

# Fundamentos de Processamento Digital de Imagens

**Agostinho Brito**

2022

# O que é processamento digital de imagens?



# O que é processamento digital de imagens

Saída	Entrada	
	IMAGEM	MODELO
IMAGEM	<b>Processamento digital de Imagens</b>	Computação gráfica
MODELO	Visão computacional	Geometria Computacional

# Primeira imagem digital

- Adquirida em 1957 por **Russel Kirsch**, cientista do NIST.
- Tinha  $176 \times 176$  pixels de tamanho.
- Diversos limiares precisaram ser ajustados para que a imagem em tom de cinza pudesse ser capturada no scanner monocromático desenvolvido.



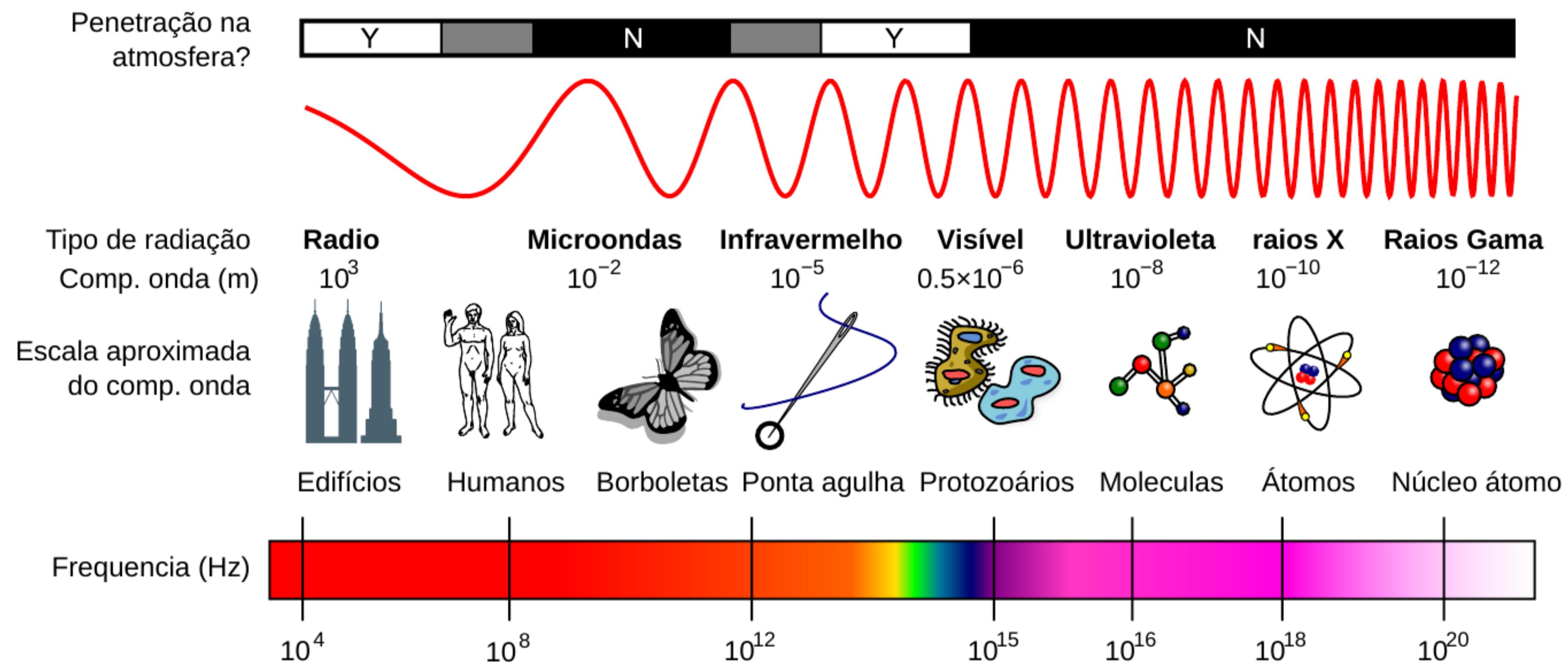
# Primeira câmera digital

- Criada em 1975 por **Steve Sasson**.
- Resolução de 0.01 Megapixels.
- A imagem foi adquirida em 23 segundos.
- Para ver as imagens foi desenvolvida uma adaptação para uma TV.



# De onde vêm as imagens?

As mais comuns provêm do espectro de ondas eletromagnéticas



(Fonte: Wikimedia)

