

Análise e Processamento de Sinal e Imagem

IV - Processamento e Análise de Imagem

António M. Gonçalves Pinheiro

Departamento de Física
Universidade da Beira Interior
Covilhã - Portugal

pinheiro@ubi.pt

Processamento e Análise de Imagem

Processamento de Imagem - Introdução

1. Aquisição e Representação de Imagem
2. Convolução espacial e Filtragem de Imagem
3. Transformadas Bidimensionais
4. Análise Espectral de Imagem

Processamento de Imagem - Filtragem

1. Filtro FIR bidimensionais
2. Filtros estimadores óptimos bidimensionais (Wiener e Kalman)

Análise de Imagem

1. Técnicas básicas de Análise de Imagem
2. Morfologia de Imagem Binária e Multinível
3. Detectores de Arestas
4. Segmentação de Imagem
5. Descrição de Imagem

Introdução à Análise e Processamento de Imagem

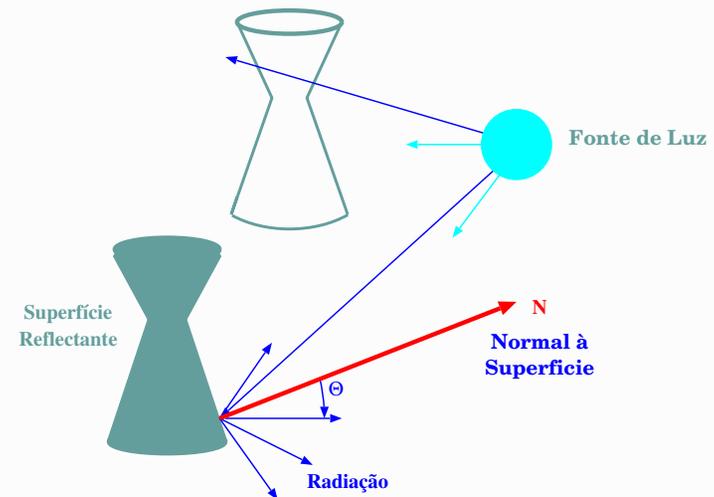
Bibliografia

1. Linda G. Shapiro and George C. Stockman, *Computer Vision*, Prentice Hall, 2001, ISBN: 0-13-030796-3
2. John W. Woods, *Multidimensional Signal, Image and Video Processing and Coding*, Elsevier Academic Press, 2006, ISBN: 0-12-088516-6.
3. William K. Pratt, *Digital Image Processing*, John Wiley & Sons, 3rd edition, 2001.
4. Páginas na Internet sugeridas:
 - ◇ <http://www.ph.tn.tudelft.nl/Courses/FIP/noframes/fip.html>
 - ◇ <http://mp7.watson.ibm.com/marvel/>
 - ◇ <http://www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm>

Introdução à Análise e Processamento de Imagem

Introdução

- Visão Humana oferece uma enorme percepção do Mundo que nos rodeia.
- A luz reflete/atraversa os objectos criando uma imagem na retina de cada um dos olhos.
- Um conjunto de componentes cria uma estrutura 3D na mente humana:
 - Cena com Objectos
 - Iluminação dos Objectos
 - Sensação da iluminação:
 - * luz refletida pelos objectos
 - * luz que atravessa os objectos



Introdução à Análise e Processamento de Imagem

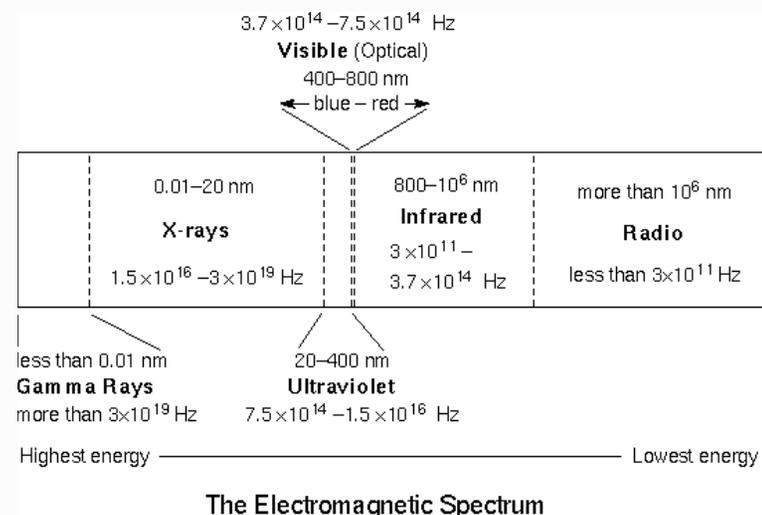
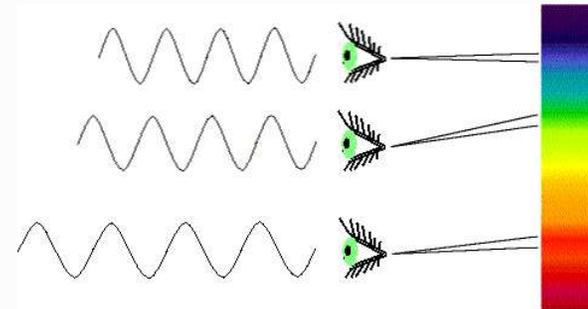
Sensores de Luz

Receptores químicos do olho humano são sensíveis à radiação (luminosa)

- Comprimentos de onda (λ) $\in [400\text{nm}, 700\text{nm}]$ (violeta ao vermelho);

Aparelhos sensíveis à Radiação Electromagnética

- Ondas de Rádio;
- Raios X;
- Microondas.

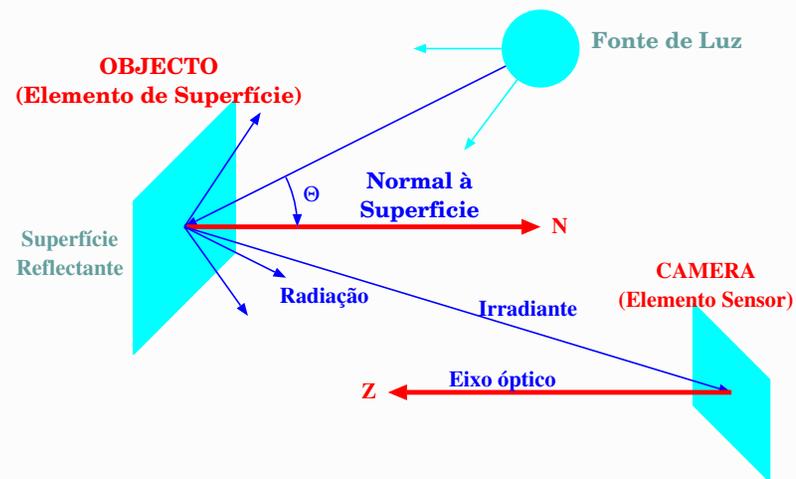


Introdução à Análise e Processamento de Imagem

Sensores de Luz

Equipamentos Detectores

- Sensores CCD (Charged Coupled Device) detectam luz visível; detectam $\lambda > 700\eta m$ (Infravermelhos);
- para λ muito pequeno: Raios X;
- para ondas muito longas.

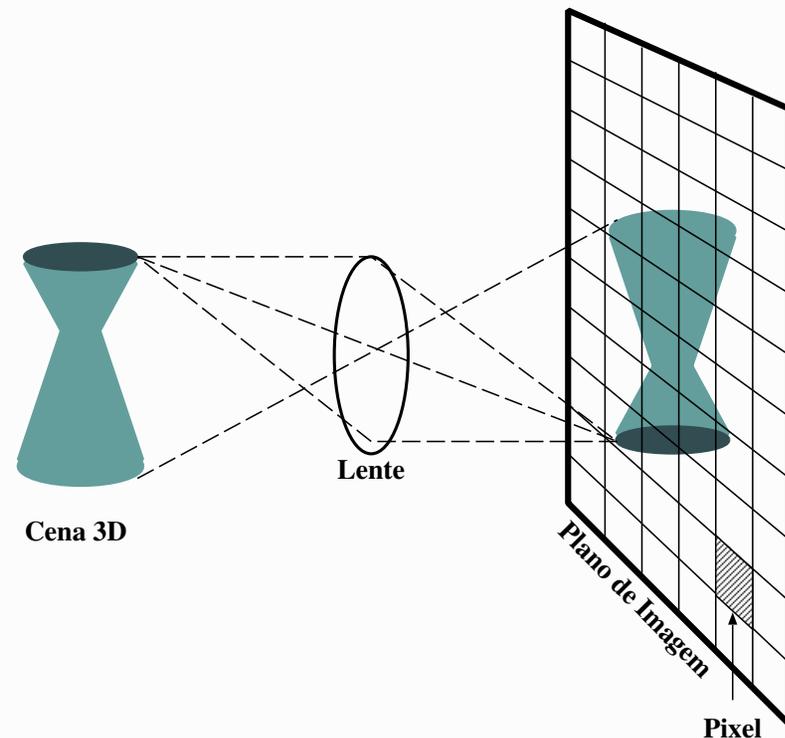


Introdução à Análise e Processamento de Imagem

Camera CCD

Composta por células de estado sólido que convertem energia da luz em energia eléctrica.

- Cada célula
 - converte a energia luminosa que recebe numa carga eléctrica;
 - gera um **pixel** de uma imagem digital;
- Forma-se uma imagem digital de $N \times M$
 - 512×512
 - 480×640
 - 768×1366
 - 1080×1920



Introdução à Análise e Processamento de Imagem

Camera CCD

Ampliação de Troços de Imagem

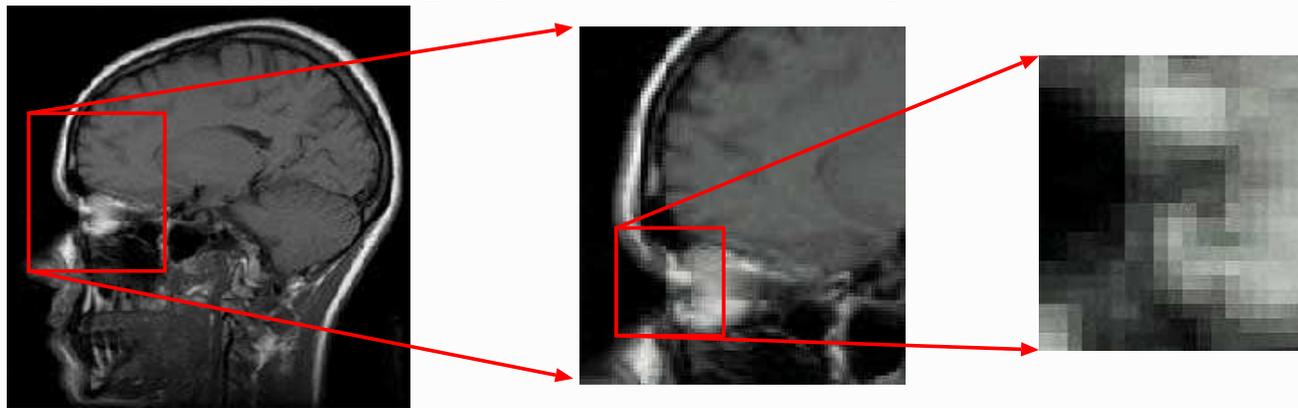


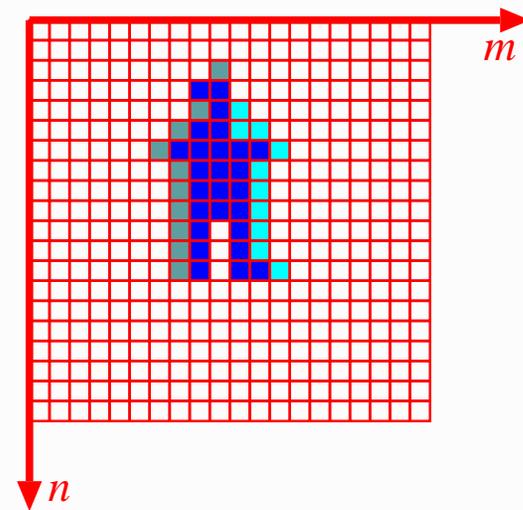
Imagem com diferentes definições



Introdução à Análise e Processamento de Imagem

Noções Básicas de Processamento de Imagem

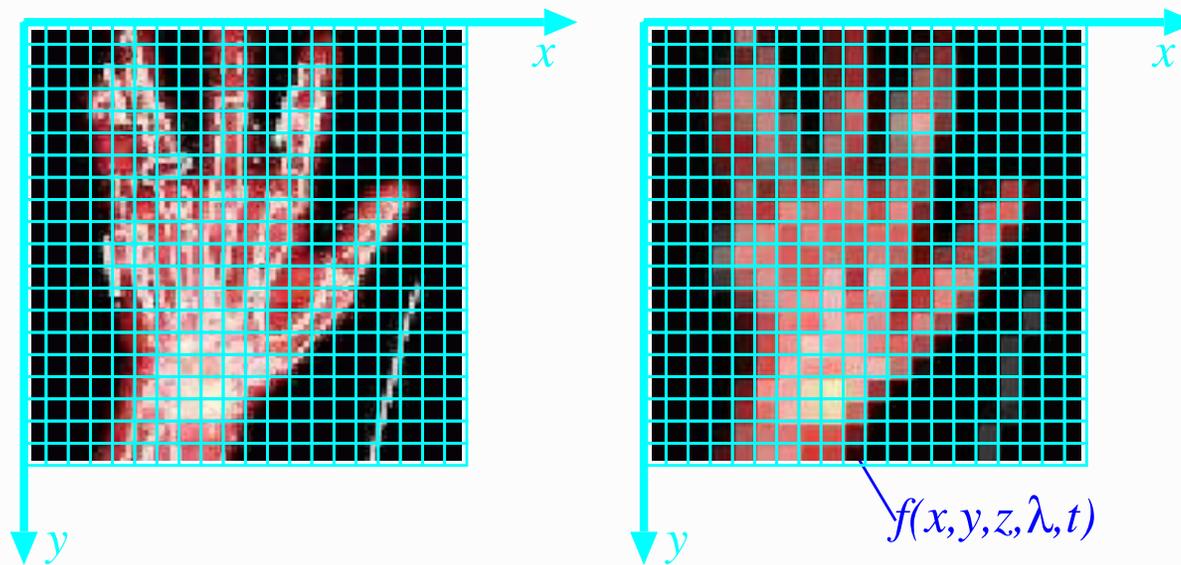
- Imagem considerada uma Matriz **I** de Pixels (**P**icture **E**lements)
- com N Linhas e M Colunas
- cada pixel $I[n, m]$ representa uma medida que depende de várias variáveis, como:
 - Cor (λ)
 - Profundidade (z)
 - tempo (t)



Introdução à Análise e Processamento de Imagem

Noções Básicas de Processamento de Imagem

A imagem obtida pelo sensor, vai passar por um processo de **quantificação** que a converte numa imagem digital de $\mathbf{N} \times \mathbf{M}$ pixels.



Introdução à Análise e Processamento de Imagem

Noções Básicas de Processamento de Imagem - Representação de Imagem

Imagem Representada por Matriz

$$\text{Img} = \begin{bmatrix}
 \text{Img}[0, 0] & \text{Img}[0, 1] & \dots & \dots & \text{Img}[0, M - 1] \\
 \text{Img}[1, 0] & \text{Img}[1, 1] & \dots & \dots & \text{Img}[1, M - 1] \\
 \text{Img}[2, 0] & \text{Img}[2, 1] & \dots & \dots & \text{Img}[2, M - 1] \\
 \dots & \dots & \dots & \dots & \dots \\
 \dots & \dots & \dots & \dots & \dots \\
 \text{Img}[N - 1, 0] & \text{Img}[N - 1, 1] & \dots & \dots & \text{Img}[N - 1, M - 1]
 \end{bmatrix}$$

em que:

- M é o número de colunas da imagem
(número de pixels de cada linha da imagem)
- N é o número de linhas da imagem
- $\text{Img}[n, m]$ representa a intensidade do pixel:

Introdução à Análise e Processamento de Imagem

Noções Básicas de Processamento de Imagem - Representação de Imagem

Intensidade do Pixel - $Img[n, m]$

1. Imagem Binária

- ◇ 0 → Fundo da imagem (Background)
- ◇ 1 → Relevo da imagem (Foreground)



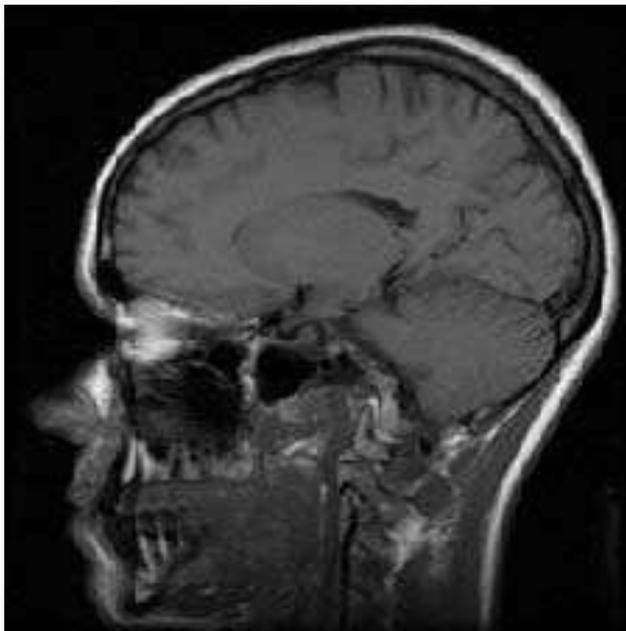
Introdução à Análise e Processamento de Imagem

Noções Básicas de Processamento de Imagem - Representação de Imagem

Intensidade do Pixel - $Img[n, m]$

2. Imagem Multinível

- ◇ 0 → Preto
- ◇ 255 → Branco
- ◇ valores intermédios originam níveis de cinzento intermédios



Introdução à Análise e Processamento de Imagem

Noções Básicas de Processamento de Imagem

Valores Comuns		
PARÂMETRO	SÍMBOLO	VALORES TÍPICOS
LINHAS	N	256, 512, 525 , 625 , 768, 900, 1024, 1080 , 2592
COLUNAS	M	256, 512, 768, 1024, 1440, 1366 , 1920 , 3888
NÚMERO DE NÍVEIS DE QUANTIFICAÇÃO	L	2, 64, 256 , 1024, 4096, $2^{16} = 65536$, 2^{24}

Introdução à Análise e Processamento de Imagem

Tecnologia Digital de Imagem

- Processamento de Imagem
Imagem à Entrada → Imagem à Saída
- Análise de Imagem
Imagem à Entrada → Medidas à Saída
- Compreensão de Imagem
Imagem à Entrada → Descrição de Alto Nível à Saída
- Codificação de Imagem
Imagem à Entrada → Fluxo Binário à Saída

Introdução à Análise e Processamento de Imagem

Noções Básicas de Processamento de Imagem

Imagem I_r que resulta de processar uma imagem I , pode resultar de um **processamento**:

- *Pontual* - $I_r[n, m] = f(I[n, m])$

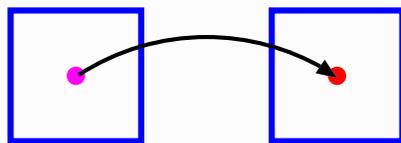
Em que o pixel resultante só depende do pixel da imagem original com as mesmas ordenadas.

