

# CIÊNCIAS DA COMPUTAÇÃO

## Processamento de Imagens e Visão Computacional

Prof. César C. Xavier

## ROTEIRO

### Fundamentos de imagens digitais

- Fontes de dados
- Resolução
- Amostragem e Quantização
- Cor de Imagens
- Formato de Imagens
- Prática: Python e OpenCV

Prof. César C. Xavier

## Fontes de dados

- Em uma pesquisa científica corresponde a se estabelecer o(s) local(is) onde far-se-á a coleta dados.
- Atenção: é impossível obter todas as informações para o objeto de estudo dado a enorme quantidade de dados, limitação de tempo ou ainda acesso a todos os dados.
- Amostra: subconjunto do universo ou de uma população.

Prof. César C. Xavier

## Fontes de dados

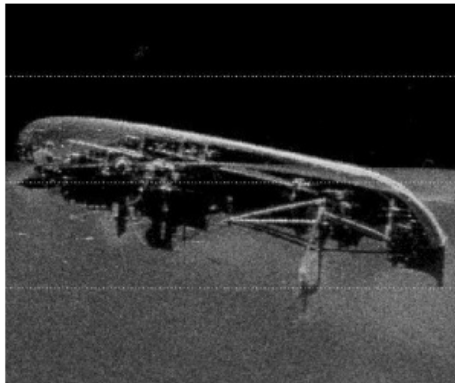
- Imagens: Ultrassom



Prof. César C. Xavier

## Fontes de dados

- Imagens: Sonar



Prof. César C. Xavier

## Fontes de dados

- Imagens: Sensor Infravermelho



Prof. César C. Xavier

## Resolução

### Resolução Espacial

- É uma medida do menor detalhe discernível em uma imagem;
- Expressa o número de pares de linha por unidade de distância e pontos (pixels) por unidade de distância;
- Por exemplo, se a largura de uma linha for 0,1 mm, temos 5 pares de linha por unidade de distância (mm);
- Nos Estados Unidos, essa medida é normalmente expressa como *dots per inch* (pontos por polegada ou dpi).

Prof. César C. Xavier

## Resolução

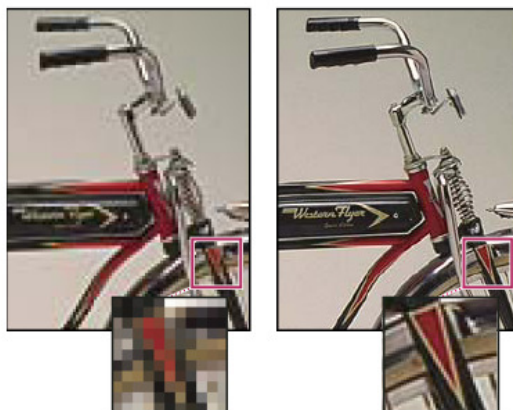
### Resolução Espacial



Prof. César C. Xavier

## Resolução

### Resolução Espacial



Mesma imagem a 72 ppi e 300 ppi; 200% de zoom de entrada

Prof. César C. Xavier

<https://helpx.adobe.com/br/photoshop/using/image-size-resolution.html>

## Resolução

### Resolução de Intensidade (de bit)

- refere-se se à menor variação discernível de nível de intensidade na imagem.
- o número de níveis de intensidade normalmente é igual a  $2^k$ , sendo k um número inteiro.
- é comum se referir ao número de bits utilizados para quantizar a intensidade como resolução de intensidade.
- Ex.: é comum dizer que uma imagem cuja intensidade é quantizada em 256 níveis tem 8 bits de resolução de intensidade.

Prof. César C. Xavier

## Resolução

### Resolução de Intensidade (de bit)



Prof. César C. Xavier

## Resolução

### Número Bits Armazenamento

**Tabela 2.1** Número de bits de armazenamento para vários valores de  $N$  e  $k$ .  $L$  é o número de níveis de intensidade.

