



# Fundamentos II

Guillermo Cámara-Chávez

# Introdução

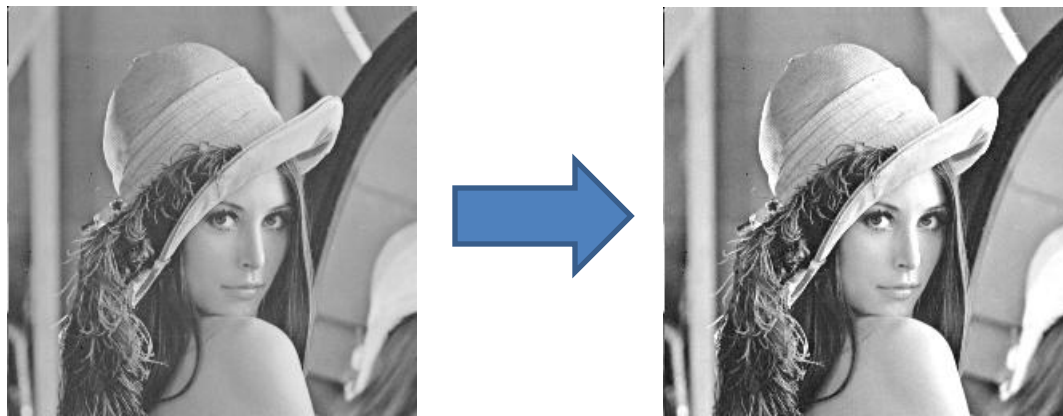


- Imagens digitais podem ser capturadas por uma variedade de sensores.
- O processamento dessas imagens possibilita diversas aplicações:
  - transmissão de vídeo,
  - diagnóstico médico,
  - controle de qualidade de processos industriais,
  - vigilância, etc.

# Introdução



- O **processamento** de imagem consiste em um conjunto de **operações matemáticas** (algoritmos) aplicadas sobre a imagem.
- O resultado do processamento é uma outra imagem.

