

# Processamento e Análise de Imagens

## Segmentação de Imagens

Felipe Augusto Lima Reis



**PUC Minas**

# Introdução

# Introdução

- **“A segmentação subdivide uma imagem em regiões ou objetos que a compõem” (Gonzalez & Woods, 2010)**
  - A segmentação particiona imagens em estruturas com conteúdo semântico relevante para a aplicação, que auxiliarão o processo de interpretação das imagens;
  - O nível de detalhe em que a subdivisão é realizada depende do problema a ser resolvido;
  - A segmentação deve parar quando os objetos ou as regiões de interesse de uma aplicação forem identificados na imagem.

# Introdução

- Tradicionalmente, a segmentação é desenvolvida com base nas seguintes características:
  - **Descontinuidade** → divide-se uma imagem com base em mudança bruscas de intensidades, como as bordas;
  - **Similaridade** → divide-se a imagem com base na similaridade de um conjunto critérios predefinidos (ex.: cor e adjacência).

# Introdução

- Processar uma imagem de modo a segmentar um número de objetos é uma tarefa difícil e extremamente dependente da correta extração de características dos objetos, especialmente em imagens ruidosas;
- Após a segmentação, cada objeto é descrito por meio de suas propriedades geométricas e topológicas (ex.: área, forma e textura, que podem ser utilizados nas etapas de análise e entendimento de imagens).

# Introdução

- O processo de análise de imagens pode ser dividido em:
  - **Supervisionado** → quando existem regiões da imagem em que se dispõe de informações que permitem a identificação de uma ou mais classes de interesse (*groundtruth*)
    - O processo utiliza a informação previamente cadastrada para aprender as regiões de interesse;
  - **Não supervisionado** → quando a região é associada a uma classe que é aprendida com base na similaridade entre as amostras de treinamento, ou seja, são utilizadas técnicas para reconhecer as classes presentes na imagem.

# Definição

# Definição

- “A segmentação subdivide uma imagem em regiões ou objetos que a compõem” (Gonzalez & Woods, 2010)



Fonte: Arbelaez et al (2011).



## Definição

- Considerando  $R$  correspondente a região ocupada por uma imagem, uma segmentação particiona  $R$  em  $n$  subregiões  $R_1, R_2, \dots, R_n$ , sujeita a propriedades lógicas  $Q(R)$ , tal que:

$$(a) \bigcup_{i=1}^n R_i = R$$

(b)  $R_i$  é um conjunto conectado,  $i = 1, 2, \dots, n$ .

(c)  $R_i \cap R_j = \emptyset$  para todo  $i$  e  $j$ ,  $i \neq j$

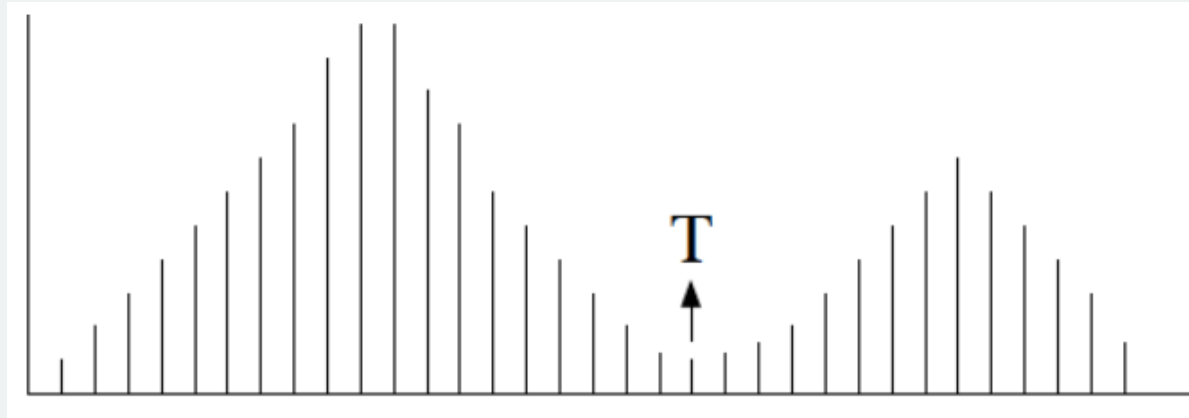
(d)  $Q(R_i) = \text{VERDADEIRA}$  para  $i = 1, 2, \dots, n$ .

(e)  $Q(R_i \cup R_j) = \text{FALSA}$  para quaisquer regiões adjacentes  $R_i$  e  $R_j$ .

# Limiarização (Thresholding)

# Limiarização

- A limiarização é uma técnica simples de segmentação que consiste na classificação dos pixels de uma imagem de acordo com a especificação de um ou mais limiares.
- Uma maneira óbvia de extrair os objetos do fundo é por meio da seleção de um limiar  $T$  que separe os dois grupos.



# Limiarização

- Seja o histograma de níveis de cinza correspondente a uma imagem  $f(x, y)$ , de maneira que os pixels dos objetos e os do fundo tenham seus níveis de cinza separados em dois grupos dominantes.
- Então, cada ponto  $(x, y)$  tal que  $f(x, y) > T$  é denominado um ponto do objeto; caso contrário, o ponto é denominado um ponto do fundo.
- A imagem limiarizada  $g(x, y)$  pode ser definida como:

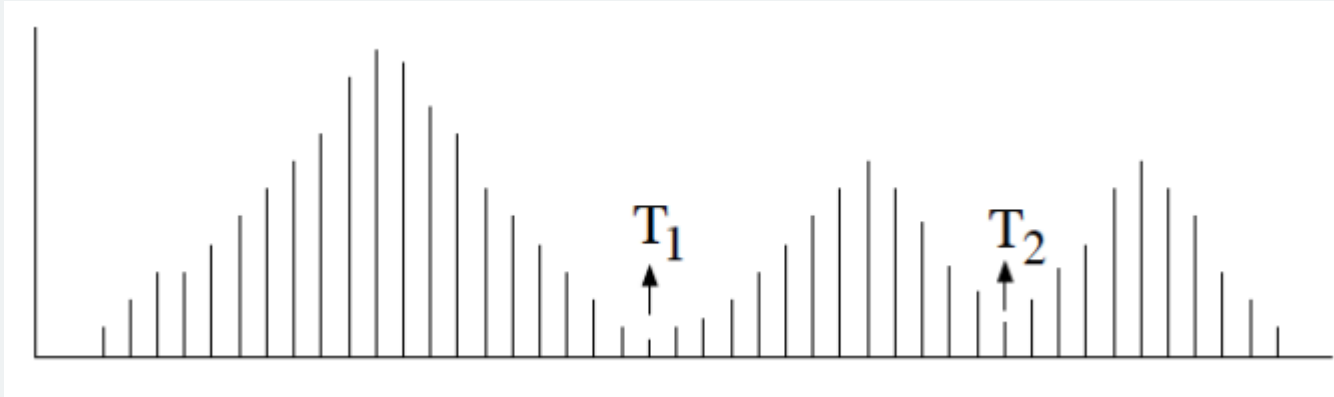
$$g(x, y) = \begin{cases} 0, & \text{se } f(x, y) \leq T \\ 1, & \text{se } f(x, y) > T \end{cases}$$

# Limiarização

- Pixels rotulados como 1 (ou qualquer outro nível de cinza conveniente) correspondem aos objetos, enquanto que aqueles rotulados como 0 correspondem ao fundo;
- Nesse caso, a limiarização é denominada binarização, pois a imagem resultante possui apenas dois valores de intensidade, 0 (preto) ou 1 (branco).

# Limiarização

- Um caso ligeiramente mais geral dessa abordagem, em que três grupos dominantes caracterizam o histograma da imagem (por exemplo, dois tipos de objetos iluminados sobre um fundo escuro).



Fonte: Adaptado de Guimarães (2020).

# Limiarização

- A limiarização da imagem pode ser definida como:

$$g(x, y) = \begin{cases} l_1, & \text{se } f(x, y) \leq T_1 \\ l_2, & \text{se } T_1 < f(x, y) \leq T_2 \\ l_3, & \text{se } f(x, y) > T_2 \end{cases}$$

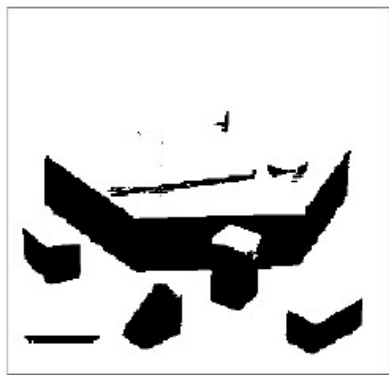
tal que, para cada intervalo, é especificado um nível de cinza correspondente.

# Limiarização

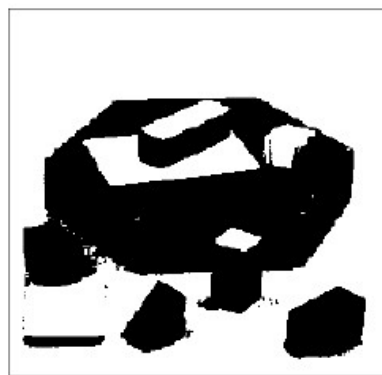
- A seleção correta do valor de limiar é fundamental para que o processo de segmentação baseada na limiarização produza bons resultados.



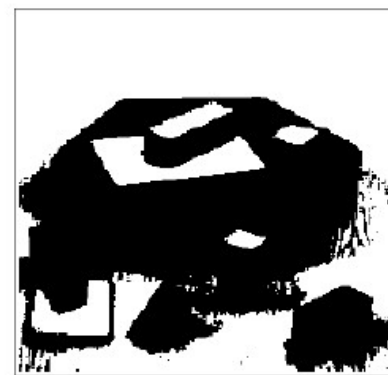
(a) imagem original



(b)  $T = 108$



(c)  $T = 179$



(d)  $T = 213$



# Limiarização

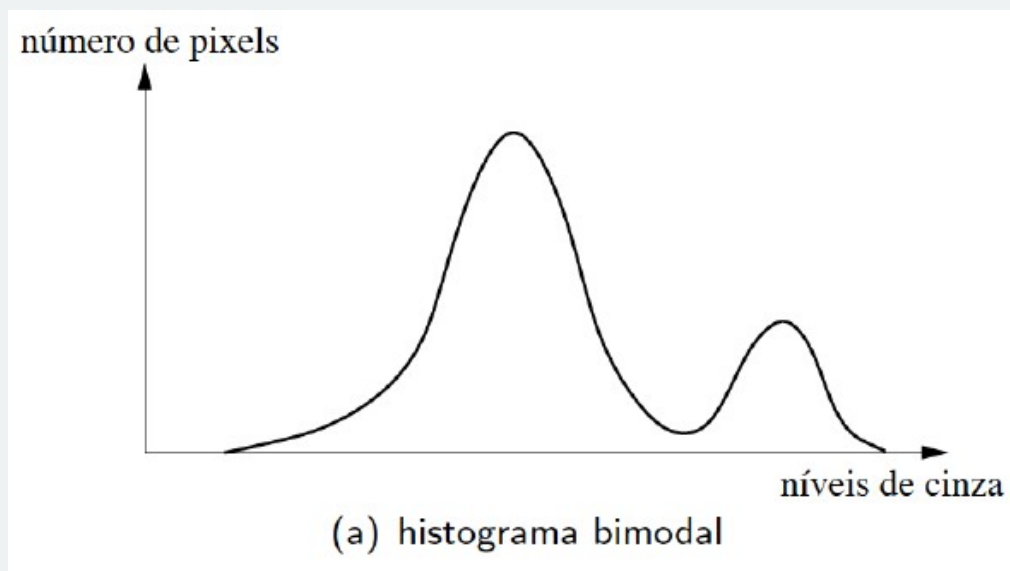
- **Limiarização global** → utilizar um único limiar para segmentar imagens não é, em geral, adequado.
  - As imagens podem conter variações nos níveis de cinza dos objetos e do fundo por causa da iluminação não-uniforme, ruído, parâmetros do dispositivo de aquisição não-uniformes ou outros fatores.
- **Limiarização local** → valores de limiar podem variar sobre a imagem como uma função de suas características locais.