# Exemplos de aplicações



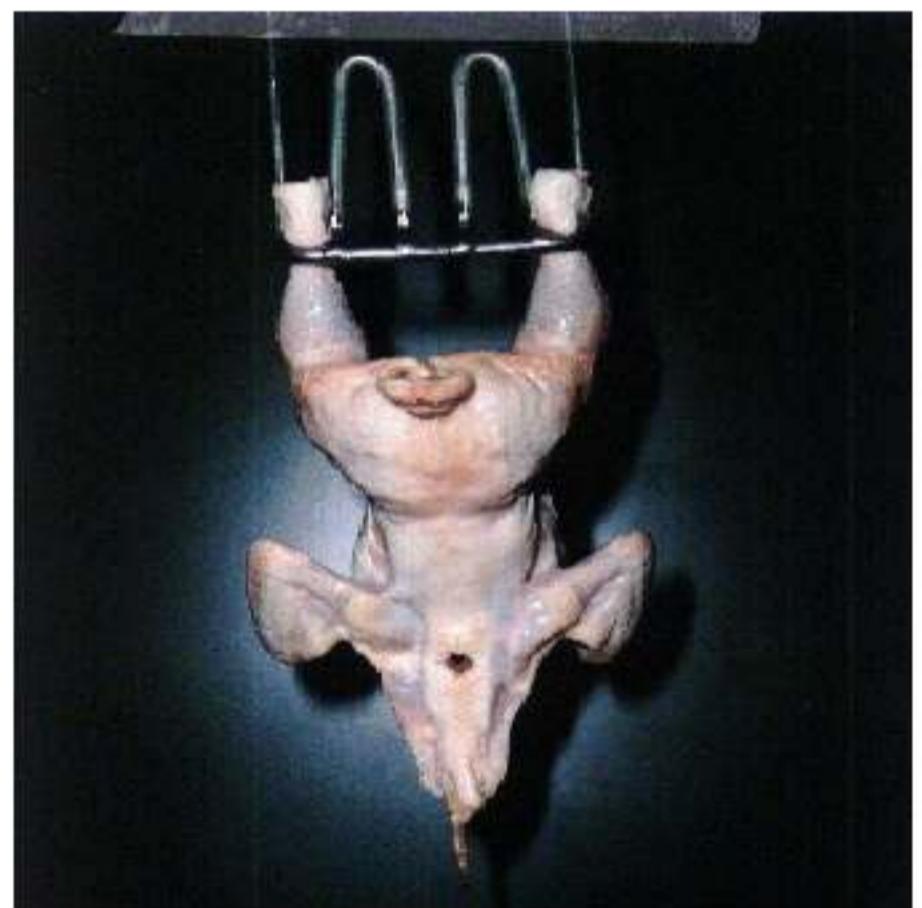
Raios X



Microscopia



Ultrassonografia



Automação

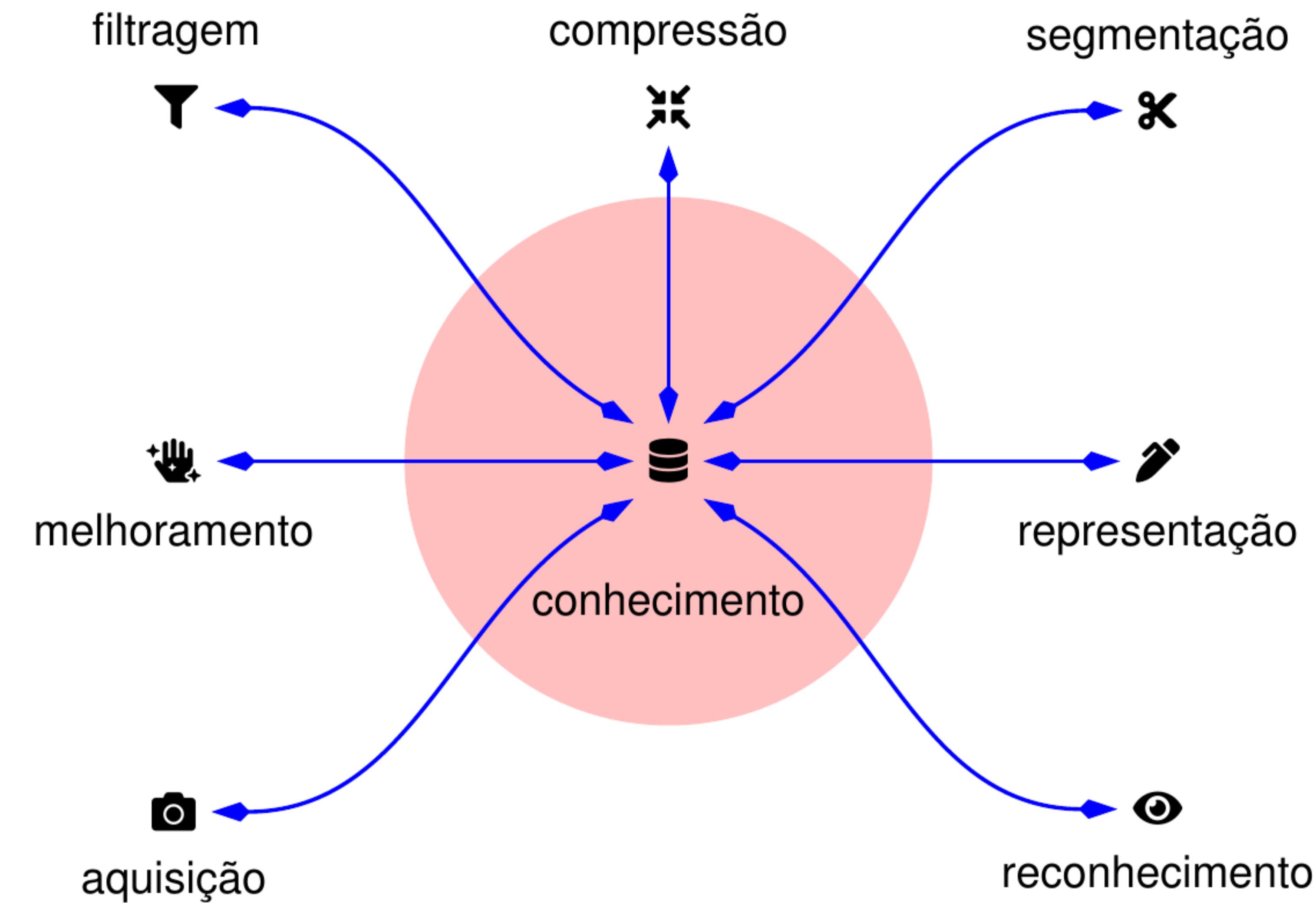


Entretenimento

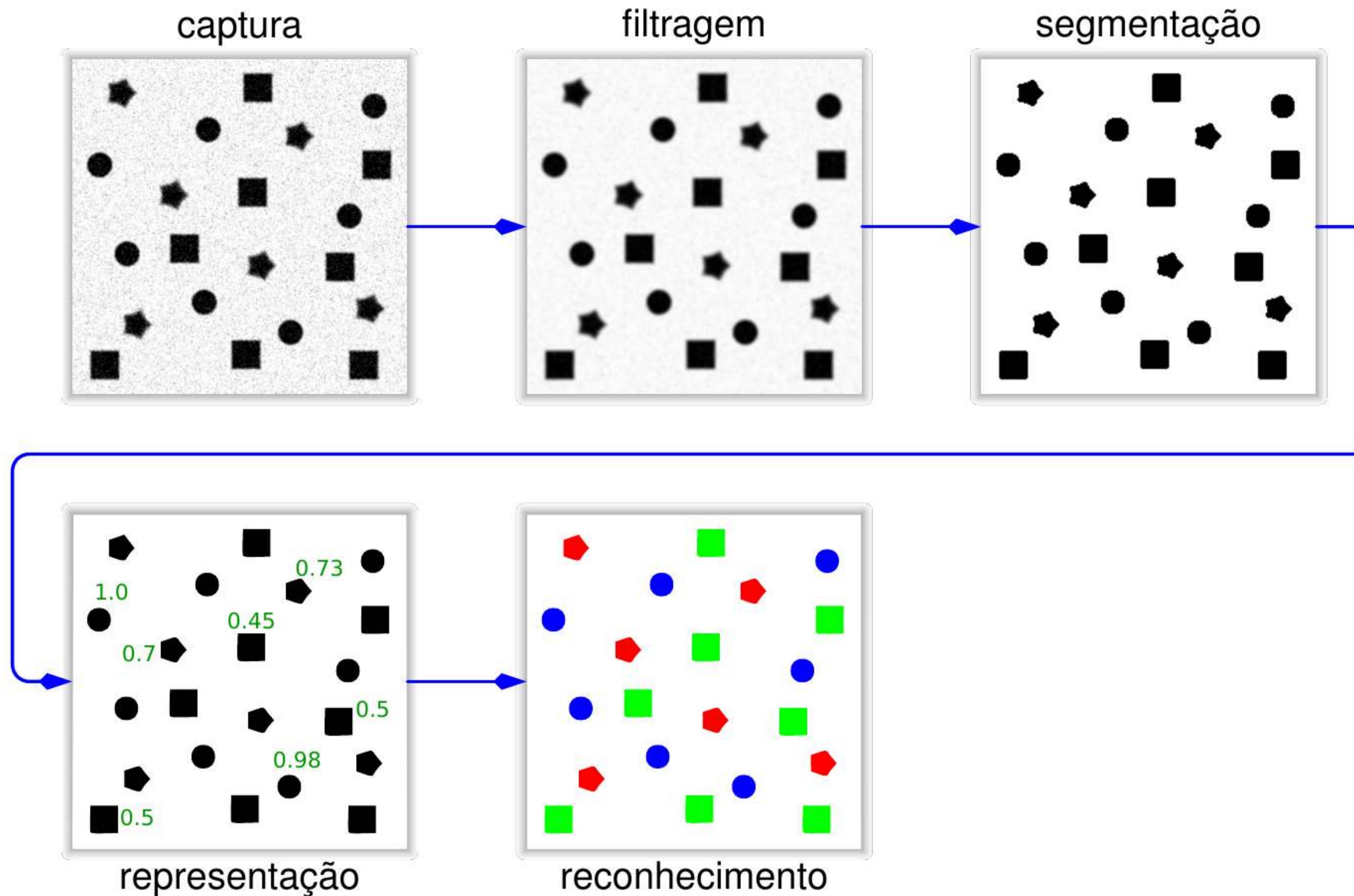


Visão

# Etapas envolvidas



# Etapas envolvidas no processamento de imagens - exemplo



# Imagen: definição

- Uma **imagem** pode ser definida como toda e qualquer visualização gerada pelo ser humano, seja em forma de objeto, de obra de arte, de registro foto-mecânico, de construção pictórica (pintura, desenho, gravura) ou até de pensamento.
- Uma imagem é uma **figura** quando é produzida por ordenação de pigmentos sobre algum suporte, geralmente utilizando técnicas de fotografia: impressão em filme de uma cena tomada com uso de uma câmera fotográfica.