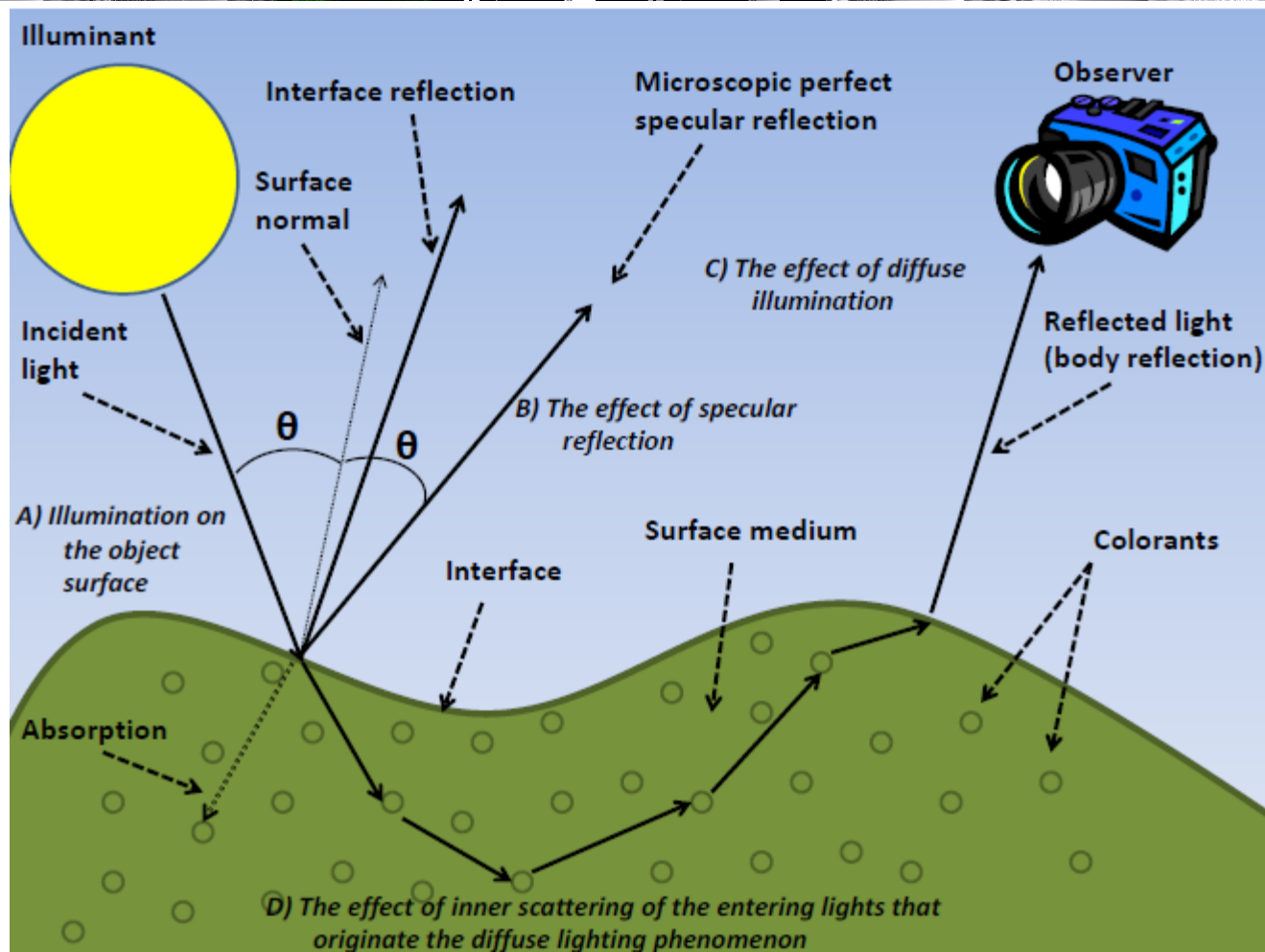


# Introdução



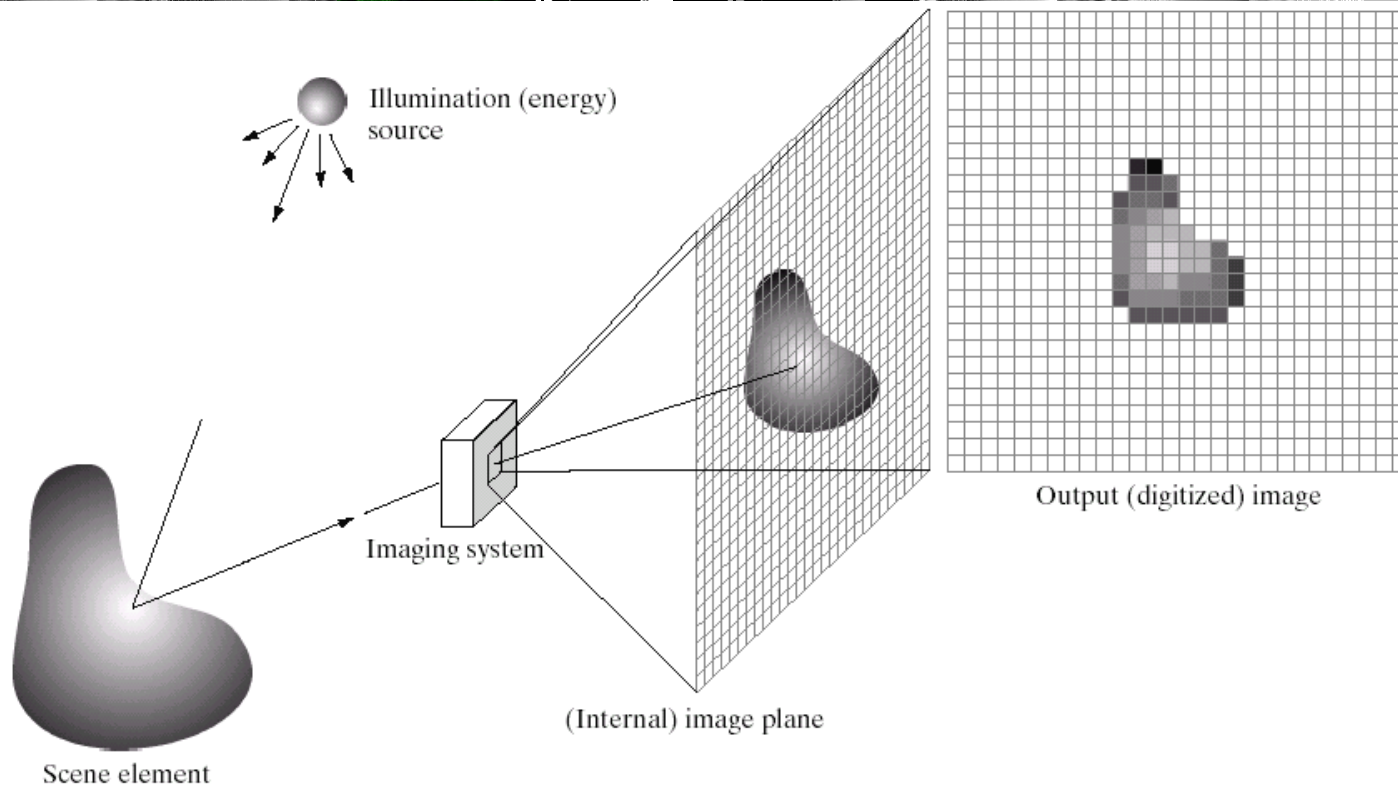
- Essas transformações são necessárias em outras disciplinas:
  - computação gráfica,
  - visualização científica,
  - Recuperação de imagens por conteúdo
  - Visão computacional

# Reflexão da luz





# Captura da imagem



a  
b c d e

**FIGURE 2.15** An example of the digital image acquisition process. (a) Energy (“illumination”) source. (b) An element of a scene. (c) Imaging system. (d) Projection of the scene onto the image plane. (e) Digitized image.