## Limiarização Global Simples

- A maneira mais simples e direta de selecionar um valor global de limiar é a partir da distribuição das intensidades dos pixels na imagem.
- Para uma imagem constituída de objetos com níveis de cinza aproximadamente iguais, diferindo apenas do nível de cinza do fundo, o histograma possuirá dois picos distintos
  - O primeiro pico será formado pelos pixels dos objetos, enquanto o outro, pelos pixels do fundo.

# Limiarização Global Simples

- Um histograma com tal distribuição é chamado bimodal.



Fonte: Guimarães (2020).

# Limiarização Global Simples

- Intuitivamente, o valor do limiar corresponde ao nível de cinza que melhor separa os dois picos.
- Vários métodos têm sido propostos para encontrar os pontos de máximo (picos) e, então, selecionar o valor de limiar como o ponto de mínimo (vale) entre eles.
- Há várias desvantagens em selecionar o limiar global por meio da distribuição das intensidades dos pixels.
  - Nem sempre as intensidades dos objetos e do fundo da imagem são bem distintas, devido a ocorrência de baixo contraste ou ruído.
  - Além disso, a determinação de picos e vales não é um problema trivial. Por exemplo, muitos pontos de mínimo ou máximo locais podem existir na imagem.

## Limiarização de Otsu

- O método de Otsu (1979) considera que o histograma de uma imagem é composto de duas classes: os pixels do objeto ou do fundo.
  - Cada classe possui suas próprias características, ou seja, sua média e desvio padrão.
  - A variância  $\sigma^2_T$  e a média  $\mu_T$  globais da imagem são calculadas.
  - O método procura maximizar a razão  $\eta(T)$  da variância entre as classes  $\sigma^2_B$  em relação à variância total, para todos os valores possíveis de limiar  $T$ .

## Limiarização de Otsu

- O método procura maximizar a razão  $\eta(T)$  da variância entre as classes  $\sigma_B^2$  em relação à variância total, para todos os valores possíveis de limiar  $T$ , sendo:

$$\eta(T) = \frac{\sigma_B^2}{\sigma_T^2}$$

## Limiarização de Otsu

- Formalmente, temos:

$$\begin{aligned} \sigma_T^2 &= \sum_{i=0}^{L-1} (i - \mu_T)^2 p_i & \mu_T &= \sum_{i=0}^{L-1} i p_i & \sigma_B^2 &= \omega_1 \omega_2 (\mu_1 \mu_2)^2 & \omega_1 &= \sum_{i=0}^T p_i \\ \omega_2 &= 1 - \omega_1 & \mu_1 &= \frac{\mu_S}{\omega_1} & \mu_2 &= \frac{\mu_T - \mu_S}{\omega_2} & \mu_S &= \sum_{i=0}^T i p_i & \sum_{i=0}^{L-1} p_i &= 1, & p_i &= \frac{n_i}{n} \end{aligned}$$

- em que  $n_i$  é o número de pixels com nível de cinza  $i$ ,  $n$  é o número total de pixels da imagem,  $p_i$  representa a probabilidade do nível de cinza  $i$  ser encontrado na imagem e  $L$  é o número de níveis de cinza da imagem.

## Limiarização de Otsu

- A razão  $\eta(T)$  é calculada para todos os valores possíveis de T.
- O limiar ótimo pode ser determinado como:

$$T = \operatorname{argmax}_T \eta(T)$$



## Limiarização de Otsu

- O método de Otsu possui bom desempenho em imagens com maior variância de intensidade.
- Uma desvantagem é que o método assume que o histograma da imagem seja bimodal.

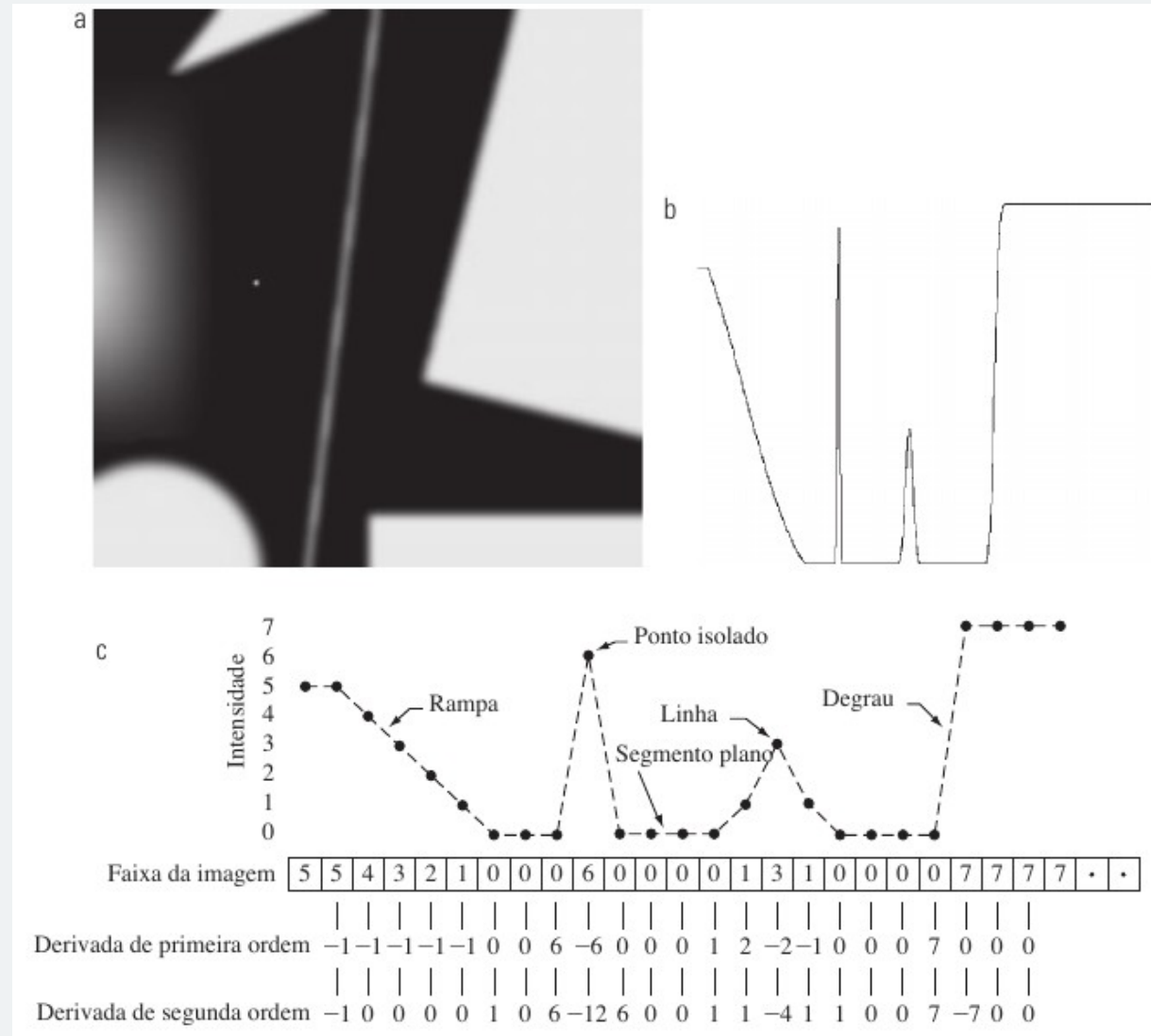
# Detecção de Descontinuidades

# Detecção de Descontinuidades

- Detecção de descontinuidades busca identificar mudanças locais abruptas de intensidade, possibilitando identificação de pontos, linhas, junções e bordas
  - **Pixels de borda** são *pixels* cuja intensidade de uma função da imagem muda abruptamente;
  - **Bordas** são conjuntos de pixels de borda conexos, que separam duas ou mais regiões de uma imagem;
  - **Linha** pode ser definida como uma borda em que a intensidade do fundo de cada lado da linha ou é muito superior/inferior à intensidade dos pixels da linha;
  - **Junções** são pontos de intersecção entre dois ou mais segmentos de borda.