# Imagen monocromática

- Função de intensidade luminosa bidimensional  $f(x, y)$ , onde os valores de  $x$  e  $y$  denotam coordenadas espaciais e o valor de  $f$  em qualquer ponto  $(x, y)$  é proporcional ao brilho da imagem naquele ponto.
- As mais comuns são as imagens cinza (fotos antigas) e as imagens preto-e-branco (impressão em livros).



## Representação

- Uma imagem é uma função bidimensional de intensidade de luz  $f(x, y)$ , onde

$$0 < f(x, y) < \infty$$

- A natureza dos tons da imagem pode ser caracterizada por duas componentes: intensidade luminosa,  $i(x, y)$ , dependendo da fonte de energia, e reflectância,  $r(x, y)$ , que depende das propriedades do material.

$$f(x, y) = i(x, y)r(x, y)$$

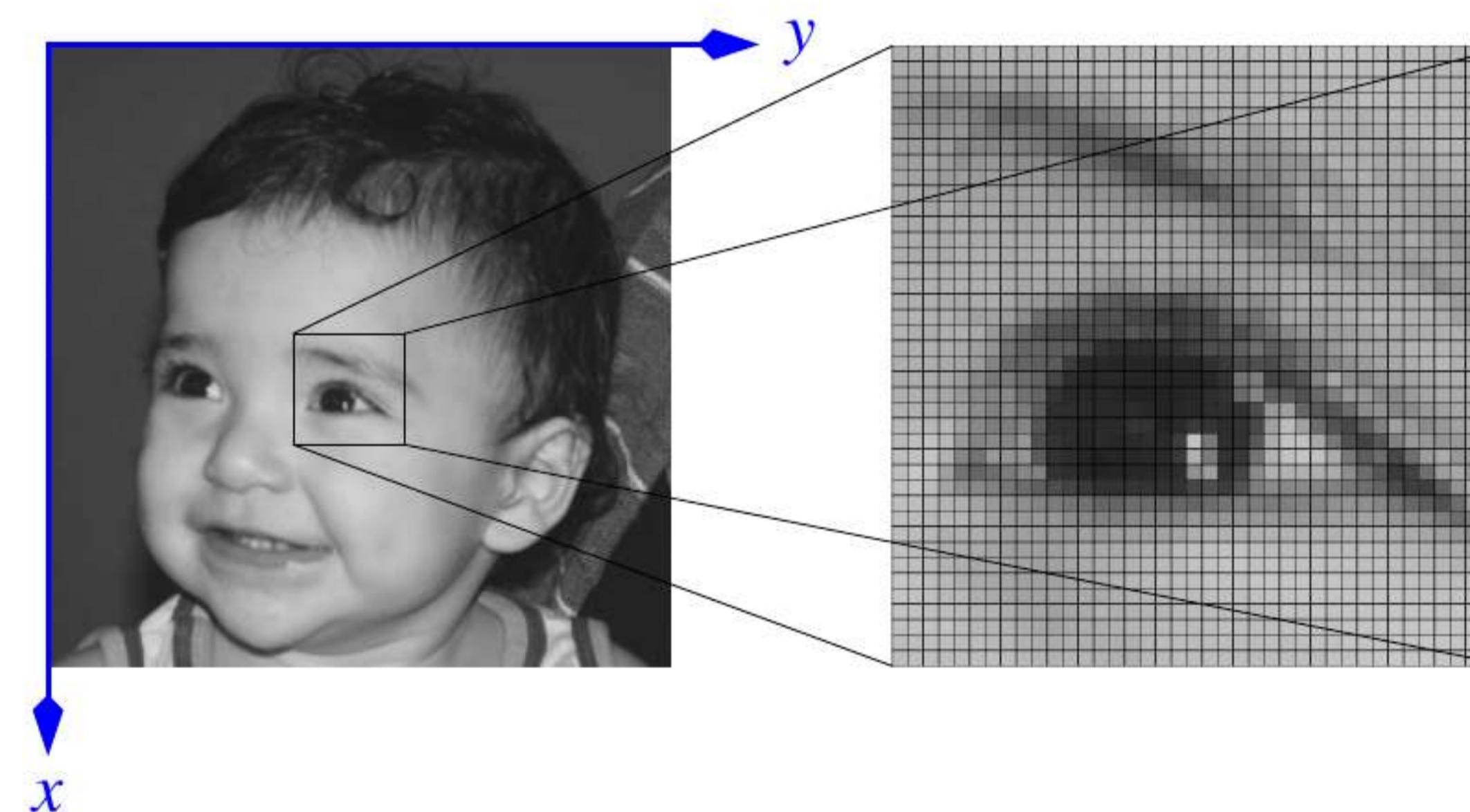
onde

$$0 < i(x, y) < \infty \quad \text{e} \quad 0 < r(x, y) < 1$$

- $r(x, y) \rightarrow 1$ : tendem a refletir a luz que incide sobre o material (Ex: superfície branca).
- $r(x, y) \rightarrow 0$ : tendem a absorver a luz que incide sobre o material (Ex: superfície preta).

# Representação de imagens monocromáticas

- Uma **imagem monocromática** é dita **digital** quando é representada por uma matriz cujos índices das linhas e colunas identificam um ponto na imagem e o elemento correspondente da matriz identifica o nível de cinza neste ponto.
- Os elementos da matriz são comumente chamados de **pixels**, ou pels, abreviaturas de *picture elements*.



## Imagen policromática

- Formada pela composição de funções de intensidade luminosa bidimensionais  $f_i(x, y)$ , onde os valores de  $x$  e  $y$  denotam coordenadas espaciais e o valor de  $f_i$  em qualquer ponto  $(x, y)$  para a função  $i$  é proporcional a uma resposta de uma região do espectro para a imagem naquele ponto.
- Normalmente, várias matrizes são necessárias para representar a noção de cor.
- As mais conhecidas são as imagens coloridas tradicionalmente armazenadas em arquivos como jpeg, png ou gif.

- A **captura de imagens** envolve um dispositivo sensível à energia eletromagnética (luz visível, ultravioleta, infravermelho, raiosX) ou eletromecânia (ultrassom), e um dispositivo digitalizador, que converte a saída elétrica contínua do sensor para níveis digitais.
- Imagens digitais possuem valores discretos de intensidade e posição espacial. Os principais tipos de sensores são os **de linha** (scanner de mesa) e os **de área** (camera filmadora).
- As imagens podem conter informação de cor ou não.
- Para imagens monocromática, apenas uma matriz é usada e a imagem é representada em **escala de cinza** (ou *grayscale*).
- Para imagens policromáticas (coloridas), três componentes de cor (ou matrizes) são utilizadas:
  - Red
  - Green
  - Blue.

Uma imagem  $f(x, y)$  precisa ser digitalizada tanto no espaço quanto em intensidade (amplitude). Este processo é chamado de quantização. Pode ser feita:

- Espacialmente ( $M_{linhas} \times N_{colunas}$ )
- No número de níveis utilizados para representar cada pixel, denotado por  $G = 2^m$ , onde  $m$  é a quantidade de bits usada para representação.
- Com  $m = 8$ , cada pixel pode representar até 256 tons de cinza, suficientes para distinção pelo olho humano.



# Efeitos da redução de elementos na matriz de pontos

- Perda de detalhes
- Efeito “tabuleiro de xadrez”.