**Look at the dot above**





**This IS a black and white image  
but you DID see the colors of the walls,  
didn't you?**



# Captura da imagem (cont.)



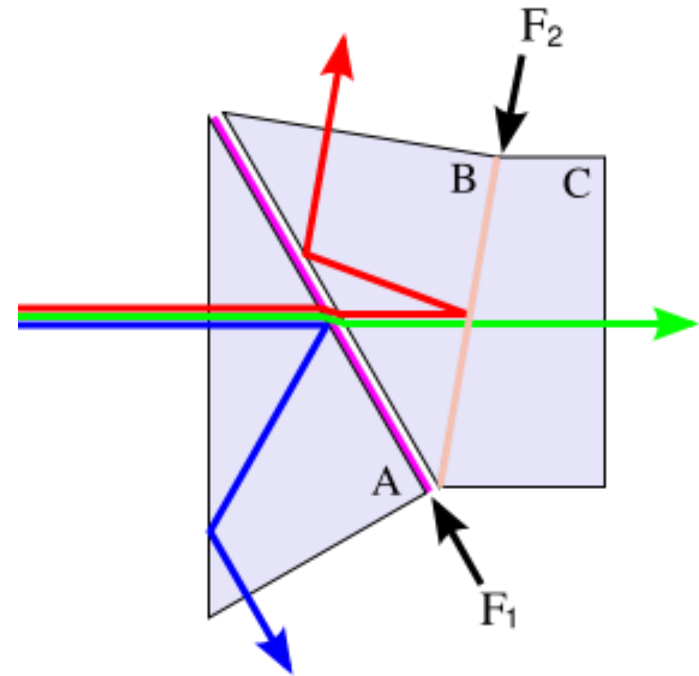
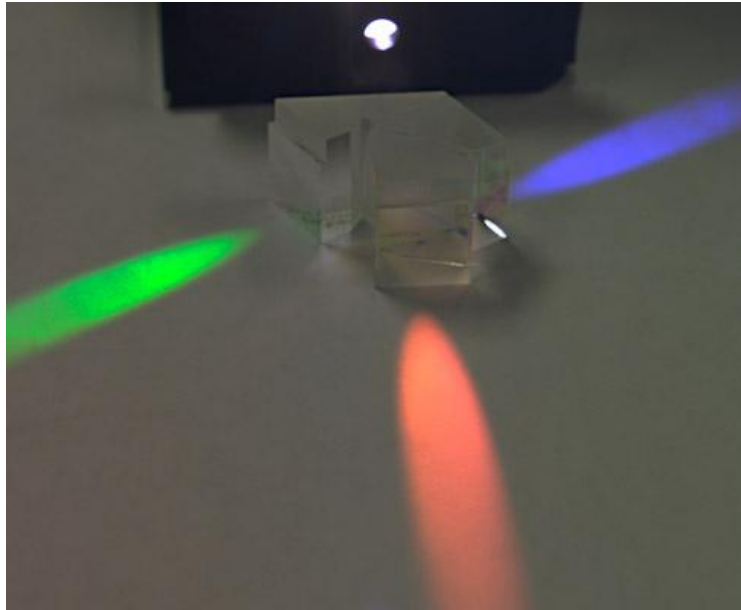
- Células foto-sensíveis (chamadas de CCD, *Charged Coupled Device*) capta a luz da cena a fotografar
- A informação é captada analogicamente e digitalizada (*frame grabber*)
- Finalmente armazenada num meio magnético

# CCD para câmeras coloridas



- Células foto-sensíveis são dispositivos monocromáticos.
- Para capturar a cor é necessário usar filtros para cores
- Abordagens:
  - Usar 3 CCDs (uma para cada cor RGB)
  - Usar um CCD com um filtro para cores

