Sistemas de Visão e Percepção Industrial

3-Processamento a Baixo Nível

Parte 3 - Deteção de Arestas

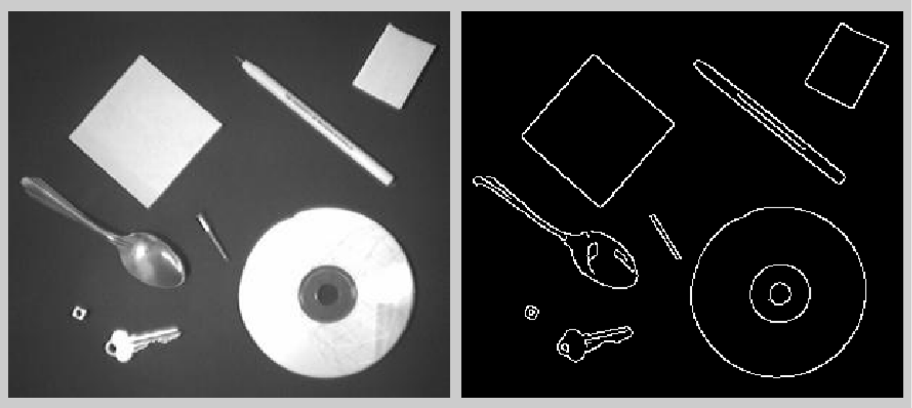
Universidade de Aveiro
2021-2022

Sumário

Conceitos e definições

Deteção de bordas (Edge detection)

- Bordas (ou "arestas") são regiões (pixels) de grande contraste nas intensidades da vizinhança.
- São importantes para a deteção e segmentação de objetos.
- Exemplo:

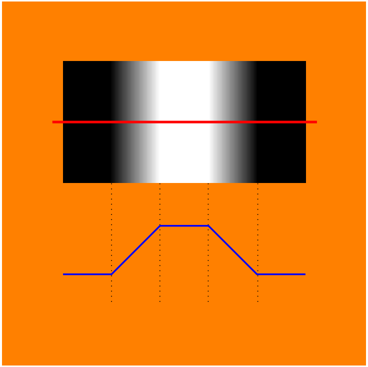


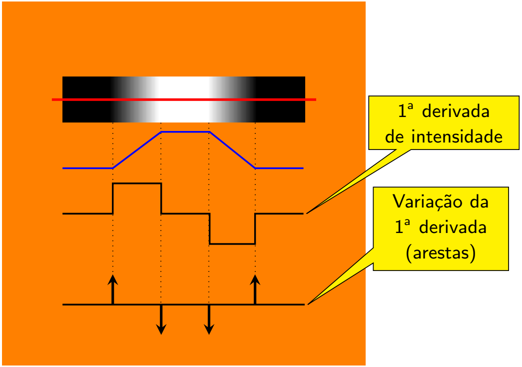
Classes de técnicas de deteção de bordas

- Técnicas baseadas em gradiente de primeira ordem
  - usam dois filtros de convolução (kernels)
  - Básico, Sobel / Prewitt, Robberts
  - Canny
  - ...
- Técnicas que usam gradientes de 2ª ordem
  - Zero Crossing (ou Marr edge detector)
  - Usa laplacianos, LoG, DoG, etc.
  - ...
- Técnicas baseadas em Template Matching (ou Prewitt Compass Edge Detection)
  - Usam em geral 8 filtros de convolução ($k \times 45°$)
  - Kirsh, Robinson, Prewitt (TM), etc...

Deteção de bordas por gradiente

- A técnica comum é a de detetar variações de intensidade usando um operador derivativo local.
- Exemplo de perfil de intensidades e suas derivadas:





Operador de gradiente para arestas (bordas)

- Definição

$$\vec{\mathrm{G}} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} \end{bmatrix}^{\mathsf{T}}$$

- Norma do gradiente e sua aproximação usual

$$||\vec{\mathrm{G}}|| = \sqrt{G_x^2 + G_y^2} \approx |G_x| + |G_y|$$

- Variante discreta...

$$G_x = f(x,y) - f(x-1,y) \qquad G_y = f(x,y) - f(x,y-1)$$

- ... e os filtros correspondentes

$$G_x = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad G_y = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Operador de gradiente (cont.)