- *Local* - $I_r[n, m] = f(I[j, i]), [j, i] \in \text{Vizinhança de } [n, m]$

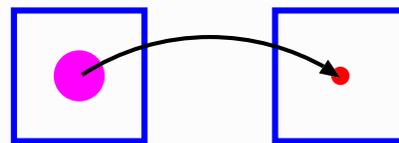
Em que o pixel resultante só depende dos pixels numa dada vizinhança do pixel da imagem original com as mesmas ordenadas.

- *Global* - $I_r[n, m] = f(I[j, i]), \forall [j, i]$

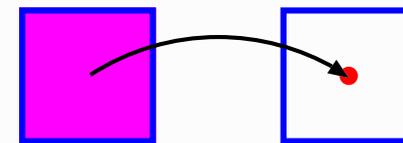
Em que o pixel resultante depende globalmente de todos os pixels da imagem original.



Pontual



Local



Global

Introdução à Análise e Processamento de Imagem

Formatos de Imagem

- *Formatos Computacionais com representação de pixel a pixel:*
 - **P**ortable **B**it **M**ap (PBM/PGM/PPM);
 - **B**MP (microsoft);
 - **T**ag **I**mage **F**ile **F**ormat (TIFF).
 - * Formato complexo e genérico;
 - * Geralmente usado pelas mesas digitalizadoras.

Introdução à Análise e Processamento de Imagem

Formatos de Imagem

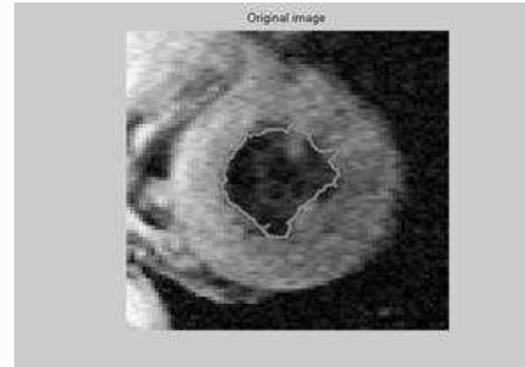
- *Formatos comprimidos:*
 - **Graphics Interchange Format (GIF)**
 - * 256 cores disponíveis (8 bits de codificação $\Rightarrow 2^8$);
 - * Compressão sem perdas Lempel-Ziv-Welch (LZW).
 - **Joint Photographic Expert Group (JPEG)**
 - * Pode codificar imagens de 64K \times 64K de 24 bits cada;
 - * Compressão com Perdas;
- *Vídeo: Moving Picture Expert Group (MPEG)*

Introdução à Análise e Processamento de Imagem

Ferramentas de Análise e Processamento de Imagem

Técnicas usadas:

- Convolução, Análise de Fourier e Análise Estatística
- Filtragem
- Análise de Cor e Textura
- Técnicas de Obtenção de Arestas e de Segmentação
- Reconhecimento de Padrões



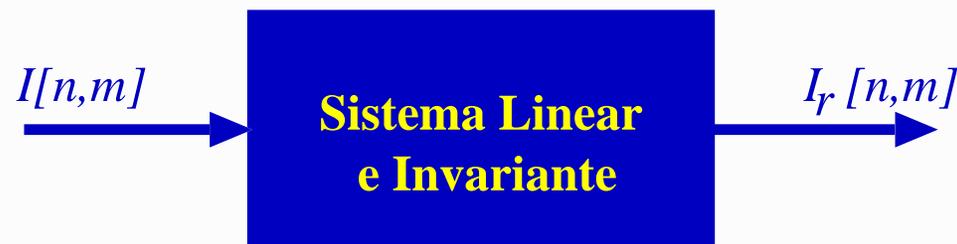
Introdução à Análise e Processamento de Imagem

Ferramentas de Processamento de Imagem

Convolução:

$$I_r[n, m] = I[n, m] \oplus h[n, m] = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} I[j, i] \cdot h[n - j, m - i]$$

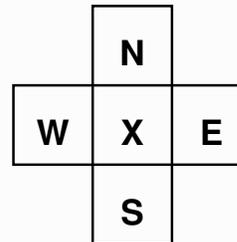
com $h[n, m]$ função impulsiva do Sistema Linear e Invariante



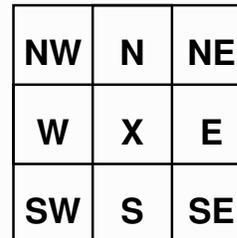
Introdução à Análise e Processamento de Imagem

Vizinhança de um pixel

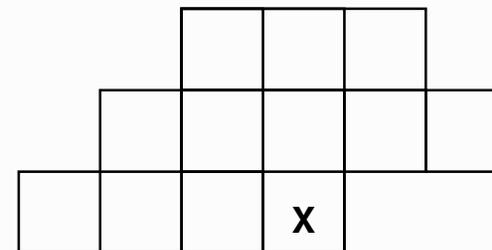
- Vizinhança de 4 pixels



- Vizinhança de 8 pixels



- Vizinhança genérica num sistema causal



Introdução à Análise e Processamento de Imagem

Exemplo de Máscaras

1	1	1
1	1	1
1	1	1

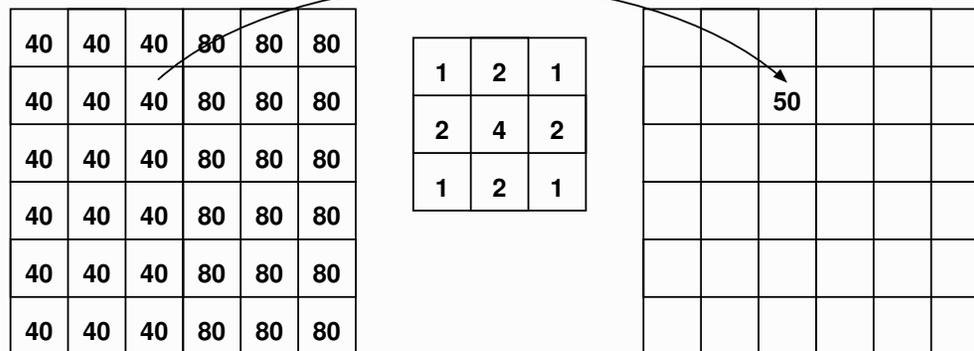
1	2	1
2	4	2
1	2	1

1
1
1
1

Introdução à Análise e Processamento de Imagem

Aplicação de Máscara a uma Imagem: Convolução

$$I_r[n, m] = I[n, m] \oplus h[n, m] = \sum_j \sum_i I[j, i] \cdot h[n - j, m - i]$$



Nota: As máscaras de convolução são normalmente normalizadas de forma a que a soma dos seus elementos seja 1. No caso da figura implica dividir todos os elementos por 16.

Introdução à Análise e Processamento de Imagem

Convolução

Exemplo usando a máscara anterior



Introdução à Análise e Processamento de Imagem

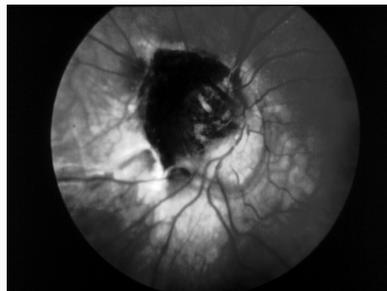
Transformada de Fourier - Resposta em Frequência

$$\mathbf{I}(\Omega, \Psi) = \mathcal{F}\{I[n, m]\} = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} I[n, m].e^{-j(\Omega n + \Psi m)}$$
$$I[n, m] = \mathcal{F}^{-1}\{\mathbf{I}(\Omega, \Psi)\} = \frac{1}{4\pi^2} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} \mathbf{I}(\Omega, \Psi).e^{j(\Omega n + \Psi m)} d\Psi d\Omega$$

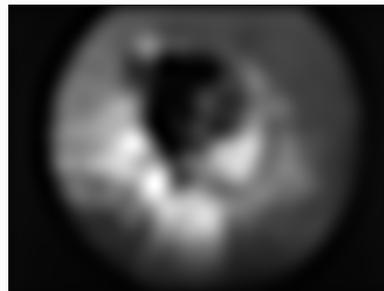
- Representa o sinal como uma soma pesada de exponenciais complexas $e^{jq} = \cos(q) + j\sin(q)$
- Representa a distribuição de Frequências da imagem

Introdução à Análise e Processamento de Imagem

Filtragem



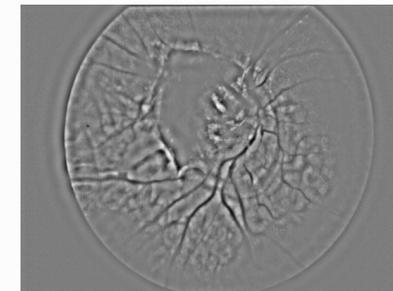
Original



passa-baixo



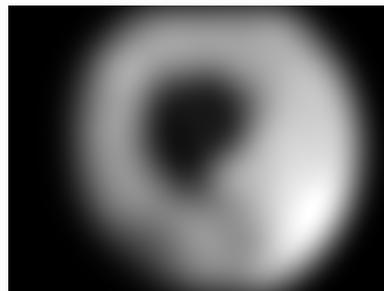
passa-banda



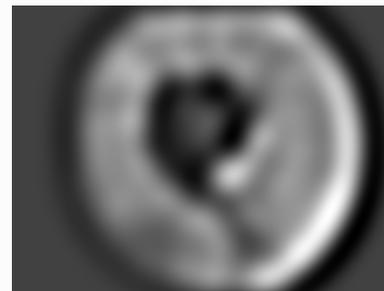
passa-alto



Original



passa-baixo



passa-banda



passa alto

Introdução à Análise e Processamento de Imagem

Ferramentas de Processamento de Imagem - Cor

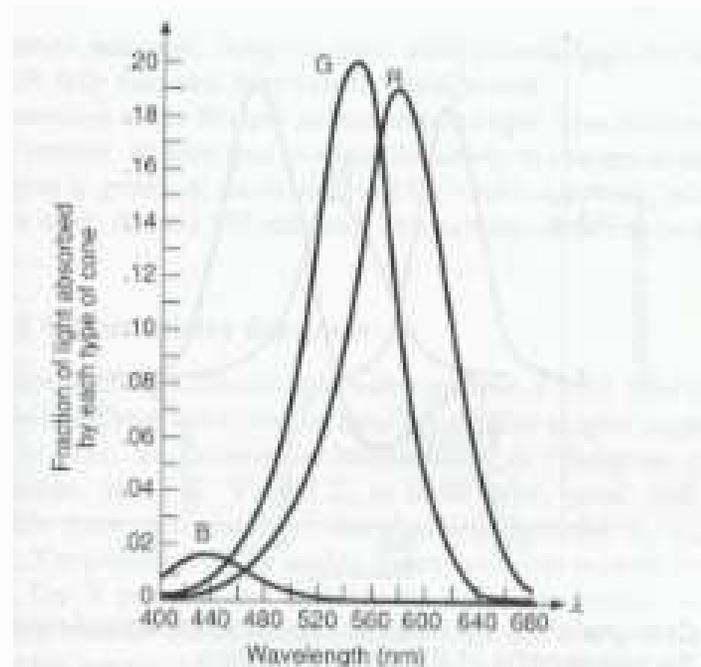
- Cor Monocromática - Resulta de apenas de um comprimento de onda
(A maioria das cores que nós vemos não são monocromáticas - Resultam da combinação de várias cores monocromáticas)
- Crominância - informação de cor;
- Luminância - mede a luz ou o brilho da cor.

Introdução à Análise e Processamento de Imagem

Ferramentas de Processamento de Imagem - Cor

Nos Humanos a percepção de cor resulta de:

- Três tipos de cones, sensíveis respectivamente à cor *Vermelha*, *Verde* e *Azul*
- Esta decomposição origina o modelo de cor RGB.

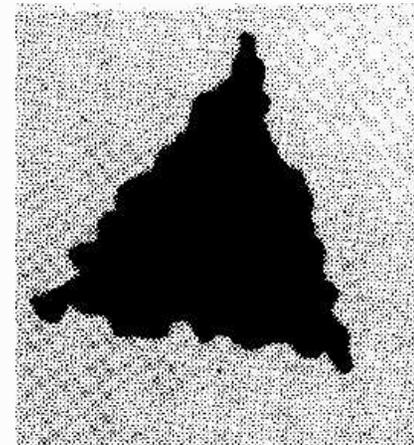
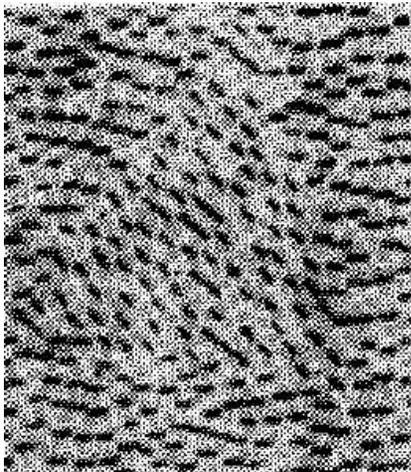


Introdução à Análise e Processamento de Imagem

Ferramentas de Processamento de Imagem - Texturas

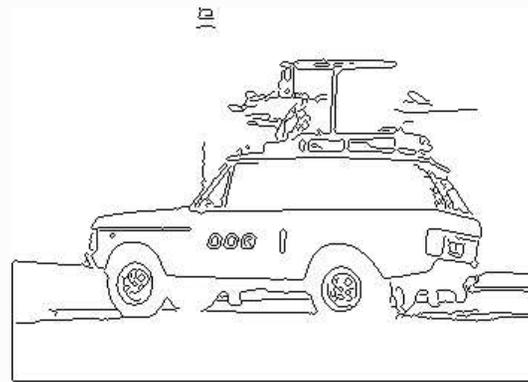
São usualmente processadas por filtros de Gabor (exemplo - reconhecimento baseado na íris do olho)
Várias características, das que se destacam:

- Frequência
- Direcção



Introdução à Análise e Processamento de Imagem

Ferramentas de Processamento de Imagem - Detecção de Arestas e Segmentação



Processamento de Imagem

Sinais Bi-dimensionais Discretos

$f[n, m]$ - sucessão complexa, $Z^2 \rightarrow C$

Impulso:

$$\delta[n, m] = \begin{cases} 1, & \text{para } [n, m] = [0, 0] \\ 0, & \text{para } [n, m] \neq [0, 0] \end{cases}$$

Função Escalão:

$$u[n, m] = \begin{cases} 1, & \text{para } n \geq 0 \text{ e } m \geq 0 \\ 0, & \text{caso contrário} \end{cases}$$

Processamento de Imagem

Sinais Separáveis

$$x[n, m] = x_1[n] x_2[m] \text{ para qualquer } n \text{ e } m$$

Matricialmente:

$$\mathbf{x} = \mathbf{x}_1 \mathbf{x}_2^T$$

Periodicidade

1 - *Periodicidade Rectangular*: $x[n, m] = x[n + N, m] = x[n, m + M]$ Período $N \times M$

2 - *Periodicidade Geral*: Sendo $\mathbf{N} = \begin{bmatrix} N_1 \\ N_2 \end{bmatrix}$ e $\mathbf{M} = \begin{bmatrix} M_1 \\ M_2 \end{bmatrix}$. Neste caso:

$$x[n, m] = x[n + N_1, m + M_1] = x[n + N_2, m + M_2] \text{ com } \det(\mathbf{N}, \mathbf{M}) \neq 0$$

Forma matricial

$$x[\mathbf{n}] = x[\mathbf{n} + \tilde{\mathbf{N}}\mathbf{r}] \text{ com } \tilde{\mathbf{N}} = [\mathbf{N} \mathbf{M}]$$

Processamento de Imagem

Sistemas Lineares e Invariantes (ao Deslocamento)

Sistema Linear:

$$T \{a_1x_1[n, m] + a_2x_2[n, m]\} = a_1T \{x_1[n, m]\} + a_2T \{x_2[n, m]\}$$

Sistema Invariante (ao Deslocamento):

$$y[n, m] = T \{x[n, m]\} \Rightarrow y[n - n_0, m - m_0] = T \{x[n - n_0, m - m_0]\}$$

Convolução Bidimensional:

$$x[n, m] \oplus h[n, m] = \sum_k \sum_l x[k, l]h[n - k, m - l]$$



Processamento de Imagem

Propriedades da Convolução

1) Comutatividade: $x \oplus y = y \oplus x$

2) Associatividade: $x \oplus (y \oplus z) = (x \oplus y) \oplus z$

3) Distributividade: $x \oplus (y + z) = x \oplus y + x \oplus z$

4) Elemento Identidade: $\delta[n, m] \Rightarrow \delta \oplus x = x$

5) Elemento Absorvente: $O[n, m] \Rightarrow O \oplus x = 0$

em que $O[n, m] = 0$ para qualquer $[n, m]$.



