

VISÃO COMPUTACIONAL

AULA 5

CARACTERÍSTICAS DE IMAGENS:
PONTOS E SUPERFÍCIES

Detecção de Bordas e Contornos

CARACTERÍSTICAS DE IMAGENS: PONTOS E SUPERFÍCIES

CARACTERÍSTICAS DE IMAGENS **(Image Features)**

- O que são?
 - Propriedades Globais de uma imagem, ou de parte dela (**característica global**). Por exemplo:
 - Médias das intensidades de pixels
 - Áreas em pixels
 - Parte de uma Imagem com algumas propriedades especiais (**característica local**). São mais importantes em Visão Computacional. Por exemplo:
 - Círculo, elipse ou linha
 - Região texturada de uma imagem de intensidade
 - Superfície plana em uma imagem de profundidade

Definição

- São partes *detectáveis*, *locais* e *significativas* de uma imagem
-

Características Significativas

- Associadas a elementos de interesse da cena via processo de formação da imagem. Por exemplo:
 - Variações agudas de intensidade criadas pelo contorno de objetos
 - Regiões da imagem com intensidade uniforme. Ex:
 - Imagens de superfícies planas
 - Invariância ou facilidade para detecção (detectabilidade)

Características Detectáveis

- Quando existe algoritmo para sua detecção, senão a característica não terá utilidade

Características diferentes ==> algoritmos diferentes

Algoritmos ==> produz grupos de palavras-chave (descritores) de características (keyword=descriptors).

Ex:

- Posição
- Coordenadas do ponto central de um segmento
- Comprimento e orientação de um segmento
- Assinaturas de um conjunto de pontos

Em Visão Computacional

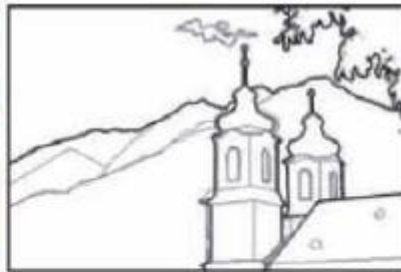
- Extração de características é passo intermediário
 - Extração de linhas ==> navegação robótica
 - Localização de objetos ==> calibrar parâmetros intrínsecos de uma câmera.
 - Descrição de geometrias ==> registro e casamento de imagens
- Algoritmo de detecção de características só pode ser avaliado dentro do contexto do sistema inteiro.

DETECÇÃO DE BORDAS E CONTORNOS

- O Básico

Definição: Bordas e Contornos

Pontos de uma borda, ou simplesmente bordas, são píxeis, ou regiões em torno deles, com valores de intensidade de imagem com variação acentuada.



DETECÇÃO DE BORDAS E CONTORNOS

- O Básico

Definição: Bordas e Contornos

Pontos de uma borda, ou simplesmente bordas, são píxeis, ou regiões em torno deles, com valores de intensidade de imagem com variação acentuada.

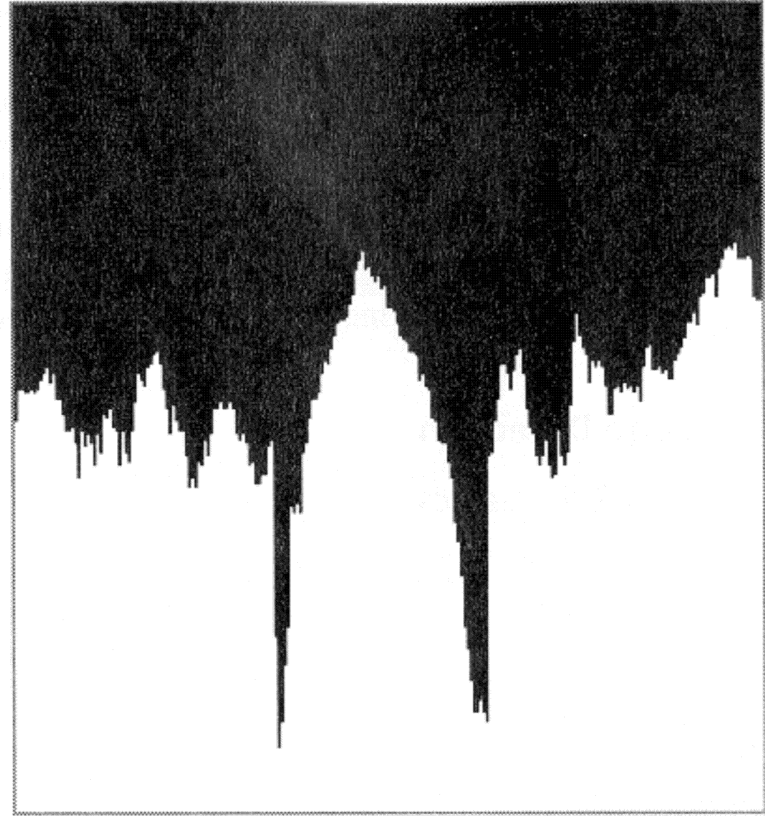
Problema da detecção de bordas

Dada uma imagem corrompida por ruído de aquisição, localizar as bordas mais prováveis de terem sido geradas pelos elementos da cena, e não por ruído

borda ==> fragmentos de um contorno

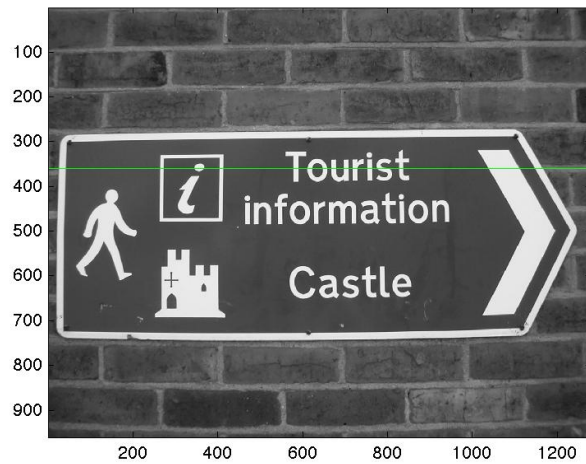


(a)

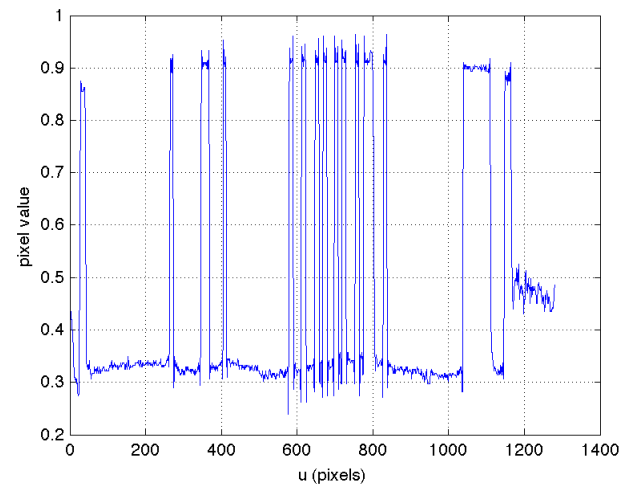


(b)