$N/k$	1 ( $L = 2$ )	2 ( $L = 4$ )	3 ( $L = 8$ )	4 ( $L = 16$ )	5 ( $L = 32$ )	6 ( $L = 64$ )	7 ( $L = 128$ )	8 ( $L = 256$ )
32	1.024	2.048	3.072	4.096	5.120	6.144	7.168	8.192
64	4.096	8.192	12.288	16.384	20.480	24.576	28.672	32.768
128	16.384	32.768	49.152	65.536	81.920	98.304	114.688	131.072
256	65.536	131.072	196.608	262.144	327.680	393.216	458.752	524.288
512	262.144	524.288	786.432	1.048.576	1.310.720	1.572.864	1.835.008	2.097.152
1.024	1.048.576	2.097.152	3.145.728	4.194.304	5.242.880	6.291.456	7.340.032	8.388.608
2.048	4.194.304	8.388.608	12.582.912	16.777.216	20.971.520	25.165.824	29.369.128	33.554.432
4.096	16.777.216	33.554.432	50.331.648	67.108.864	83.886.080	100.663.296	117.440.512	134.217.728
8.192	67.108.864	134.217.728	201.326.592	268.435.456	335.544.320	402.653.184	469.762.048	536.870.912

$N$ : número de pixels

Prof. César C. Xavier

## Resolução

### Resolução Temporal

- Relaciona-se ao sistema de captura de imagens de forma contínua;
- Unidade: quadros por segundo (qps) ou *frame per second* (fps).

Prof. César C. Xavier

## Amostragem e Quantização

### Objetivo Aquisição de imagens

- Gerar imagens a partir de dados captados pelos sensores.
- Saída: tensão contínua.
- Conversão: dados contínuos → formato digital.

### Conceitos Básicos

- Amostragem: digitalização dos valores de coordenada.
- Quantização: digitalização dos valores de amplitude.

### Produção de Imagem Digital: varredura, linha a linha, a partir da parte superior.

- Precisão Quantização: depende do conteúdo do ruído.

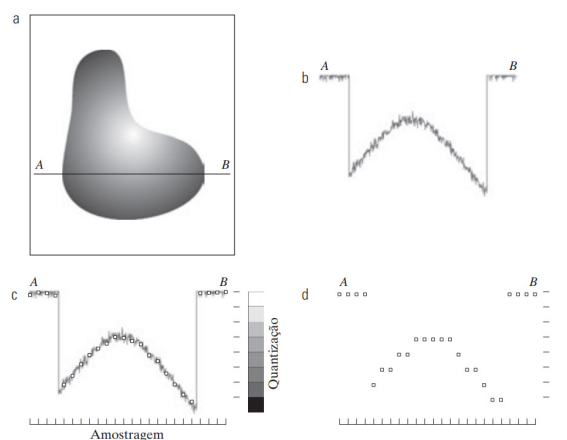


Figura 2.16 Produzindo uma imagem digital. (a) Imagem contínua. (b) Linha de varredura de A a B na imagem contínua utilizada para ilustrar os conceitos de amostragem e quantização. (c) Amostragem e quantização. (d) Linha de varredura digital.

Prof. César C. Xavier



## Amostragem e Quantização

- Sensores de varredura de linha
  - ♦ Há movimento mecânico
- Sensores de Matriz
  - ♦ Não há movimento.
  - ♦ Número de sensores determina os limites de amostragem nas duas direções (x, y).
- Qualidade Imagem
  - ♦ Número de amostras.
  - ♦ Níveis de discretização de intensidade na amostragem e quantização.