Processamento de Imagem

Estabilidade de Sistemas 2-D

Um Sistema Estável é aquele em que uma pequena alteração na entrada, resulta numa pequena alteração à saída.

Estabilidade BIBO:

BIBO - “Bounded Input Bounded Output”

Num sistema LSI corresponde a dizer que:

$$\sum_k \sum_l |h[k, l]| < +\infty$$

Processamento de Imagem

Transformada de Fourier 2D

$$X(V, U) = \sum_{n=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} x[n, m] e^{-j(Vn+Um)}$$

Transformada de Fourier de Sinal Discreto é periódica:

$$X(V, U) = X(V \pm 2\pi, U \pm 2\pi)$$

Transformada Inversa:

$$x[n, m] = \frac{1}{(2\pi)^2} \int_{-\pi}^{+\pi} \int_{-\pi}^{+\pi} X(V, U) e^{+j(Vn+Um)} dV dU$$

Processamento de Imagem

Propriedades da Transformada de Fourier 2D

1) Linearidade

$$\alpha x[n, m] + \beta y[n, m] \xleftrightarrow{TF} \alpha X(V, U) + \beta Y(V, U)$$

2) Convolução

$$x[n, m] * y[n, m] \xleftrightarrow{TF} X(V, U) Y(V, U)$$

3) Multiplicação

$$x[n, m] y[n, m] \xleftrightarrow{TF} (X * Y)(V, U) = \frac{1}{(2\pi)^2} \int_{-\pi}^{+\pi} \int_{-\pi}^{+\pi} X(v, \nu) Y(V - v, U - \nu) d\nu dv$$

4) Modulação

$$x[n, m] e^{j(\nu n + \nu m)} \xleftrightarrow{TF} X(V - \nu, U - \nu), \text{ com } \nu, \nu \text{ inteiros}$$

5) Deslocamento

$$x[n - k, m - l] \xleftrightarrow{TF} X(V, U) e^{-j(Vk + Ul)}, \text{ com } k, l \text{ inteiros}$$



Processamento de Imagem

Propriedades da Transformada de Fourier 2D

6) Diferenciação no domínio da Frequência

$$-jnx[n, m] \stackrel{TF}{\leftrightarrow} \frac{\partial X(V, U)}{\partial V} \quad -jm x[n, m] \stackrel{TF}{\leftrightarrow} \frac{\partial X(V, U)}{\partial U}$$

7) Valor inicial

$$x[0, 0] = \frac{1}{(2\pi)^2} \int_{-\pi}^{+\pi} \int_{-\pi}^{+\pi} X(V, U) dV dU$$

8) Valor DC

$$X(0, 0) = \sum_{n=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} x[n, m]$$

9) Sinal Separável

$$x[n, m] = x_1[n] x_2[m] \stackrel{TF}{\leftrightarrow} X_1(V) X_2(U)$$

Processamento de Imagem

Propriedades da Transformada de Fourier 2D

10) Teorema de Parseval

$$\sum_{n=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} x[n, m] y^*[n, m] = \frac{1}{(2\pi)^2} \int_{-\pi}^{+\pi} \int_{-\pi}^{+\pi} X(V, U) Y^*(V, U) dV dU$$

Potência

$$\sum_{n=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} |x[n, m]|^2 = \frac{1}{(2\pi)^2} \int_{-\pi}^{+\pi} \int_{-\pi}^{+\pi} |X(V, U)|^2 dV dU$$

Processamento de Imagem

Pares de Transformada de Fourier 2D

1) Filtro passa-baixo ideal (banda quadrada)

$$H_s(V, U) = I_{U_c}(V)I_{U_c}(U) \text{ com } (V, U) \in [-\pi, \pi] \times [-\pi, \pi],$$

$$\text{com } I_{U_c} = \begin{cases} 1 & |U| < U_c, |V| < U_c \\ 0 & \text{caso contrário} \end{cases}$$

$$h_s[n, m] = \frac{\text{sen}(U_c n)}{\pi n} \frac{\text{sen}(U_c m)}{\pi m} \quad -\infty < n, m < +\infty$$

2) Filtro passa-baixo ideal (banda circular)

$$H_c(V, U) = \begin{cases} 1 & \sqrt{V^2 + U^2} < U_c \\ 0 & \text{caso contrário} \end{cases} \text{ com } (V, U) \in [-\pi, \pi] \times [-\pi, \pi],$$

$$h_c[n, m] = \frac{1}{(2\pi)^2} \int_0^{U_c} u \int_{-\pi}^{+\pi} e^{j u r \cos(\theta)} d\theta du = \frac{U_c}{2\pi\sqrt{n^2 + m^2}} J_1 \left(U_c \sqrt{n^2 + m^2} \right)$$

Processamento de Imagem

Transformada Z Bidimensional

$$X(Z, W) = \sum_{n=-\infty}^{+\infty} \sum_{m=-\infty}^{+\infty} x[n, m] Z^{-n} W^{-m}$$

Inversa:

$$x[n, m] = \frac{1}{(2\pi j)^2} \oint_{C_1} \oint_{C_2} X(Z, W) Z^{n-1} W^{m-1} dZ dW$$

Introdução à Análise e Processamento de Imagem

Transformada Discreta de Fourier - DFT - Bidimensional

$$F[K, L] = \begin{cases} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} I[n, m] e^{-j2\pi\left(\frac{nK}{N} + \frac{mL}{M}\right)}, & (K, L) \in [0, N-1] \times [0, M-1] \\ 0 & \text{caso contrário} \end{cases}$$

Transformada Discreta de Fourier Bidimensional - DFT - Inversa

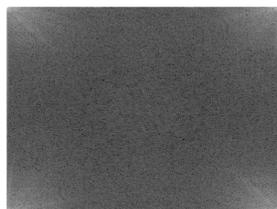
$$I[n, m] = \begin{cases} \frac{1}{NM} \sum_{K=0}^{N-1} \sum_{L=0}^{M-1} F[K, L] e^{j2\pi\left(\frac{nK}{N} + \frac{mL}{M}\right)}, & [n, m] \in [0, N-1] \times [0, M-1] \\ 0 & \text{caso contrário} \end{cases}$$

Introdução à Análise e Processamento de Imagem

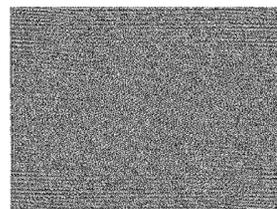
Transformadas Discretas de Fourier



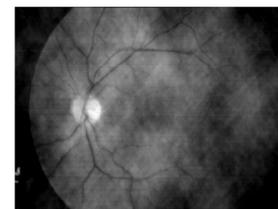
Imagem 1



Log do módulo da DFT da imagem 1



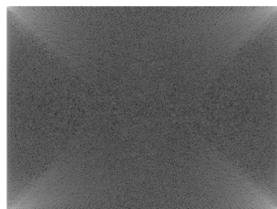
Fase da DFT da imagem 1



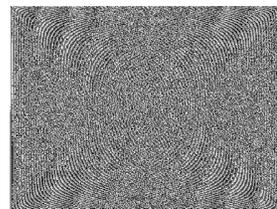
DFT^{-1} obtida com módulo da image 2 e fase da imagem 1



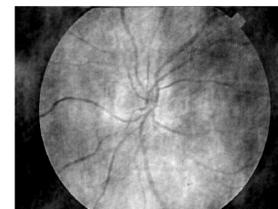
Imagem 2



Log do módulo da DFT da imagem 2



Fase da DFT da imagem 2



DFT^{-1} obtida com módulo da image 1 e fase da imagem 2

Desenhos de Filtros FIR

- Filtro Ideal - $h_I[n, m]$
- Janela - $w[n, m]$
- Filtro FIR - $h[n, m]$

$$h[n, m] = w[n, m]h_I[n, m] \xleftrightarrow{\text{TF}} H(V, U) = W(V, U) \circledN H_I(V, U)$$

com \circledN - convolução periódica com período fundamental $[-\pi, \pi] \times [-\pi, \pi]$

$$H(V, U) = \frac{1}{(2\pi)^2} \int \int_{[-\pi, \pi] \times [-\pi, \pi]} W(S, R) H(V - S, U - R) dS dR$$

Dimensionamento de Filtros Bidimensionais

Desenhos de Filtros FIR

Propriedades desejadas para $w[n, m]$

- $w[n, m]$ deve estar num rectângulo de $N \times M$.
- $w[n, m]$ deve aproximar uma função com simetria circular (sob o seu centro) e deve ser real.
- O volume de $w[n, m]$ deve ser concentrado no domínio espacial.
- O volume de $W(V, U)$ deve ser concentrado no domínio de frequência.

Dimensionamento de Filtros Bidimensionais

Janelas Rectangulares separáveis

$$w_S[n, m] = w_1[n] w_2[m]$$

Nota: Utilização de janelas 1-D

Janelas Circulares

$$w_C[n, m] = w_1 \left[\sqrt{n^2 + m^2} \right]$$

Nota: Oferece melhor simetria circular

Algoritmos básicos de Processamento de Imagem

Contagem do Número de Objectos numa Imagem Binária

1. *Cantos Externos*

Um canto externo é contabilizado cada vez que um pixel e a sua vizinhança coincida com uma destas máscaras:

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

2. *Cantos Internos*

Um canto interno é contabilizado cada vez que um pixel e a sua vizinhança coincida com uma destas máscaras:

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

$$\text{Número de Objectos} = \|\text{Núm. de Cantos Externos} - \text{Núm. de Cantos Internos}\|/4$$

Algoritmos básicos de Processamento de Imagem

Etiquetar Componentes Conectadas

Permite definir áreas conectadas entre si e referi-las por uma Etiqueta. Dois Algoritmos:

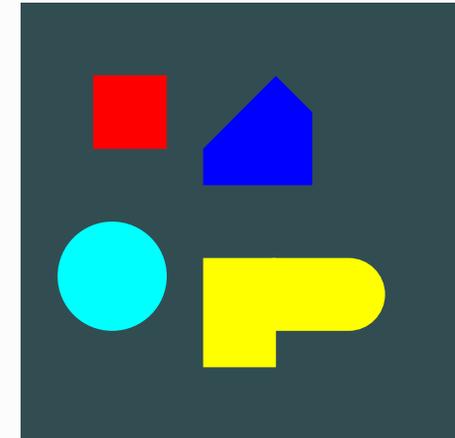
1) *Algoritmo Recursivo*

Faz pesquisa numa vizinhança de quatro pixels das componentes conectadas, até que todos os pixels conectados tenham a mesma etiqueta.

2) *Algoritmo Linha a Linha*

Vai pesquisar pixel a pixel, linha a linha por conexões.

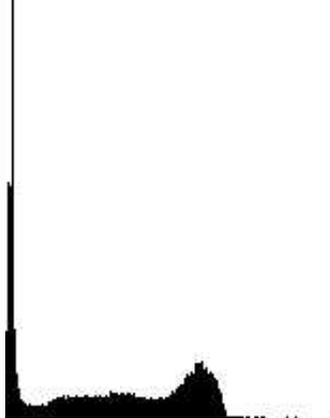
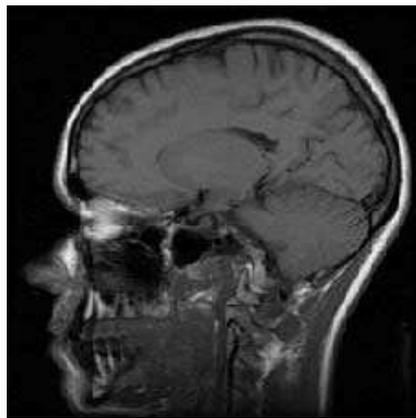
Depois de executado, faz uma re-etiquetagem de regiões, que sendo contíguas, e tendo o mesmo valor de pixel, foram colocadas em etiquetas diferentes.



Algoritmos básicos de Processamento de Imagem

Histograma da imagem

Gráfico que contabiliza o número de vezes que aparece cada cor.



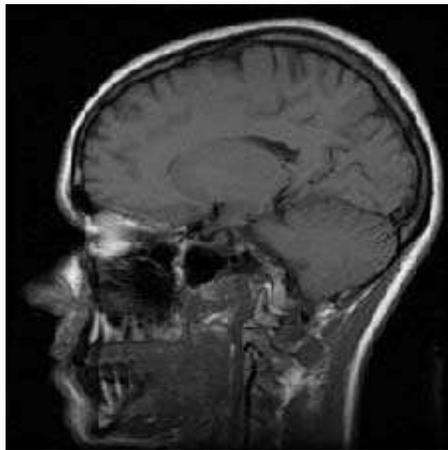
Algoritmos básicos de Processamento de Imagem

Quantificação de imagem

Um algoritmo de quantificação selecciona as zonas do histograma que vão ser quantificados na mesma cor, baseada na distribuição de cores.

Exemplo:

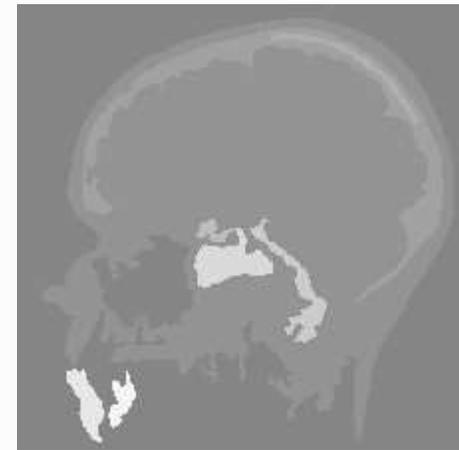
M. I. Sezan, “A peak detection algorithm and it’s application to histogram-based image data reduction,” *Computer Vision, Graphics, and Image Processing*, vol. 49, pp. 36–51, 1990.



original



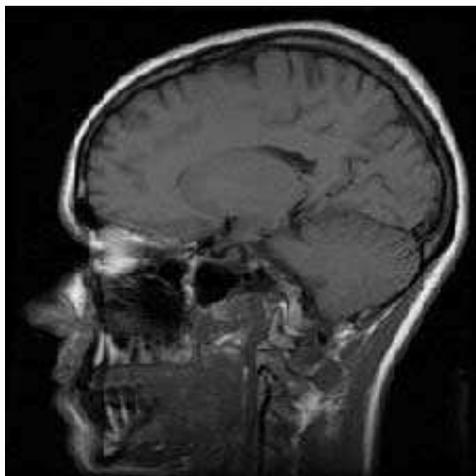
2 níveis



8 regiões

Algoritmos básicos de Processamento de Imagem

Quantificação de imagem



original



3 níveis



24 regiões

Técnicas de Enriquecimento de Imagem

Mapeamento dos Cinzentos

Alterar o mapeamento dos cinzentos pode tornar uma imagem muito mais nítida:

- Espalhamento do mapa de cores
- Correção Gama (e outras)
- Equalização do histograma

Definição de Operador de Pixel

Um Operador de Pixel determina cada pixel da imagem resultante como uma função do Pixel da imagem original, ou seja,

$$I_r[n, m] = f(I[n, m])$$

Técnicas de Enriquecimento de Imagem

Espalhamento do mapa de cores

Quando uma imagem é representada por um número de cores inferior ao número total de cores, concentradas entre um valor $\text{Min} \{I[n, m]\}$ e $\text{Max} \{I[n, m]\}$, estas cores são espalhadas pelo conjunto de cores disponíveis. Isto permite que as diferentes cores sejam realçadas entre si.

Operador de Pixel:

$$I_r[n, m] = (I[n, m] - \text{Min} \{I[n, m]\}) \times \frac{255}{\text{Max} \{I[n, m]\} - \text{Min} \{I[n, m]\}}$$

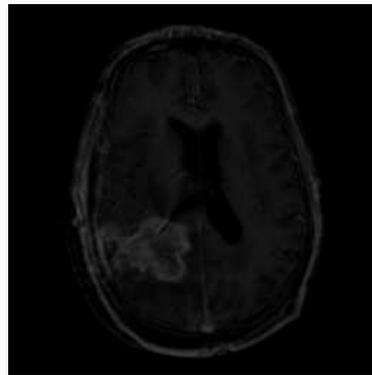


Imagem representada por 16 cores.

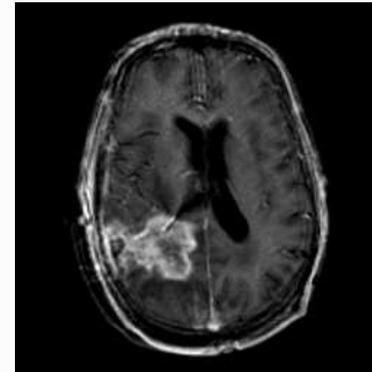


Imagem representada com espalhamento das 16 cores.

Técnicas de Enriquecimento de Imagem

Correcção Gamma

Muitas vezes devido às condições de aquisição as imagens aparecem muito escurecidas. A correcção Gamma atribui às cores mais escuras cores mais claras.