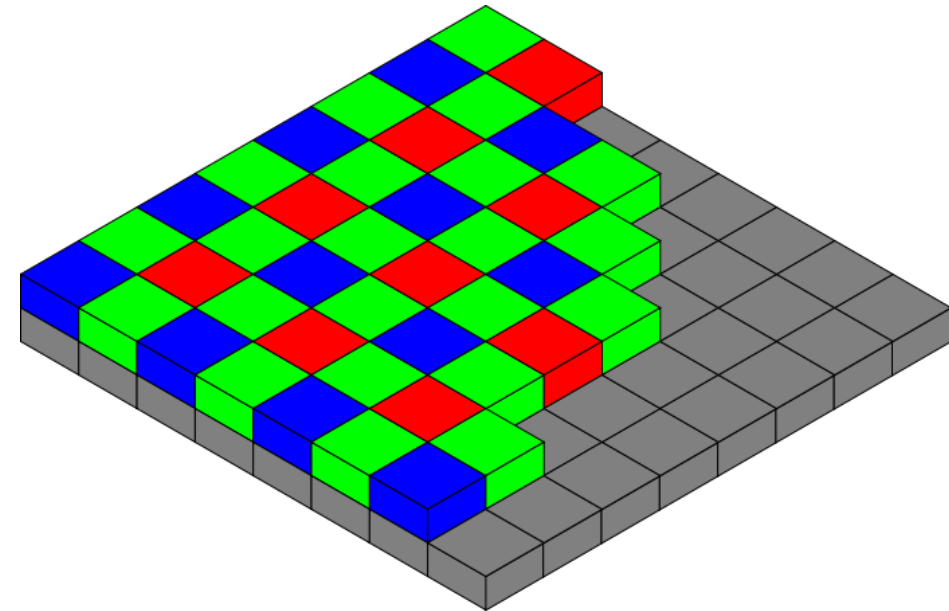
# CCD para câmeras coloridas



# CCD para câmeras coloridas

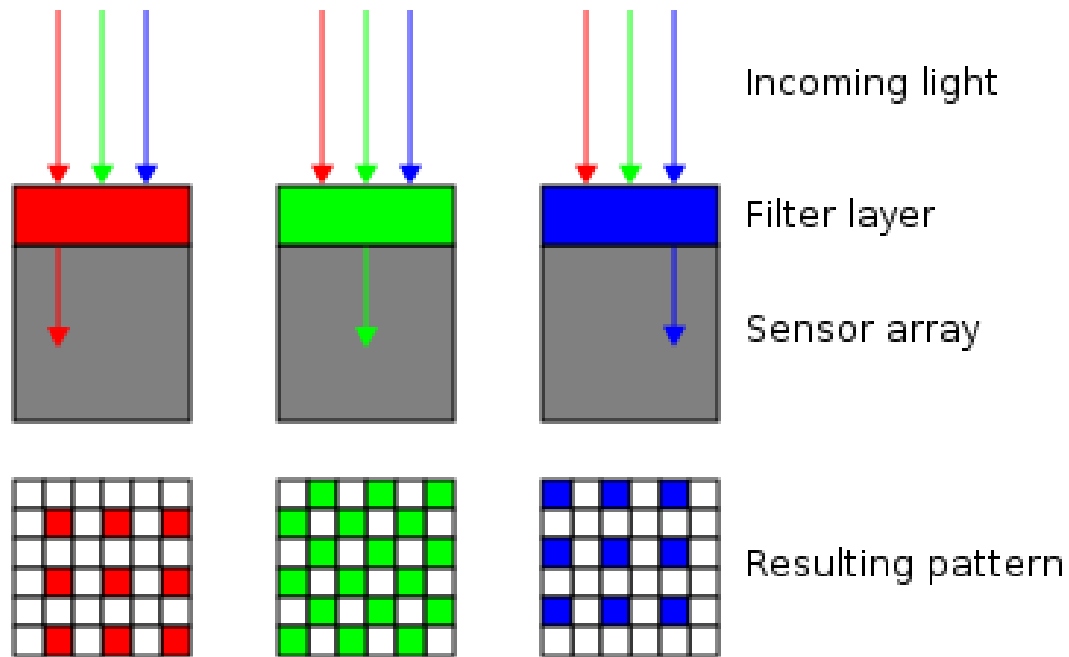


- Bayer filter pattern
- O sistema visual humano é mais sensível a cor verde
  - $Y = 0.299 * R + 0.587 * G + 0.114 * B$





# CCD para câmeras coloridas



# O que é uma imagem?



- Uma imagem é função bidimensional de intensidade da luz  $f(x,y)$ , onde  $x$  e  $y$  denotam as coordenadas espaciais.
- O valor de  $f$  em qualquer ponto  $(x,y)$  é proporcional ao brilho (níveis de cinza) da imagem nesse ponto.

# O que é uma imagem?



- Uma **imagem digital** é uma imagem  $f(x,y)$  **discretizada** tanto em **coordenadas espaciais** quanto em **brilho**.



# O que é uma imagem?



- Uma imagem digital pode ser considerada como uma matriz cujos **índices de linhas e colunas** identificam um **ponto na imagem**.
- Os elementos dessa matriz são chamados de *pixels* (abreviação de *picture elements*)



# Resolução da imagem



- É igual ao número de pontos que a compõe dividida pela área ocupada pela mesma.
- Quanto maior a resolução da imagem, melhor é sua definição.