# Detecção de Descontinuidades

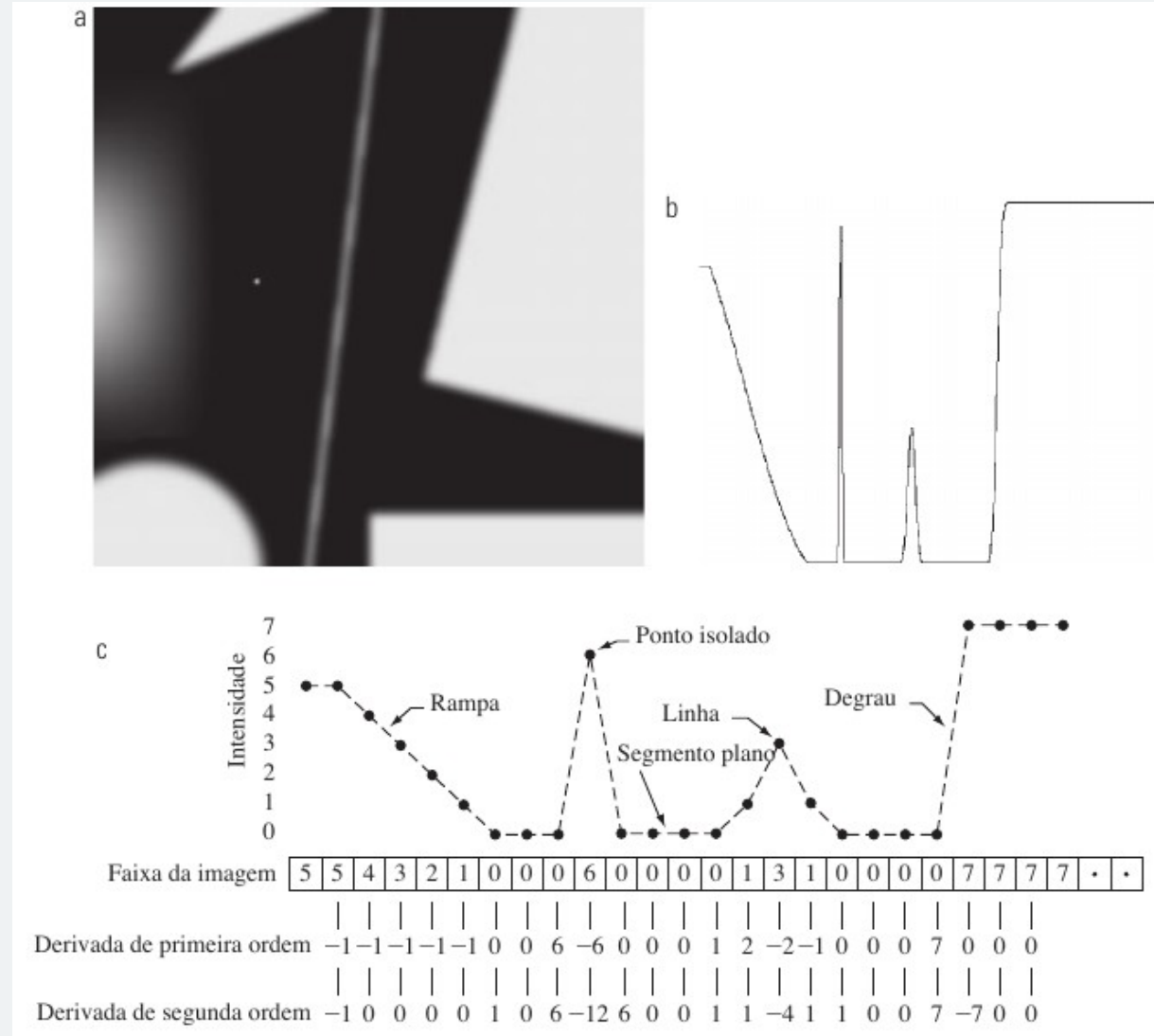
- Um exemplo de detecção de pontos, linhas, bordas e segmentos pode ser vista na figura ao lado.



Fonte: Gonzalez & Woods (2018).

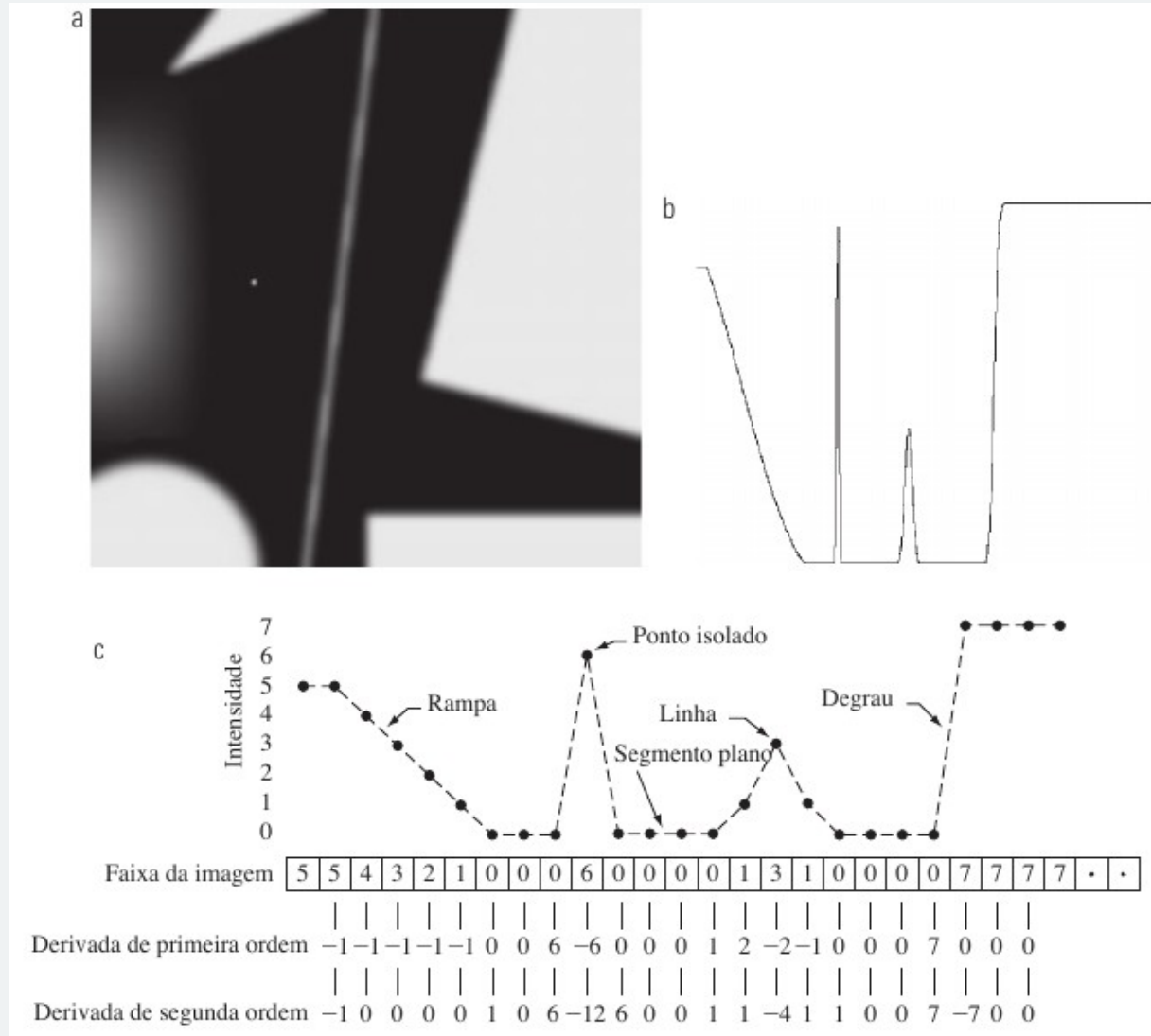
# Detecção de Descontinuidades

- Pode-se observar que as derivadas de primeira e segunda ordem podem ser utilizadas para identificação de descontinuidades.
- A derivada primeira é diferente de zero nas transições e na rampa;
- A derivada segunda ordem é mais agressiva e precisa, diferente de zero somente nas transições;



# Detecção de Descontinuidades

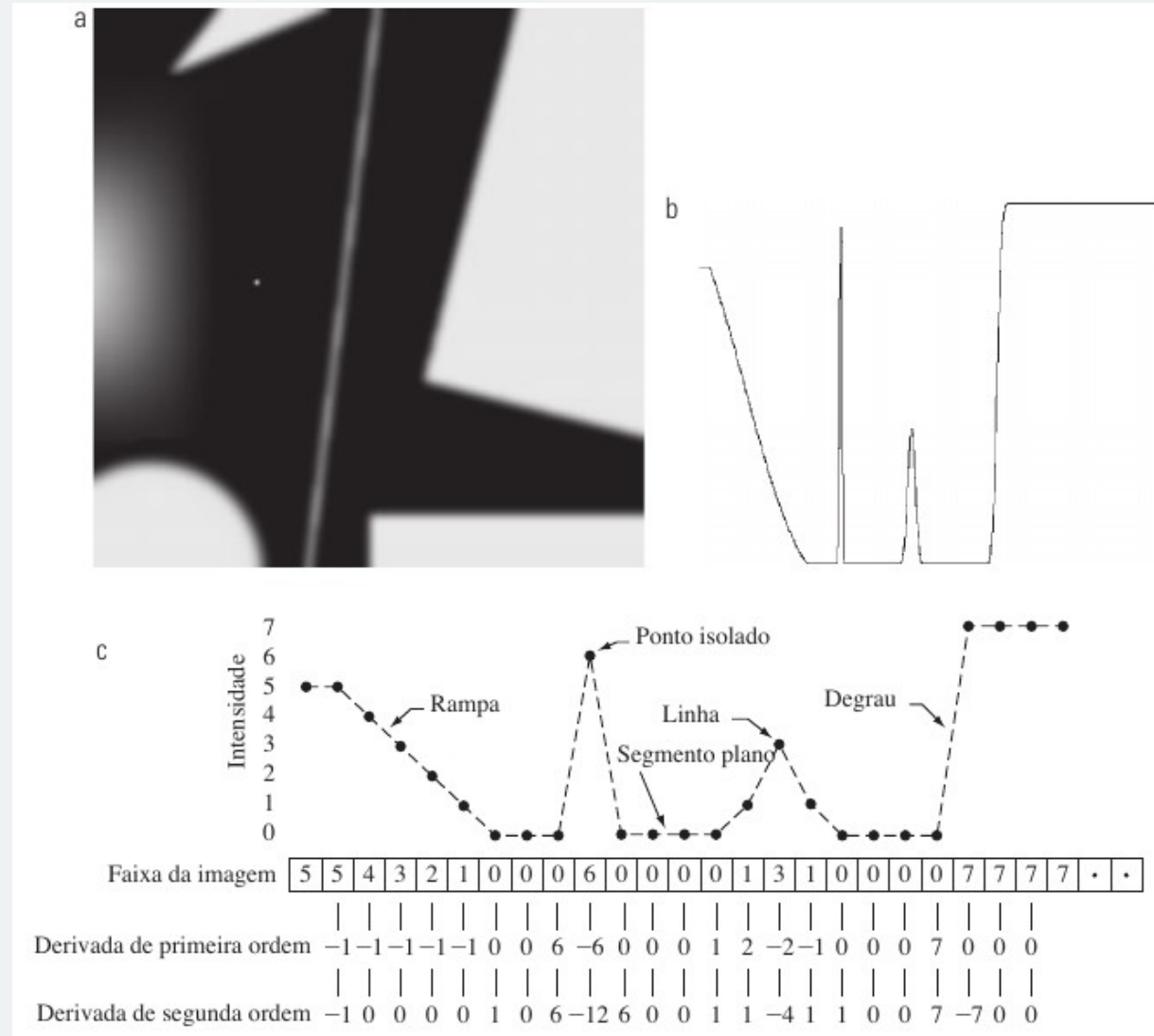
- A derivada primeira produz bordas mais grossas enquanto a derivada segunda produz bordas mais finas.
- Pode-se observar ainda que a magnitude da derivada segunda é muito maior, facilitando a identificação e separação das descontinuidades;



Fonte: Gonzalez & Woods (2018).