Operador de Pixel:

$$f(x) = x^{1/\gamma} \Rightarrow I_r[n, m] = 256 \left(\frac{I[n, m]}{256} \right)^{1/\gamma}$$

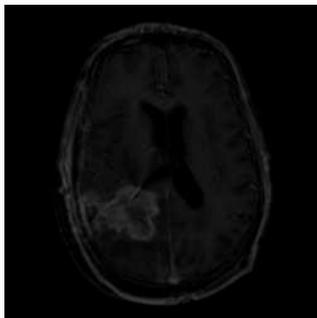
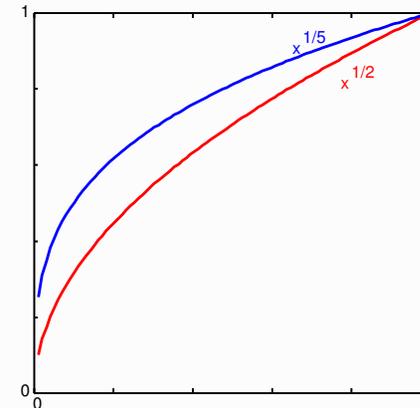
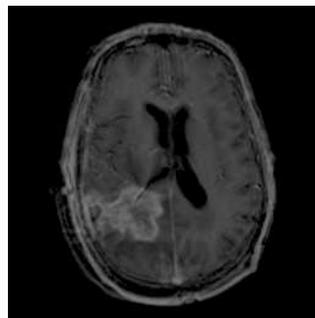
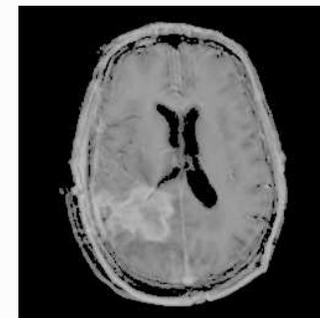


Imagem original.



Transformação gamma
($\gamma = 2$).



Transformação gamma
($\gamma = 5$).

Técnicas de Enriquecimento de Imagem

Equalização de imagem

Um algoritmo de quantificação selecciona as zonas do histograma que vão ser quantificados na mesma cor, baseada na distribuição de cores.

Exemplo:

M. I. Sezan, “A peak detection algorithm and it’s application to histogram-based image data reduction,” *Computer Vision, Graphics, and Image Processing*, vol. 49, pp. 36–51, 1990.

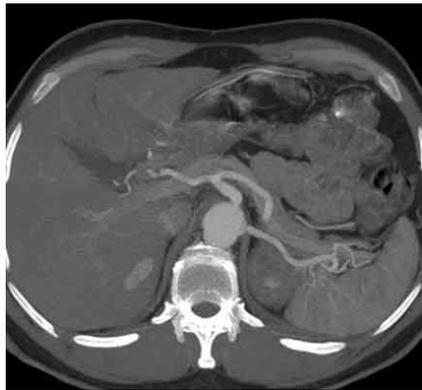
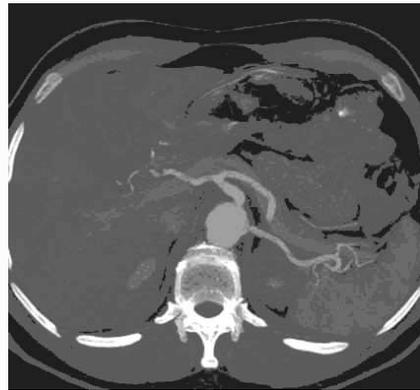
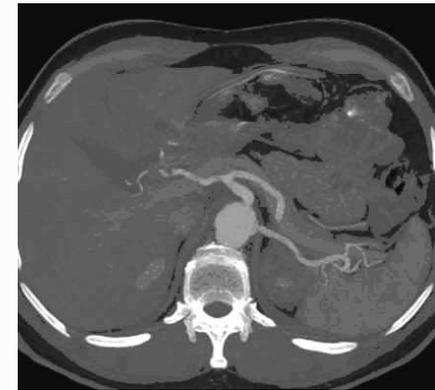


Imagem original.



Equalização em 26 cores.



Equalização em 43 cores.

Enriquecimento e Filtragem de Imagem

Remoção de Ruído

Imagens binárias são muitas vezes corrompidas por ruído “Salt & Pepper”.

Usa-se uma máscara com os oito vizinhos; Se os oito vizinhos são da mesma cor e o pixel tiver uma cor diferente é considerado que se está na presença de ruído e a cor do pixel é alterada.

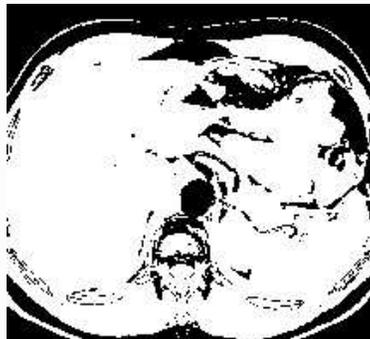
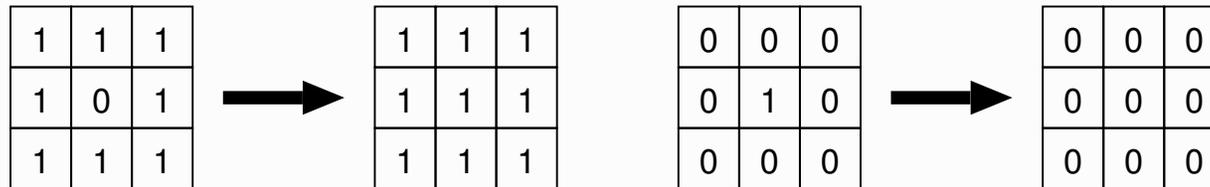


Imagem original.

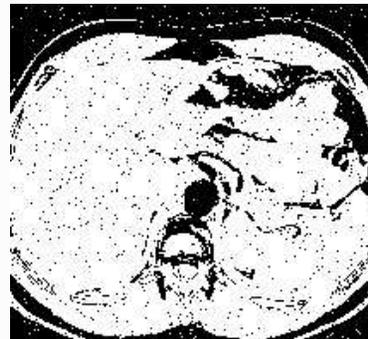


Imagem com Ruído Salt & Pepper.

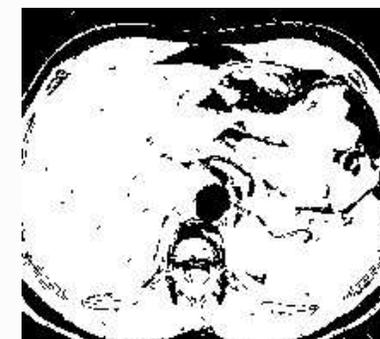


Imagem depois da remoção do Ruído

Enriquecimento e Filtragem de Imagem

Suavização da Imagem

Suavizar uma imagem permite suavizar o efeito do ruído.

Filtro “box”(médio):

$$I_r[n, m] = \frac{1}{K^2} \sum_{i=-K}^K \sum_{j=-K}^K I[n - j, m - i]$$

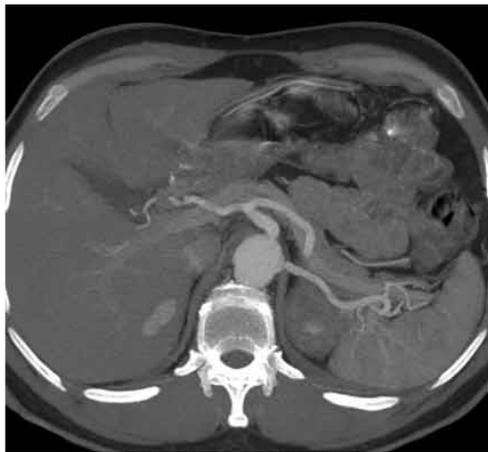


Imagem original



Imagem com aditivo ruído
gaussiano

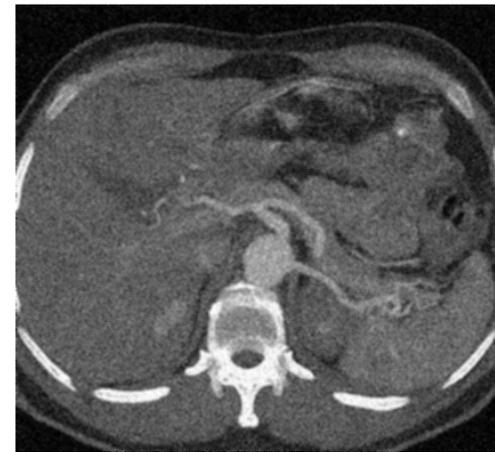


Imagem filtrada com filtro
médio de dimensão 3 - $K = 1$

Enriquecimento e Filtragem de Imagem

Filtro Mediano

- ◇ Os pixels na janela do filtro são ordenados por ordem crescente.
- ◇ O pixel do meio dessa ordem é seleccionado para substituir o pixel $I[n, m]$.



Imagem original



Imagem com aditivo ruído gaussiano



Imagem resultante com filtro mediano de dimensão 3

Enriquecimento e Filtragem de Imagem

Filtro Gaussiano

O filtro Gaussiano faz uma média ponderada pela função gaussiana

$$g(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-d^2/(2\sigma^2)} \quad \text{com} \quad d = \sqrt{(x - x_c)^2 + (y - y_c)^2}$$

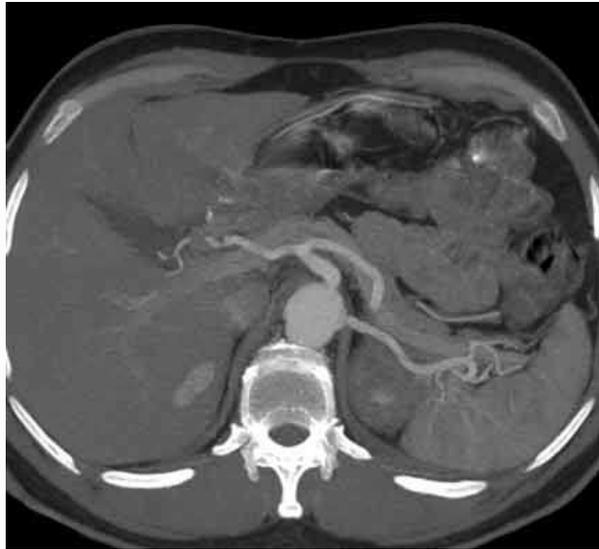


Imagem original

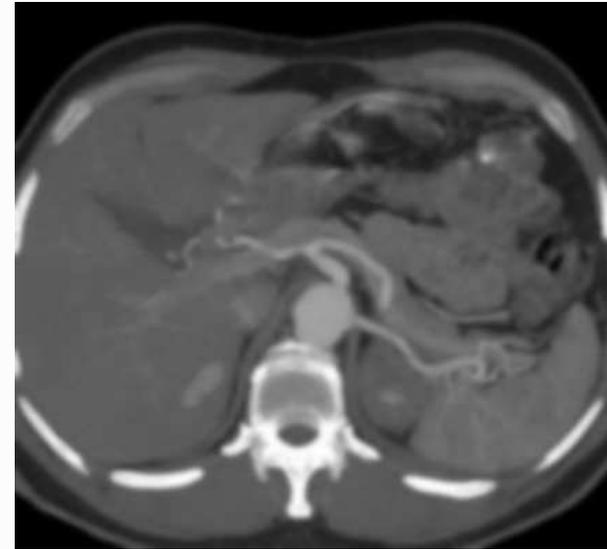


Imagem filtrada ($\sigma = 2$)

Enriquecimento e Filtragem de Imagem

Filtro Gaussiano



Imagem original

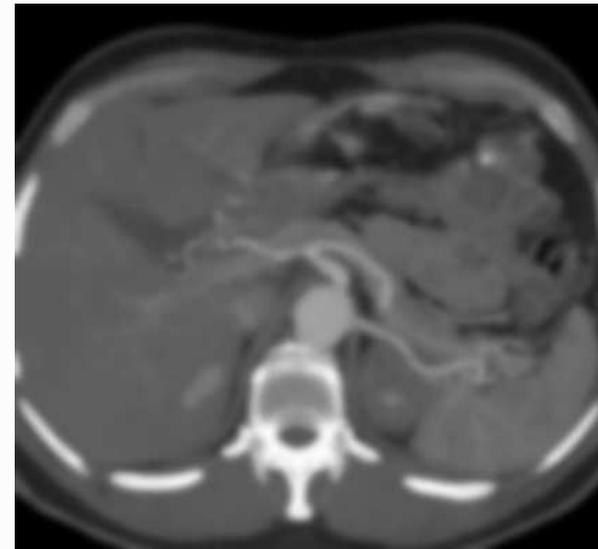


Imagem filtrada ($\sigma = 2\sqrt{2}$)

Enriquecimento e Filtragem de Imagem

Filtro Gaussiano



Imagem original

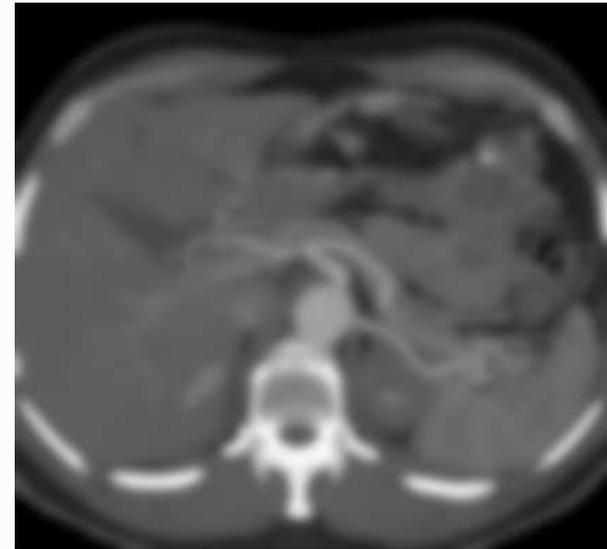


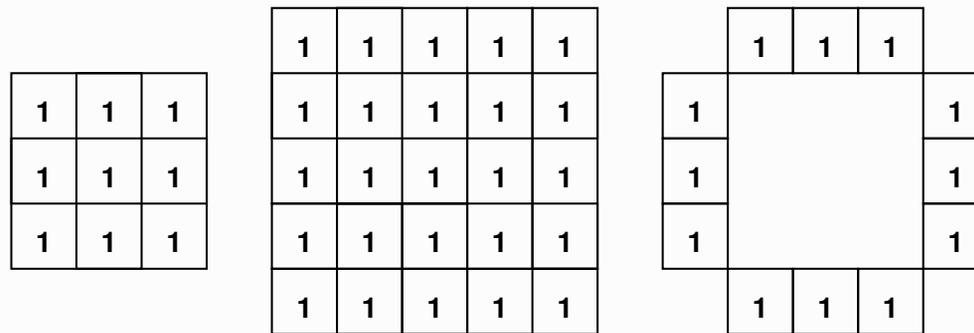
Imagem filtrada ($\sigma = 4$)

Morfologia de Imagem

- Binária
- Multinível

Elementos Estruturantes - S

Baseia-se num elemento estruturante S que representa uma forma



Exemplos de elementos estruturantes binários.

Morfologia de Imagem

Morfologia de Imagem Binária

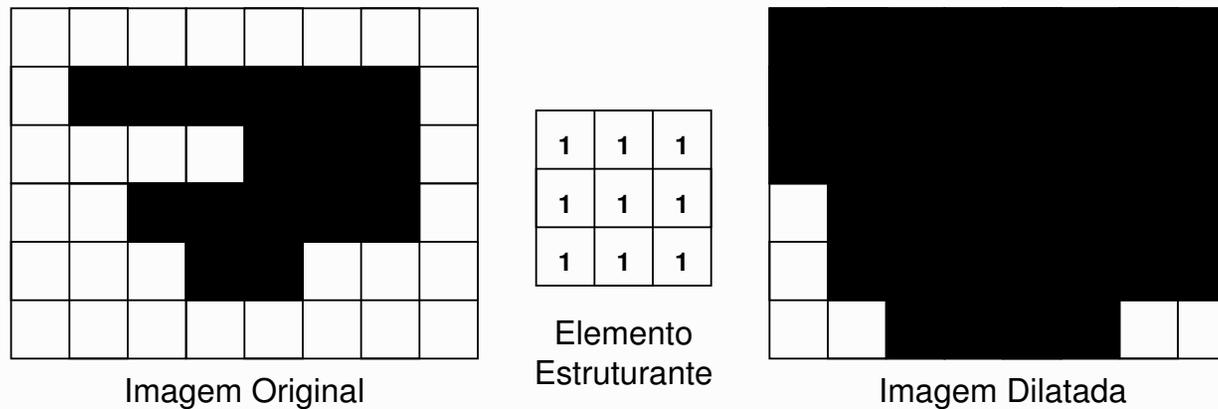
As Operações Morfológicas sob Imagens Binárias mais importantes são:

- Dilatação → *Alarga as regiões*
- Erosão → *Diminui as regiões*
- Fecho (“Closing”) → *Tende a fechar buracos interiores interiores à região e a eliminar as baías nos limites da região*
- Abertura (“Opening”) → *Tende a retirar pequenas porções ou regiões que saem dos limites*

Morfologia de Imagem

Morfologia de Imagem Binária - Dilatação

A dilatação binária coloca cada pixel de imagem resultante a 1 desde que a imagem original tenha dentro do Elemento Estruturante um pixel a 1.



Exemplo de Dilatação Morfológica Binária.

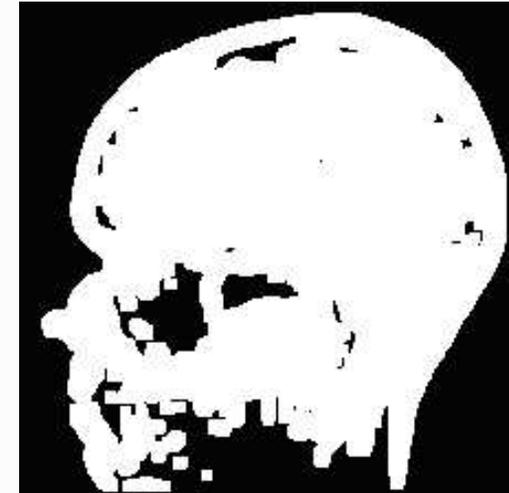
Morfologia de Imagem

Morfologia de Imagem Binária - Dilatação



1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Elemento
Estruturante

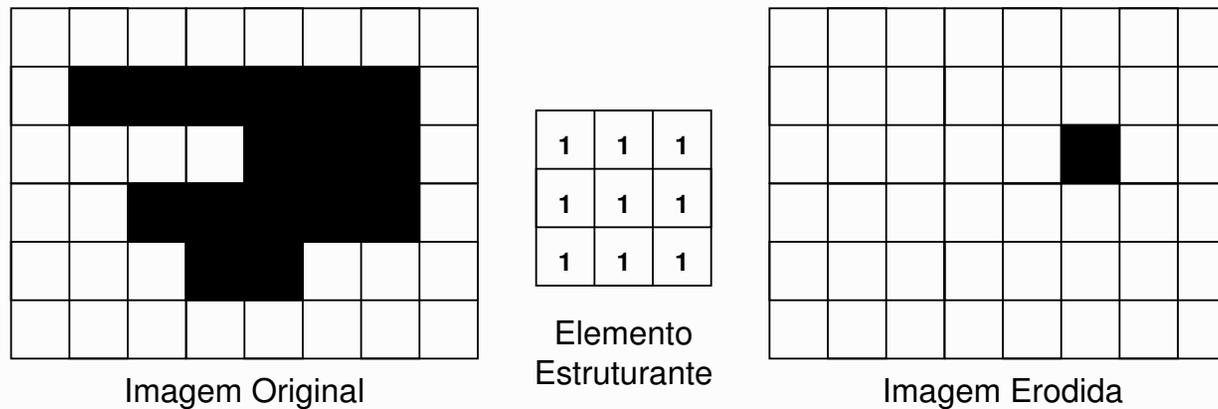


Exemplo de Dilatação Morfológica Binária.