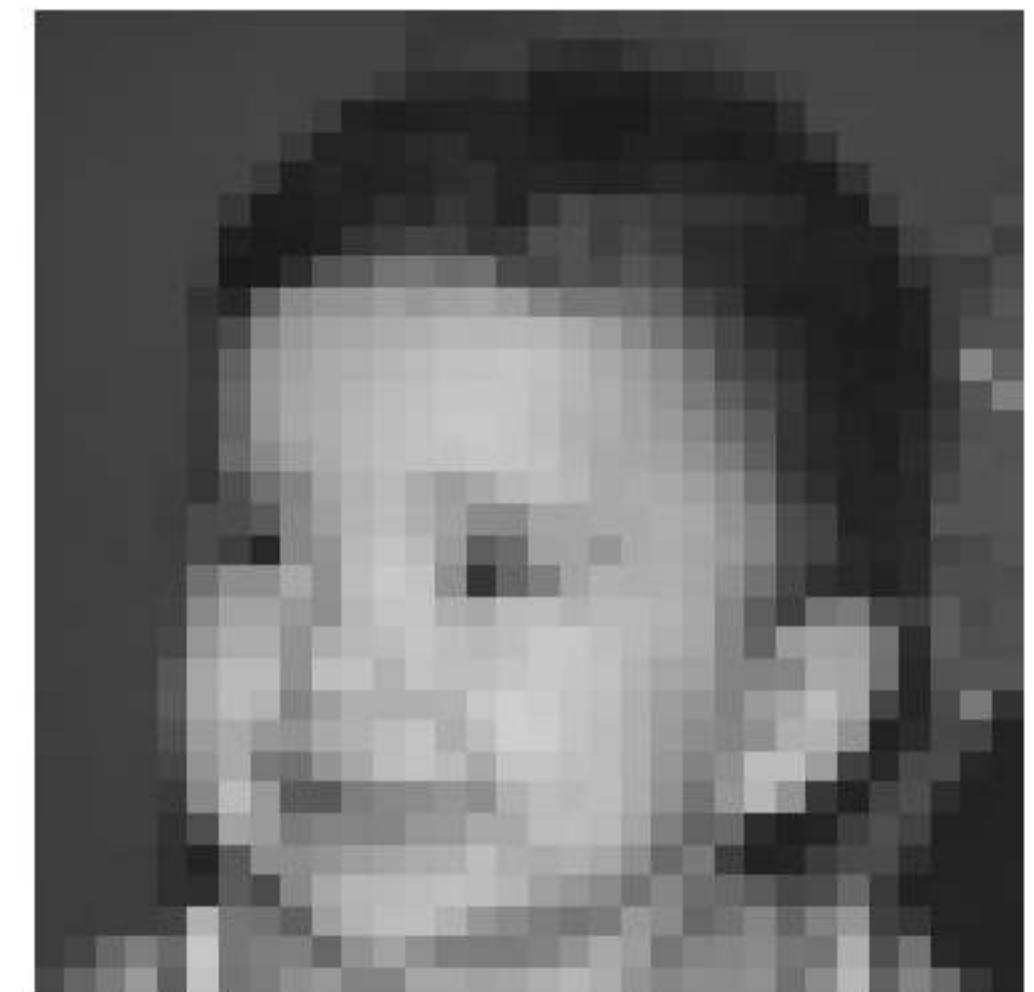
256



128



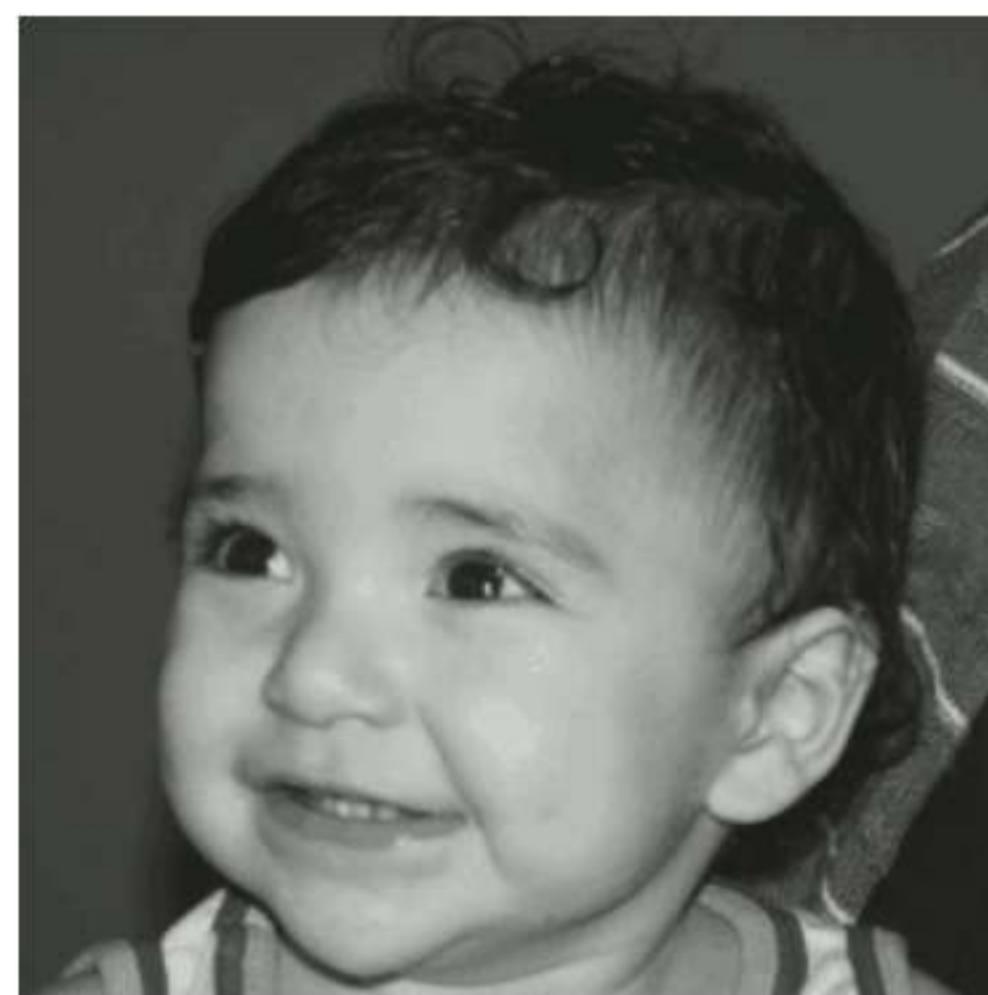
64



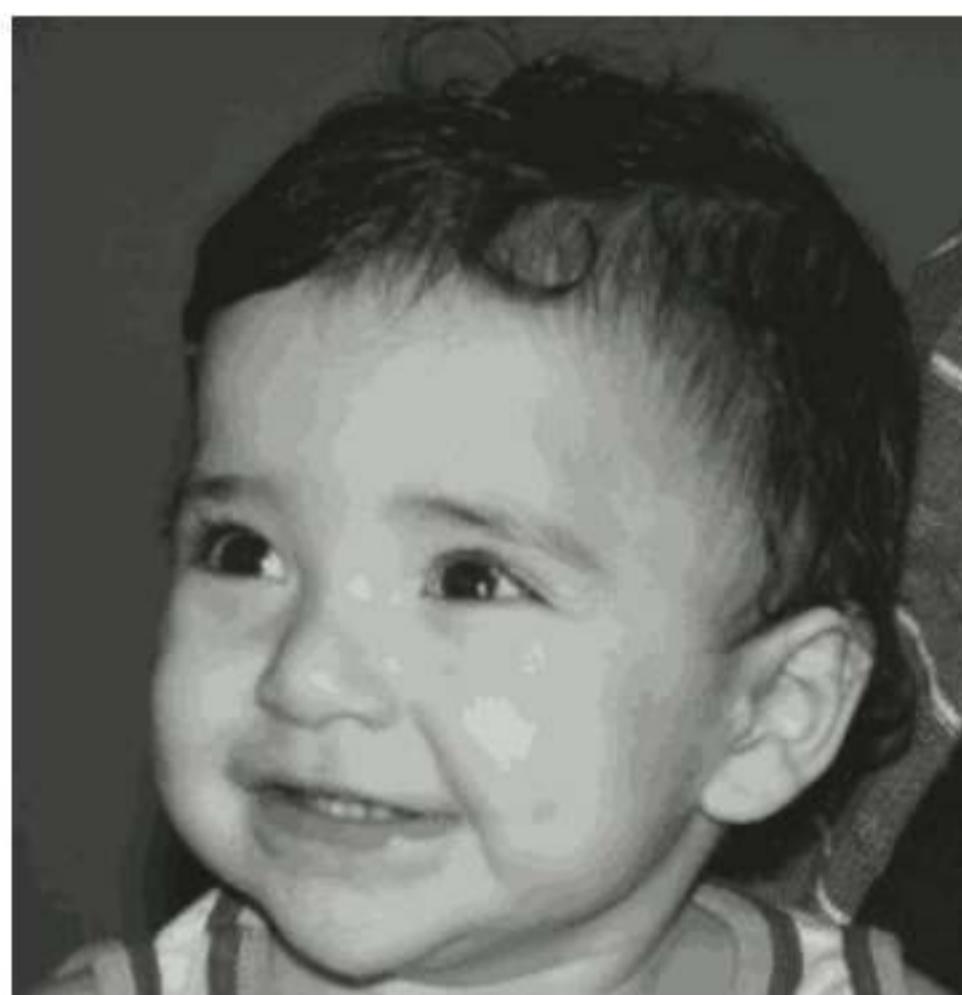
32

# Efeitos da redução da quantidade de bits para representação

- Efeito de falso contorno.
- Visível em imagens com 16 tons de cinza ou menos.



5 bits



4 bits



3 bits



2 bits

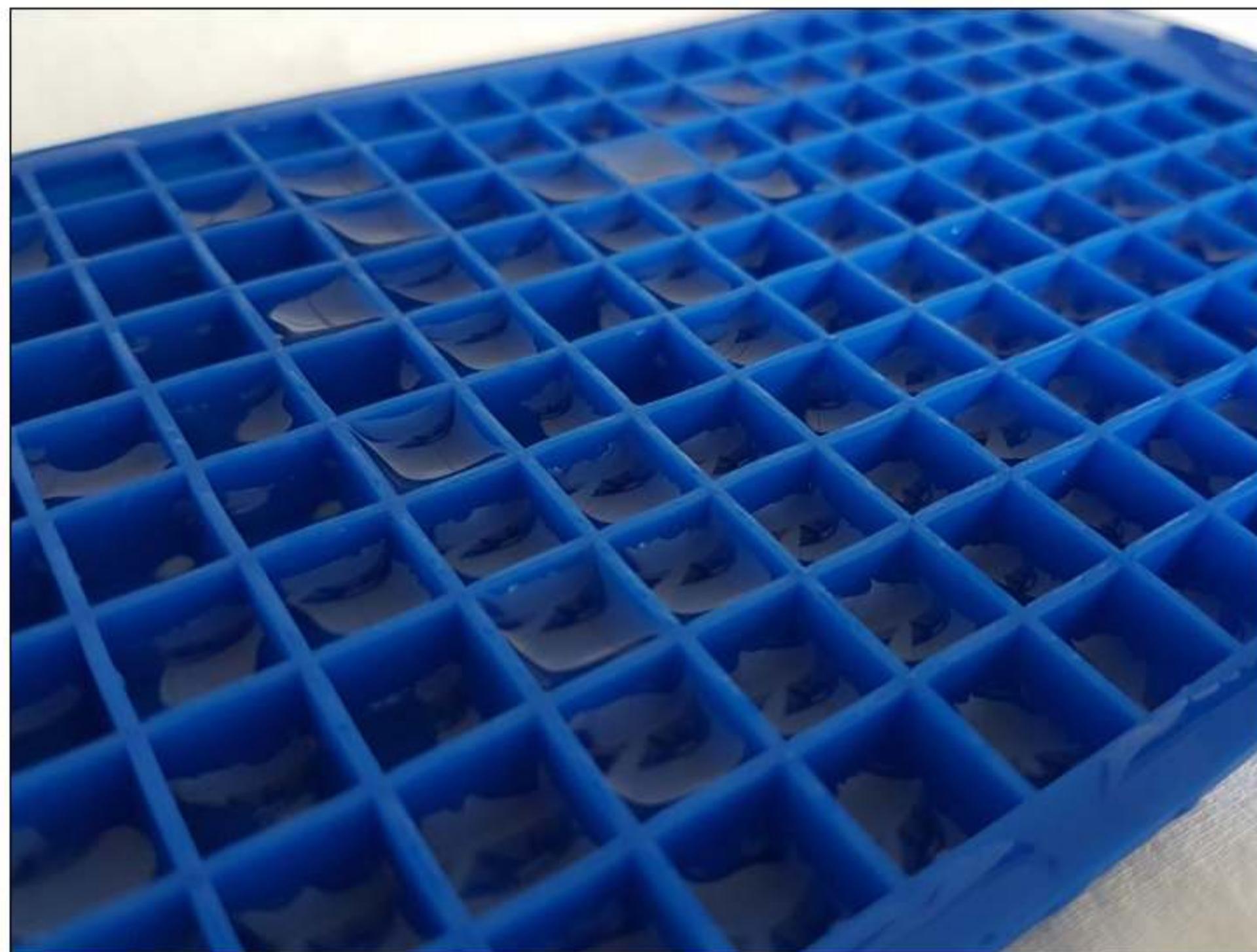
- Número de elementos nos registradores.
- Quanto maior, melhor a riqueza de detalhes.
- Fator relacionado com a quantidade de memória necessária para guardar uma imagem.
- Imagem cinza com resolução de 640x480 pixels necessita de 300Kbytes para ser armazenada.  
Para uma imagem com resolução de 1024x1024, a memória necessária é de 1Mbyte.
- Métodos de compressão reduzem estes números.

## Cuidado

Maior resolução não necessariamente implica em melhor qualidade.

## Resolução espacial

- A qualidade do sensor faz toda a diferença.
- Sensores grandes **normalmente** produzem imagens melhores.
- Lá vem a **chuva de fótons** no “sensor” de  $16 \times 10$  “pixels”.



- Os circuitos de tratamento de ruídos nos sensores atuais têm melhorado bastante a qualidade das imagens.

# A tarefa do processamento

- O processamento de imagens digitais é geralmente expresso na forma algorítmica. A maioria das funções de processamento pode ser implementada via *software*.
- O processamento via *hardware* geralmente só é necessário quando a velocidade é fator preponderante nos resultados e não pode ser alcançada via software. Ex: **PIXY cam**.
- Câmeras de vídeo comuns capturam imagens a 30 quadros por segundo.