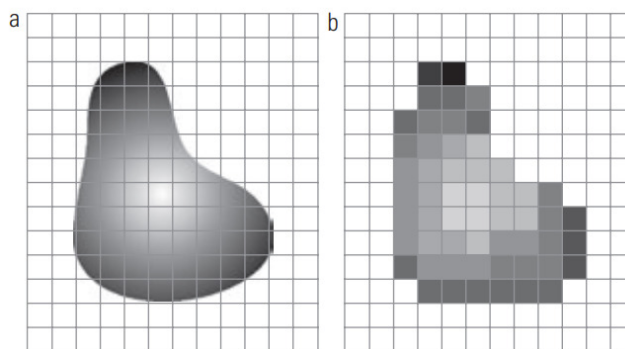
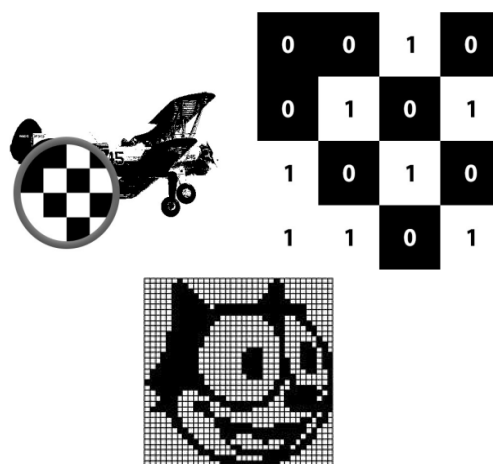


Figura 2.17 (a) Imagem contínua projetada em uma matriz de sensores. (b) Resultado da amostragem e quantização da imagem.

Prof. César C. Xavier

## Cor de Imagens

- Imagem Binária
  - Pixel admite apenas dois valores: 0 e 1.
- Largamente utilizado em VC:
  - Etapa de Segmentação; e
  - Extração Características.



Exemplos de imagens binárias.

Prof. César C. Xavier



## Cor de Imagens

- Imagem Tons de Cinza
  - Possibilidade de armazenar valor intermediários.
  - 8 bits: 256 tons de cinza, variando de 0 (preto) até 255 (branco).



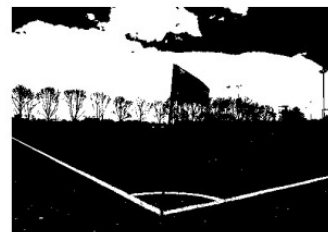
107	085	076	079
121	102	090	093
134	123	110	105
114	110	095	089

Exemplo de imagem tom cinza.

Prof. César C. Xavier

## Cor de Imagens

- Imagem Tons de Cinza
  - Possibilidade de armazenar valor intermediários.
  - 8 bits: 256 tons de cinza, variando de 0 (preto) até 255 (branco).



(a)



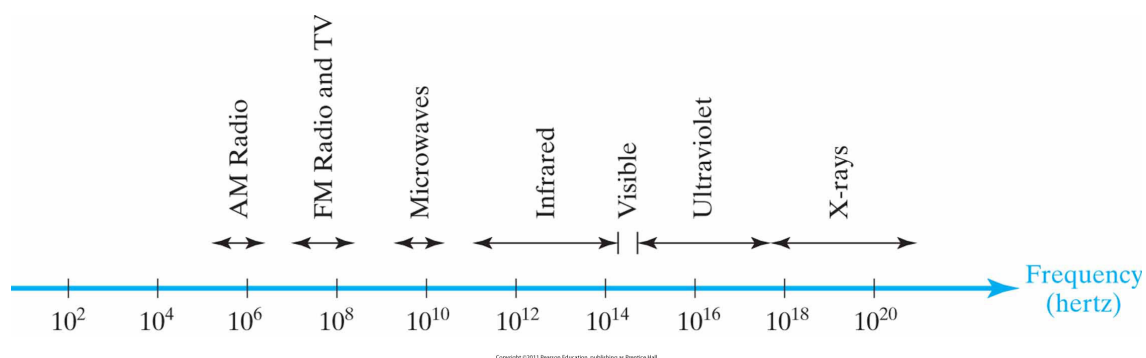
(b)

Comparação entre a mesma imagem na representação binária (a) e em tom cinza (b).