Morfologia de Imagem

Morfologia de Imagem Binária - Erosão

A Erosão Binária coloca cada pixel de imagem resultante a 0 desde que a imagem original tenha dentro do Elemento Estruturante um pixel a 0.



Exemplo de Erosão Morfológica Binária.

Morfologia de Imagem

Morfologia de Imagem Binária - Erosão



1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Elemento
Estruturante



Exemplo de Erosão Morfológica Binária.

Morfologia de Imagem

Morfologia de Imagem Binária - Fecho (“Closing”)

A Operação Morfológica de Fecho consiste na Aplicação da Operação de Dilatação seguida da Operação de Erosão sobre a imagem Dilatada.

Origina o Fecho de Buracos nas Regiões e a Eliminação de Baías nos limites da regiões



1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Elemento
Estruturante



Exemplo de Fecho Morfológico Binário.

Morfologia de Imagem

Morfologia de Imagem Binária - Abertura (“Opening”)

A Operação Morfológica de Abertura consiste na Aplicação da Operação de Erosão seguida da Operação de Dilatação sobre a imagem Erodida.

Retira pequenas porções ou regiões que saem dos limites



1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Elemento
Estruturante



Exemplo de Abertura Morfológica Binária.

Morfologia de Imagem

Morfologia de Imagem Multinível

As Operações Morfológicas sob Imagens Multinível mais importantes são:

- Dilatação, Erosão, Fecho (“Closing”) e Abertura (“Opening”)
- Gradiente → *Realça os limites das Regiões*

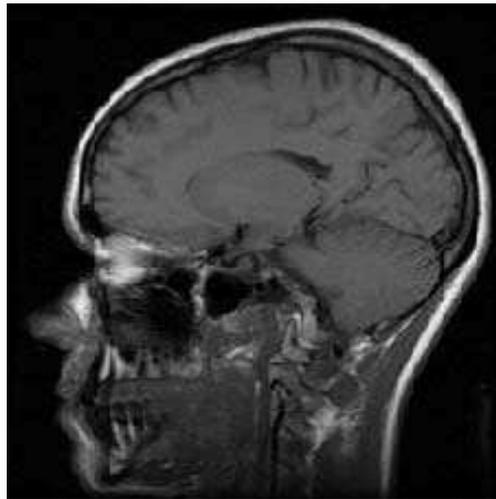
Elementos Estruturantes

São imagens Multinível com uma forma arbitrária.

Morfologia de Imagem

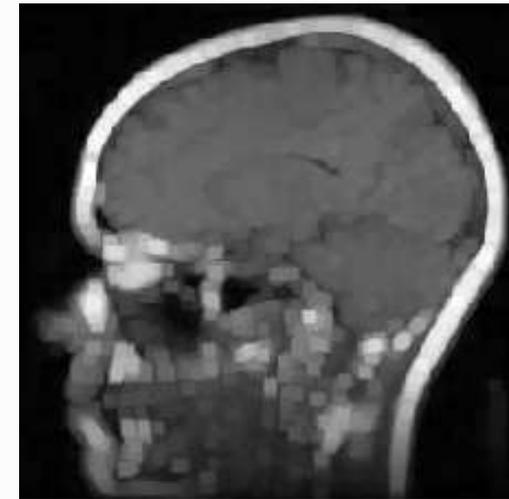
Morfologia de Imagem Multinível - Dilatação

$$D(Img, S) = \max_{[j, i] \in S} \{Img[n - j, m - i] + S[j, i]\}$$



0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Elemento
Estruturante

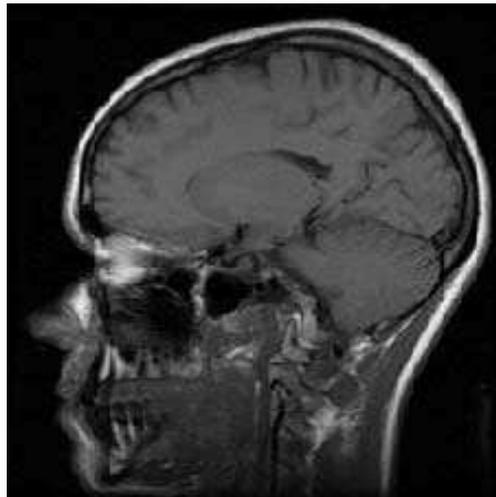


Exemplo de Dilatação Morfológica Multinível.

Morfologia de Imagem

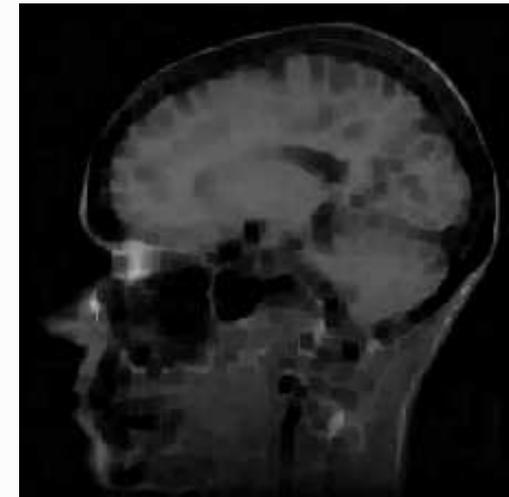
Morfologia de Imagem Multinível - Erosão

$$E(Img, S) = \min_{[j, i] \in S} \{Img[n + j, m + i] - S[j, i]\}$$



0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Elemento
Estruturante

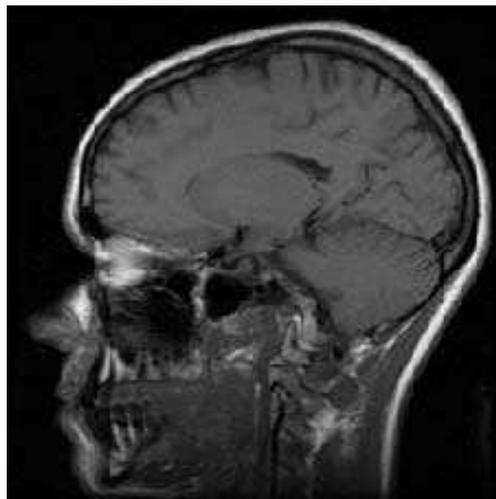


Exemplo de Erosão Morfológica Multinível.

Morfologia de Imagem

Morfologia de Imagem Multinível - Fecho (“Closing”)

$$F(Img, S) = E(D(Img, S), S)$$



0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Elemento
Estruturante

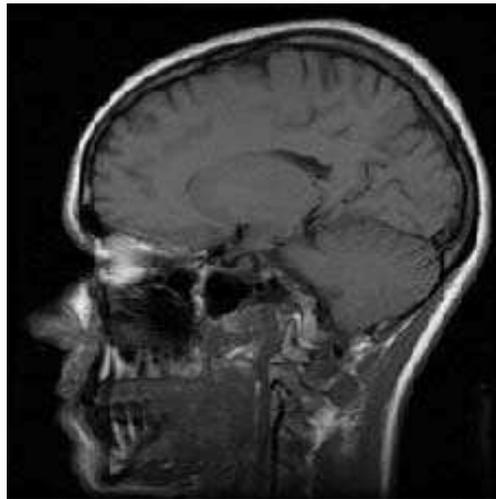


Exemplo de Fecho Morfológico Multinível.

Morfologia de Imagem

Morfologia de Imagem Multinível - Abertura (“Opening”)

$$F(Img, S) = D(E(Img, S), S)$$



0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

Elemento
Estruturante



Exemplo de Abertura Morfológica Multinível.

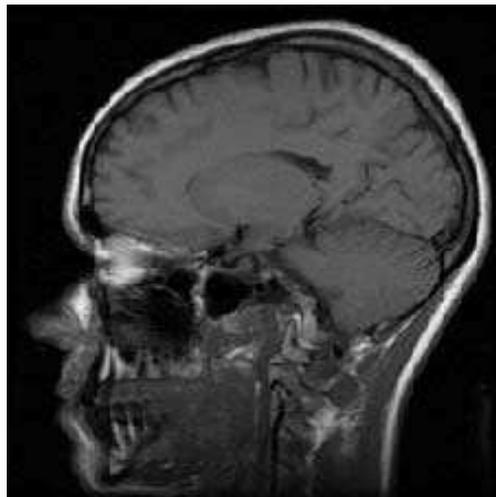
Morfologia de Imagem

Morfologia de Imagem Multinível - Gradiente

A Operação Morfológica Gradiente consiste na subtração da imagem Dilatada pela imagem Erodida.

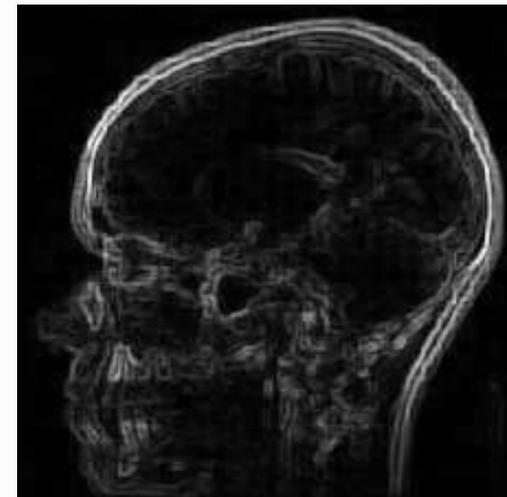
Realça os limites das regiões

$$F(Img, S) = D(Img, S) - E(Img, S)$$



0	0	0
0	0	0
0	0	0

Elemento
Estruturante



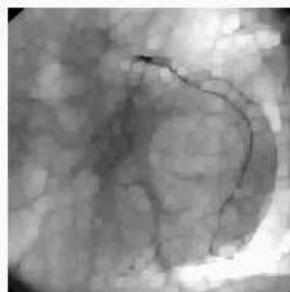
Exemplo de Gradiente Morfológico.

Morfologia de Imagem

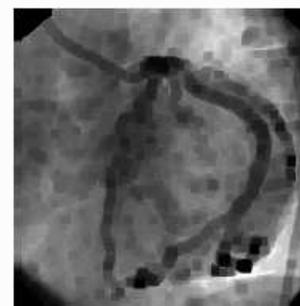
Morfologia de Imagem Multinível - Exemplo



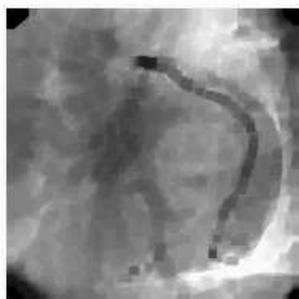
Original



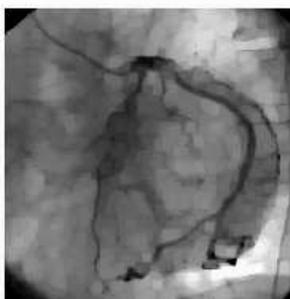
Dilatada



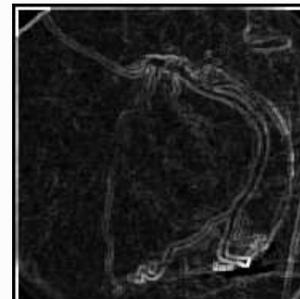
Erodida



Fecho



Abertura



Gradiente

Detecção de Arestas

A detecção de arestas permite separar as diferentes zonas de uma imagem.

Entre os diferentes métodos vamos considerar aqueles que provavelmente são os mais utilizados:

- Operador Diferenciais
- O Detector de Canny

Detecção de Arestas

Operadores Diferenciais

Uma máscara calcula uma aproximação das derivadas $\frac{\partial f}{\partial y}, \frac{\partial f}{\partial x}$ Um ponto representa

um ponto Aresta se o módulo do gradiente $|\nabla f| = \sqrt{\left(\frac{\partial f}{\partial y}\right)^2 + \left(\frac{\partial f}{\partial x}\right)^2}$ tiver um

máximo segundo a direcção do gradiente $\theta = \arctg\left(\frac{\partial f}{\partial y}\right) / \left(\frac{\partial f}{\partial x}\right)$

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

Mx

Operador de Prewitt

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

My

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

Mx

Operador de Sobel

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

My

0	1
-1	0

Mx

Operador de Roberts

1	0
0	-1

My

Operadores Diferenciais

Nota: O operador de Sobel é o operador diferencial mais popular

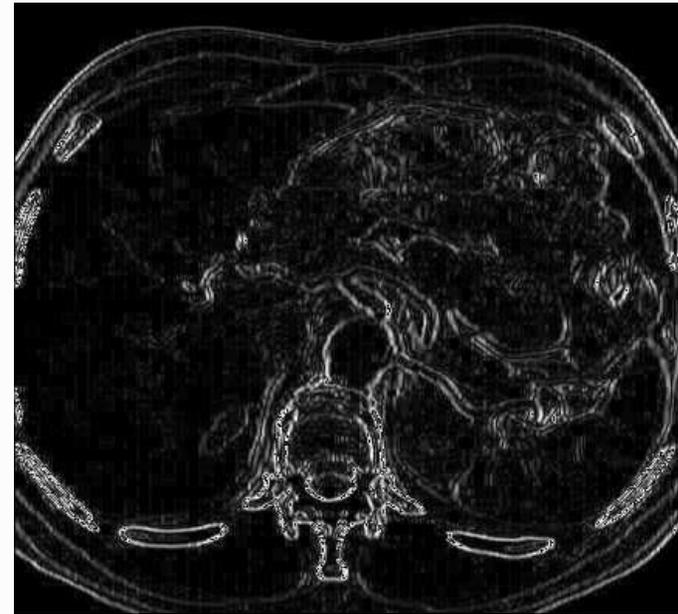


Detecção de Arestas

Exemplo de Aplicação do Operador de Sobel



Imagem original



Módulo do Gradiente

Detecção de Arestas

Exemplo de Aplicação do Operador de Sobel

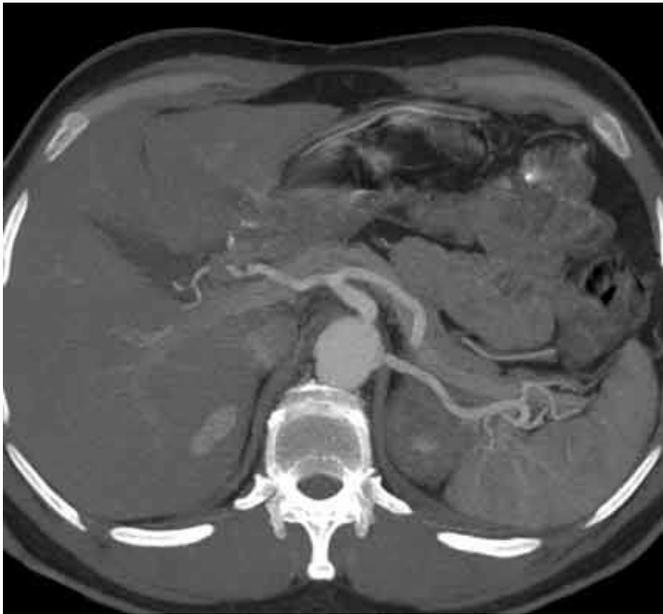
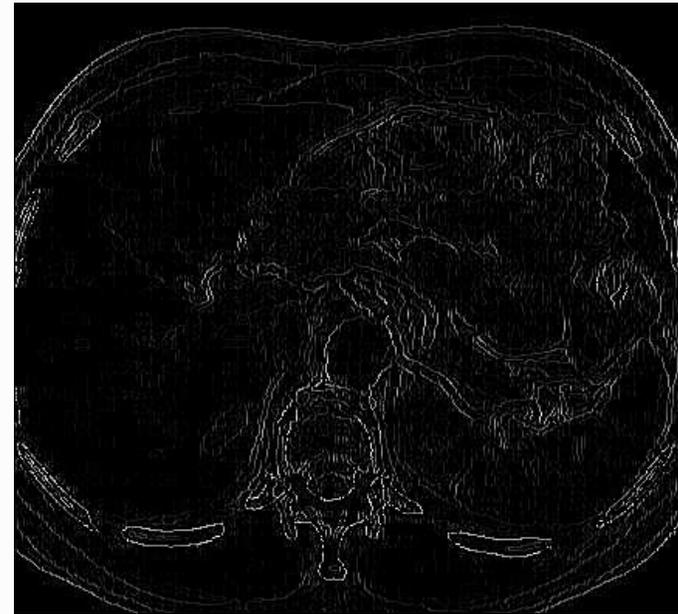