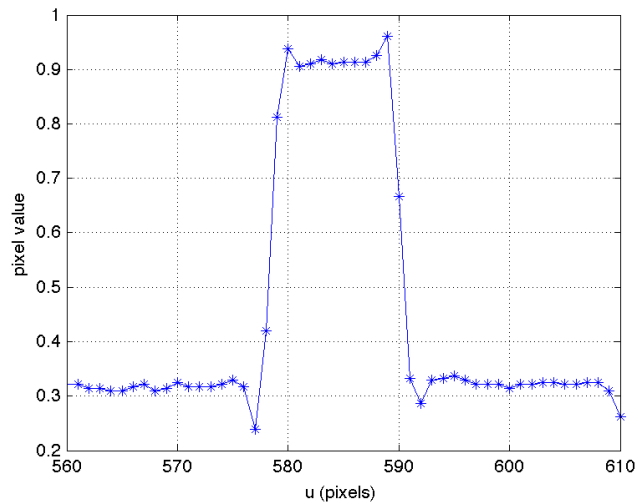
(a) Imagem 325x237 pixels, com linha em $i = 56$ mostrada. (b) perfil de intensidade ao longo da linha



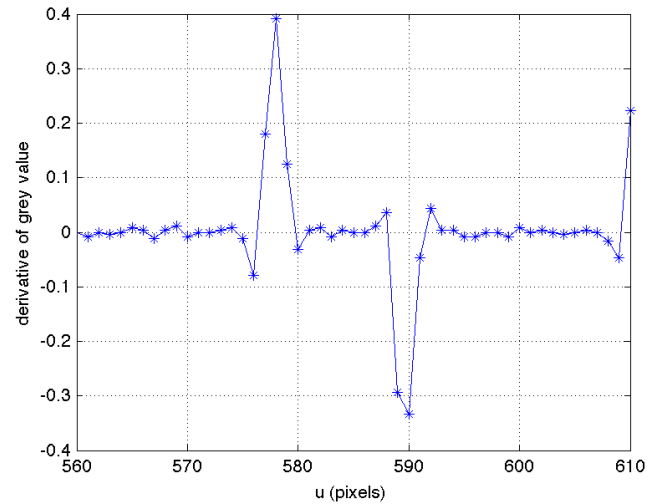
a) Imagem original com linha em 360



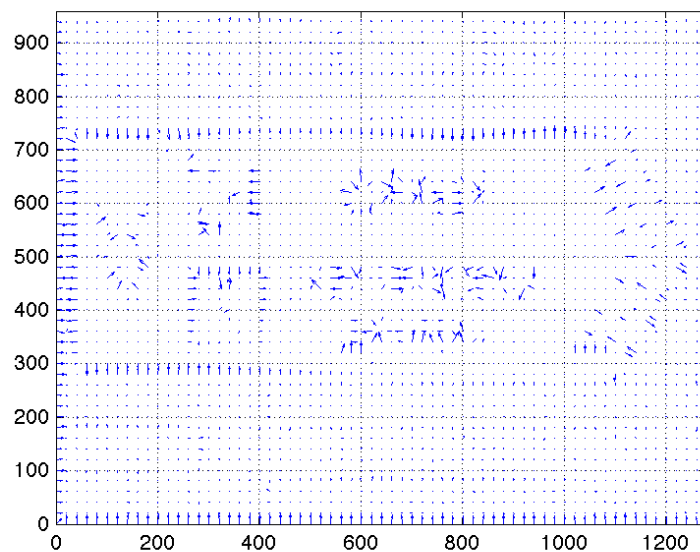
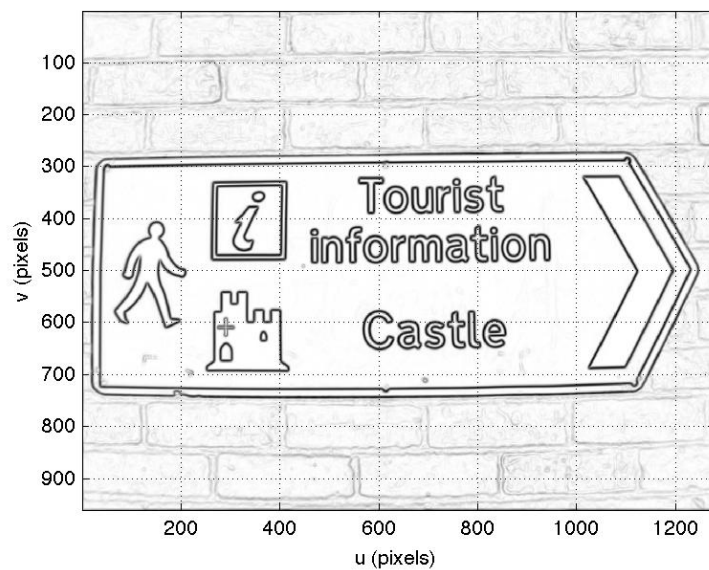
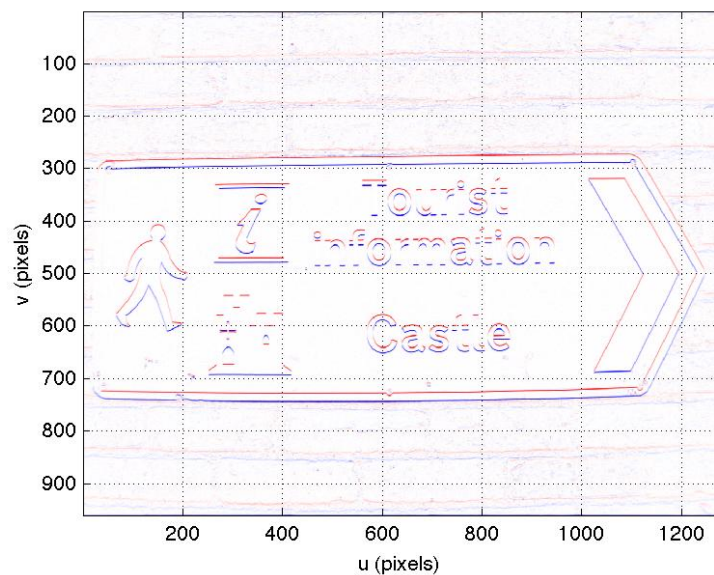
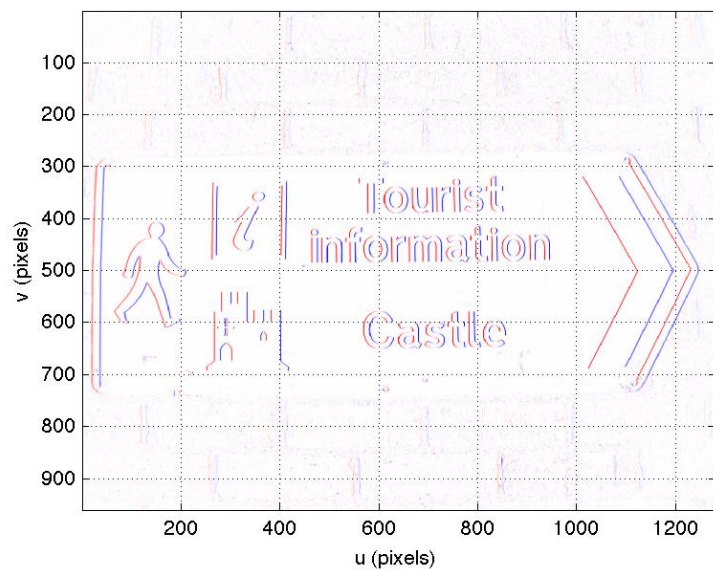
b) perfil de intensidade em $v = 360$.



c) Close-up em $u=580$;



d) Derivada de c)

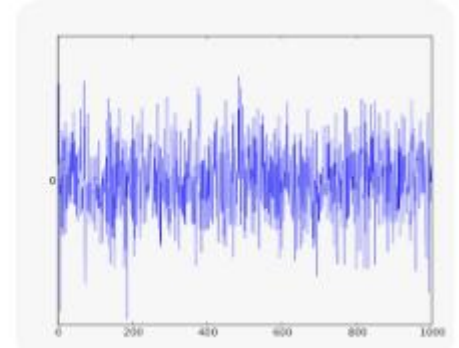
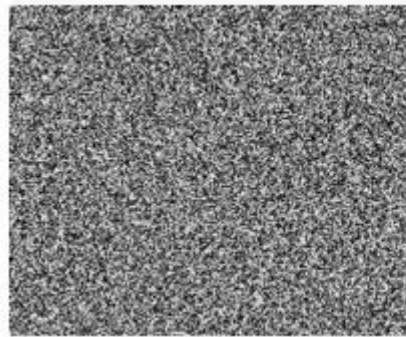


Os 3 Passos da Detecção de Bordas

- **Atenuação de Ruído**

- Suprimir o máximo possível de ruídos de imagens, sem destruir a borda. Na ausência de informação assuma ruído branco e Gaussiano.