- Uma vez obtido o valor do gradiente para todos os pixels...
  - ...procuram-se os que têm um máximo local ou que excedem um dado limiar estabelecido por um critério que poderá ter algumas variantes...
  - ...esses pontos serão "arestas"!
  - **NB**: Existe alguma diversidade de algoritmos para definir se um dado valor de gradiente é "suficiente" ou não para classificar o ponto como "aresta". Uma referência interessante sobre o assunto é:
    - Digital Image Processing, W. Pratt, Wiley, 2007.
- Além do valor (intensidade ou "força") da "aresta", em certos casos é usual também considerar a "orientação" da "aresta" dada por:

$$\theta = \arctan\left(\frac{G_y}{G_x}\right)$$

Operadores específicos

Um operador muito usado

- Operador de Sobel
  - Alternativa usualmente mais eficaz para detetar "arestas" do que gradiente simples.
  - Atente-se ao "peso" dos diversos pixels para os validar como "aresta":

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \qquad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Outros operadores para "arestas"

- Operador de Prewitt

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad G_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

- Operador de Roberts (2×2)

$$G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \qquad G_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

  - Mais em concreto, se P_1, P_2, P_3, P_4 forem os pixels dispostos no seguinte arranjo numa dada imagem, a operação de cálculo do gradiente por Roberts no pixel P_1 é equivalente a:

P_1	P_2
P_3	P_4

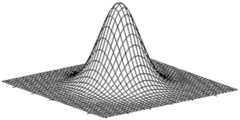
$$G = |P_1 - P_4| + |P_2 - P_3|$$

- Note-se que este operador é muito rápido mas é muito sensível ao ruído!

Um algoritmo mais sofisticado

- Método de Canny (dos mais populares)
  - Operação com estas fases principais:
    - Aplicação de uma Convolução Gaussiana
    - Aplicação de Filtro de Roberts (por vezes outros como Sobel)
    - Seguimento da aresta para evitar fragmentações

- Filtragem Gaussiana



$$\frac{1}{273}\begin{bmatrix}1 & 4 & 7 & 4 & 1\\ 4 & 16 & 26 & 16 & 4\\ 7 & 26 & 41 & 26 & 7\\ 4 & 16 & 26 & 16 & 4\\ 1 & 4 & 7 & 4 & 1\end{bmatrix}$$

- Roberts

$$G_x = \begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}$$

$$G_y = \begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix}$$

- Dois limiares de binarização T1 e T2 usados com histerese no "seguimento" da aresta.

- Mais detalhes em `http://homepages.inf.ed.ac.uk/rbf/HIPR2/canny.htm`

vs_edges.m

Operadores segunda ordem

Operadores de segunda derivada

- Os detetores de bordos de primeira derivada...
  - ... podem gerar arestas "largas" em regiões de variação constante!
  - ... são relativamente direcionais (anisotrópicos).
- Alternativa:
  - usar a segunda derivada.
- Exemplo
  - Laplaciano
    - Assemelha-se a operação de 2ª derivada.
    - Ainda mais sensível ao ruído que a 1ª derivada.
  - Definição e filtro correspondente:

$$\mathcal{L}[f(x,y)] = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \qquad \mathcal{L} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Demonstração do filtro do Laplaciano

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

- Os valores da primeira derivada na horizontal:

$$G_{1x} = f(x, y) - f(x-1, y) \text{ e } G_{2x} = f(x+1, y) - f(x, y)$$

- Estas duas diferenças permitem uma segunda operação de diferenças entre elas:

$$G_{XX} = G_{2x} - G_{1x} = [f(x+1, y) - f(x, y)] - [f(x, y) - f(x-1, y)] = f(x+1, y) - 2f(x, y) + f(x-1, y)$$

- Raciocínio similar para a derivada na vertical:

$$G_{YY} = f(x, y+1) - 2f(x, y) + f(x, y-1)$$

Demonstração do filtro do Laplaciano (conc.)