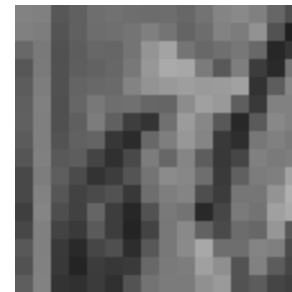
128x128



64x64

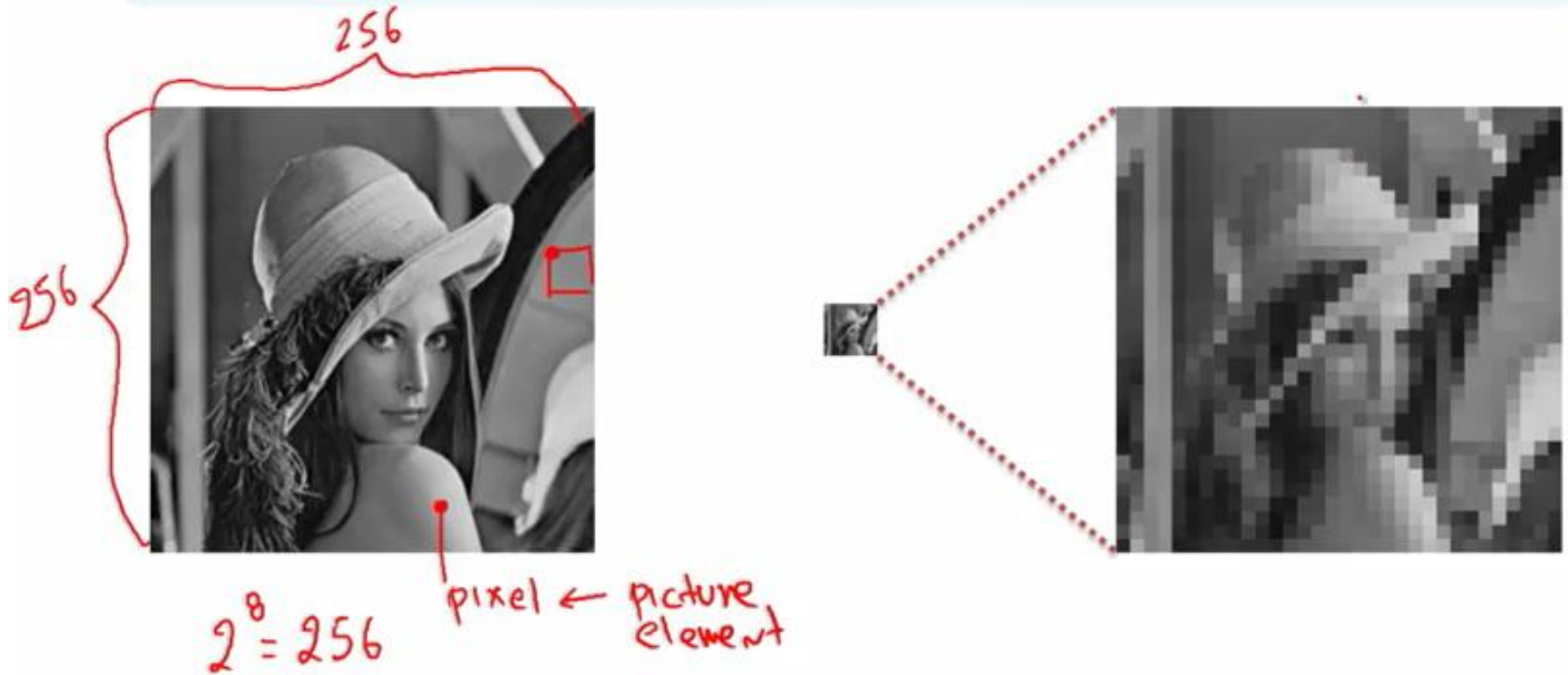


32x32



16x16

# Resolução da imagem



# Resolução da imagem (cont.)



- Se uma imagem possui **1000 x 1000** pixels, significa que possui um **milhão** de pixels.
- Atenção: isso **não necessariamente** quer dizer que está imagem é de **alta qualidade**, ainda falta a **relação** com a **quantidade de pixels por polegada** (DPIs) da imagem.

# Resolução da imagem (cont.)



- Para uma boa definição é preciso que uma imagem tenha 300 DPIs (*dots per inches* ou pontos por polegadas).
- Isso permite uma copia de alta qualidade em papel fotográfico



# Escala de cinzas (*grayscale*)



- São imagens compostas de tons de cinza, variando do preto (0) até o branco (255).
- O valor da intensidade está normalmente no intervalo  $[0, 2^b-1]$ , onde  $b$  é o número de bits por pixel.