Ruído branco →



- **Realce de Bordas**

- Projetar um filtro que responda à borda, isto é, a saída do filtro seja maior nos pontos de borda e menor em outros locais, de modo que a borda possa ser detectada como máximo local na saída do filtro.

○ **Localização de Borda**

- Decidir quais máximos locais na saída do filtro são bordas e quais são apenas causados por ruído. Isto envolve:
 - Afinar bordas grossas para a largura de 1 pixel (supressão de não-máximos)
 - Estabelecer o mínimo valor para declarar um máximo local como ponto de borda (thresholding).

Detector CANNY EDGE

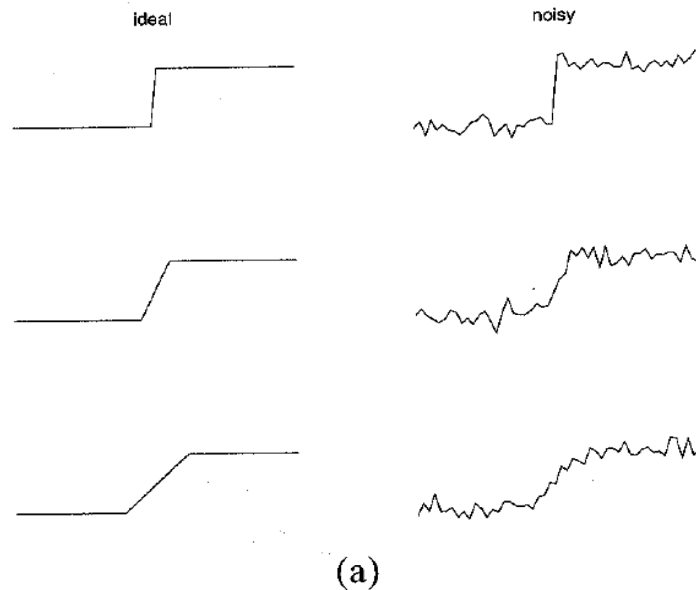
- Provavelmente o mais utilizado hoje em V. C.
- É ótimo, em um sentido matemático

Para aplicação de um detector de bordas é necessário:

- I. Formular um modelo matemático de bordas e ruído;
- II. Formular critérios de desempenho quantitativo, formalizando propriedades desejáveis do detector; (ex.: boa imunidade ao ruído)
- III. Sintetizar o melhor filtro dados os modelos e critério de desempenho. Filtros lineares são mais fáceis de manipular e implementar.

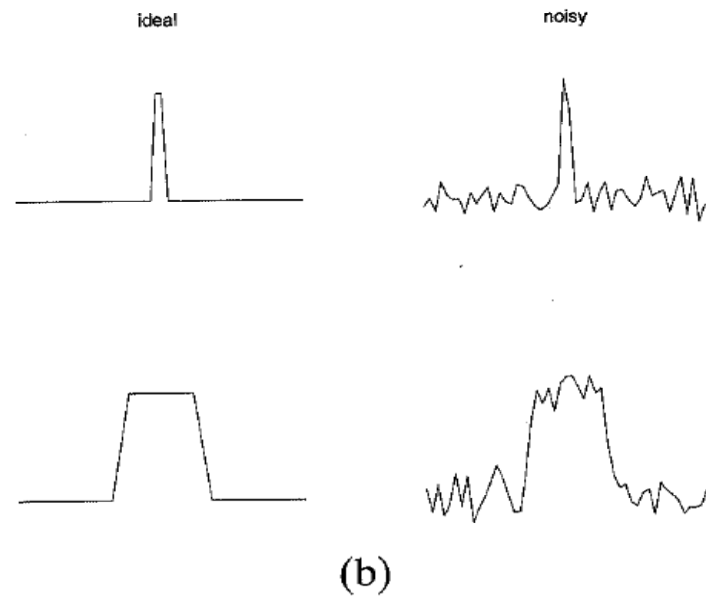
I. Modelagem de Bordas e Ruído

- Borda
 - De acordo com os perfis de intensidade
 - Borda em Degrau (step)
 - O tipo mais comum
 - Pode ocorrer na forma de rampa



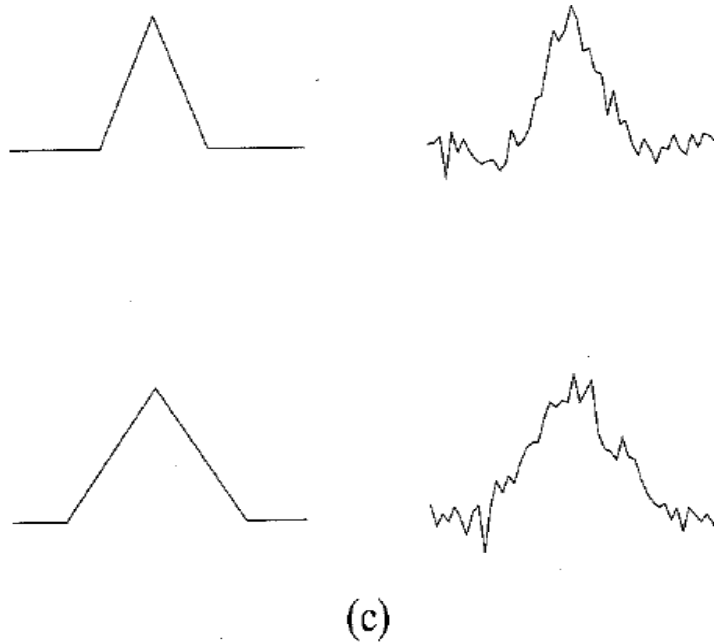
Esquerda: Bordas em Degrau ideais. Direita: versões ruidosa, com adição de ruído gaussiano com $\sigma=5\%$ da altura do degrau

- Borda em Cristas (ridge)
 - Gerados por linhas finas
 - Podem ser modelados como dois degraus



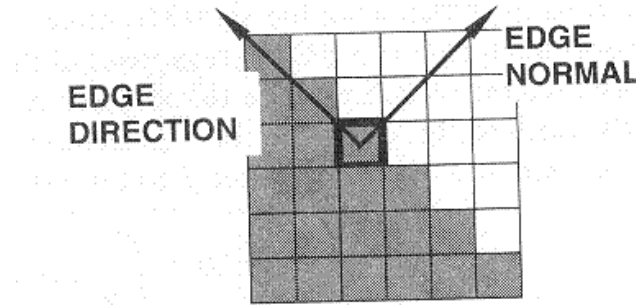
Esquerda: Bordas em Crista ideais. Direita: versões ruidosa, com adição de ruído gaussiano com $\sigma=5\%$ da altura do degrau

- Borda em Triângulo (roof)
 - Raros
 - Podem aparecer ao longo de interseções de superfícies
 - Perfil de primeira derivada



Esquerda: Bordas em Triângulo ideais. Direita: versões ruidosa, com adição de ruído gaussiano com $\sigma=5\%$ da altura do degrau

O Essencial de um Descritor de Borda (Edge Descriptor)



Normal a uma borda e direção da borda. Posição em (2,2), com origem no canto superior esquerdo.