# Detecção de Descontinuidades

- A derivada segunda melhora pequenos detalhes (inclusive ruídos) de forma mais eficiente;
- Por fim, pode-se observar que a derivada segunda possui valores opostos nos pontos em degrau
  - Tal característica permite a conhecimento do tipo de variação do sinal – de escuro para claro ou vice-versa.



# Detecção de Descontinuidades

- Para cálculo de derivadas, é comum a varredura da imagem por um filtro espacial (máscara), em um processo de convolução / correlação;
- No caso de uma máscara  $w$  com tamanho de  $3 \times 3$  *pixels* esse procedimento envolve o cálculo da soma dos produtos dos coeficientes da máscara pelos níveis de cinza da região delimitada pela máscara;
- A resposta da máscara é definida em relação à sua posição central
  - Quando a máscara é posicionada em um pixel na fronteira da imagem, a resposta é calculada, utilizando-se a vizinhança parcial apropriada.



# Detecção de Descontinuidades

- A variação de intensidade em imagens está representada na figura ao lado.
- A variação suave, indicada na Figura C, possui como vantagem a continuidade, possibilitando a derivação em todos os pontos da imagem.

Fonte: Guimarães (2020).



(a) imagens contendo uma região clara e uma região escura



(b) perfil mostrando variação abrupta de intensidade de cinza ao longo de uma linha de varredura horizontal

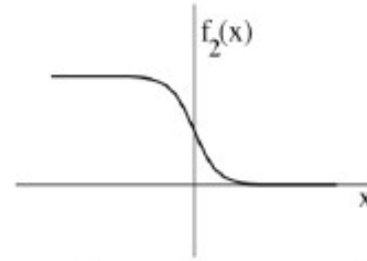


(c) perfil mostrando variação suave dos níveis de cinza

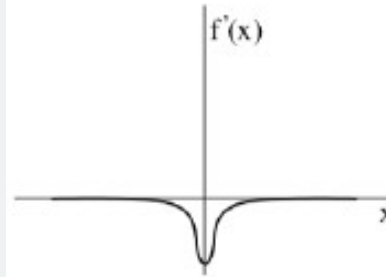
# Detecção de Descontinuidades

- A derivada primeira é positiva nas transições da região escura para clara, negativa nas transições da região clara para escura e nula nas áreas de nível de cinza constante;
- A derivada segunda é positiva na parte da transição associada ao lado escuro da borda, negativa na parte da transição associada ao lado claro da borda e nula nas áreas de nível de cinza constante.

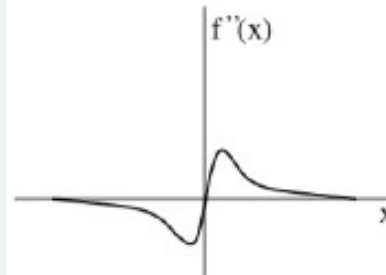
Fonte: Guimarães (2020).



(c) perfil mostrando variação suave dos níveis de cinza



(d) derivada primeira do perfil de níveis de cinza



(e) derivada segunda do perfil de níveis de cinza

## Detecção de Descontinuidades

- A magnitude da derivada primeira pode ser utilizada na detecção de uma borda em uma imagem, enquanto o sinal da derivada segunda possui um cruzamento em zero, ou seja, uma indicação de que há uma mudança de sinal na transição dos níveis de cinza, permitindo a localização das bordas em uma imagem.
- A derivada primeira em qualquer ponto da imagem é obtida usando-se a magnitude do **gradiente** naquele ponto.
- A derivada segunda é obtida similarmente utilizando-se o operador **Laplaciano**.

# Detecção de Pontos

- A detecção de pontos isolados em uma imagem pode ser realizada pela aplicação direta da máscara  $h$ , definida como:

1	1	1
1	-8	1
1	1	1

## Detecção de Pontos

- Um ponto é detectado na posição central da máscara se  $|R| > T$  em que  $T$  é um limiar não-negativo e  $R$  é dado pela equação:

$$R = \omega_1 z_1 + \omega_2 z_2 + \cdots + \omega_9 z_9 = \sum_{k=1}^9 \omega_k z_k$$

- Um ponto é detectado se houver uma discrepância entre seu valor de nível de cinza e de seus vizinhos.
- Quando a máscara  $h$  é posicionada sobre uma região homogênea da imagem, ou seja, todos os pixels pertencentes à região possuem a mesma intensidade, a resposta da máscara é nula ( $R = 0$ ).

# Detecção de Linhas e Retas

- Segmentos de retas também podem ser detectados pelo uso de máscaras
  - Diferentes máscaras podem ser utilizadas para detecção de retas em múltiplas orientações.

-1	-1	-1
2	2	2
-1	-1	-1

Horizontal

2	-1	-1
-1	2	-1
-1	-1	2

+45°

-1	2	-1
-1	2	-1
-1	2	-1

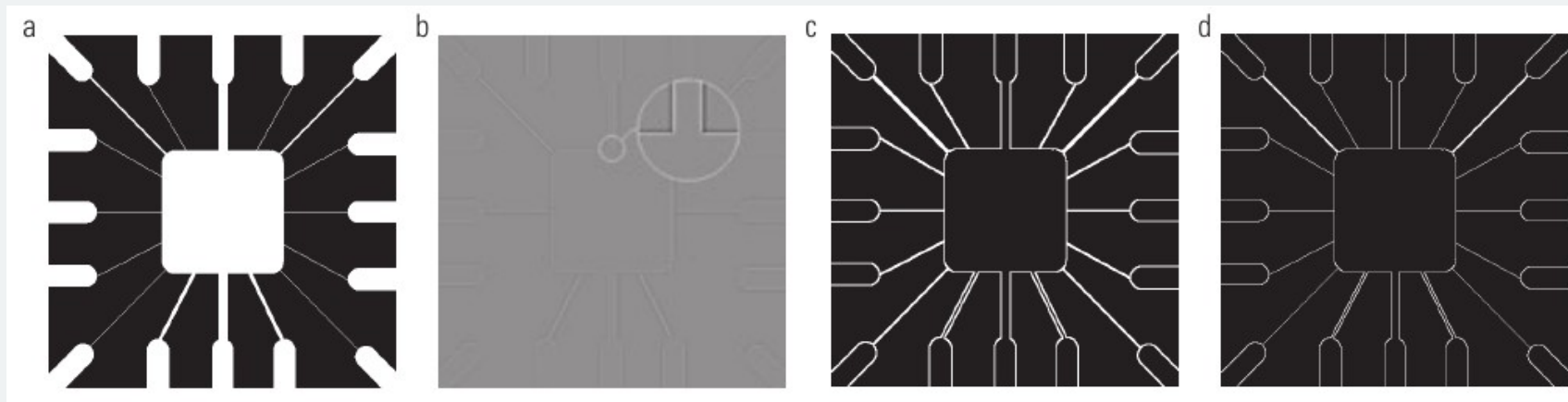
Vertical

-1	-1	2
-1	2	-1
2	-1	-1

-45°

## Detecção de Linhas e Retas

- A detecção de linhas e retas pode ser feita com auxílio do filtro Laplaciano.
- Na imagem abaixo (b), o cinza médio representa o valor zero, os tons mais escuros representam valores negativos e tons claros, valores positivos.
- As imagens (c) e (d) representam os valores absolutos e positivos do filtro.



# Detecção de Bordas



# Detecção de Bordas

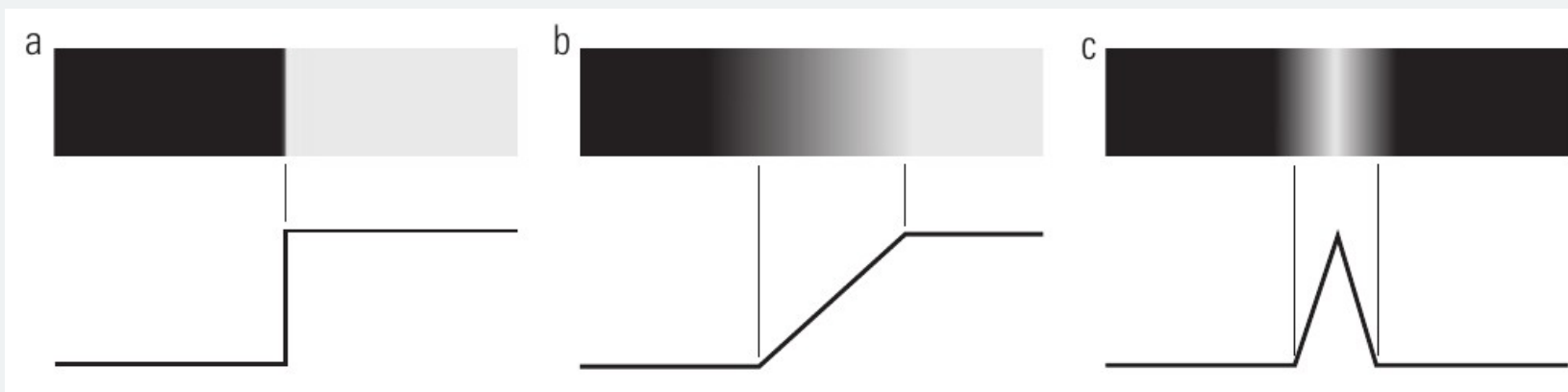
- **Uma borda é o limite ou a fronteira entre duas regiões** com propriedades relativamente distintas de nível de cinza
  - É utilizado como método de segmentação de imagens com base na variação abrupta de intensidade;
- Assume-se que as regiões em questão são suficientemente homogêneas, de maneira que a transição entre duas regiões pode ser determinada com base apenas na descontinuidade dos níveis de cinza.
- Basicamente, a ideia associada à maioria das técnicas para detecção de bordas é o cálculo de um operador local diferencial.

# Detecção de Bordas

- Bordas podem ser classificadas de acordo com seu perfil de intensidade
  - Uma **borda em degrau** (ideal) envolve uma transição entre dois níveis de intensidade que ocorrem idealmente com uma distância de 1 pixel;
  - Uma **borda em rampa** é aquela que possui uma sequência de pontos (segmento de borda), em geral desfocados e ruidosos, cujo um ponto de borda agora é qualquer ponto contido na rampa;
  - Uma **borda em forma de telhado** é um modelo de linhas em uma região, com a base (largura) de uma borda em forma de telhado determinada pela espessura e a nitidez da linha.

# Detecção de Bordas

- Os tipos de borda estão detalhados na imagem abaixo:
  - a) Borda em degrau;
  - b) Borda em rampa;
  - c) Borda em forma de telhado.