Prof. César C. Xavier

## Cor de Imagens

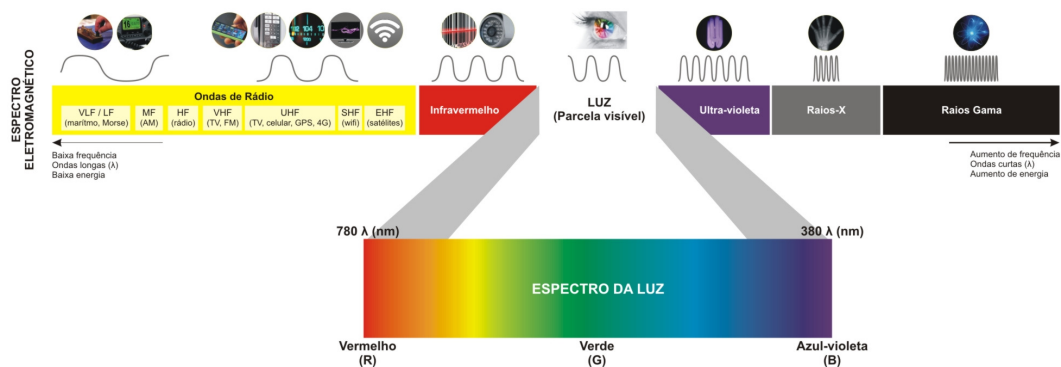
- Imagem em RGB
- Espectro Eletromagnético



Prof. César C. Xavier

## Cor de Imagens

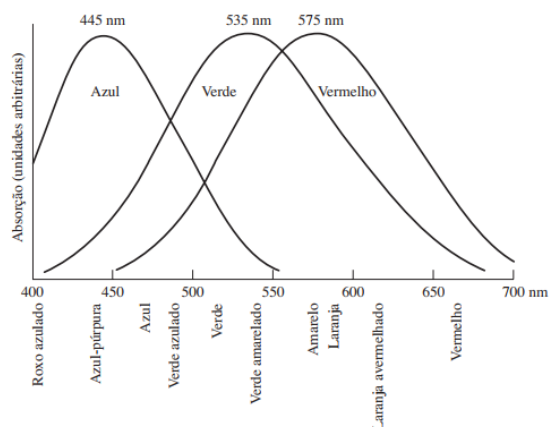
- Imagem em RGB
- Espectro Eletromagnético



Prof. César C. Xavier

## Cor de Imagens

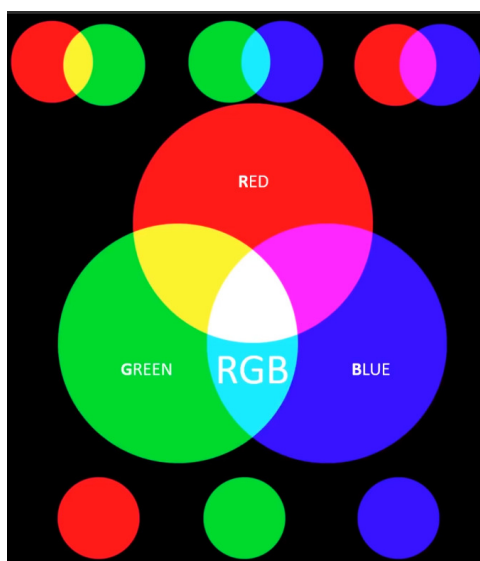
- Imagem em RGB
  - Espectro Eletromagnético



Prof. César C. Xavier

## Cor de Imagens

- Imagem em RGB
  - Sistema Aditivo

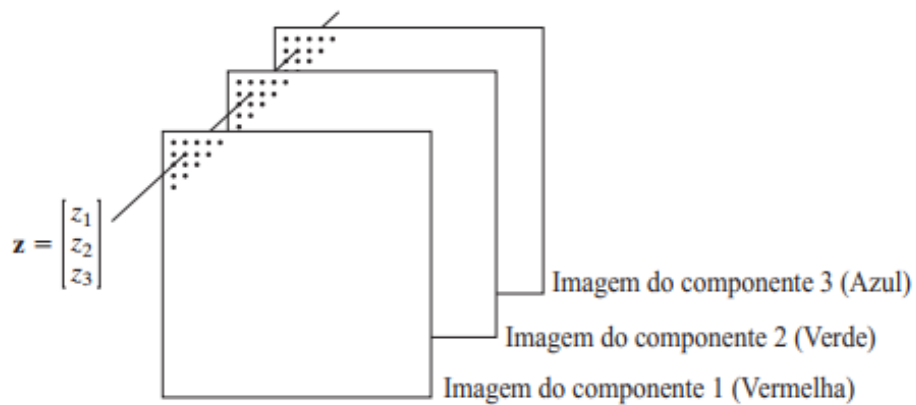


Prof. César C. Xavier

## Cor de Imagens

- Imagem colorida pode ser entendida como uma matriz 3D:

$$L \times C \times 3$$



Prof. César C. Xavier

## Cor de Imagens

- Imagem RGB



Prof. César C. Xavier

## Formato de Imagens

- BMP
- JPEG
- GIF
- PNG
- TIF/TIFF

Prof. César C. Xavier

## Formato de Imagens

- BMP – Bitmap
  - Desenvolvido pela Microsoft
  - Não tem compressão de dados
  - Formatos (bit/pixel):
    - 1 / 4 / 8 / 24 e 32.
  - Pouco usado na WWW

Prof. César C. Xavier

## Formato de Imagens

- JPEG – *Joint Photographic Experts Group*
  - Desenvolvido em 1983
  - Uso de compressão de dados
    - Taxa de compressão variável
  - Pode trabalhar até 24 bits / pixel
  - Largamente utilizado WWW

Prof. César C. Xavier

## Formato de Imagens

- GIF – *Graphics Interchange Format*
  - Desenvolvido pela empresa CompuServe (1987)
  - Admite compactação da imagem
  - Limitado a 256 cores.
  - Permite armazenamento de diversas imagens em um mesmo arquivo.
  - Excelente solução para pequenas animações.
  - Suporte fundo transparente.

Prof. César C. Xavier

## Formato de Imagens

- PNG – *Portable Network Graphics*
  - Desenvolvido em 1996 pela empresa W3C.
  - Faz uso de algoritmo de compressão de dados.
  - Não está limitado a 256 cores.
  - Opera com cores de até 24 bits / pixel.
  - Não suporta animações (estático).
  - Compactação < compactação JPEG.
  - Suporte fundo transparente.

Prof. César C. Xavier

## Formato de Imagens

- TIF/TIFF – *Tagget Image File Format*
  - Desenvolvido pela empresa Aldus em 1987 (adquirido pela Adobe).
  - Baixa taxa de compressão.
  - Excelente para edição de imagens.
  - Admite até 32 bits / pixel.
  - Suporte fundo transparente.

Prof. César C. Xavier



## Prática: Python e OpenCV

- Carregando Imagens de Arquivos

```
import cv2

imagem = cv2.imread("imagem.jpg")
cv2.imshow("Imagem", imagem)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Prof. César C. Xavier

## Prática: Python e OpenCV

- Carregando Imagens de Vídeos

```
import cv2

captura = cv2.VideoCapture("video.mp4")

while True:
    Ret, frame = captura.read( )
    cv2.imshow("Imagem", frame)

    if cv2.waitKey(10) & 0xFF == ord("q"):
        break

captura.release()
cv2.destroyAllWindows()
```

Prof. César C. Xavier

## Prática: Python e OpenCV

- Carregando Imagens da Webcam

```
import cv2

captura = cv2.VideoCapture(0)

while True:
    Ret, frame = captura.read( )
    cv2.imshow("Imagem", frame)

    if cv2.waitKey(1) & 0xFF == ord("q"):
        break

captura.release()
cv2.destroyAllWindows()
```

Prof. César C. Xavier

## Prática: Python e OpenCV

### Exercício #1

- Desenvolva um programa em Python que reproduz/gera GIF.
  - Entrega:
    - 1 arquivo zipado com: 2 programas em python ( geração de GIF e outros reprodutor de GIF); imagens a serem utilizadas para testar os programas.
    - Os programas deverão indicar os nomes dos integrantes do grupo
    - Enviar para o e-mail: cesar.xavier@docente.unip.br
  - Prazo: Prova P1

Prof. César C. Xavier