- Normal
 - Para contornos de imagens 2-D, a direção (vetor unitário) da máxima variação de intensidade no ponto de borda. Isto identifica a direção perpendicular à borda.
- Direção
 - Direção perpendicular à normal à borda, e portanto tangente ao contorno do qual a borda é parte. Isto identifica a direção tangente à borda.
- Posição ou Centro
 - Posição na imagem na qual a borda é localizada, ao longo da perpendicular à borda. Isto normalmente é gravado em imagem binária (1 para contorno, 0 para não contorno)

- Intensidade (Edge Strength)
 - Medida do contraste local da imagem; módulo da variação de intensidade através da borda (ao longo da normal)

O modelo considerado ideal para o contorno de degrau na imagem é

$$G(x) = \begin{cases} 0 & \text{se } x < 0 \\ A & \text{se } x \geq 0 \end{cases}$$

Box Filter ou Filtro Caixa

II. Critérios para Detecção Ótima de Bordas

Critério 1: Boa Detecção

- O detector ótimo deve minimizar a probabilidade de falsos positivos (detectando bordas espúrias causadas por ruído), bem como de minimizar a perda de bordas verdadeiras (falsos negativos).
- Pode-se maximizar o Signal-to-Noise Ratio (SNR)

$$\text{SNR} = \sqrt{\frac{\text{Potencia da Resposta do filtro ao degrau ideal}}{\text{Potencia do Ruído}}}$$

Potência de um sinal $s(t) = E\{s^2(t)\}$

$$E[X] = \sum_{i=1}^{\infty} x_i p(x_i)$$

x_i = valores de X ; probabilidade de ocorrência de x_i

filtro

A resposta a um filtro 1-D linear $f(t)$, de largura $2W$, sobre um degrau ideal

degrau

$$G(t) = \begin{cases} 0 & \text{se } t < 0 \\ A & \text{se } t \geq 0 \end{cases}$$

é

Sinal centrado em $x = 0$

$$\int_{-W}^W G(-t) \cdot f(t) dt = A \cdot \int_{-W}^0 f(t) dt$$

resposta

Assume-se que o ruído é aditivo, branco e gaussiano

$$\Rightarrow SNR = \frac{A \left\| \int_{-W}^0 f(t) dt \right\|^2}{n_o \cdot \int_{-W}^W f^2(t) dt}$$

e n_o^2 = RMS amplitude do ruído por unidade de comprimento

Critério 2: Boa Localização

- As bordas detectadas devem estar o mais próximo possível das bordas reais. Deve-se maximizar

$$LOC = \frac{A \|f'(0)\|}{n_o \cdot \sqrt{\int_{-W}^W f'^2(t) dt}}$$

Relação entre a potência da resposta à derivada do filtro e a potência da derivada do ruído.

$$SNR = \frac{A \left\| \int_{-W}^0 f(t) dt \right\|^2}{n_o \cdot \int_{-W}^W f^2(t) dt}$$

- SNR e LOC dependem apenas do filtro

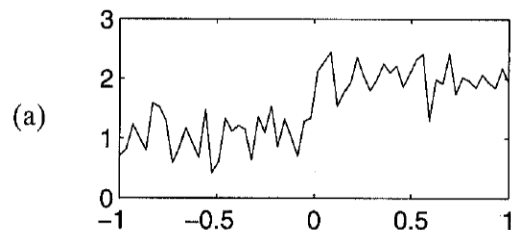
$$LOC = \frac{A \|f'(0)\|}{n_o \cdot \int_{-W}^W f'^2(t) dt}$$

- $SNR = \frac{A}{n_o} \cdot \Sigma f$ e $LOC = \frac{A}{n_o} \cdot \Lambda(f')$

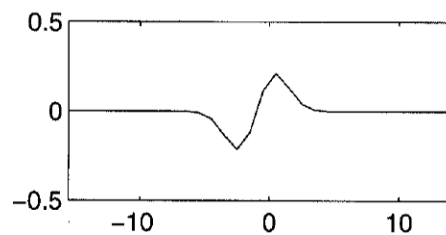
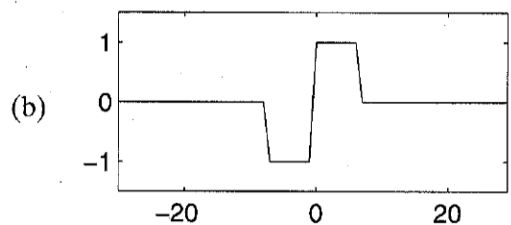
- O produto $\Sigma \cdot \Lambda$ deve ser máximo para f

Pode-se demonstrar que o produto acima pode ser maximizado para $f(x) = G(-x)$ em $[-W, W]$

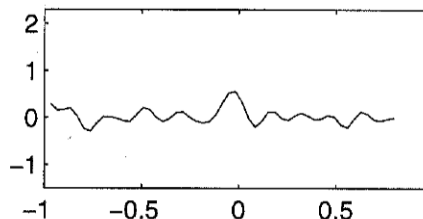
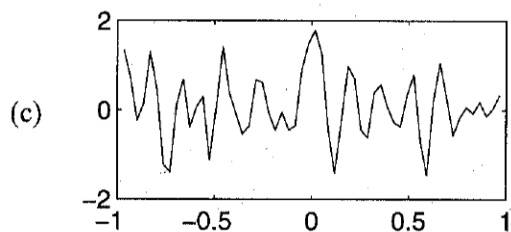
➔ O detector ótimo, portanto, é um simples operador de diferença, ou filtro caixa. (detector de degrau 1-D)



(a) Borda em degrau corrompida com ruído gaussiano aditivo, com $\sigma=15\%$ da altura do degrau não corrompido.



(b) Esquerda: Filtro Caixa
Direita: Filtro DoG



(c) Esquerda: Resposta ao Filtro Caixa → maior amplitude e mais máximos locais
Direita: Resposta ao Filtro DoG → menor amplitude e menos máximos locais

No entanto, aparecem muitos Máximos Locais. Isso sugere um terceiro critério.

Critério 3: Resposta Única

- O detector deve retornar apenas 1 ponto para cada ponto da borda verdadeira, isto é, minimizar o número de máximos locais em torno da borda verdadeira criados pelo ruído.
- Isto é formalizado impondo que a distância média entre máximos locais causados pelo ruído seja apenas uma fração da metade da largura do filtro W .