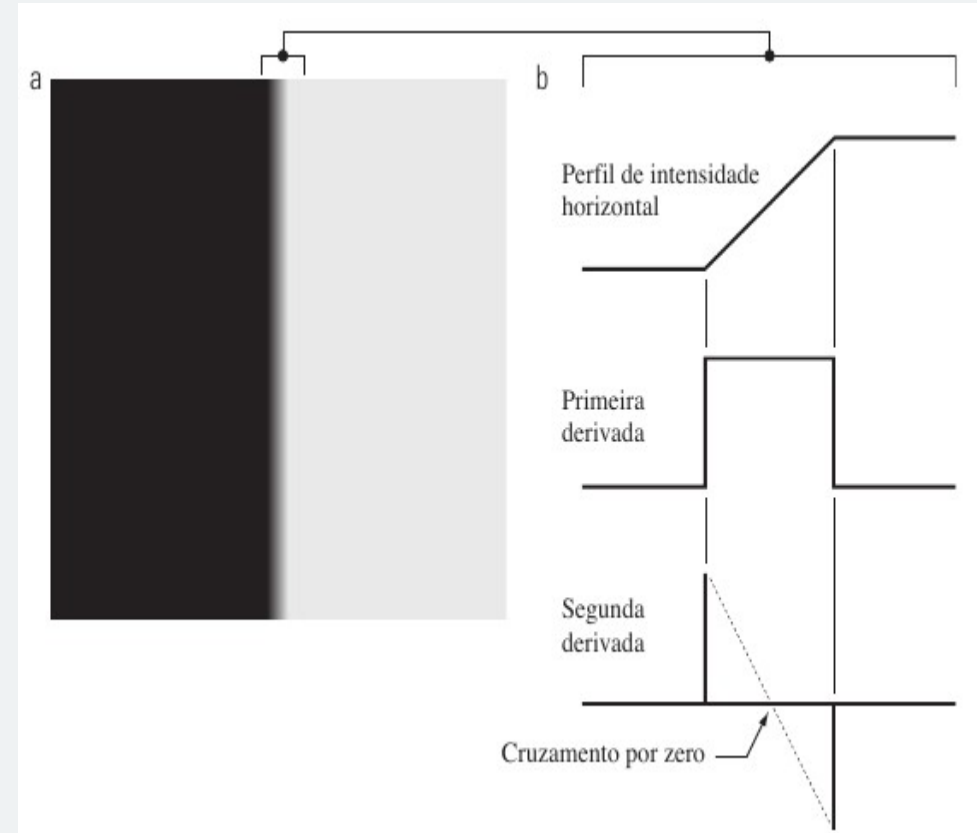


# Detecção de Bordas

- A produção do tipo de borda está relacionado ao método de produção das imagens:
  - a) **Borda em degrau**: imagens geradas por computador, comum em modelagem de sólidos e animações;
  - b) **Borda em rampa**: normalmente geradas por limitações no mecanismo de foco da captura de imagem - nível de ruído é determinado pelos componentes eletrônicos do sistema de imagens;
  - c) **Borda em forma de telhado**: surgem, por exemplo, nas imagens em profundidade, quando os objetos finos estão mais próximos do sensor do que seu fundo equidistante, fazendo com que objetos finos pareçam mais claros.

# Detecção de Bordas

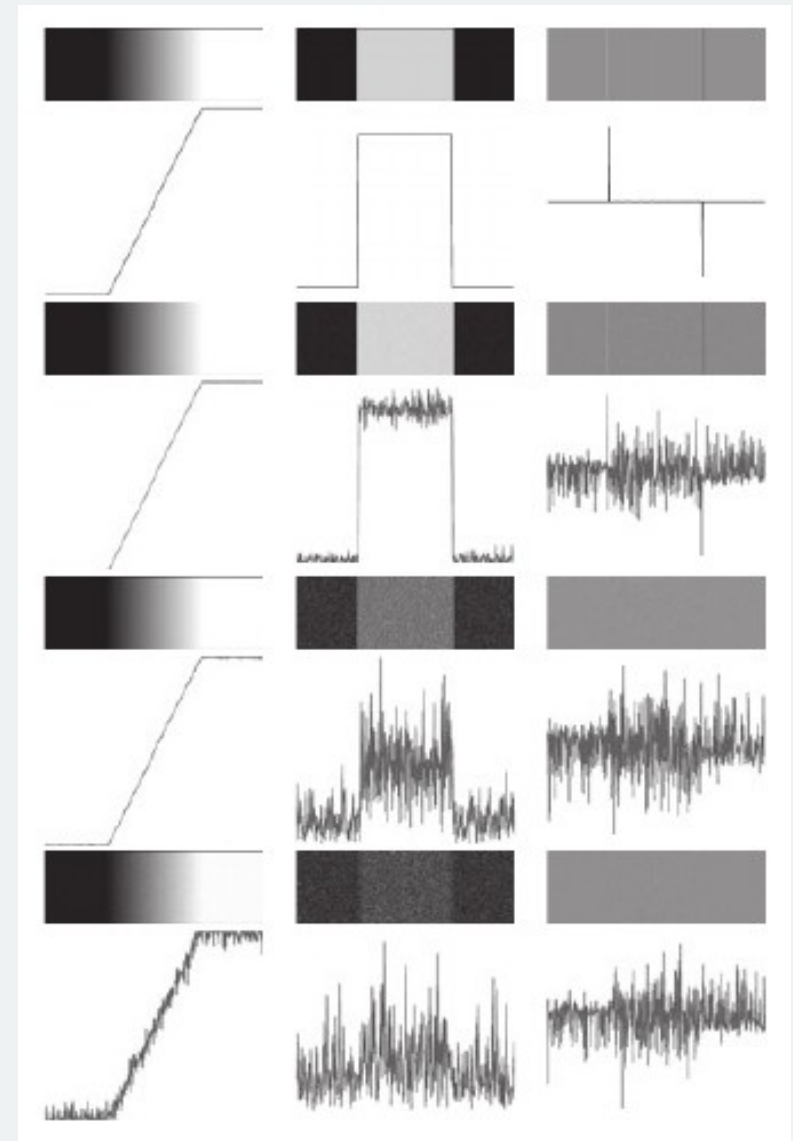
- A segunda derivada possui as seguintes características na detecção de bordas, em especial das bordas em rampa pois:
  - a) produz dois valores para cada borda em uma imagem;
  - b) seus cruzamentos por zero podem ser usados para localizar o centro de bordas espessas, como mostraremos mais adiante nesta seção.



# Detecção de Bordas

- A segunda derivada, no entanto, possui um problema grave: é sensível ao ruído
  - A primeira coluna contém uma borda em declive corrompida por ruído gaussiano aleatório de média zero e desvio padrão 0,0, 0,1, 1,0 e 10,0 níveis de intensidade;
  - A segunda coluna possui a derivada primeira e seus níveis de intensidade;
  - A segunda coluna possui a derivada segunda e seus níveis de intensidade.

Fonte: Gonzalez & Woods (2018).



# Detecção de Bordas

- De forma geral, o processo de detecção de bordas é composto pelas seguintes etapas fundamentais:
  - a) **Suavização da imagem para redução do ruído**: aplicação, em geral, de filtros espaciais à imagem;
  - b) **Detecção dos pontos de borda**: operação local que tira de uma imagem todos os pontos que são candidatos potenciais a se tornarem pontos de borda;
  - c) **Localização da borda**: seleciona, dentre os possíveis pontos de borda, apenas aqueles que de fato pertencem ao conjunto de pontos que formam uma borda.

# Detecção de Bordas

## Operadores de Gradiente

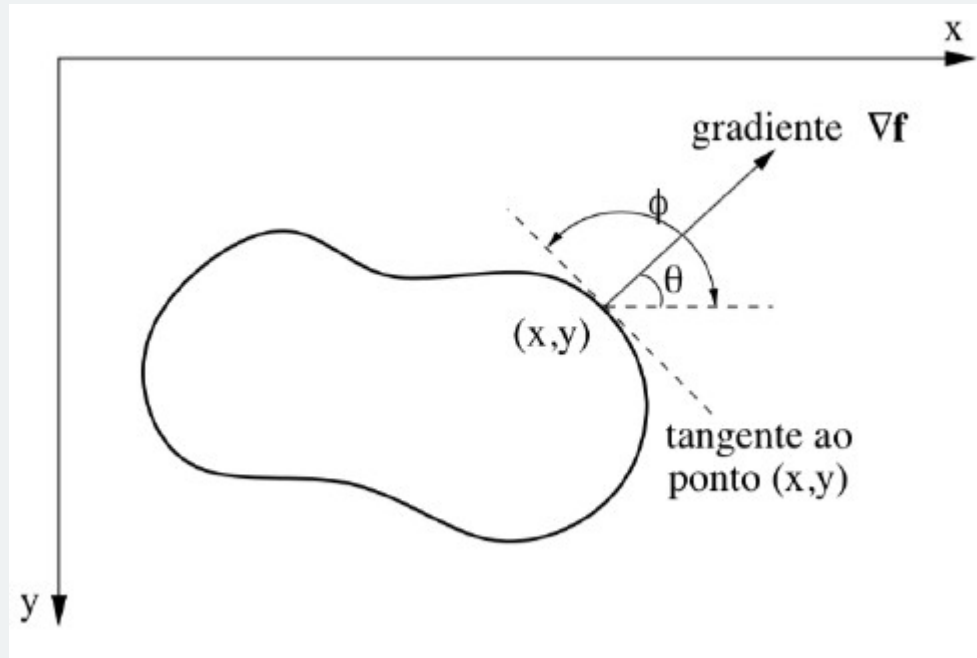


# Operadores de Gradiente

- A detecção de bordas é, essencialmente, a operação de identificação de **mudanças locais** significativas nos níveis de cinza da imagem;
- Essas mudanças podem ser descritas por meio do conceito de derivada;
- Como uma imagem depende de duas coordenadas espaciais, as bordas da imagem podem ser expressas por derivadas parciais;
- Um operador comumente utilizado em diferenciação de imagens é o **gradiente**, que é um vetor cuja direção indica os locais nos quais os níveis de cinza sofrem maior variação.

# Operadores de Gradiente

- Direção do gradiente é sempre perpendicular à direção tangente da borda.



# Operadores de Gradiente

- O **gradiente**, denotado por  $\nabla f$ , é uma ferramenta simples para encontrar a força (intensidade) e a direção da borda na posição  $(x, y)$  de uma imagem  $f$ ,
- O gradiente é definido como o vetor:

$$\nabla f = \text{grad}(f) = \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

# Operadores de Gradiente

- O gradiente aponta a direção de maior taxa de variação e a magnitude (tamanho) corresponde ao valor da taxa de variação na direção do vetor gradiente

$$M(x, y) = \text{mag}(\nabla f) = \sqrt{g_x^2 + g_y^2}$$

- A direção do vetor gradiente, medido em relação ao eixo  $x$ , é dada pelo ângulo:

$$\alpha(x, y) = \text{tg}^{-1} \left[ \frac{g_y}{g_x} \right]$$

- A direção de uma borda em um ponto arbitrário  $(x, y)$  é ortogonal à direção,  $\alpha(x, y)$ , do vetor gradiente no ponto.