# Escala de cinzas (*grayscale*)



118	97	98	173	152	16
95	93	130	152	158	162
93	97	167	157	165	171
95	111	168	168	171	176
96	153	167	174	169	104
110	163	170	167	144	23

# *True color*



- Utiliza 24 bits por pixel com 8 bits para cada cor que possibilita 256 níveis de intensidade e um total de 16 milhões de cores por pixel.
- O valor (0,0,0) de RGB equivale ao preto, e o valor (255,255,255) de RGB equivale ao branco

# True color



RED						
175	177	175	184	178	176	
177	<b>174</b>	179	175	178	179	
174	175	178	177	175	183	
			174	179	183	
			177	177	175	
			176	176	177	

GREEN						
72	63	63	62	64	72	
69	<b>63</b>	63	67	62	66	
67	56	65	64	62	67	
				63	69	
				63	65	
				66	63	

BLUE						
81	74	78	72	76	79	
73	<b>70</b>	72	75	76	72	
71	71	71	78	76	77	
79	73	72	78	70	76	
77	73	71	72	76	74	
81	72	73	80	83	75	



# *True color*



**R**



**G**



**B**



# Paleta de cores



- Imagens que utilizam paleta de cores
  - É um método onde o valor de um pixel pode estar relacionado a uma combinação de cores.
  - A principal vantagem é a grande variedade de cores que pode ser suportada em um espaço limitado de memória utilizando apenas 8 bits por pixel.

# Imagens indexadas



- Imagens com **256** cores ou menos, são denominadas indexadas.
- As imagens indexadas têm uma paleta de cores associada, denominada **color lookup table** em inglês, ou **CLUT** - tabela de consulta de cores.
- A **paleta** define até 256 cores, dando um número a cada cor. A imagem então refere-se a cada cor pela sua posição na paleta.

# Imagens indexadas



78	89	89	8
48	89	89	89
208	89	89	5 9
89	5	89	96 8
89	89	96	96 89
26	89	89	26 96

0.2863	0.0353	0.2000
0.2980	0.5098	0.6118
0.6314	0.0471	0.2235
0.3765	0.2784	0.3137
0.6784	0.5333	0.3804
0.8431	0.7765	0.3922



# Imagens



- Para carregar imagens em matlab usamos a função **imread**

**[x,map] = imread(nome\_imagem);**

- Para imagens em níveis de cinza e coloridas  
`img = imread('lenna.png');`  
`imshow(img);`



# Imagens (cont.)



- Para imagens indexadas

```
[img, map] = imread('lenna.gif');  
imshow(img, map);
```

# Imagens (cont.)



- Mostrando imagens

`imshow(f, G)`

onde  $f$  é uma imagem e  $G$  é o número de níveis de intensidade. Por *default* o valor de  $G$  é 256.

# Imagens (cont.)



- Usando a sintaxe: `imshow(f, [low high])`
  - preto: valores menores ou iguais a low
  - branco: valores maiores ou iguais a high
- Usando a sintaxe: `imshow(f, [])`
  - low: o menor de f
  - high: o maior valor de f

# Imagens (cont.)



- Escrevendo imagens:  
imwrite(f, 'filename')  
>> imwrite (f, 'lenna.png', 'png');