Pode-se mostrar que $(f(at) \Leftrightarrow \frac{1}{|a|} F(\frac{\omega}{a})) \rightarrow$ Teorema da Similaridade)

$$\text{Se } f_w = f(x/w)$$

$$\Rightarrow \Sigma(f_w) = \sqrt{w} \cdot \Sigma(f)$$

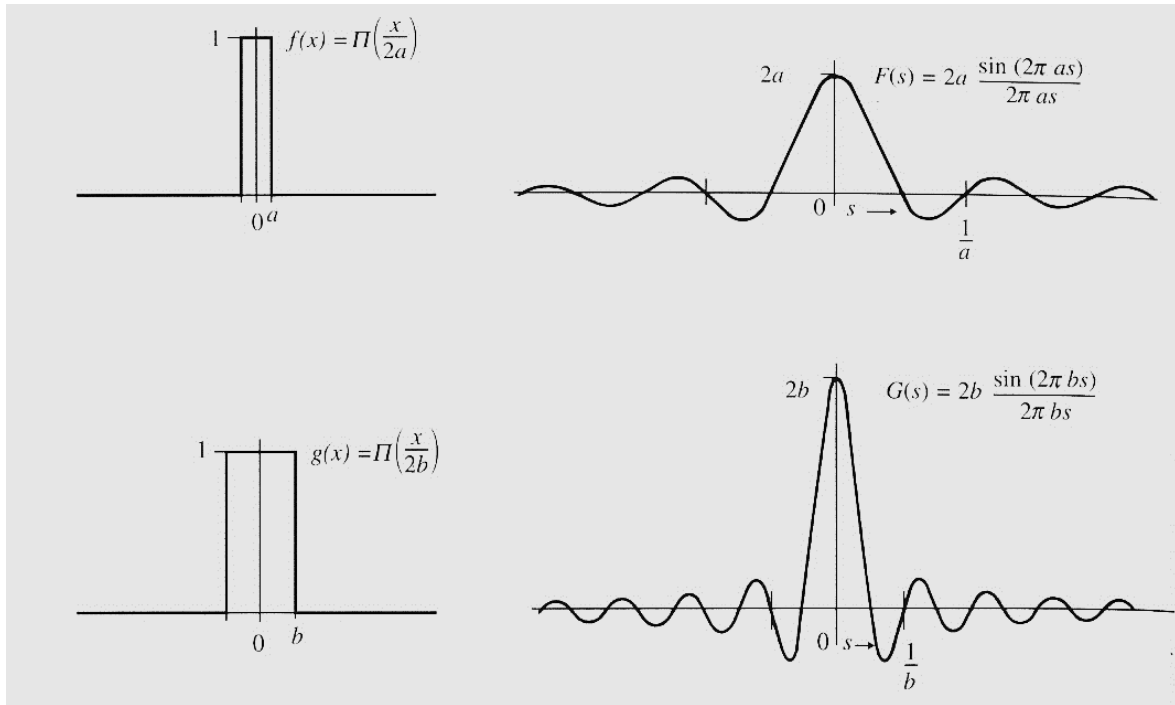
$$\Rightarrow \Lambda(f'_w) = \frac{1}{\sqrt{w}} \cdot \Lambda(f')$$

Um filtro mais largo melhora a detecção.

Um filtro mais largo piora a localização no mesmo fator

$$SNR = \frac{A \left\| \int_{-W}^0 f(t) dt \right\|}{n_o \cdot \sqrt{\int_{-W}^W f^2(t) dt}}$$

$$LOC = \frac{A \|f'(0)\|}{n_o \cdot \sqrt{\int_{-W}^W f'^2(t) dt}}$$



$$f(at) \Leftrightarrow \frac{1}{|a|} F\left(\frac{\omega}{a}\right) \rightarrow$$

**Teorema da
Similaridade**

\Rightarrow Um filtro mais largo melhora a detecção

(F mais estreita = reduz passagem de altas frequências de ruído)

\Rightarrow Um filtro mais estreito melhora a localização

(F mais larga = aumenta passagem altas frequências das derivadas)

É possível otimizar a detecção ou localização modificando a escala do filtro, mas não as duas simultaneamente.

III. Realce Ótimo da Borda em Degrau

- Filtro de Realce Ótimo f
 - Maximizar $\Lambda(f) \cdot \Sigma(f)$, sob a restrição de resposta única. (critérios 1, 2 e 3) (Boa detecção, boa localização, resposta única)
 - Não há soluções de forma fechada (closed-form solutions). (solução deve ser numérica)
 - Uma boa aproximação de um filtro em degrau ótimo é
 - Primeira Derivada do Gaussiano
 - 80% do ideal sob os critérios 1 e 2
 - 90% do ideal sob o critério 3

- Um filtro 2-D poderia ser substituído assim:
 - Aplica-se o filtro 1-D na direção normal à borda (não se sabe ainda onde está a normal)
 - Aplica-se o filtro 1-D em várias direções

Pode-se simplificar o procedimento acima

- Usa-se um filtro Gaussiano circular seguido da estimativa do gradiente; (filtro gaussiano tem máscara com valores que decrescem radialmente)
- Usa-se o gradiente para as derivadas direcionais; (vertical e horizontal)
- É menos preciso do que filtros direcionais, mas é adequado em muitos casos.

Pode-se, então, qualificar

- Intensidade da Borda

$$s(i, j) = \|\nabla(G * I)\|$$

norma do gradiente de uma imagem suavizada por um filtro gaussiano

- Normal à borda

$$\vec{n} = \frac{\nabla(G * I)}{\|\nabla(G * I)\|}$$

$$\vec{\nabla} f = \frac{\partial f}{\partial x} \hat{i} + \frac{\partial f}{\partial y} \hat{j} + \frac{\partial f}{\partial z} \hat{k},$$

Gradiente ==> vetorial

$$\vec{\nabla} \cdot \vec{v} = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z},$$

Divergente ==> escalar

$$\vec{\nabla} \times \vec{v} = \left(\frac{\partial v_z}{\partial y} - \frac{\partial v_y}{\partial z} \right) \hat{i} + \left(\frac{\partial v_x}{\partial z} - \frac{\partial v_z}{\partial x} \right) \hat{j} + \left(\frac{\partial v_y}{\partial x} - \frac{\partial v_x}{\partial y} \right) \hat{k}.$$

Rotacional ==> vetorial

Algoritmo CANNY_EDGE_DETECTOR

Dada uma Imagem I

1. Aplica-se CANNY_ENHANCER a I
 2. Aplica-se NONMAX_SUPPRESSION à saída do CANNY_ENHANCER
 3. Aplica-se HYSTERESIS_THRES à saída do NONMAX_SUPPRESSION
-

Algoritmo CANNY_ENHANCER

A entrada é I , corrompida por ruído. G é uma gaussiana com média nula e desvio padrão σ .

1. Aplicar a máscara Gaussiana (LINEAR_FILTER) G ,

$$J = I * G$$

2. Para cada pixel (i, j)

- a. Computar as componentes do gradiente, J_x e J_y ;

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0 - h)}{2h} + O(h^2) = \text{derivada}$$