# A etapa da comunicação

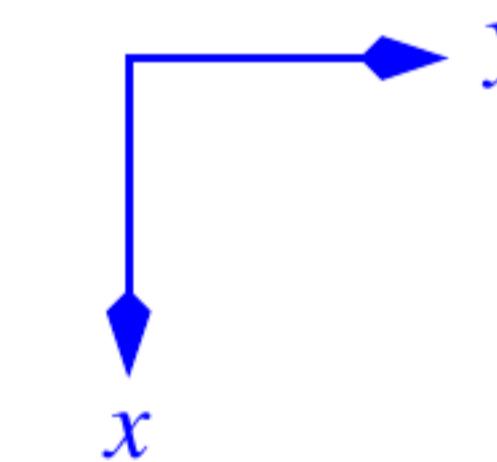
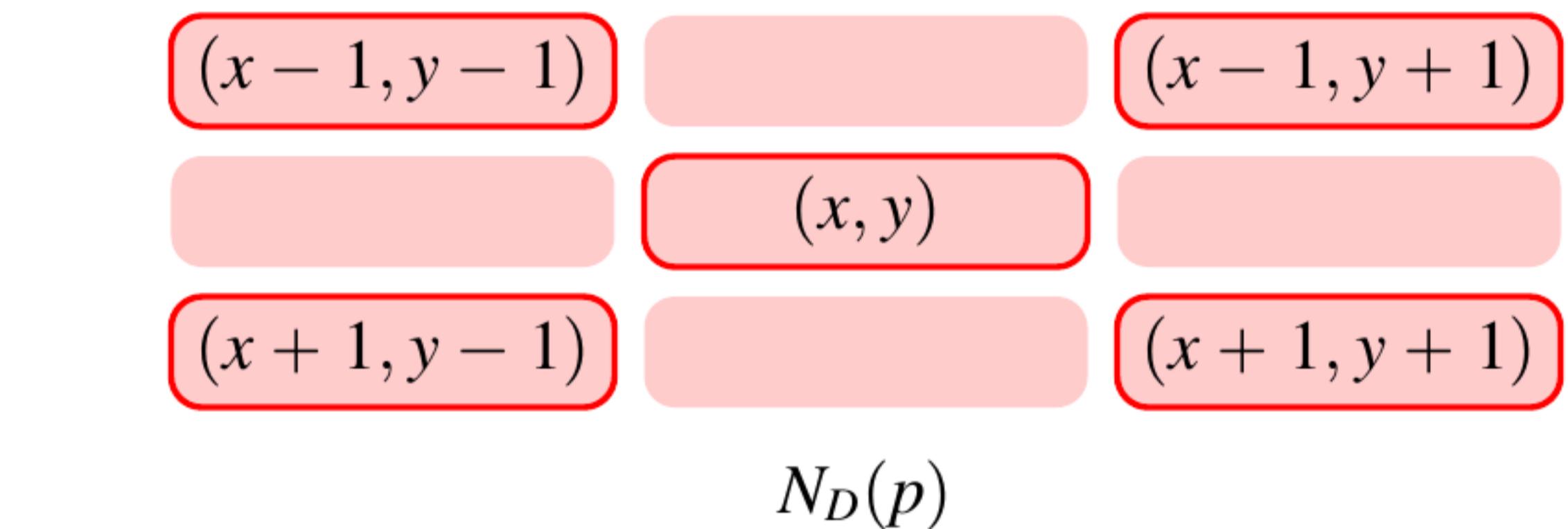
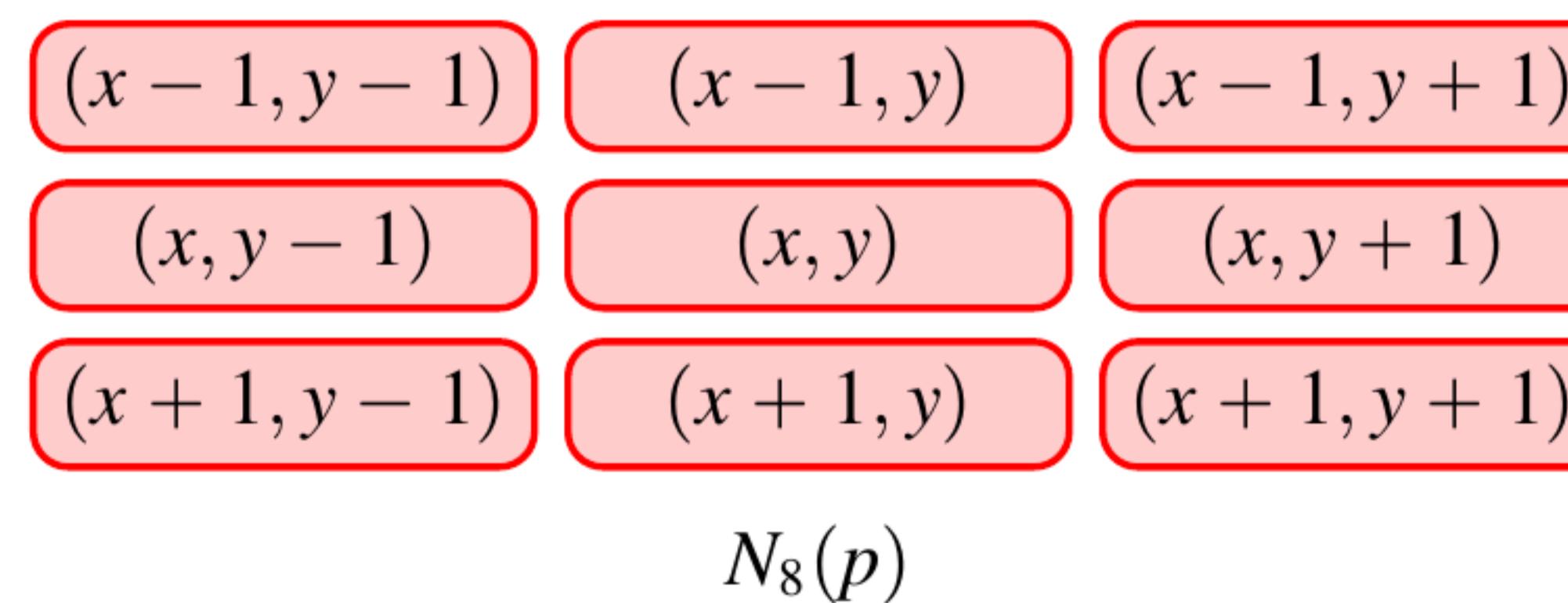
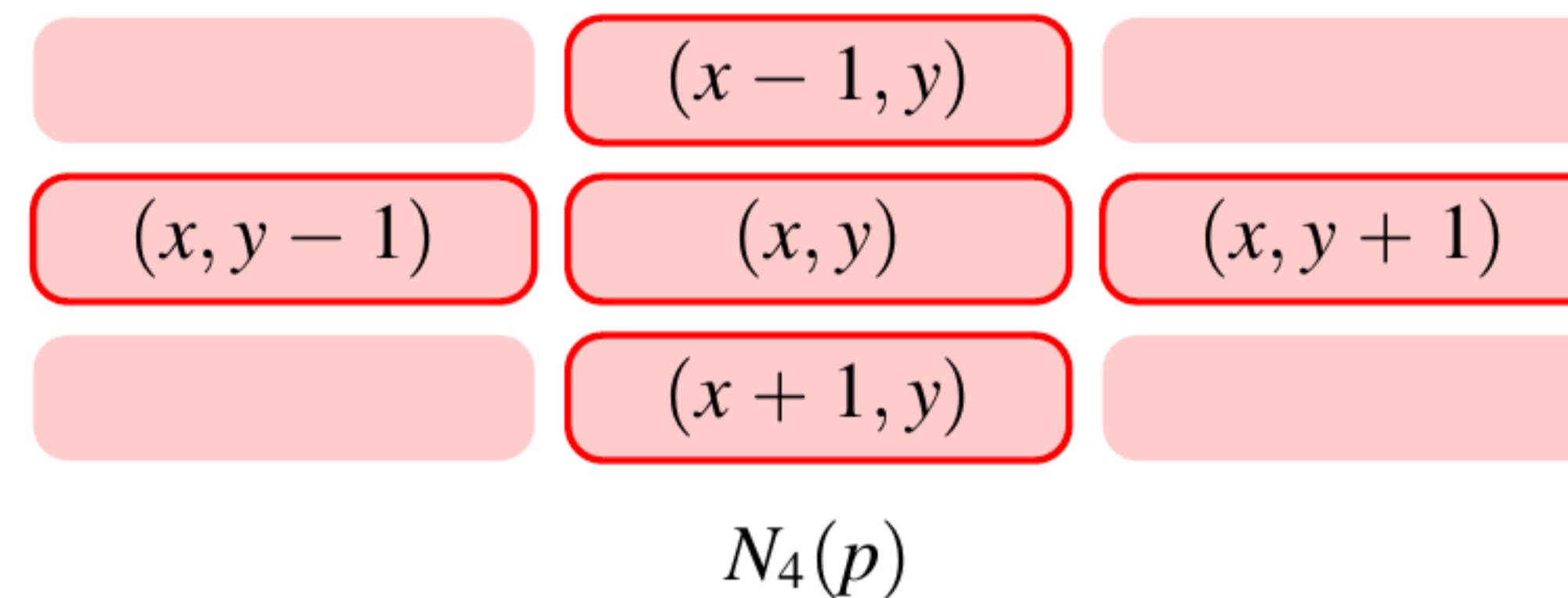
- Envio de dados entre estações de processamento.
- Técnicas de compressão aceleram a transmissão de dados com ou sem perda de informações.
- Praticamente todas as comunicações na Internet que envolvem transmissão de imagens (ou vídeo) utilizam alguma técnica de compressão de dados.