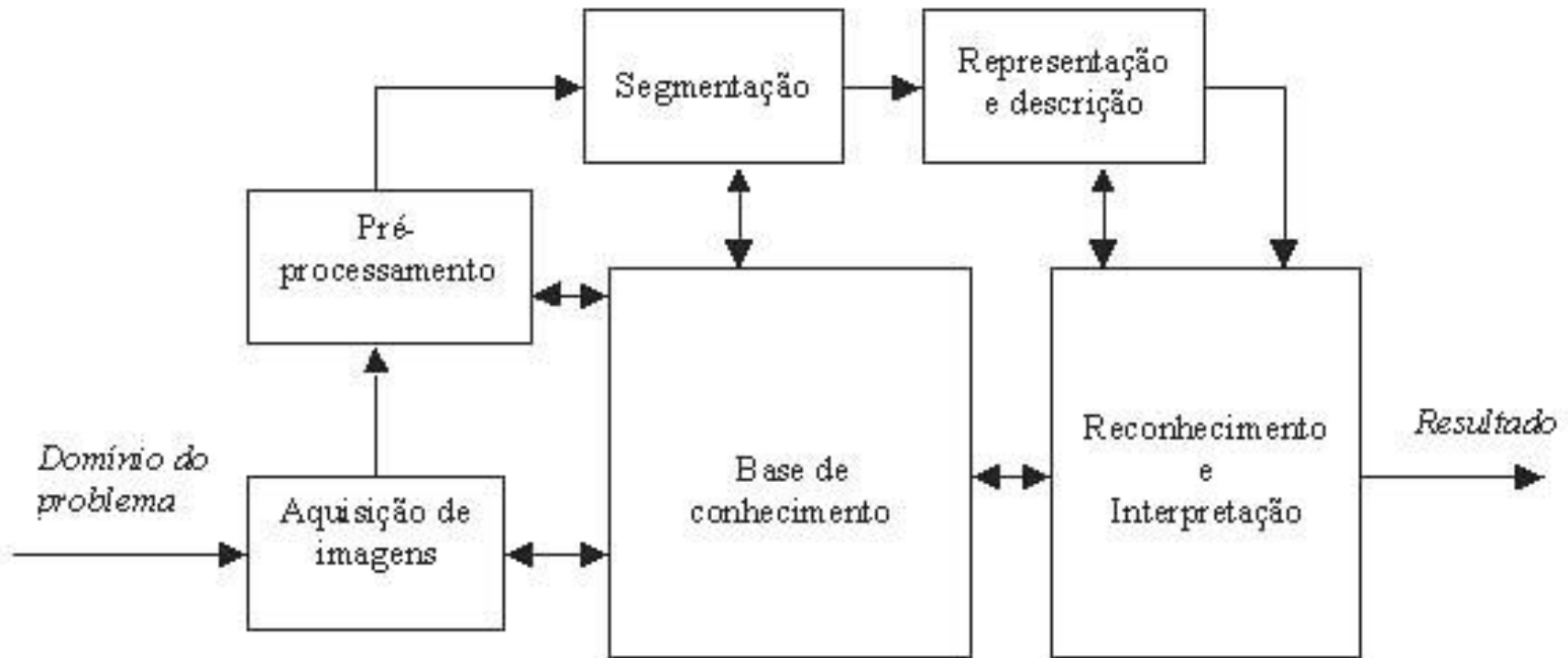
# Conversão de imagens



- `gray2ind` – imagem em níveis de cinza a indexada
- `im2bw` – imagem a binária
- `im2double` – imagem a precisão dupla
- `im2uint8` – imagem a inteiro de 8bits sem sinal
- `im2uint16` – imagem a inteiro de 16bits sem sinal
- `ind2gray` – imagem indexada a níveis de cinza
- `mat2gray` – matriz a imagem em níveis de cinza
- `rgb2gray` – imagem colorida a níveis de cinza
- `rgb2ind` – imagem colorida a indexada



# Passos Fundamentais do Processamento de Imagens



# Profundidade de bits



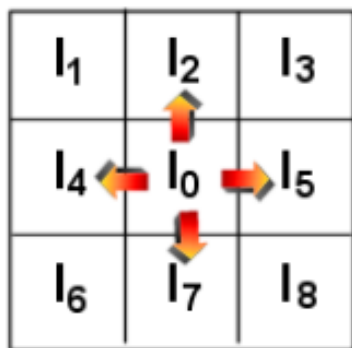
- É determinada pelo número de bits utilizados para definir cada pixel;
- Quanto maior a profundidade de bits, maior será o número de tons representados, em escala de cinza ou cor.



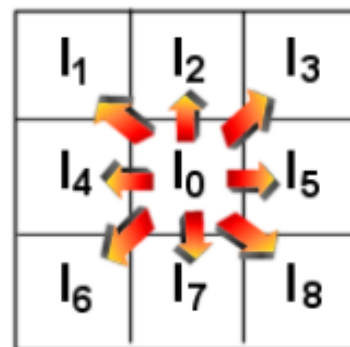
# Relacionamento básico entre pixels



- Vizinhos de um pixel



Vizinhança-4



Vizinhança-8

# Relacionamento básico entre pixels



- Conectividade
  - Para dizer que dois pixels **são conectados** é preciso determinar se eles são adjacentes e se seus níveis de cinza **satisfazem** um certo **critério de similaridade**
- Rotulação de componentes conexos
  - Atribuir rótulos diferentes a componentes conexos disjuntos de uma imagem

# Relacionamento básico entre pixels



Algoritmos simples e seqüencial de rotulação

1. Percorrer a imagem da esquerda para direita e de cima para baixo
2. Assumindo vizinhança  $N_4(p)$
3. Sejam os vizinhos superior e esquerdo de  $p$  representador por  $s$  e  $e$
4. Se o valor de  $p$  é 0, siga para a próxima posição



# Relacionamento básico entre pixels

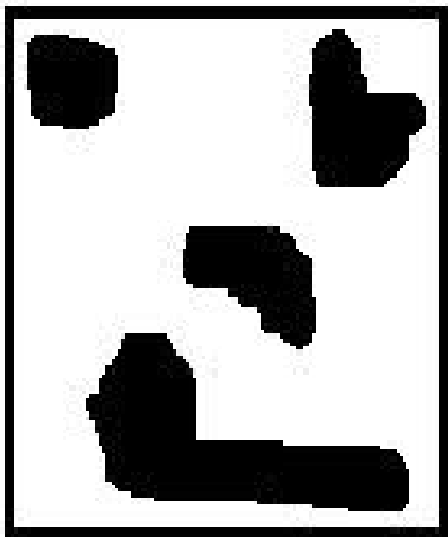


5. Se o valor de  $p$  é 1, examine  $s$  e  $e$ :
- a) Se ambos forem iguais a zero, atribua a  $p$  um novo rótulo
  - b) Se apenas um dos vizinhos de  $p$  for igual a 1, atribua a  $p$  o seu rótulo
  - c) Se ambos forem 1 e com o mesmo rótulo, atribua a  $p$  este rótulo
  - d) Se ambos forem 1 e com rótulos diferentes, atribua um dos rótulos a  $p$  e anote  $q$  os dois rótulos são iguais

