados.

# Compressão com perdas: JPEG

- A codificação **JPEG** (*Joint Photographic Experts Group*) é atualmente a **técnica mais importante de compressão**, principalmente de imagens com gradações suaves de intensidade geradas por captura de fotos ou síntese 3D.
- Etapas da compressão JPEG em forma bem simplificada:
  - Obtenção do espectro bidimensional da imagem (DCT)
  - Truncamento dos componentes de espectro
  - Codificação entrópica dos componentes

# Compressão com perdas: JPEG



# Compressão com perdas: JPEG

- O padrão JPEG é complexo, com suporte para diversos tipos de algoritmos de compressão.
- Existe uma versão sem perdas do JPG, usada em aplicações em que mesmo perdas mínimas não são toleráveis (ex.: imagens médicas e para edição e processamento).
- Também é possível gerar imagens JPEG progressivas, nas quais são transmitidos primeiro os pixels correspondentes a resoluções mais baixas e, em seguida, são transmitidos os pixels para dobrar, progressivamente, a resolução da imagem.

# Outras técnicas de compressão: Wavelets

- Um tipo de transformada espectral que permite altas taxas de compressão, principalmente em imagens com componentes de alta frequência, como em algumas imagens astronômicas.
- A variante **JPEG 2000** combina os conceitos do JPEG original com o uso das transformadas de *wavelets*.



# Outras técnicas de compressão: Wavelets

Original image



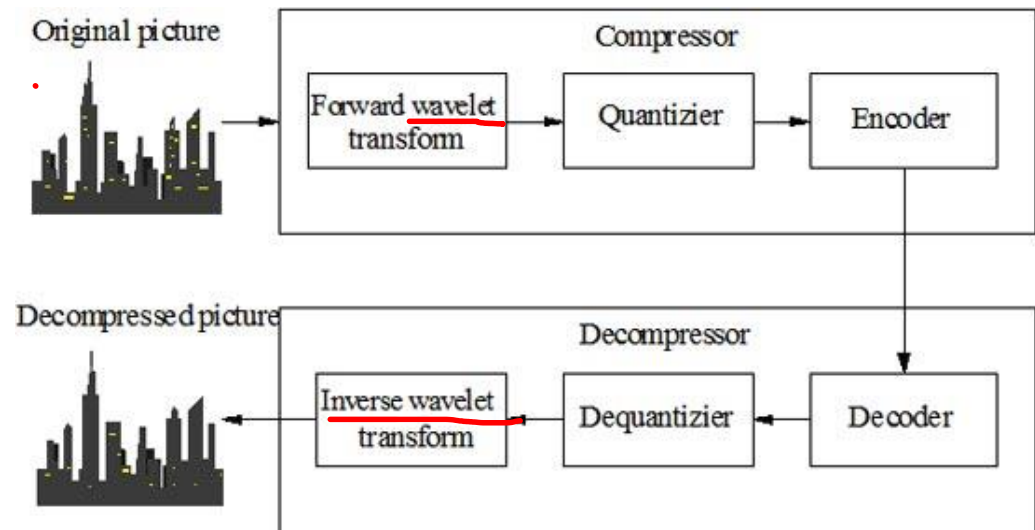
DCT-based  
image compression



Wavelet-based  
image compression



<http://disp.ee.ntu.edu.tw/meeting/%E4%BF%8A%E5%BE%B7/Wavelet/Wavelet.ppt>

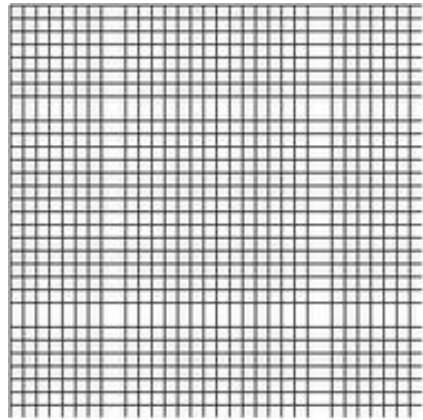


<https://www.pantechsolutions.net/image-processing-projects/matlab-code-for-image-compression-using-spiht-algorithm>

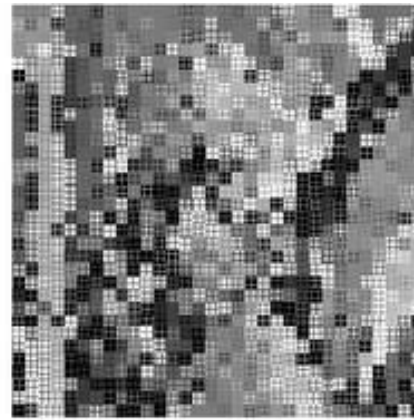
# Outras técnicas de compressão: Fractais

- Baseada no uso de fractais, um conceito matemático para a descrição de imagens complexas, tais como as encontradas na natureza.
- Basicamente, funciona com a inversão da elaboração de fractais; esta gera imagens a partir dos parâmetros de um **algoritmo de geração de fractais**, e a compressão faz o processo inverso, determinando os parâmetros que gerariam uma dada imagem.

# Outras técnicas de compressão: Fractais



1



2



3



6



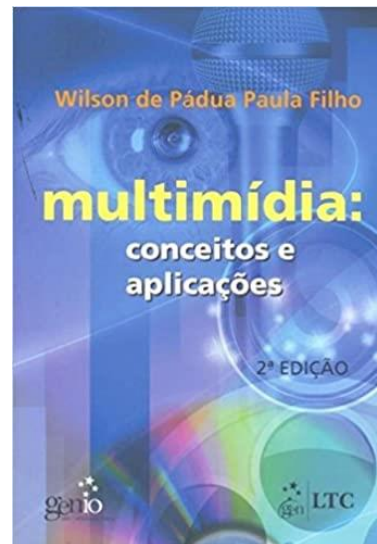
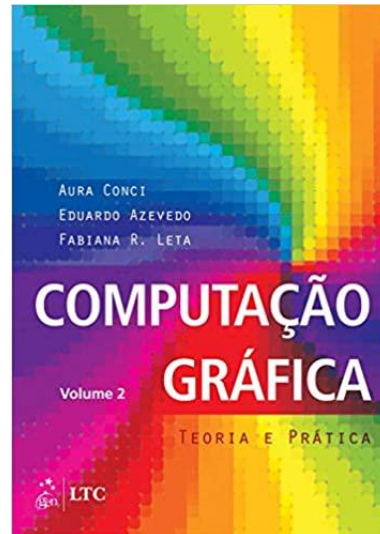
5



4

<http://developers-club.com/posts/126653/>

# Referências & Links Interessantes



- AZEVEDO, Eduardo; CONCI, Aura, Computação gráfica volume 1: geração de imagens. Rio de Janeiro, RJ. Editora Campus, 2003, 353 p. ISBN 85-352-1252-3.
- AZEVEDO, Eduardo; CONCI, Aura; LETA, Fabiana R. Computação gráfica volume 2: teoria e prática. Rio de Janeiro, RJ: Editora Elsevier, 2007, 384 p. ISBN 85-352-2329-0.
- PAULA FILHO, Wilson de Pádua, Multimídia: Conceitos e aplicações. Rio de Janeiro, RJ: LTC, 2000, 321 p. ISBN 978-85-216-1222-3.