Imagem original



Não Máximos Suprimidos

Detecção de Arestas

Exemplo de Aplicação do Operador de Sobel

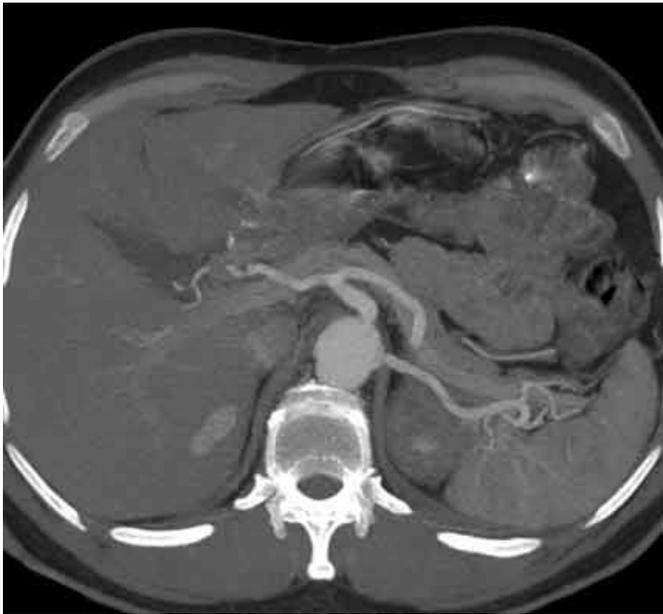
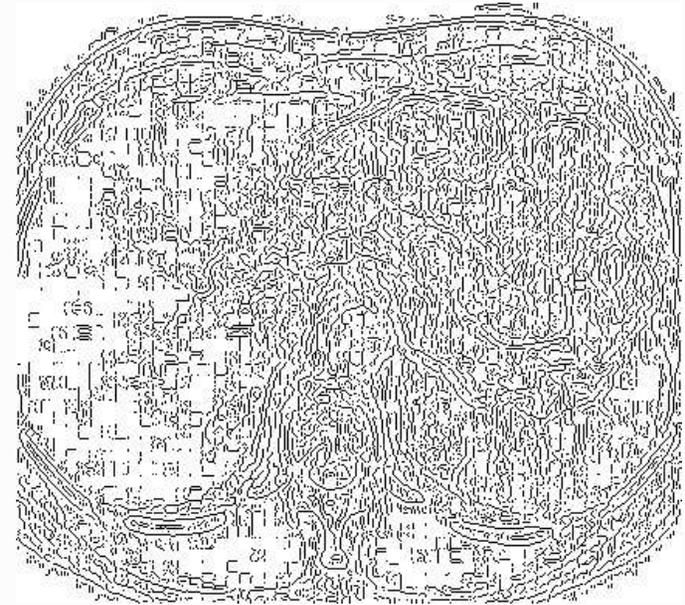


Imagem original



Arestas da Imagem

Detecção de Arestas

Detector de Arestas de Canny

Pretende seguir estes critérios de funcionamento:

- Boa Detecção
- Boa Localização
- Uma só resposta para uma única Aresta

Detecção de Arestas

Detector de Arestas de Canny

Pode ser definido pelos seguintes passos:

- Filtragem Gaussiana

- Diferenciação

Usualmente apenas se usa

$$\frac{\partial f}{\partial y} = I(n+1, m) - I(n-1, m) \quad e \quad \frac{\partial f}{\partial x} = I(n, m+1) - I(n, m-1)$$

- Supressão de não máximos (Só se consideram máximos na direcção do gradiente)
- Os máximos são seleccionados para Arestas por um processo de histerese

Detecção de Arestas

Detector de Arestas de Canny - Supressão de não máximos

Extracção de máximos locais na direcção do gradiente.

• Se $(\Delta_x \times \Delta_y \geq 0)$

– Se $|\Delta_x| \geq |\Delta_y|$

$$B = N_3 + \frac{\Delta_y}{\Delta_x} (N_4 - N_3)$$

$$C = N_7 + \frac{\Delta_y}{\Delta_x} (N_0 - N_7)$$

– Caso contrário

$$B = N_5 + \frac{\Delta_x}{\Delta_y} (N_4 - N_5)$$

$$C = N_1 + \frac{\Delta_x}{\Delta_y} (N_0 - N_1)$$

• Se $(\Delta_x \times \Delta_y < 0)$

– Se $|\Delta_x| \geq |\Delta_y|$

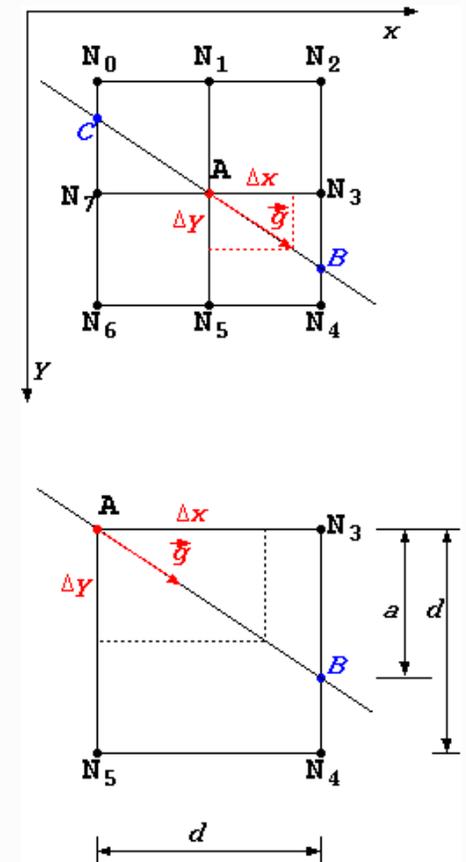
$$B = N_3 - \frac{\Delta_y}{\Delta_x} (N_2 - N_3)$$

$$C = N_7 - \frac{\Delta_y}{\Delta_x} (N_6 - N_7)$$

– Caso contrário

$$B = N_1 - \frac{\Delta_x}{\Delta_y} (N_2 - N_1)$$

$$C = N_5 - \frac{\Delta_x}{\Delta_y} (N_6 - N_5)$$



Verificação se pixel A é ou não máximo na direcção do gradiente \vec{g} .

A é um máximo local na direcção do gradiente se $A > B$ e $A > C$.

Detecção de Arestas

Detector de Arestas de Canny

Processo de histerese

- Só são considerados Arestas que contenham pelo menos um ponto acima de um valor ht . Este valor é escolhido de forma a que uma percentagem $100 - Fh$ de pontos máximos (usualmente cerca de $Fh = 80\%$) estejam acima desse limiar.
- Todos os pontos conectados aos Arestas com um ponto acima de ht e que tenham um valor acima de um valor lt são também considerados como pontos Arestas. lt é escolhido como uma fracção de ht ($lt = Fl \times ht$).

Detecção de Arestas

Detector de Arestas de Canny

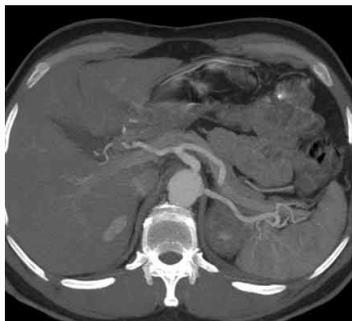
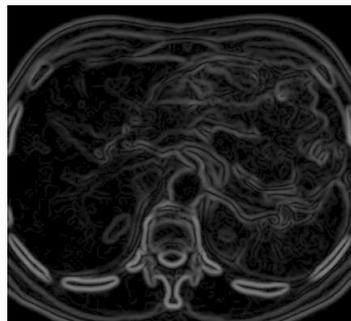


Imagem original



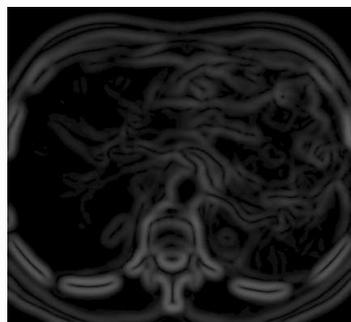
Módulo do Gradiente ($\sigma = 2$)



Não Máximos Suprimidos



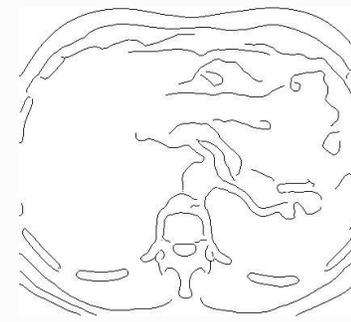
Arestas da Imagem (ht=70% e lt=20%)



Módulo do Gradiente ($\sigma = 4$)



Não Máximos Suprimidos



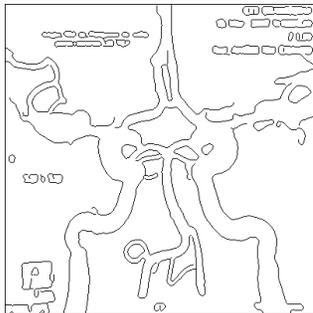
Arestas da Imagem (ht=70% e lt=20%)

Detecção de Arestas

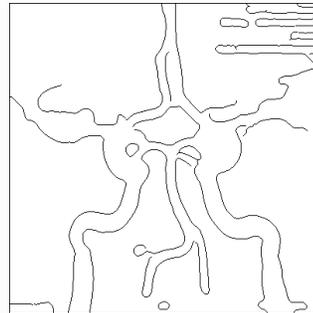
Detector de Arestas de Canny - Influência da parametrização



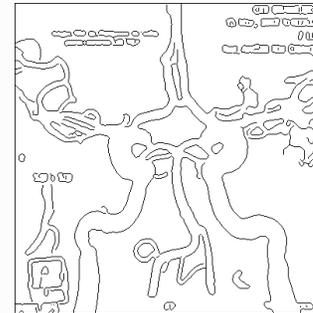
Original



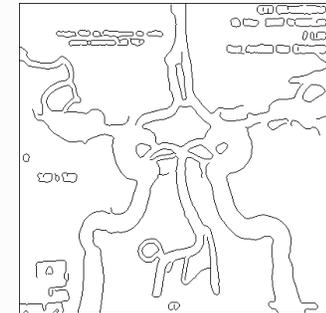
Referencia



$\sigma \uparrow$



$ht \downarrow$



$lt \uparrow$

Segmentação de Imagem

- Introdução
- Segmentação
 - Identificação de Regiões
 - Estratégias de Segmentação
- Técnicas de Segmentação
 - Quantificação de Cores
 - Difusão Anisotrópica
 - Segmentação por “Watershed”
 - “Snakes- Modelos de Contornos Activos
 - Métodos baseados na detecção de Arestas
- Técnicas de Reconhecimento de Padrões Aplicadas à Segmentação
 - Reconhecimento Supervisionado
 - Reconhecimento Não-supervisionado
 - “Clustering”

Segmentação de Imagem

Objectivos

1. Decompor a imagem em partes para análise posterior;
2. Proporcionar uma alteração da representação, que tenha maior significado ou permita um análise mais eficiente.

Exemplos de Aplicações

- Localizar um Tumor
- Medir a dimensão de um Tumor

Segmentação de Imagem

Identificação de Regiões

- As Regiões devem ser uniformes relativamente a uma (ou mais) característica(s), como sejam, o nível de cinzento, a cor ou a textura.
- Os interiores das Regiões devem ser simples e não conterem vários pequenos buracos.
- Regiões adjacentes devem ter valores significativamente diferentes da(s) característica(s) que as identificam.
- Os limites das regiões devem ser suaves, e devem ser espacialmente precisos.

Segmentação de Imagem

Estratégias de Segmentação

- Métodos baseados na Região (conectada)
 - As regiões são localmente homogêneas relativamente a um propriedade.
 - As regiões satisfazem uma determinada propriedade.

- Métodos baseados nos detecção de Arestas
 - Regiões são limitadas por uma determinada característica.
 - As características contêm uma elevado contraste duma propriedade.

Segmentação de Imagem

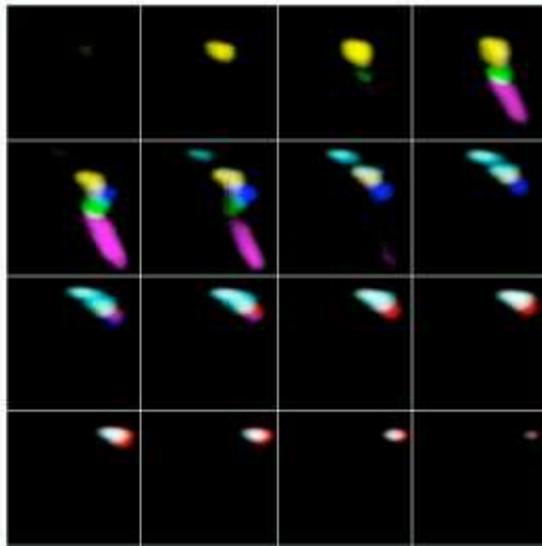
Características usadas na Classificação dos Pixels

- Intensidade
- Derivadas (eventualmente tiradas em Diferentes Escalas)
- Estatísticas da Vizinhança
 - Média, Variância
 - Histograma da Vizinhança
 - Textura (baseada em filtros Passa-Banda: Gabor, “Wavelets”)
- Dados multivariados
 - Cor
 - MRI Espectrais

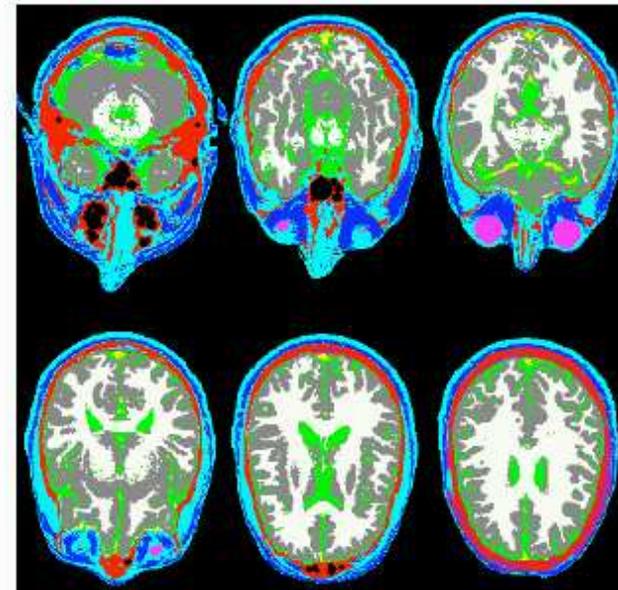
Segmentação de Imagem

Exemplo de Classificação baseada no MRI Espectral

T1, T2, PD



Espaço de Características



Classificação

Retirado de *Tasdizen et al*

Segmentação de Imagem - Métodos baseados na Região

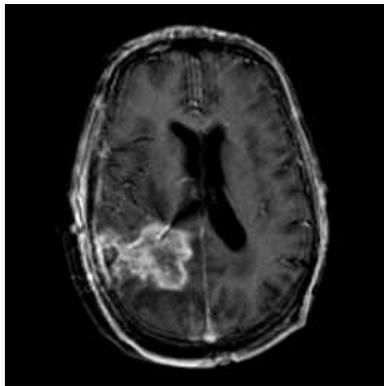
Separação de Cores Baseada no Histograma

ALGORITMO

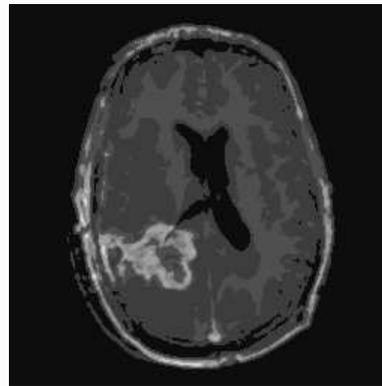
1. Usa-se um Algoritmo de **Quantificação de imagem** baseado no histograma. Exemplo:
M. I. Sezan, “A peak detection algorithm and it’s application to histogram-based image data reduction,” *Computer Vision, Graphics, and Image Processing*, vol. 49, pp. 36–51, 1990.
No fundo, a quantificação estabelece um conjunto de “*Clusters*” e atribui uma classe a cada pixel.
2. Um algoritmo de Etiquetagem das componentes conectadas estabelece as regiões.
3. Regiões pequenas podem ser eliminadas por inclusão noutras.

Segmentação de Imagem - Métodos baseados na Região

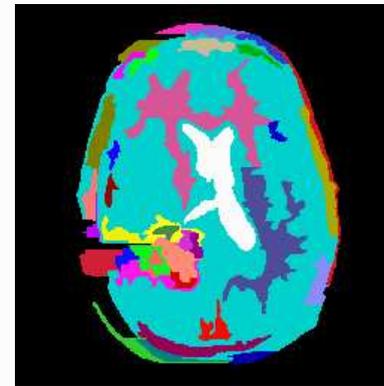
Separação de Cores Baseada no Histograma



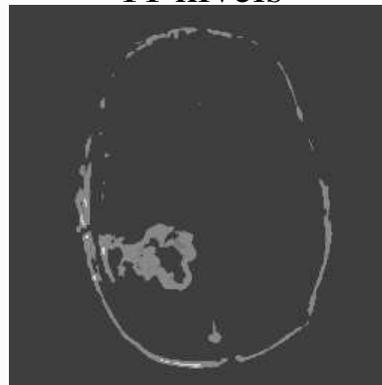
original



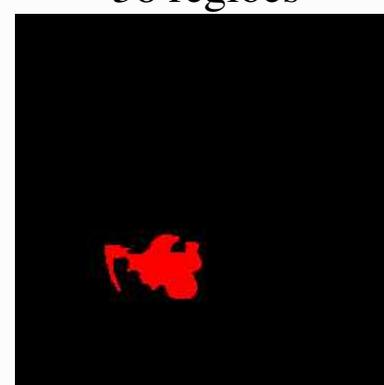
11 níveis



38 regiões



4 níveis



2 regiões

Segmentação de Imagem - Métodos baseados na Região

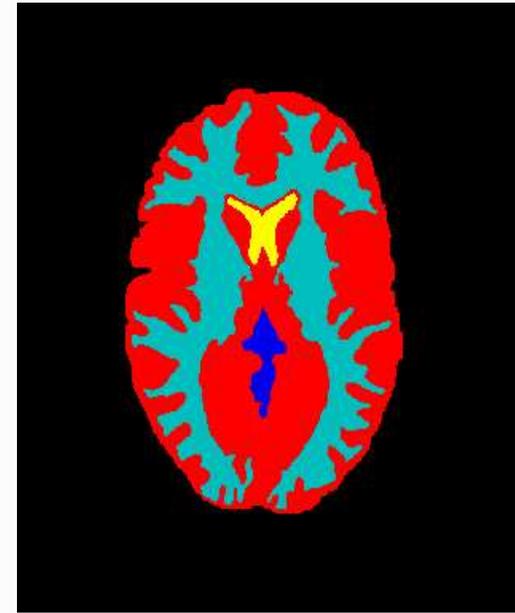
Separação de Cores Baseada no Histograma



original



4 níveis



5 regiões

Segmentação de Imagem - Métodos baseados na Região

Difusão Não Linear

Baseado em: P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-12, no. 7, pp. 629–639, July 1990.