# Relacionamento básico entre pixels



6. Ao final alguns rótulos poderão ser equivalentes. Portanto, é necessário reorganizar os rótulos equivalentes em valores iguais



1	1	0	0	1	0
1	1	0	0	1	1
0	0	0	0	1	1
0	0	1	1	0	0
0	1	0	1	0	0
0	1	0	0	0	0
0	1	1	1	1	1

1	1	0	0	2	0
1	1	0	0	2	2
0	0	0	0	2	2
0	0	3	3	0	0
0	4	0	3	0	0
0	4	0	0	0	0
0	4	4	4	4	4

# Tipos de ficheiros de imagem digital



- GIF (Graphics Interchange Format) – a cair em desuso
- PNG (Portable Network Graphics)
- JPEG (Joint Photographics Experts Group)
- TIFF (Tagged Image File Format) – a cair em desuso
- MPEG (Moving Picture Experts Group)

# GIF



- GIF87a, GIF89a
- Proposto pela Unisys Corp. e pela CompuServe.
- Objetivo inicial: transmitir imagens pelas linhas telefônicas via modem.
- Norma para ficheiros de imagem sem compressão.
- Imagens limitadas a cores de 8-bits (paleta de 256 cores).
- Suporta transparências (de uma só cor) e animação básica.



# PNG



- Foi desenvolvido para substituir o formato GIF em ambiente Web.
- É um formato sem-perdas, pois a compressão de ficheiros é feita sem perder quaisquer dados.
- Flexibilidade.
- As imagens podem ser:
  - indexadas (ou baseadas em paleta)
  - em tons de cinza
  - true-color (ou RGB) com resolução até 48-bits.

# PNG



- Outras características:
  - cor independente da plataforma
  - correção gamma
  - transparências variáveis
  - Não suporta animação.

# PNG



- Vantagens em ambiente Web:
  - Tamanho de ficheiro. Mais pequenos que os ficheiros GIF em 5-25%.
  - Transparência alpha total.
  - Correção gamma. Compensação automática de brilho entre monitores Macintosh e PC.
- Desvantagens:
  - Nem todos os browsers suportam completamente o formato PNG, em particular as características de transparência e correção de brilho

# JPEG



- Criada pelo Joint Photographics Experts Group.
- Uma norma para compressão **com-perdas**.
- Tira vantagem das limitações do sistema de visão humana por forma a atingir elevadas taxas de compressão.

# JPEG



- Ao salvar um ficheiro JPEG, podemos escolher um nível baixo ou elevado de compressão. Um nível baixo implica maior qualidade de imagem, mas também um tamanho maior do ficheiro.
- Resolução de cor: até 24 bits.



# JPEG



- Aplicabilidade:
  - Imagens com variações graduais de cor ou tonalidade; por exemplo, fotografia.
- Desvantagem
  - Degradação visível em imagem com contornos bem definidos, quando sujeita a elevados níveis de compressão

# TIFF



- Uma norma para ficheiros de imagem sem compressão.
- Armazena vários tipos de imagem: monocromática, tons de cinzento, 8-bits, 24-bits, etc.
- Desenvolvida pela Aldus Corp., 1980, e mais tarde suportada pela Microsoft.
- Compressão sem perdas que permite ao utilizador activar o nível desejado de qualidade/compressão.
- Não tem qualquer vantagem sobre o JPEG

# Exercícios propostos



1. Uma forma de transformar uma imagem colorida em níveis de cinza é:  $G = (R+G+B)/3$ , implemente o algoritmo (não utilizar a função `rgb2gray`)
2. Considere duas imagens cinzas, A e B, com  $|DA| = 100 \times 200$  e  $|DB| = 200 \times 400$  pixels respectivamente. A primeira cobre uma área de  $10 \times 20 \text{ cm}^2$  e a segunda uma área de  $100 \times 200 \text{ cm}^2$ .
  - a. Qual imagem possui maior resolução espacial?

# Exercícios propostos (cont.)



- b. Qual deveria ser o tamanho da imagem  $A$  para possuir a mesma resolução espacial da imagem  $B$ ?
- c. Qual é o comprimento de um segmento de reta em  $A$ , que vai do pixel  $p = (0, 0)$  ao pixel  $q = (50, 100)$ ?
- d. Quantos bytes são necessários para armazenar  $A$ , se os valores dos pixels vão de 0 a 2000? Qual é a profundidade de  $A$  neste caso?

# Exercícios propostos (cont.)



3. Qual a profundidade de uma imagem com 65536 níveis de cinza?
4. Qual é o tamanho de uma imagem gerada pela amostragem de uma região de  $100 \times 200$  cm<sup>2</sup> a intervalos  $dx = 0,1$  mm,  $dy = 0,2$  mm? Se esta imagem fosse a foto de um objeto, que tipo de distorção você veria? Por quê?