# Operadores de Gradiente

- O algoritmo ao lado pode ser utilizado para encontrar bordas em uma imagem;
- Para cada pixel da imagem, calcular o gradiente e verificar se a magnitude é superior a um limiar  $T$ .

Fonte: Guimarães (2020).

---

**Algoritmo 1** Determinação de pontos de borda em uma imagem

---

```
1: entradas: uma imagem de entrada  $f$  com dimensões  $M \times N$  pixels e um  
   limiar  $T$ .  
2: for  $x = 0$  até  $M - 1$  do  
3:   for  $y = 0$  até  $N - 1$  do  
4:     // calcular a magnitude do gradiente  $\nabla f(x, y)$   
5:     
$$\nabla f(x, y) = \sqrt{\left(\frac{\partial f(x, y)}{\partial x}\right)^2 + \left(\frac{\partial f(x, y)}{\partial y}\right)^2}$$
  
6:     // efetuar a limiarização  
7:     if  $\nabla f(x, y) > T$  then  
8:        $(x, y)$  é um ponto da borda  
9:     end if  
10:   end for  
11: end for
```

# Operadores de Gradiente

- Devido ao custo computacional de cálculo de valores de gradiente, são utilizadas outras técnicas que requerem menor poder computacional;
  - Uma mudança em intensidade pode ser detectada pela diferença entre os valores de pixels adjacentes;
  - **Bordas verticais** podem ser detectadas pela **diferença horizontal** entre pontos, enquanto **bordas horizontais** podem ser detectadas pela **diferença vertical** entre pontos adjacentes da imagem.

# Operadores de Gradiente

- Seja a região da imagem mostrada na figura abaixo, em que os valores denotam os níveis de cinza dos pixels.

$f(x-1, y-1)$	$f(x, y-1)$	$f(x+1, y-1)$
$f(x-1, y)$	$f(x, y)$	$f(x+1, y)$
$f(x-1, y+1)$	$f(x, y+1)$	$f(x+1, y+1)$

Fonte: Guimarães (2020).

# Operadores de Gradiente

- Para facilitar a compreensão dos cálculos, podemos ainda utilizar a seguinte notação abaixo:

$z_1$	$z_2$	$z_3$
$z_4$	$z_5$	$z_6$
$z_7$	$z_8$	$z_9$

Fonte: Gonzalez & Woods (2018).



## Operadores de Gradiente

- A magnitude do gradiente pode ser aproximada no ponto  $f(x, y)$  de várias maneiras;
- Uma forma simples consiste em usar a diferença  $f(x, y) - f(x+1, y)$  na direção  $x$  e  $f(x, y) - f(x, y+1)$  na direção  $y$ , combinadas como

$$\nabla f \approx \sqrt{[f(x, y) - f(x + 1, y)]^2 + [f(x, y) - f(x, y + 1)]^2}$$

- Outra abordagem para aproximação é usar as diferenças cruzadas:

$$\nabla f \approx \sqrt{[f(x, y) - f(x + 1, y + 1)]^2 + [f(x, y + 1) - f(x + 1, y)]^2}$$

$$\nabla f \approx |f(x, y) - f(x + 1, y + 1)| + |f(x, y + 1) - f(x + 1, y)|$$

## Operadores de Gradiente

- Da mesma forma, as equações anteriores podem ser representadas por  $g_x$  e  $g_y$ , respectivamente nas direções  $x$  e  $y$ :

$$g_x = \frac{\partial f}{\partial x} = (z_9 - z_5) \quad g_y = \frac{\partial f}{\partial y} = (z_8 - z_6)$$

# Gradiente de Roberts

- Esses mesmos cálculos podem ser representados por máscaras de  $2 \times 2$  *pixels*;
- Essas máscaras são chamadas de operadores cruzados de **gradiente de Roberts**.

-1	0
0	1

0	-1
1	0

Fonte: Gonzalez & Woods (2018).

## Operadores de Prewitt

- Uma aproximação para o ponto  $f(x, y)$  usando uma vizinhança de  $3 \times 3$  pixels é dada por:

$$g_x = \frac{\partial f}{\partial x} = (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)$$

$$g_y = \frac{\partial f}{\partial y} = (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)$$

Fonte: Gonzalez & Woods (2018).

# Operadores de Prewitt

- Esses mesmos cálculos podem ser representados por máscaras de  $3 \times 3$  *pixels*;
- Essas máscaras são chamadas de **Operadores de Prewitt**.

-1	-1	-1	-1	0	1
0	0	0	-1	0	1
1	1	1	-1	0	1

Fonte: Gonzalez & Woods (2018).

# Operadores de Sobel

- A utilização de valores na posição central produz a suavização das imagens.

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$g_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

Fonte: Gonzalez & Woods (2018).

# Operadores de Sobel

- Esses mesmos cálculos podem ser representados por máscaras de  $3 \times 3$  *pixels*;
- Essas máscaras são chamadas de **Operadores de Sobel**.

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

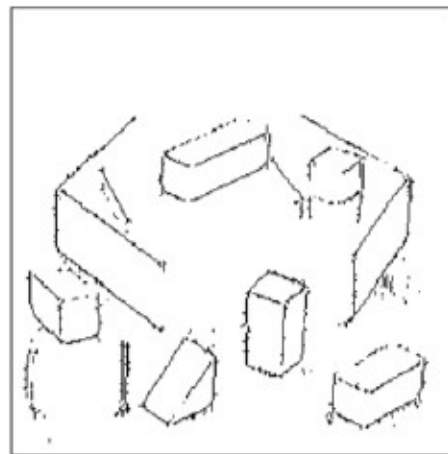
Fonte: Gonzalez & Woods (2018).

# Operadores de Sobel

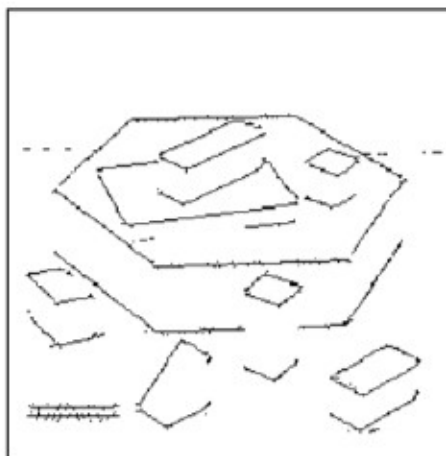
- A imagem ao lado contém a ilustração da detecção de bordas com os operadores de Sobel.



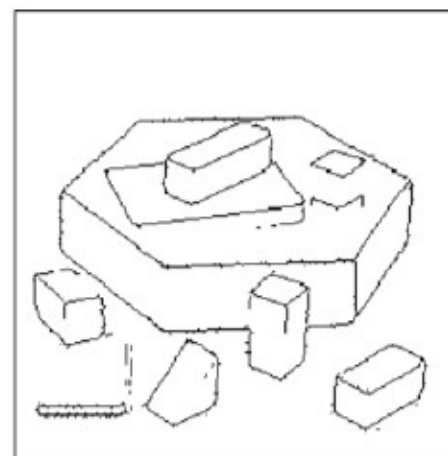
(a) imagem original



(b) resultado de  $G_x$



(c) resultado de  $G_y$



(d) combinação de  $G_x$  e  $G_y$

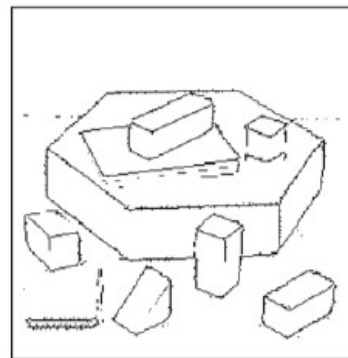


# Operadores de Gradiente

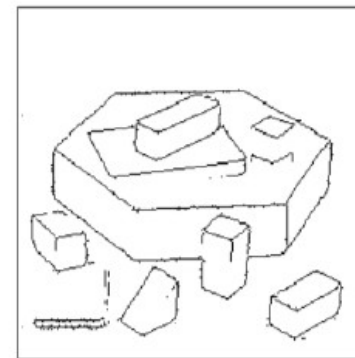
- A imagem ao lado contém a comparação de diferentes operadores de Gradiente, propostos na literatura.



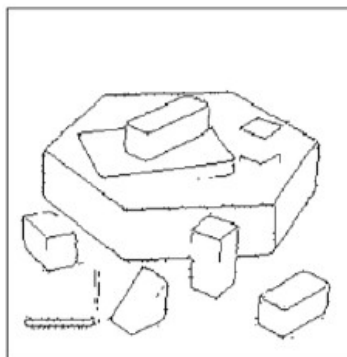
(a) imagem original



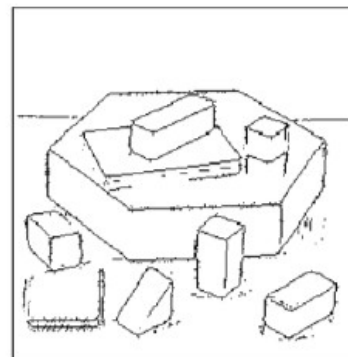
(b) Roberts



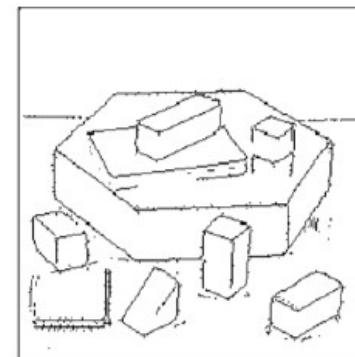
(c) Prewitt



(d) Sobel



(e) Kirsch



(f) Frei-Chen

Fonte: Guimarães (2020).

# Detecção de Bordas

## Operador Laplaciano

# Operador Laplaciano

- O operador Laplaciano de uma função bidimensional contínua  $f(x, y)$  é definido por uma derivada de segunda ordem como:

$$\nabla^2 G(x, y) = \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2}$$

- Assim como no caso do gradiente, o Laplaciano ser aproximada na forma discreta de maneiras diferentes.

## Operador Laplaciano

- O operador Laplaciano pode ser aproximado para uma região de  $3 \times 3$  pixels e também generalizado segundo as equações abaixo:

$$\frac{\partial^2 f}{\partial x^2} \approx f(x-1, y) - 2f(x, y) + f(x+1, y)$$

$$\frac{\partial^2 f}{\partial y^2} \approx f(x, y-1) - 2f(x, y) + f(x, y+1)$$

$$\nabla^2 f \approx f(x-1, y) + f(x+1, y) + f(x, y-1) + f(x, y+1) - 4f(x, y)$$

# Operador Laplaciano

- A exigência para a definição do Laplaciano na forma discreta é que o coeficiente associado ao pixel central seja positivo e que os outros pixels externos sejam negativos;
- A resposta do operador Laplaciano é nula quando o ponto central e seus vizinhos tiverem os mesmos valores de intensidade, ou seja, pertencerem a uma região homogênea da imagem.