Equação de Difusão:
$$I_t(y, x) = \nabla \cdot (f(\|\nabla I\|) \nabla I(y, x)),$$

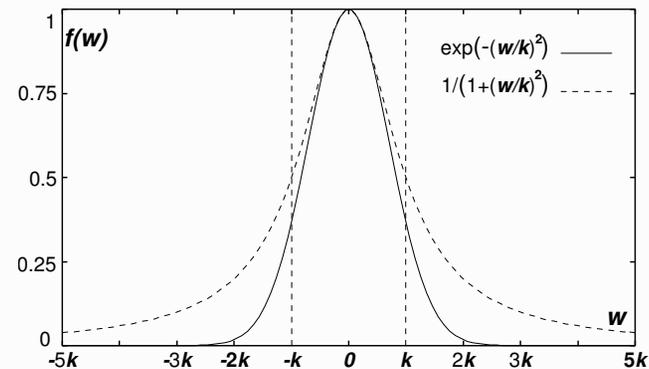
- Se $f(\|\nabla I\|)$ for constante resulta numa filtragem gaussiana!
- Sugestão:

$$f(w) = e^{-\left(\frac{w}{K}\right)^2}$$

ou

$$f(w) = \frac{1}{1 + \left(\frac{w}{K}\right)^2}$$

Nota: se $f(w) = K$ (constante) resulta num filtro gaussiano.



Funções de Difusão propostas por Perona e Malik.

Segmentação de Imagem - Métodos baseados na Região

Difusão Não Linear - Resolução Numérica

Discretização do operador Laplaciano

(4 vizinhos mais próximos):

$$I_{t+1}[n, m] = I_t[n, m] + \lambda \left(\begin{aligned} &C_t^N \Delta_N I_t[n, m] + \\ &C_t^S \Delta_S I_t[n, m] + \\ &C_t^E \Delta_E I_t[n, m] + \\ &C_t^W \Delta_W I_t[n, m] \end{aligned} \right)$$

Os coeficientes de Condução são aproximados por:

$$\begin{aligned} C_t^N &= f(|\Delta_N I_t[n, m]|) \\ C_t^S &= f(|\Delta_S I_t[n, m]|) \\ C_t^E &= f(|\Delta_E I_t[n, m]|) \\ C_t^W &= f(|\Delta_W I_t[n, m]|) \end{aligned}$$

onde $f(\cdot)$ é a função de difusão,

λ constante de tempo para cada iteração:

$$\lambda \in (0, 1/4)$$

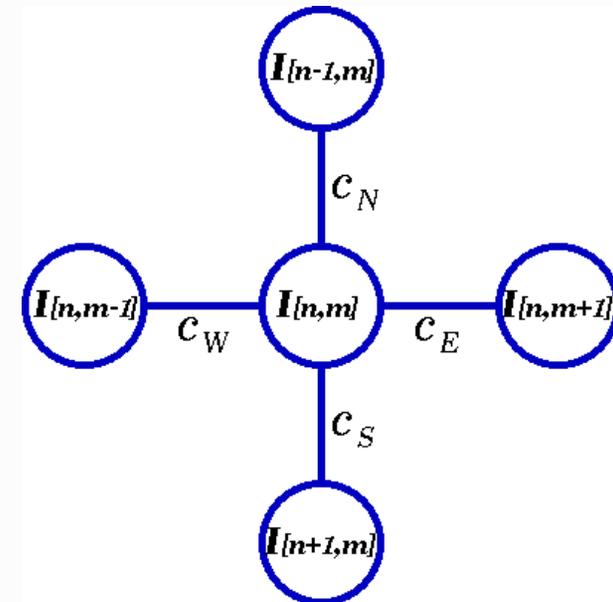
ΔI_t diferença entre níveis da imagem gradiente na escala t nas direções Norte (N), Sul (S), Este (E) e Oeste (W):

$$\Delta_N I_t[n, m] = I_t[n - 1, m] - I_t[n, m]$$

$$\Delta_S I_t[n, m] = I_t[n + 1, m] - I_t[n, m]$$

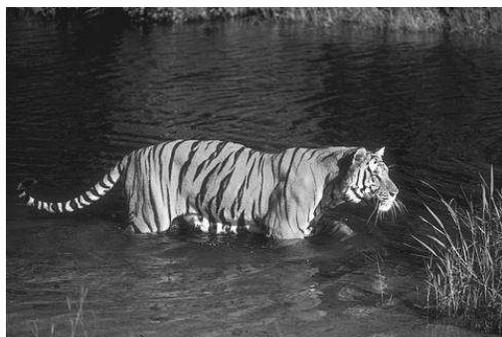
$$\Delta_E I_t[n, m] = I_t[n, m + 1] - I_t[n, m]$$

$$\Delta_W I_t[n, m] = I_t[n, m - 1] - I_t[n, m]$$



Segmentação de Imagem - Métodos baseados na Região

Difusão Não Linear



original



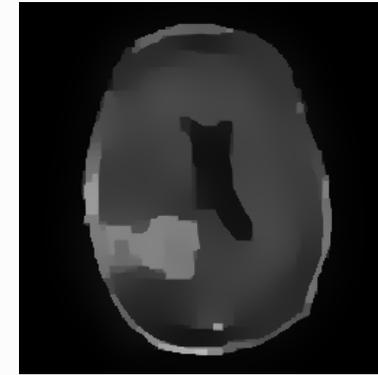
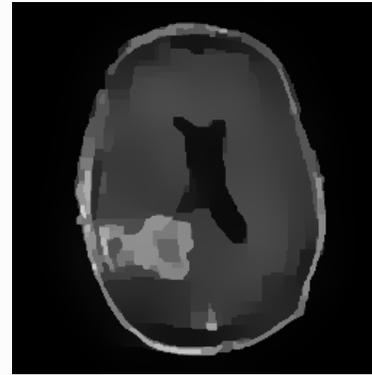
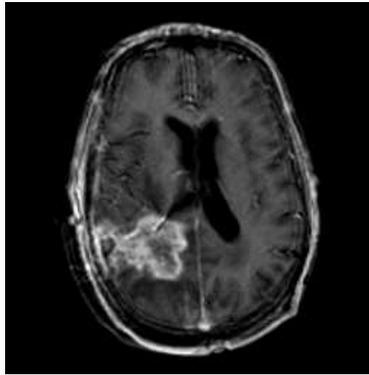
Perona & Malik



Pinheiro 2008

Segmentação de Imagem - Métodos baseados na Região

Difusão Não Linear



original

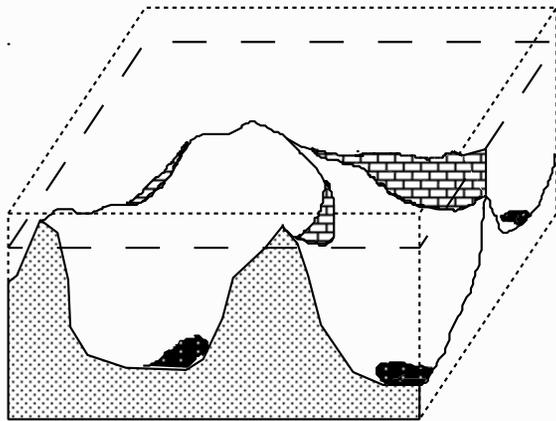
Perona & Malik

Pinheiro 2008

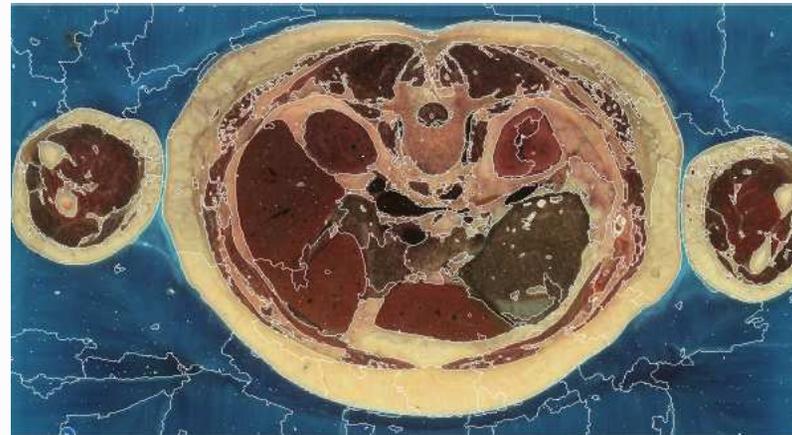
Segmentação de Imagem - Métodos baseados na Região

“Watershed”

- ◇ Algoritmo que vê a imagem como um mapa topográfico.
- ◇ Primeiro Aplica-se um Gradiente à imagem.
- ◇ Consiste no enchimento dos vales do mapa.
- ◇ Quando duas regiões de enchimento se tocam, constroi-se uma barragem.
- ◇ No final, várias regiões foram definidas, separada por um conjunto de barragens (contornos das regiões).



Enchimento e Barragens



Exemplo

Segmentação de Imagem - Métodos baseados nas Arestas

“Snakes” - Contornos Activos

M. Kass, A. Witkin, and D. Terzopoulos, “Snakes - Active Contour Models” *International Journal of Computer Vision*, 1(4): 321-331, 1987.

Um Contorno inicial vai convergir iterativamente para uma zona de elevado gradiente.

Formulação Matemática:

Considerando a “Snakes” dada por $\vec{v}(s) = (y(s), x(s))$, pode-se escrever a sua energia por:

$$E_{Snakes}^* = \int_0^1 E_{Snakes}(\vec{v}(s)) ds = \int_0^1 [E_{int}(\vec{v}(s)) + E_{imag}(\vec{v}(s)) + E_{con}(\vec{v}(s))] ds$$

em que

E_{int} representa a energia interna,

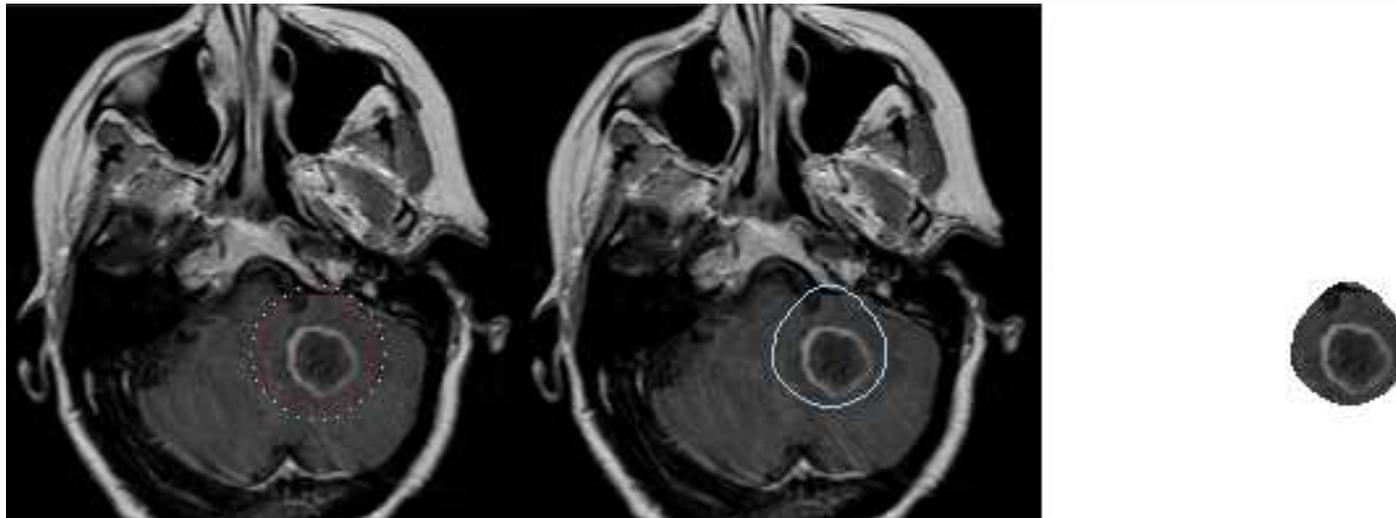
E_{imag} representa as forças da imagem e

E_{con} representa as forças externas de constrangimento.

Segmentação de Imagem - Métodos baseados nas Arestas

“Snakes” - Contornos Activos

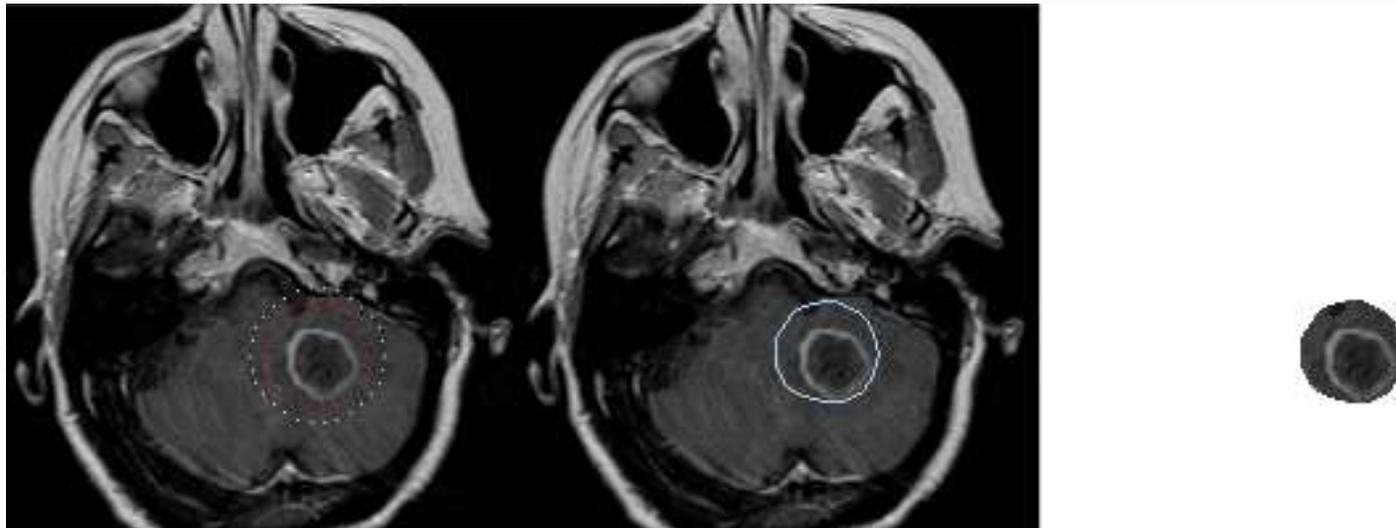
Exemplo (João Machado)



Segmentação de Imagem - Métodos baseados nas Arestas

“Snakes” - Contornos Activos

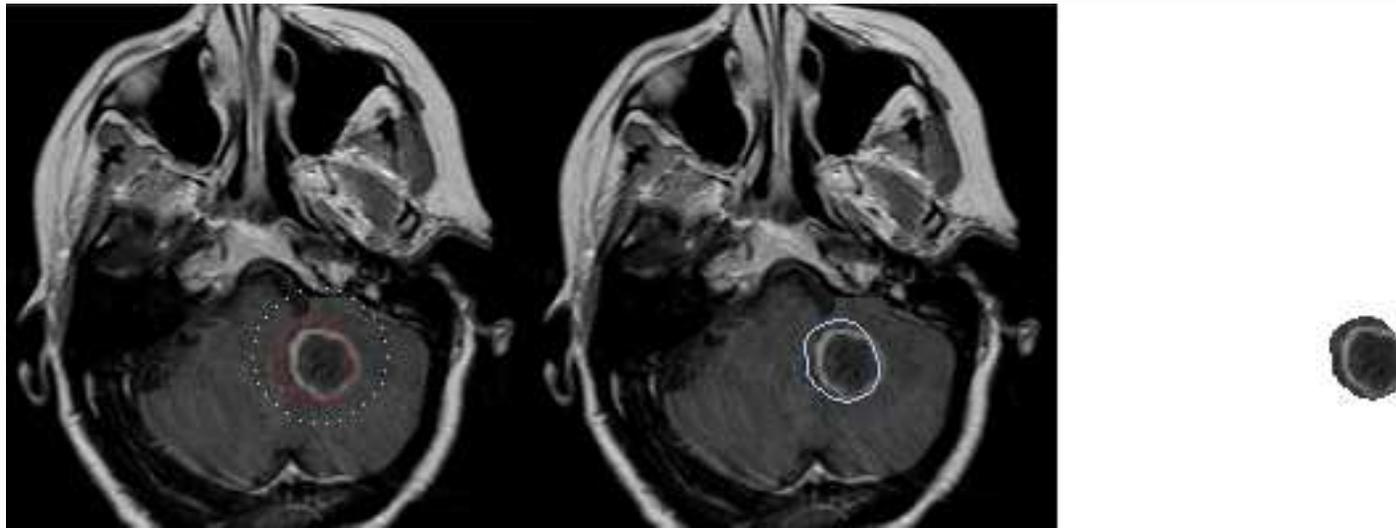
Exemplo (João Machado)



Segmentação de Imagem - Métodos baseados nas Arestas

“Snakes” - Contornos Activos

Exemplo (João Machado)



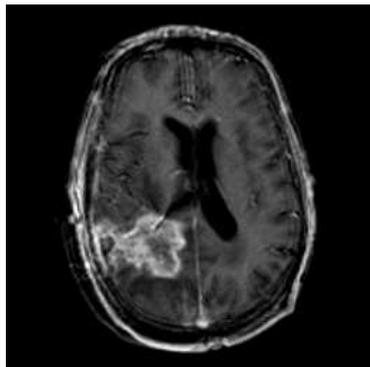
Segmentação de Imagem - Métodos baseados nas Arestas

Técnicas de multiresolução

Baseia-se na determinação das Arestas em diferentes resoluções:

- ◇ Alta resolução implica muitas Arestas posicionalmente precisos.
- ◇ Baixa resolução implica Arestas menos importantes suprimidos (com importância local) mas Arestas resultantes posicionalmente imprecisos.