- Os filtros de convolução (2D) são então:

$$G_{XX} = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix} \qquad G_{YY} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

- Como a operação de convolução é distributiva em relação à adição, virá:
  - $\mathcal{L}[A(x,y)] = G_{XX} * A + G_{YY} * A = (G_{XX} + G_{YY}) * A = \mathcal{L} * A$
    - onde se tem finalmente:

$$\mathcal{L} = G_{XX} + G_{YY} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Ainda a segunda derivada - Laplaciano

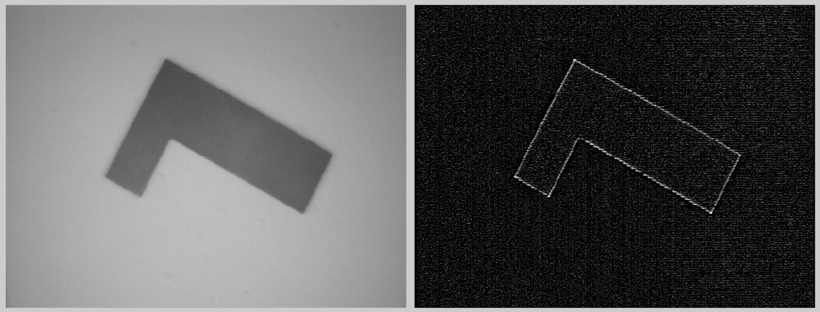
- Como a segunda derivada é sensível ao ruido, é comum aplicar uma técnica de suavização antes de usar o operador de Laplace (2ª derivada).
- Laplaciano da Gaussiana - LoG
  - A segunda derivada de $f(x, y)$ é suavizada
  - Expressão do LoG (Laplace of Gaussian):

$$\nabla^2[G(x, y, \sigma) * f(x, y)] = \mathrm{LoG}(x, y) = -\frac{1}{\pi\sigma^4}\left(1 - \frac{x^2 + y^2}{2\sigma^2}\right)\mathrm{e}^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Outras aproximações discretas comuns do operador Laplaciano:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \qquad \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}$$

Exemplo de resultado de um LoG



```
A=im2double(imread('part.png'));
H=fspecial('log'); %creates the right filter weights
C=imfilter(A,H);   %applies the filter
C=C/max(max(C));   %normalize result to 1
subplot(1,2,1); imshow(A); subplot(1,2,2); imshow(C)
```

- A imagem à direita não é a de arestas, é apenas a do gradiente normalizado a 1.
  - Para obter as arestas recorre-se à técnica de "Zero-crossing".
  - As arestas estarão nas zonas onde o gradiente de 2ª ordem passa de negativo a positivo e vice-versa.

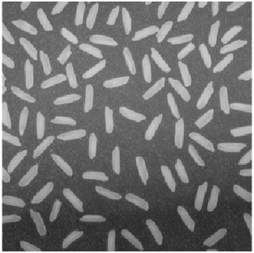
vs_log.m

Exemplos em Matlab

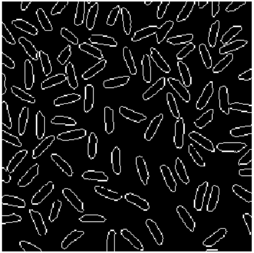
Operadores em Matlab

- Em Matlab existe uma função que calcula o gradiente e determina os valores das arestas.
- Função `edge()` com seis algoritmos possíveis:
  - `BW = edge(I,'sobel')`
  - `BW = edge(I,'prewitt')`
  - `BW = edge(I,'roberts')`
  - `BW = edge(I,'log')`
  - `BW = edge(I,'zerocross')` ['`log`' e outros filtros]
  - `BW = edge(I,'canny')`
    - `BW = edge(I,'approxcanny')` (variante mais rápida de Canny)
- Aceita outros parâmetros opcionais variáveis conforme o operador escolhido.
- Pode retornar outros valores tais como o limiar automático usado na decisão do que é "aresta".