- b. Estimar a intensidade da borda;

$$e_s(i, j) = \sqrt{J_x^2(i, j) + J_y^2(i, j)}$$

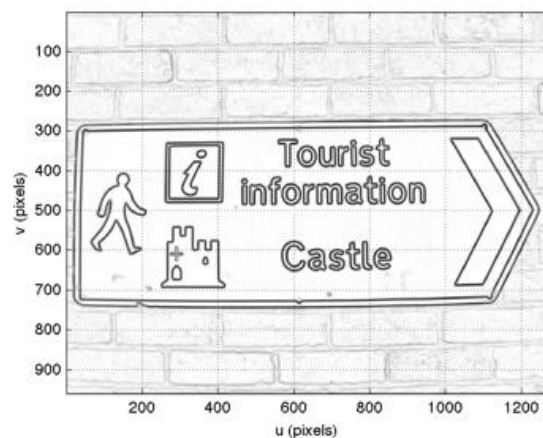
c. Estimar a orientação da normal ao contorno;

$$e_o(i, j) = a \tan \frac{J_y}{J_x}$$

Saídas => Intensidade da Imagem , E_s , formada pelos $e_s(i, j)$

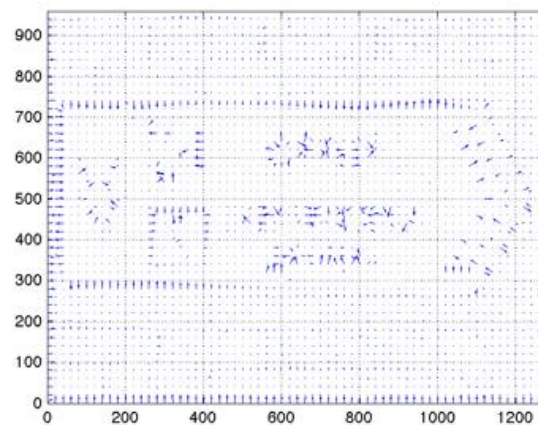
Orientação da Imagem , E_o , formada pelos $e_o(i, j)$

E_s



Magnitude do gradiente

E_o



Direção do gradiente

Algoritmo NONMAX_SUPPRESSION

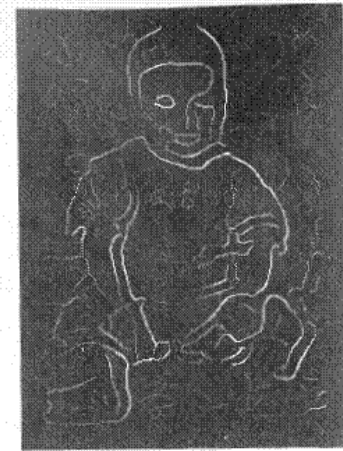
A matriz E_s pode conter muitas cristas de ruídos em torno do máximo local. Pode-se reduzir o contorno à largura de 1 pixel.

- A entrada é E_s e E_o .
- 4 direções d_1, \dots, d_4 , identificadas por $0^\circ, 45^\circ, 90^\circ$ e 135° com respeito ao eixo horizontal do sistema de ref. da imagem.

Para cada pixel (i, j) :

1. Ache a direção , d_k , que melhor aproxima a direção $E_o(i, j)$ (normal à borda)
2. Se $E_s(i, j)$ é menor do que pelo menos um de seus vizinhos ao longo de d_k , faça $I_N(i, j) = 0$ (supressão); senão faça $I_N(i, j) = E_s(i, j)$.

Saída \Rightarrow Imagem $I_N(i, j)$ dos pontos do contorno afilado
(i.e., $E_s(i, j)$ depois da supressão dos pontos não-máximos)



Saída de Intensidade de imagens pelo algoritmo CANNY_ENHANCER, depois da supressão de não-máximos, mostrando o efeito de tamanhos diversos de filtros gaussianos, com $\sigma = 1, 2, 3$ respectivamente.

A imagem I_N ainda pode conter máximos locais criados por ruído

Algoritmo HYSTERESIS_THRES

Thresholding pode aceitar falsas bordas (se o limiar for baixo), e fragmentar o contorno verdadeiro (valores de máximos reais podem oscilar acima e abaixo do limiar).

A solução é o procedimento seguinte:

I_N , saída de NONMAX_SUPPRESSION,
 E_o , matriz de orientação da imagem,
 τ_l e τ_h dois limiares tal que $\tau_l < \tau_h$.

Para todos os pontos do contorno em I_N , e varrendo I_N em uma ordem fixa:

1. Localize o próximo pixel da borda não visitado, $I_N(i, j)$, tal que $I_N(i, j) > \tau_h$;
2. Iniciando de $I_N(i, j)$, siga as cadeias de máximos locais conectados, em ambos os sentidos perpendiculares à normal (tangente) ao contorno, enquanto $I_N > \tau_l$.
3. Marque todos os pontos visitados, e salve a localização dos pontos conectados do contorno.

Saídas => Conjunto de listas descrevendo posição e orientação de contornos contínuos na imagem;

Intensidade e orientação da imagem, descrevendo as propriedades dos pontos do contorno.



Saída de HISTERESIS_THRESH, mostrando o efeito da variação do tamanho do filtro. Esquerda para a direita: $\sigma = 1, 2, 3$ píxeis

- Outros Detectores de Bordas

Algoritmo ROBERTS_EDGE_DET

A entrada é formada por uma imagem , I, e um limiar τ .

1. Aplique um filtro para atenuar ruído (Gaussiano na ausência de informações), obtendo uma nova imagem I_s .
2. Filtrar I_s (LINEAR_FILTER) com as máscaras abaixo, obtendo-se duas imagens, I_1 e I_2 , respectivamente

$$I_1 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad I_2 = \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix}$$

3. estimar a magnitude do gradiente em cada pixel (i, j) com

$$G(i, j) = \sqrt{I_1^2(i, j) + I_2^2(i, j)}$$

obtendo uma imagem de magnitude de gradientes, G.

Para filtros passa-alta, em geral a soma dos elementos da máscara é zero

4. marque todos os pontos da borda (i, j) tal que $G(i, j) > \tau$.


Saída ==> localização dos pontos da borda obtidas no último passo.

Algoritmo SOBEL_EDGE_DET

O mesmo que ROBERTS_EDGE_DET, alterando o passo 2 para:

2. Filtrar I_s (LINEAR_FILTER) com as máscaras

$$I_1 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad I_2 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$


transpostas

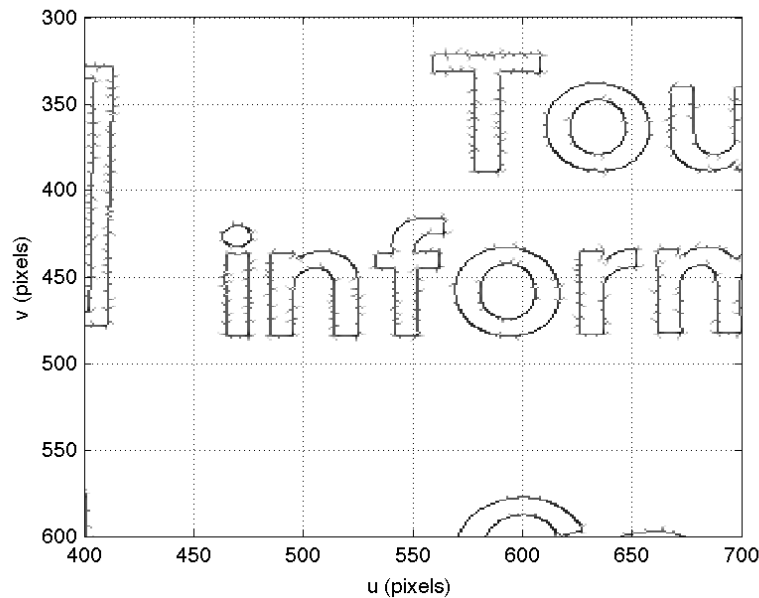
obtendo 2 imagens I_1 e I_2

- Os dois detectores anteriores funcionam também como filtros de realce de bordas.