# Operador Laplaciano

- A máscara *h1* pode ser usada na implementação tal que as duas matrizes que compõem a máscara correspondem às derivadas segundas ao longo de todas as linhas e colunas, respectivamente, assim como no Laplaciano contínuo.

$$\begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -1 & 0 \\ 0 & 2 & 0 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Fonte: Guimarães (2020).

# Operador Laplaciano

- Em certas situações, é desejável dar maior peso aos pontos vizinhos mais próximos do pixel central;
- Uma aproximação do Laplaciano com tal característica é dada pela máscara  $h_2$ .

-1	-4	-1
-4	20	-4
-1	-4	-1

Fonte: Guimarães (2020).

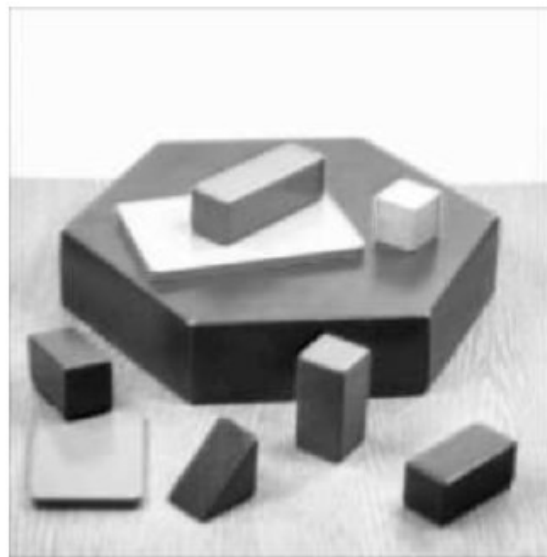
# Operador Laplaciano

- Embora o operador Laplaciano responda a transições de intensidade, ele raramente é utilizado na prática para detecção de bordas;
- Por ser uma derivada de segunda ordem, o Laplaciano é tipicamente sensível a ruído de maneira inaceitável;
- Além disso, o Laplaciano usualmente cumpre papel secundário como um detector para determinar se um pixel está no lado claro ou escuro de uma borda;
- Um uso mais geral do Laplaciano é a localização de bordas baseada na propriedade de mudança de sinal da derivada segunda, conhecida como cruzamento em zero.

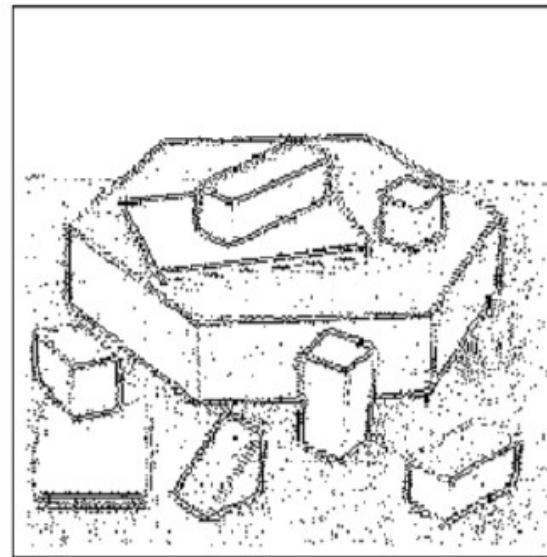


# Operador Laplaciano

- Um exemplo do resultado do operador Laplaciano pode ser visto na figura abaixo:



(a) imagem original



(b) mapa de bordas

## Laplaciano do Gaussiano (Marr-Hildreth)

- Marr e Hildreth (1980) desenvolveram um mecanismo mais sofisticado para detecção de bordas;
- Eles acreditavam que:
  - A detecção deve ser invariante à escala da imagem, sendo necessários operadores de diferentes tamanhos;
  - Mudanças súbitas de intensidade causam picos/vales na primeira derivada e cruzamento por zero na segunda derivada;
- Para isso, desenvolveram um operador que utilizava o Laplaciano e evitavam ruídos, utilizando um filtro gaussiano prévio;
- O operador foi denominado **Laplaciano do Gaussiano (LoG)**.

## Laplaciano do Gaussiano (Marr-Hildreth)

- Após a suavização da imagem por meio de um filtro Gaussiano, as bordas são identificadas pela presença de um cruzamento em zero na derivada segunda com um pico acentuado correspondente à derivada primeira;
- A resposta do operador Laplaciano do Gaussiano é obtida pela operação de convolução, definida abaixo, em que  $f(x, y)$  é uma imagem suavizada por uma função Gaussiana.

$$\nabla^2((G(x, y) * f(x, y)))$$

## Laplaciano do Gaussiano (Marr-Hildreth)

- Após a suavização da imagem por meio de um filtro Gaussiano, as bordas são identificadas pela presença de um cruzamento em zero na derivada segunda com um pico acentuado correspondente à derivada primeira.
- A resposta do operador Laplaciano do Gaussiano é obtida pela operação de convolução, definida abaixo, em que  $f(x, y)$  é uma imagem suavizada por uma função Gaussiana.

$$\nabla^2((G(x, y) * f(x, y)))$$

## Laplaciano do Gaussiano (Marr-Hildreth)

- Considerando o operador Laplaciano:

$$\nabla^2 G(x, y) = \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2}$$

- E o operador Gaussiano:

$$G(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

## Laplaciano do Gaussiano (Marr-Hildreth)

- Temos que o operador Laplaciano do Gaussiano é definido por:

$$\begin{aligned}\nabla^2 G(x, y) &= \frac{\partial^2 G(x, y)}{\partial x^2} + \frac{\partial^2 G(x, y)}{\partial y^2} \\ &= \frac{\partial}{\partial x} \left[ \frac{-x}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \right] + \frac{\partial}{\partial y} \left[ \frac{-y}{\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \right] \\ &= \left[ \frac{x^2}{\sigma^4} - \frac{1}{\sigma^2} \right] e^{-\frac{x^2+y^2}{2\sigma^2}} + \left[ \frac{y^2}{\sigma^4} - \frac{1}{\sigma^2} \right] e^{-\frac{x^2+y^2}{2\sigma^2}} \\ &= \nabla^2 G(x, y) = \left[ \frac{x^2 + y^2 - \sigma^2}{\sigma^4} \right] e^{-\frac{x^2+y^2}{2\sigma^2}}\end{aligned}$$

## Laplaciano do Gaussiano (Marr-Hildreth)

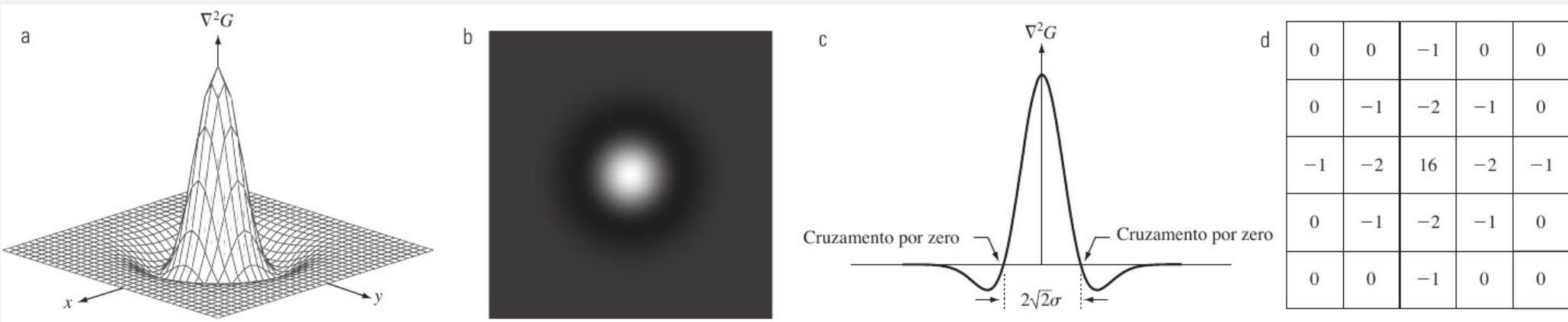
- O operador Laplaciano do Gaussiano pode ser representado de forma aproximada por uma máscara  $5 \times 5$ , conforme imagem abaixo:

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

Fonte: Gonzalez & Woods (2018).

# Laplaciano do Gaussiano (Marr-Hildreth)

- De forma resumida, o operador LoG pode ser ilustrado pela figura abaixo;
- A imagem abaixo mostra (a) do negativo da função LoG, (b) sua imagem correspondente, (c) a seção transversal de (a) e (d) a máscara correspondente.



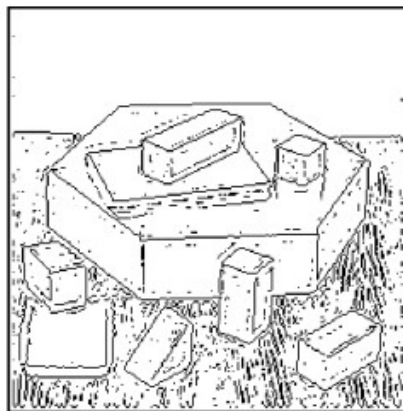


## Laplaciano do Gaussiano (Marr-Hildreth)

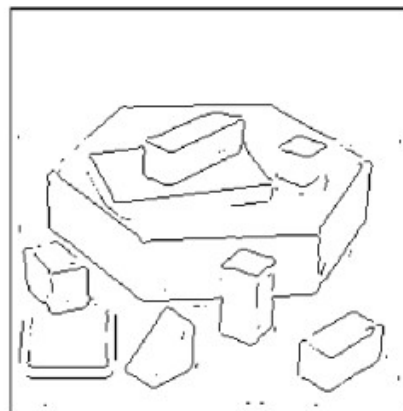
- O resultado do operador Laplaciano do Gaussiano pode ser visto na imagem abaixo, para diferente valores do parâmetro  $\sigma$ .