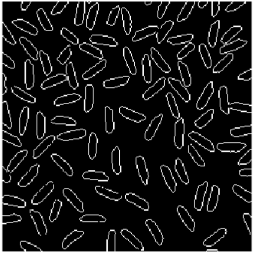
Exemplos de deteção de arestas - 1



original

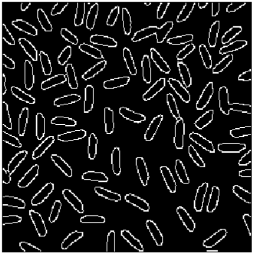


prewitt

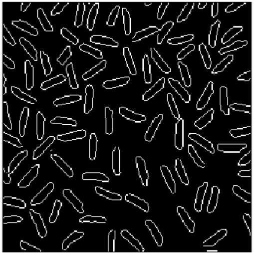


sobel

roberts



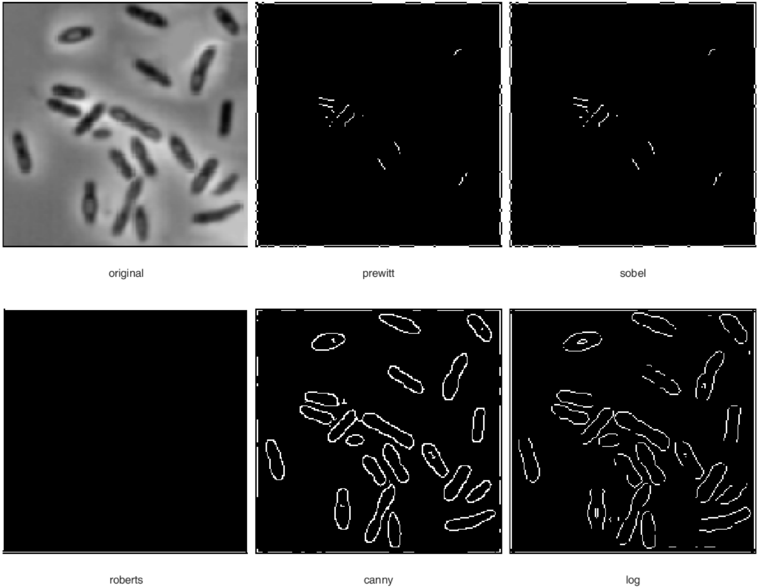
canny



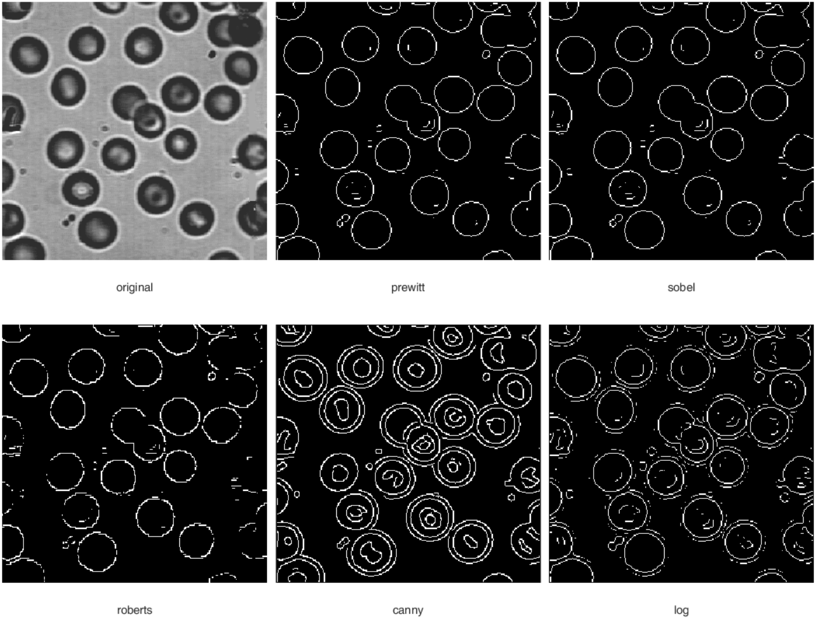
log

vs_edge_var.m

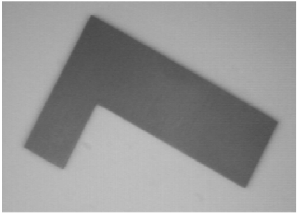
Exemplos de deteção de arestas - 2



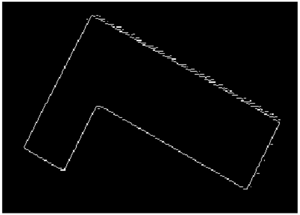
Exemplos de deteção de arestas - 3



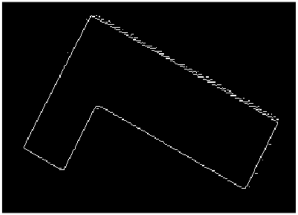
Exemplos de deteção de arestas - 4



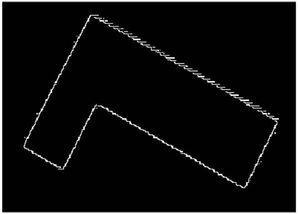
original



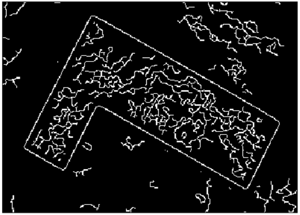
prewitt



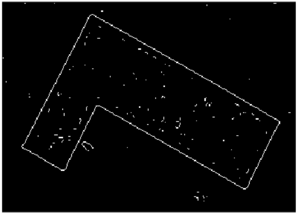
sobel



roberts



canny



log

Técnicas de *template matching*

Alternativa de "template matching" (compass operators)

- Em alternativa aos métodos diferenciais usam-se em cada ponto múltiplos filtros de convolução (mais de dois, e tipicamente de 8 a 12)
- Esses filtros representam as diversas possibilidades de tipologias de arestas.
- Aplicam-se todos os filtros de um dado conjunto, e em cada ponto é escolhido aquele que resultar no maior valor do cálculo de convolução:
  - Seja a imagem A e os filtros $H_j(j = 1, ..., n)$
  - Sejam as convoluções: $D_j = H_j * A$
  - $G(u, v) = max(Dj(u, v))$ para $j = 1, ..., n$
  - Ou seja, o valor da operação de TM para arestas (G) é o máximo dos n filtros testados em cada ponto.
- Exemplos de filtros de TM (3×3):
  - Kirsh, Robinson "nivel-3", Robinson "nível-5", Prewitt (para TM), etc.
- As variantes são em múltiplos de 45° ("rotação" em torno do elemento central)

Exemplos de filtros usados em TM

- Prewitt 0°
$$\begin{bmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{bmatrix}$$
- Prewitt 45°
$$\begin{bmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{bmatrix}$$
- Prewitt 90°
$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix}$$
- Kirsh 0°
$$\begin{bmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{bmatrix}$$
- Kirsh 45°
$$\begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix}$$
- Kirsh 90°
$$\begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}$$
- Robinson L-3 0°
$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$
- Robinson L-3 45°
$$\begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 1 \end{bmatrix}$$
- Robinson L-3 90°
$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

`vs_compass_edges.m`