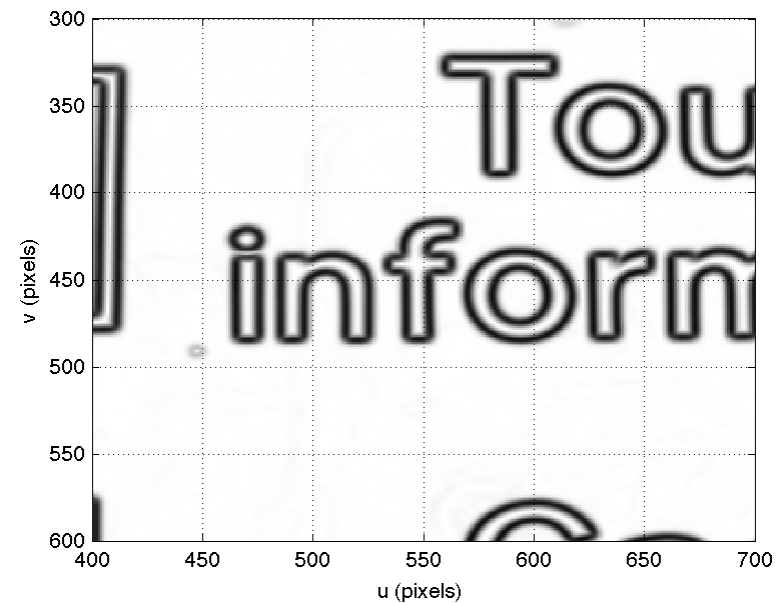


Esquerda: Saída do detector de bordas Sobel. Meio: bordas detectadas usando limiar 35 na imagem realçada. Direita: mesmo com limiar 50. Alguns contornos têm largura maior do que 1 pixel.

Operador DoG



Operador Canny com parâmetros padrão



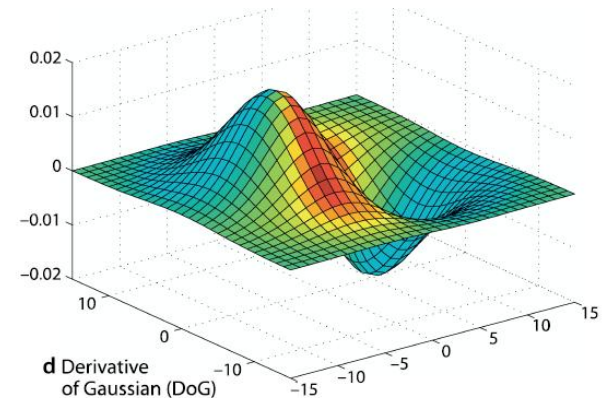
Magnitude do DoG, com $\sigma = 2$.

$$I_u = D \otimes I$$

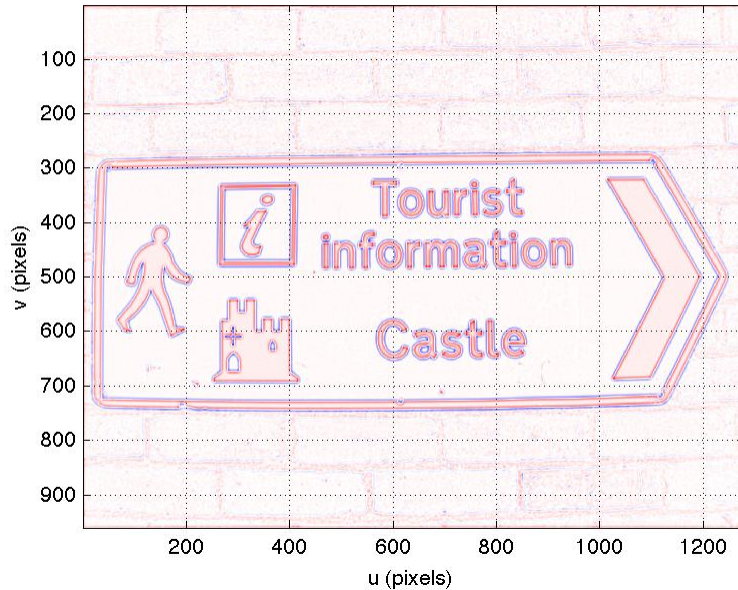
$$I_v = D^T \otimes I \quad D = \text{máscara como Sobel}$$

$$\nabla I = D \otimes (G(\sigma) \otimes I) = (D \otimes G(\sigma)) \otimes I = \text{DoG} \otimes I$$

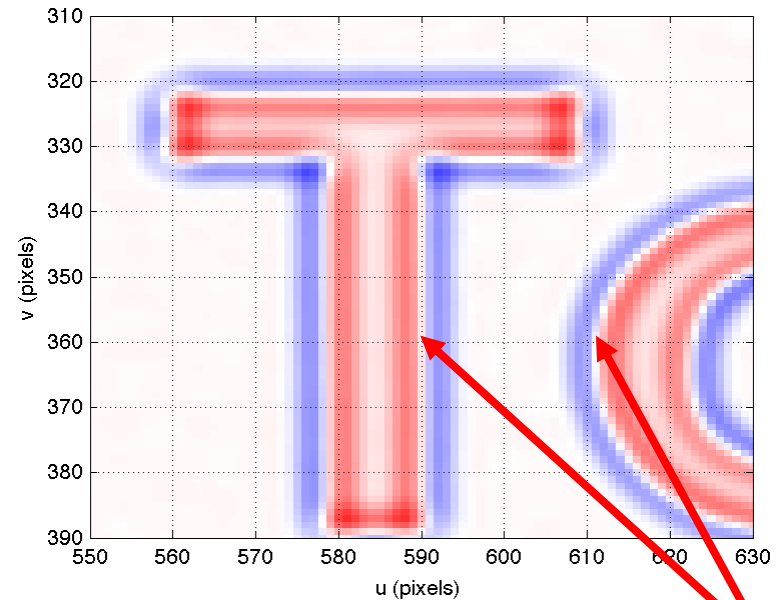
O operador $|\text{DoG}|$ exige menor custo computacional do que o Canny, mas gera bordas mais grossas