(a) imagem original



(b)  $\sigma = 1.0$



(c)  $\sigma = 2.0$



(d)  $\sigma = 3.0$

# Detecção de Bordas

## Operador de Canny

# Detector de Bordas de Canny

- O detector de Bordas de Canny (1986) é um mecanismo superior àqueles discutidos até o momento
  - Ele é capaz de localizar bordas mesmo na presença de ruídos;
- Bastante utilizado ainda hoje, devido a sua simplicidade, popularidade e disponibilidade nas bibliotecas de processamento de imagens;

# Detector de Bordas de Canny

- Baseia-se em três objetivos:
  - **Baixa taxa de erro:** todas as bordas deverão ser encontradas e as bordas detectadas devem ser o mais próximas possível das bordas verdadeiras.
  - **Os pontos de borda devem estar bem localizados:** as bordas detectadas devem ser o mais próximas possível das bordas verdadeiras
    - A distância entre um ponto marcado como uma borda pelo detector e o centro da borda verdadeira deve ser mínima.
  - **Resposta de um único ponto de borda:** o detector deve retornar apenas um ponto para cada ponto de borda verdadeiro.
    - O detector não deve identificar múltiplos pixels de borda.

# Detector de Bordas de Canny

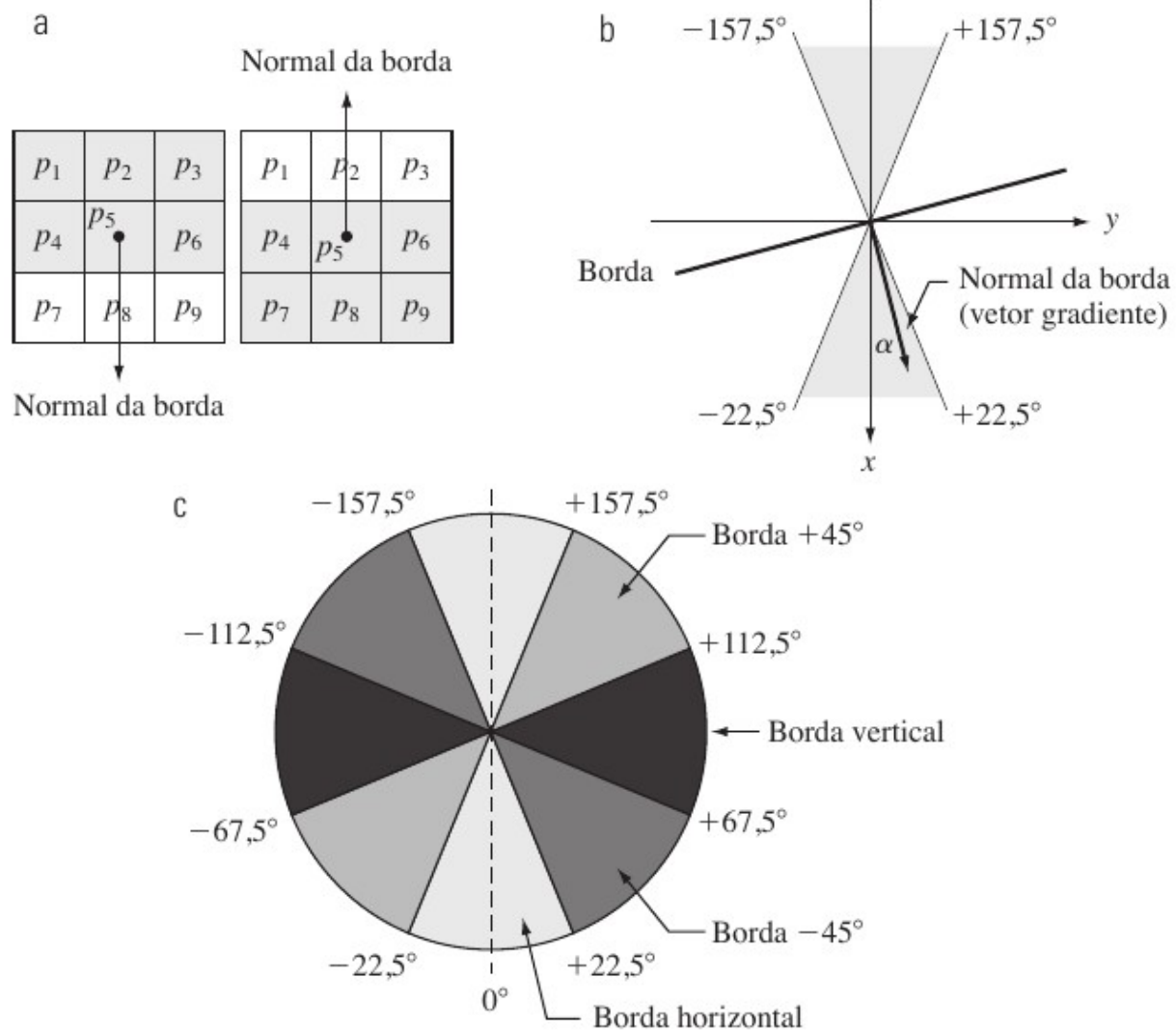
- Procedimento para detecção de bordas:
  1. Suavizar a imagem por meio de um filtro Gaussiano;
  2. Calcular a magnitude e a direção do gradiente utilizando aproximações baseadas em diferenças finitas para as derivadas parciais, de maneira similar aos métodos de gradiente apresentados anteriormente.
  3. Localizar a borda, tomando-se apenas os pontos cuja magnitude sejam localmente máximos na direção do gradiente.
    1. Essa operação é chamada de **supressão dos não máximos** e reduz a espessura das bordas.
  4. Limiarização, para redução de pontos falsos de bordas.

# Detector de Bordas de Canny

- A etapa de **supressão dos não máximos** pode ser feita da seguinte maneira:
  - Especificar um número de orientações discretas da normal de borda (vetor gradiente);
  - Definir direções (orientações) básicas de bordas que passem pelo ponto central
  - Formular o seguinte procedimento centrada em  $(x, y)$  de  $\alpha(x, y)$ , na qual  $g_N(x, y)$  é imagem com supressão de não máximos.
    1. Encontrar a direção  $d_k$  que está mais perto de  $\alpha(x, y)$ .
    2. Se o valor de  $M(x, y)$  for inferior a pelo menos um dos seus dois vizinhos ao longo de  $d_k$ , deixar  $g_N(x, y) = 0$  (supressão);
    3. Caso contrário, deixar  $g_N(x, y) = M(x, y)$ .

# Operador de Canny

- As direções possíveis das bordas estão representadas pela imagem ao lado
- As direções são tomadas em relação ao ponto central da máscara  $3 \times 3$ .



## Detector de Bordas de Canny

- A etapa de **limiarização** é utilizada para remoção de fragmentos espúrios, causados pela presença de ruídos ou textura fina;
- Para remoção dos pontos falsos de borda, pode ser escolhido um limiar  $T$ , sujeito aos seguintes problemas:
  - **Valor de  $T$  baixo:** causa manutenção de bordas falsas (falsos positivos);
  - **Valor de  $T$  alto:** causa eliminação de pontos de borda reais (falsos negativos).



## Detector de Bordas de Canny

- Para resolver o problema de definição do valor  $T$  e evitar bordas fragmentadas, Canny utiliza um processo denominado **limiarização por histerese**;
- Este processo utiliza dois limiares,  $T_1$  e  $T_2$ , onde  $T_2 > T_1$ ;
- Realiza-se o processo de limiarização, utilizando os valores de  $T_1$  e  $T_2$ ;
  - Todos os *pixels* de borda fortes, acima de  $T_2$  são definidos como válidos e indicados como borda;
  - Pontos com valores superiores a  $T_1$  somente serão considerados pontos de borda se estiverem conectados a outros pontos de borda;
  - Pontos com valores inferiores a  $T_1$  não são considerados pontos de borda.

# Detector de Bordas de Canny

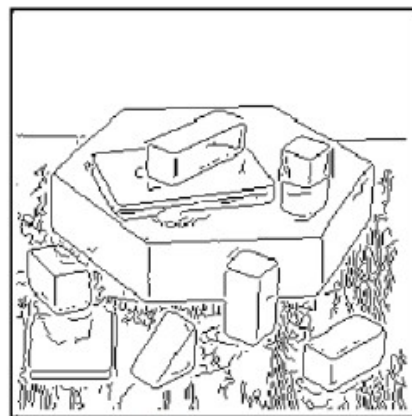
- Para definir os valores dos limiares  $T_1$  e  $T_2$ , Canny sugeriu que a razão do limiar alto (2) para o baixo (1) deve ser de dois ou três para um (Gonzalez & Woods, 2018).
  - Alguns autores recomendam a definição dos parâmetros com base na estimativa de sinal-ruído;
- Comentários finais:
  - Apesar do processo de limiarização com histerese reduzir a largura das bordas, caso estas mantenham largura superior a 1 pixel ao final do processo, pode ser utilizado um mecanismo de afinamento de bordas;
  - O algoritmo de Marr-Hildreth pode ser utilizado para definição suavização da imagem de entrada, no passo 1 do algoritmo.

# Detector de Bordas de Canny

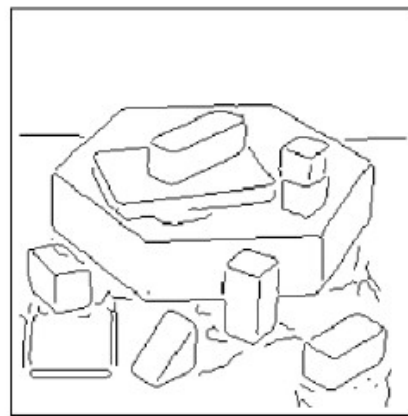
- O resultado do detector de bordas de Canny pode ser visto na imagem abaixo, para diferente valores do parâmetro  $\sigma$ .



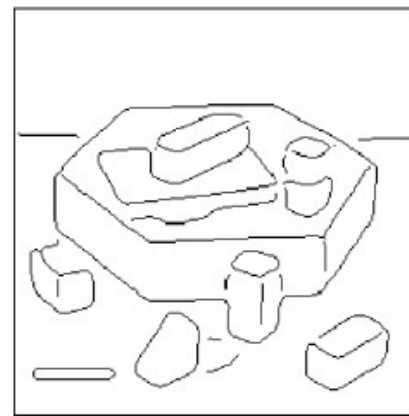
(a) imagem original



(b)  $\sigma = 0.5$



(c)  $\sigma = 1.0$



(d)  $\sigma = 2.0$

# Referências

## Referências

- Rafael C. Gonzalez, Richard E. Woods. **Digital Image Processing – 4th Edition**. 2018. Pearson. ISBN: 978-9353062989.
- Rafael C. Gonzalez, Richard E. Woods. **Processamento digital de imagens - 3ª Edição**. 2010. Pearson Prentice Hall. ISBN: 978-8581435862.
- Agostinho Brito Jr. **Processamento digital de imagens – Slides de Aula**. 2018.
- Silvio Jamil F. Guimarães. **Digital Image Processing – Segmentation – Slides de Aula**. 2020.

## Referências

- Arbelaez, P., Maire, M., Fowlkes, C., and Malik, J. (2011). **Contour detection and hierarchical image segmentation.** IEEE Trans. Pattern Anal. Mach. Intell., 33(5):898–916.