- Exibição da imagem em monitores de vídeo ou dispositivos de impressão.
- Imagens coloridas / *pseudocolor*.

índice	R	G	B
0	10	230	20
1	15	23	179
...	...	...	...
255	200	10	68

# Relações de vizinhança entre pixels

- Cada pixel possui um conjunto de outros pixels que participam de sua vizinhança.
- O conceito de vizinhança é utilizado no estabelecimento de fronteiras e conectividade de regiões em uma imagem.



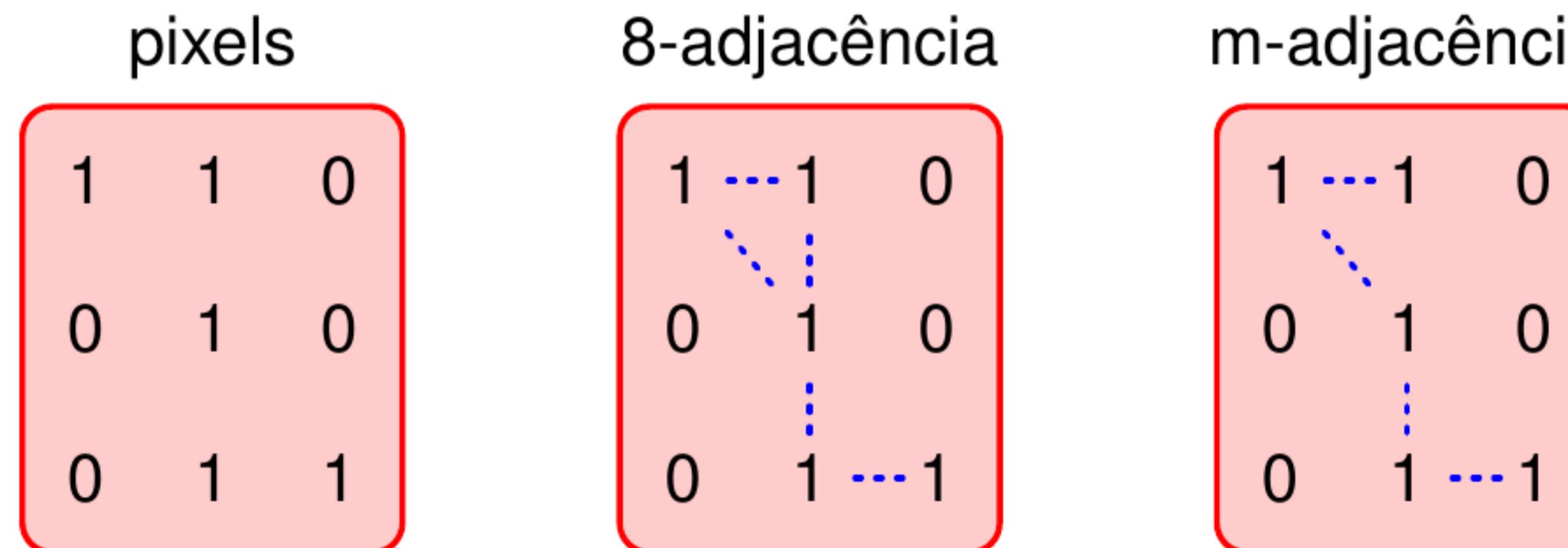
# Relações entre pixels

- As relações de adjacência entre pixels são baseadas em um critério de similaridade (níveis de cinza iguais).
- Seja  $\mathbb{V}$  o conjunto dos níveis de cinza utilizados para definir conectividade.
- Para uma imagem binária,  $\mathbb{V} = \{1\}$ .
- Para uma imagem em tons de cinza, o conjunto é determinado por uma faixa de tons de cinza. Por exemplo,  $\mathbb{V} = \{16, 17, \dots, 32\}$ .

**conectividade 4** - dois pixels p e q com valores de  $\mathbb{V}$  são 4-conectados se q está em  $N_4(p)$ .

**conectividade 8** - dois pixels p e q com valores de  $\mathbb{V}$  são 8-conectados se q está em  $N_8(p)$ .

**conectividade mista** - dois pixels p e q com valores de  $\mathbb{V}$  são m-conectados se q está em  $N_4(p)$  e  $N_D(p) \cup N_4(q)$  não contenha pixels pertencentes a  $\mathbb{V}$ .



- A rotulação de componentes conectadas exerce importante papel no processamento automático de imagens binárias. Permite atribuir a cada componente um rótulo para diferenciá-la das outras componentes na imagem.

---

## Algoritmo 1 floodfill(x, y, rotulo)

---

- 1: Iniciar pilha de posições com a posição  $(x, y)$  do pixel p
- 2: **while** Pilha não estiver vazia **do**
- 3:     Retirar elemento da pilha.
- 4:     Para cada ponto  $p'$  vizinho do pixel p.
- 5:         **if**  $p'(x', y') = 255$  **then**
- 6:             Empilhar a sua posição  $(x', y')$  na pilha de posições.
- 7:         **end if**
- 8:     Alterar o valor do pixel  $(x, y)$  para o valor do rótulo.
- 9: **end while**

---

# Rotulação de componentes conectadas

- Assumindo que os pixels de fundo têm valor igual a 0 e os pixels das várias componentes têm valor igual a 255, algoritmo de rotulação é o seguinte:

---

## Algoritmo 2 labeling

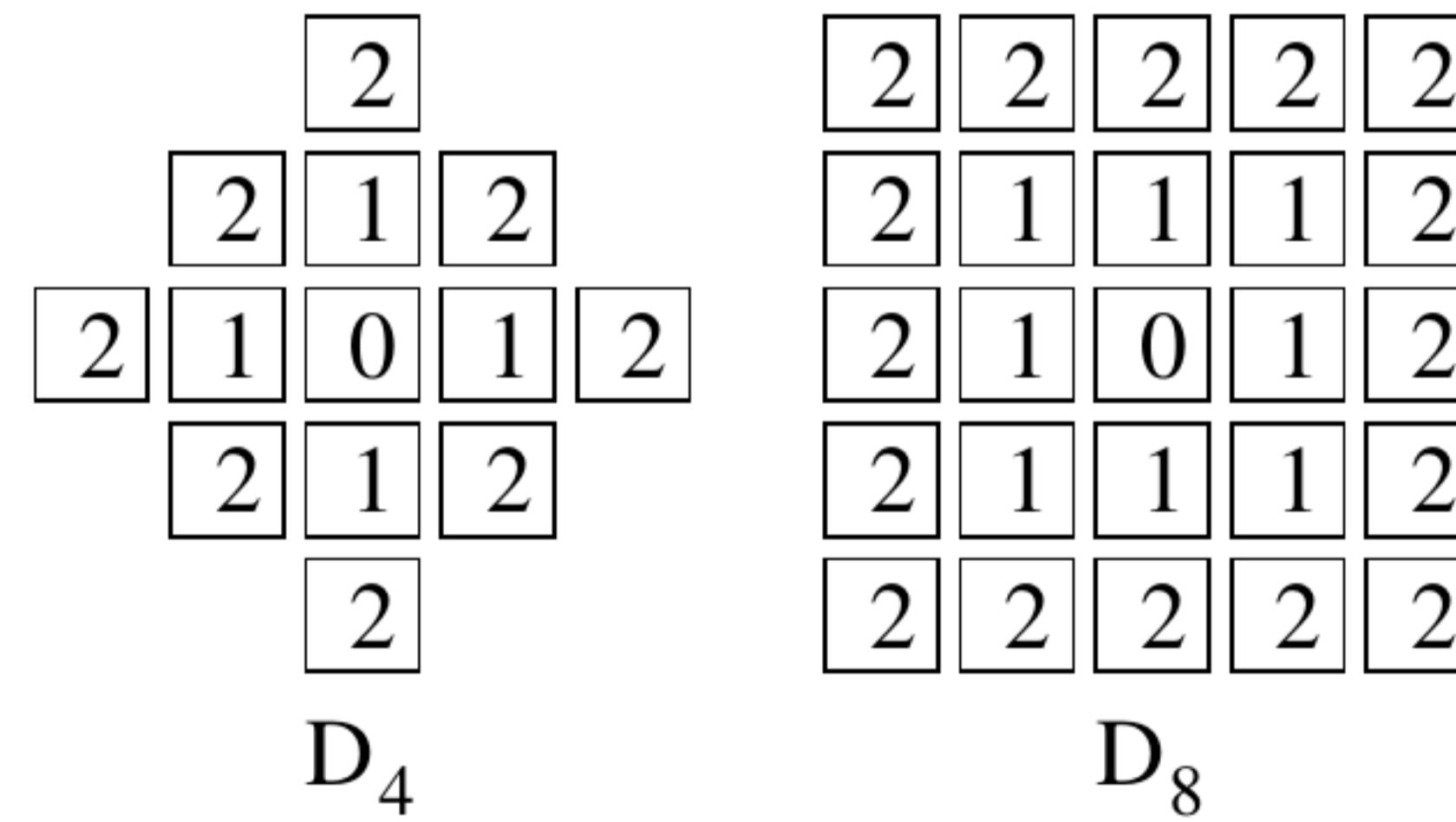
---

```
1: rotulo = 1
2: for  $x \leftarrow 0, nlinhas - 1$  do
3:   for  $y \leftarrow 0, ncolunas - 1$  do
4:     if  $p(x, y) = 255$  then
5:       FLOODFILL( $x, y, rotulo$ )
6:       rotulo = rotulo + 1
7:     end if
8:   end for
9: end for
```

---

# Distância entre pixels

Designação	Representação	Cálculo
Euclidiana	$D_e[p(x, y), q(s, t)]$	$\sqrt{(x - s)^2 + (y - t)^2}$
Quarteirão	$D_4[p(x, y), q(s, t)]$	$\ x - s\  + \ y - t\ $
Tabuleiro	$D_8[p(x, y), q(s, t)]$	$\max(\ x - s\ , \ y - t\ )$



## Translação

- deslocamento de um ponto com coordenadas  $(x, y)$  para uma nova localização  $(x', y')$  por  $(x_0, y_0)$  pixels, ou seja,

$$x' = x + x_0$$

$$y' = y + y_0$$

- O pixel na nova posição  $(x', y')$  assumirá a cor do pixel que existia na posição  $(x, y)$  antes da operação de translação.
- Costuma-se geralmente representar as transformações em coordenadas homogêneas, uniformizando-as em uma matriz. Para a translação, é feita a

representação

$$\begin{bmatrix} x' \\ y' \\ s \end{bmatrix} = \begin{bmatrix} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- A matriz de translação em coordenadas homogêneas será então

$$T = \begin{bmatrix} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Em coordenadas cartesianas, ter-se-á o ponto

$$x_{cart} = x'/s$$

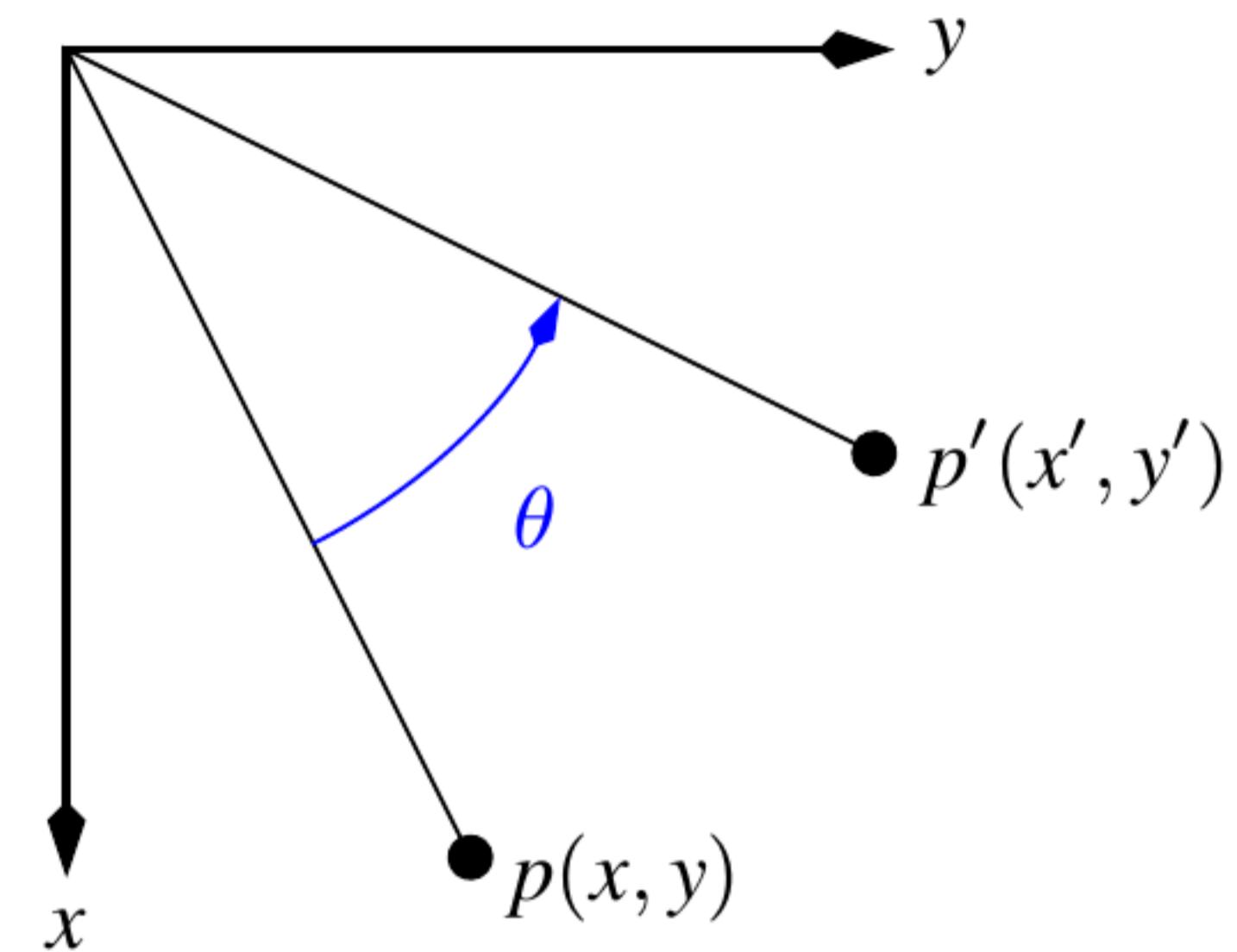
$$y_{cart} = y'/s$$

## Escalamento

$$T = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

## Rotação

$$T = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$



- O mapeamento das transformações deve ser feito no sentido inverso
- Para cada pixel  $p'(x', y')$  da imagem rotacionada, procura-se sua posição na imagem original,  $p(x, y)$  e atribui-se o valor deste pixel à imagem processada.
- Se o mapeamento for realizado no sentido direto, a imagem resultante poderá conter pontos com valores indefinidos.



Obrigado