Operador LoG



LoG



Azul positivo, vermelho negativo

Valores nulos

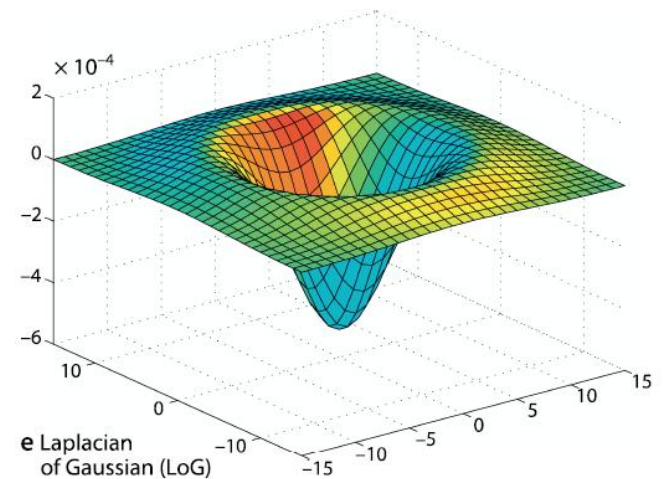
Operador LoG

$$\nabla^2 I = I_{uu} + I_{vv}$$

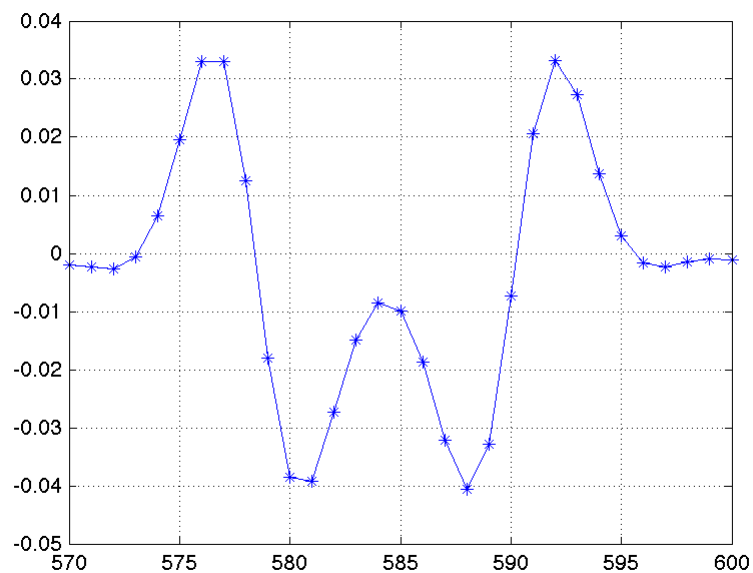
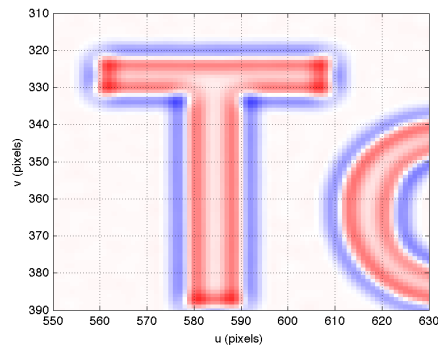
$$\nabla^2 I = L \otimes (G(\sigma) \otimes I) = (L \otimes G(\sigma)) \otimes I = \text{LoG} \otimes I$$

$$\text{LoG}(u, v) = \frac{\partial^2 G}{\partial u^2} + \frac{\partial^2 G}{\partial v^2} = \frac{1}{\pi \sigma^4} \left(\frac{u^2 + v^2}{2\sigma^2} - 1 \right) e^{-\frac{u^2 + v^2}{2\sigma^2}}$$

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \quad \text{Laplaciano}$$

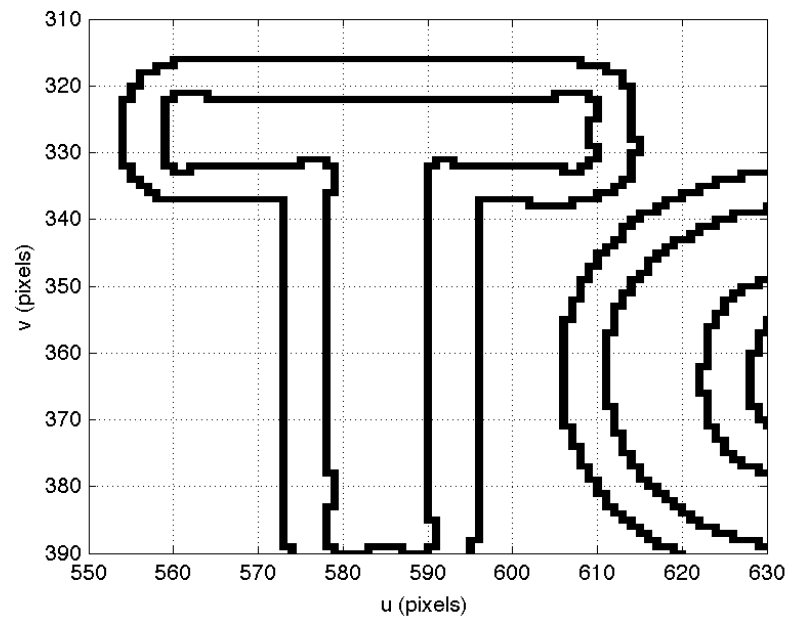


Operador LoG

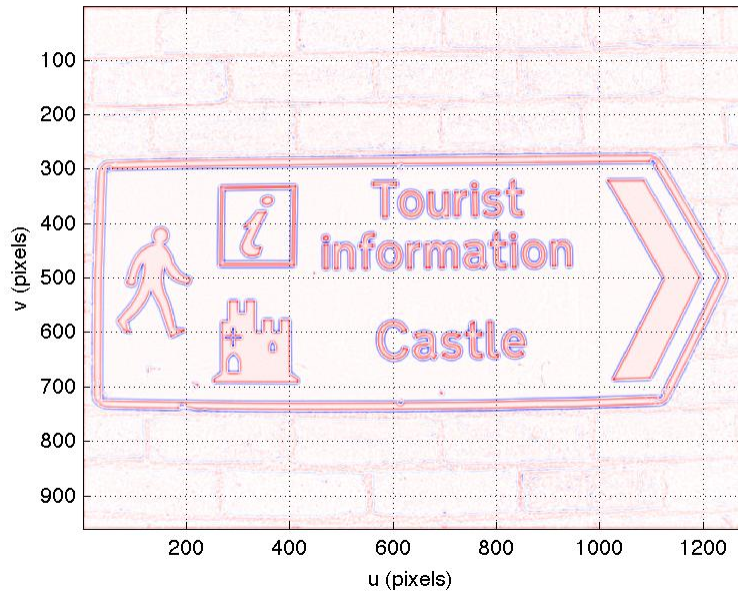


Perfil de seção horizontal do LoG na
seção mais estreita do “T”

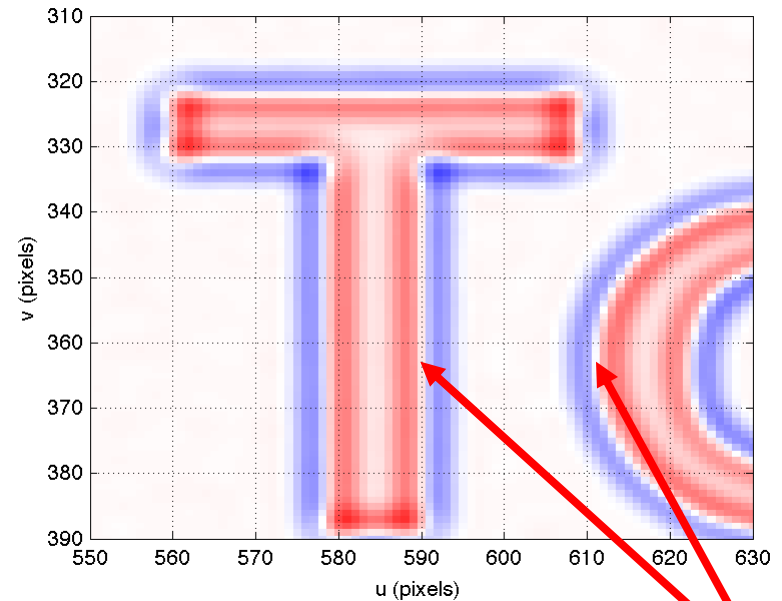
Máximo gradiente se localiza no cruzamento da segunda
derivada em zero.



Detector “zero-crossing”



LoG



Azul positivo, vermelho negativo

Valores nulos

Operador Diferença de Gaussianos

Aproxima o LoG

$$\text{DiffG}(u, v; \sigma_1, \sigma_2) = G(\sigma_1) - G(\sigma_2) = \frac{1}{2\pi\sigma_1^2\sigma_2^2} \left(\sigma_2^2 e^{-\frac{u^2+v^2}{2\sigma_1^2}} - \sigma_1^2 e^{-\frac{u^2+v^2}{2\sigma_2^2}} \right)$$

onde $\sigma_1 > \sigma_2$ e usualmente $\sigma_1 = 1,6 \cdot \sigma_2$