△ Usar Imagem de Arestas de baixa resolução para seleccionar Arestas mais importantes, e desloca-los para a posição na imagem de alta resolução, de forma a serem mais precisos.



Segmentação de Imagem

Técnicas de Reconhecimento de Padrões Aplicadas

Noção Geral

Da imagem retira-se um vector característico

$$\vec{\mathbf{x}} = (x_0, x_1, \dots, x_{M-1}) \in \mathfrak{R}^n$$

que representa um conjunto de medidas de uma imagem.

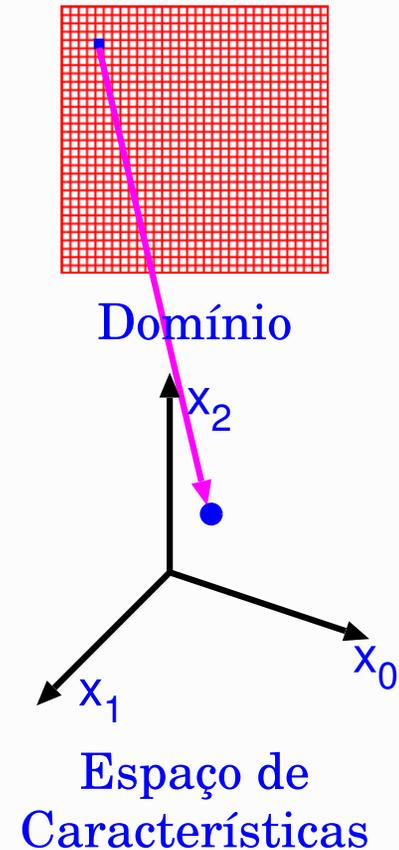
Tipicamente, estabelece-se uma função f , que pondera os diferentes características medidas, tendo em conta o seu custo relativo:

$$f(\vec{\mathbf{x}}) : \mathfrak{R}^n \rightarrow \mathfrak{R}$$

Exemplo:

$$f(\vec{\mathbf{x}}) = \sum_{k=0}^{M-1} \omega_k x_k,$$

em que ω_k representa o peso relativo da característica x_k .



Segmentação de Imagem

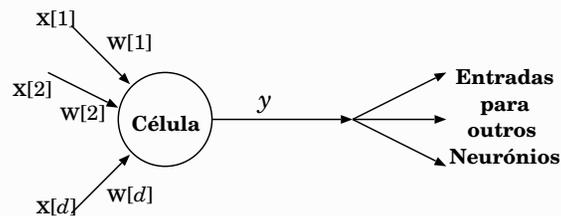
Reconhecimento Supervisionado

- Exemplos previamente classificados são usados para estabelecer f .
 - Uso de *Protótipos*: A classificação é feita usando o *protótipo* mais próximo.
 - Estatístico: Usam-se funções de Densidade de Probabilidade, escolhendo-se a classificação mais provável para o vector \vec{x} .
 - Redes Neurais: Programada por um processo de aprendizagem que estabelece valores para os pesos.
- Não-supervisionado: Decisão é feita exclusivamente a partir dos dados usando função pre-definida de f .

Segmentação de Imagem

Rede Neuronal

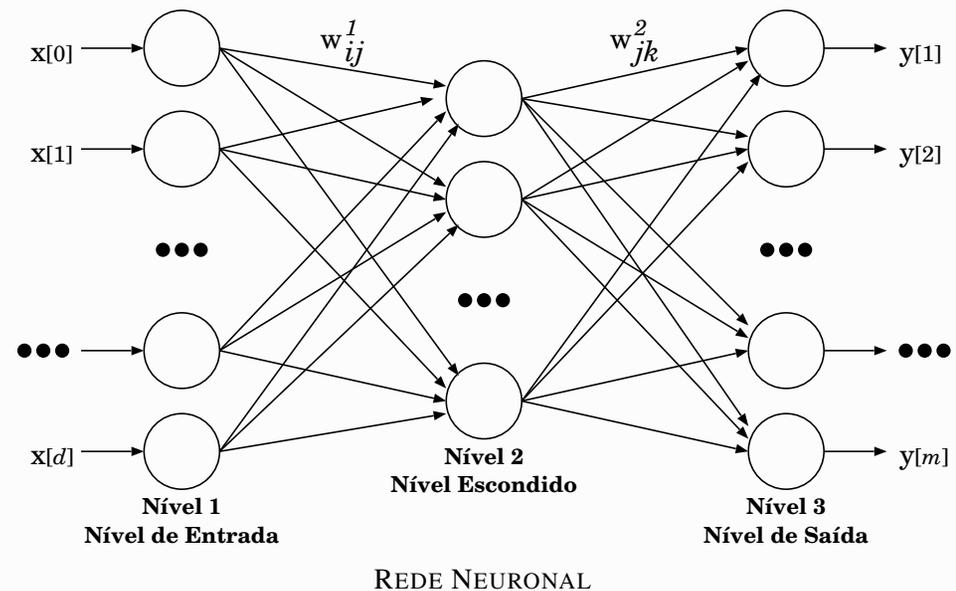
Baseadas no modelo simplificado de um neurónio humano.



$$y = g \left(\sum_{j=1}^d w[j]x[j] \right)$$

onde $g(\alpha)$ pode ser dado por:

- ◇ $g(\alpha) = \begin{cases} 1 & \text{se } \alpha > t \\ 0 & \text{c. c.} \end{cases}$
- ◇ $g(\alpha) = 1/(1 + e^{-\beta(\alpha-t)})$



Segmentação de Imagem

Reconhecimento Não-supervisionado

- Decisão é feita exclusivamente a partir dos dados usando função predefinida de f .
 - Baseado em estruturas naturais dos dados - Ex.: “*Clustering*”.
 - Algoritmo de “*Clustering K-means*”.

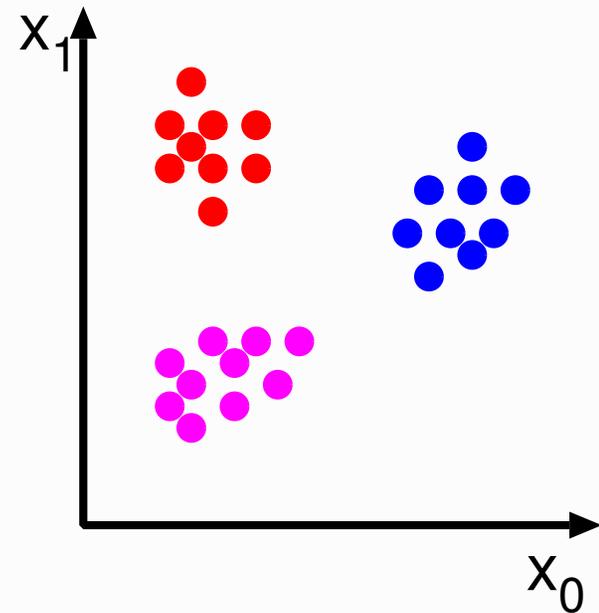
Segmentação de Imagem

“Clustering”

Processo de partição dos vectores característicos em sub-conjuntos, chamados “*Clusters*”.

Uma forma normal de formar “*Clusters*” é associar pontos que estão próximos entre si no espaço euclideano considerado.

Pretende-se portanto, criar partições de um conjunto de vectores, em grupos que apresentam valores similares.



EXEMPLOS DE “*Clusters*” NUM ESPAÇO BIDIMENSIONAL.

Segmentação de Imagem

Algoritmos Clássicos de “Clustering”

Os vectores característicos, podem incluir componentes como:

- ◇ Valores de Intensidade
- ◇ Valores das componentes de Cor (RGB, HSV,...)
- ◇ Propriedades calculadas
- ◇ Medidas de Texturas

A escolha dos **CLUSTERS** pode ser definida

- ◇ com base no número de “Clusters”, K .
- ◇ mantendo a variância para cada “Clusters”, abaixo de um determinado valor.

Segmentação de Imagem

Algoritmos Clássicos de “Clustering”

Tipicamente existem K clusters, C_1, C_2, \dots, C_K , com médias m_1, m_2, \dots, m_K .
A medida do *erro quadrático mínimo* pode ser definida como:

$$D = \sum_{k=1}^K \sum_{x_i \in C_k} \|x_i - m_k\|^2,$$

que mede a proximidade dos dados aos clusters que lhe foram atribuídos.

ALGORITMO BASE

- ◇ Pixels são agrupados em “Clusters”.
- ◇ Um algoritmo de Etiquetagem permite encontrar regiões conectadas.

Segmentação de Imagem

Algoritmo de “Clustering” por “K-means” Iterativo

1. Fazer o número de iterações $i_C = 1$.
2. Escolher aleatoriamente um conjunto de K “Clusters” com médias $m_1(1), m_2(1), \dots, m_K(1)$.
3. Para cada vector x_i calcular $D(x_i, m_k(i_C))$, para cada $k = 1, 2, \dots, K$ e atribuir x_i ao cluster C_j com a média mais próxima.
4. Incrementar i_C , somando 1, e actualizar as médias para obter um novo conjunto $m_1(i_C), m_2(i_C), \dots, m_K(i_C)$.
5. Repetir os passos 3. e 4. até que $C_k(i_C) = C_k(i_C + 1)$ para todos os k .

NOTA: O algoritmo é convergente, embora possa não levar à solução óptima.

Usualmente para-se quando $|C_k(i_C) - C_k(i_C + 1)|$ é menor que um determinado limiar.

Segmentação de Imagem

Algoritmo de “*Clustering*” por “*K-means*” Iterativo



Original



Máscara

Exemplo de “*Clustering*” por “*K-means*”

Transformadas de Imagem

DCT - “Discrete Cosine Transform”

- ◇ A transformada de Fourier resulta num número Complexo.
- ◇ Em algumas aplicações, este facto torna-se uma grande desvantagem.
- ◇ Por isso, surge a aplicação da **DCT - Transformada Discreta do Cosseno**, que é *REAL*.
- ◇ Entre as aplicações, destaca-se o uso em Normas de compressão como a JPEG e a MPEG.

Considerando uma Imagem $I[n, m]$ com dimensão $M \times N$, a **DCT 2-D** é dada por:

$$I_{DCT}(U, V) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} 4I[n, m] \cos\left(\frac{\pi U}{2N}(2n+1)\right) \cos\left(\frac{\pi V}{2M}(2m+1)\right)$$

para $(V, U) \in [0, M-1] \times [0, N-1]$. Caso contrário $I_{DCT}(V, U) = 0$.

Nota: A **DCT 2-D** é separável, logo pode ser aplicada de forma independente sobre as linhas e depois sobre as colunas.

Transformadas de Imagem

DCT - “Discrete Cosine Transform”

A transformada inversa, **IDCT 2-D**, para $[n, m] \in [0, M - 1] \times [0, N - 1]$, é dada por:

$$I[n, m] = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} w[n]w[m]I_{DCT}(U, V) \cos\left(\frac{\pi U}{2N}(2n+1)\right) \cos\left(\frac{\pi V}{2M}(2m+1)\right)$$

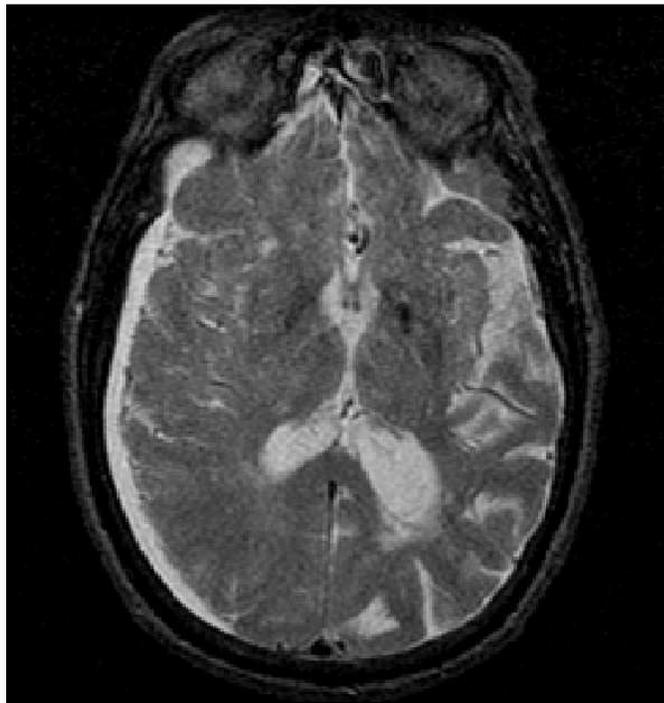
As funções de peso $w(k)$ são dadas por:

$$w(k) = \begin{cases} 1/2, & k = 0 \\ 1 & k \neq 0 \end{cases}$$

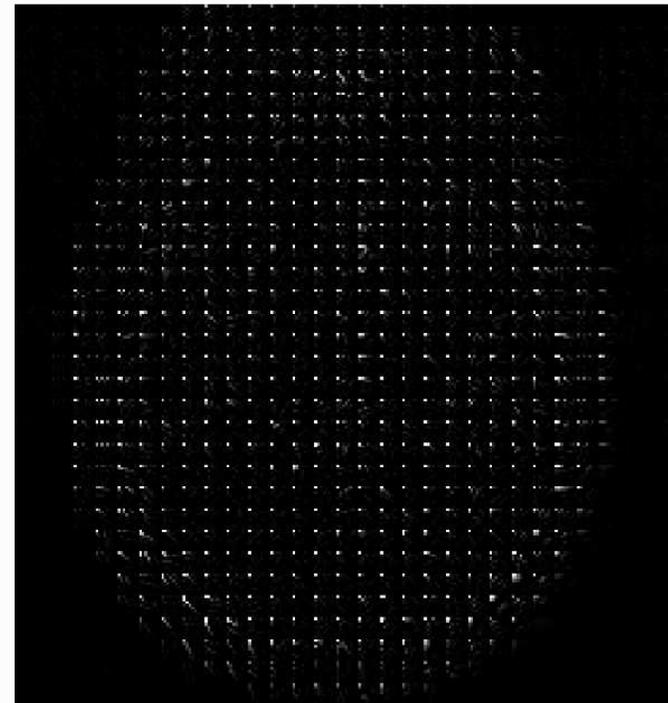
Transformadas de Imagem

DCT - “Discrete Cosine Transform”

Nas normas de imagem, a DCT é aplicada a pequenos blocos de imagem de 8×8 .



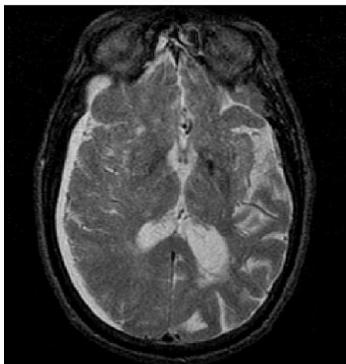
imagem



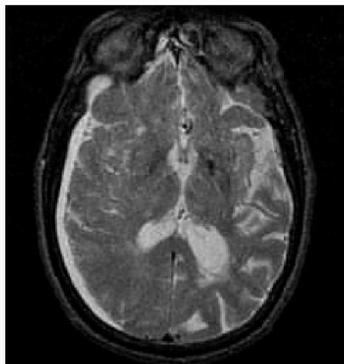
DCT aplicada a blocos de 8×8 .

Transformadas de Imagem

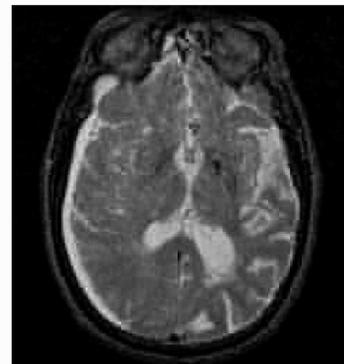
DCT - “Discrete Cosine Transform”



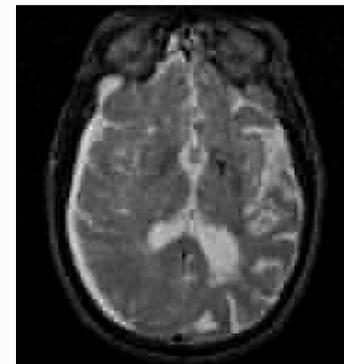
Máscara Aplicada

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$


Máscara Aplicada

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$


Máscara Aplicada

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$


Máscara Aplicada

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$


Transformadas de Imagem

DWT - “Discrete Wavelet Transform”

A Imagem $I(y, x)$ com dimensão $M \times N$, é filtrada por quatro filtros ideais de dimensão $\pi/2 \times \pi/2$, que no seu conjunto dividem a banda em quatro bandas distintas e exclusivas.

Isto significa:

- ◇ Filtro H_{LL} : passa a banda $[0, \pi/2] \times [0, \pi/2]$
- ◇ Filtro H_{LH} : passa a banda $[0, \pi/2] \times [\pi/2, \pi]$
- ◇ Filtro H_{HL} : passa a banda $[\pi/2, \pi] \times [0, \pi/2]$
- ◇ Filtro H_{HH} : passa a banda $[\pi/2, \pi] \times [\pi/2, \pi]$

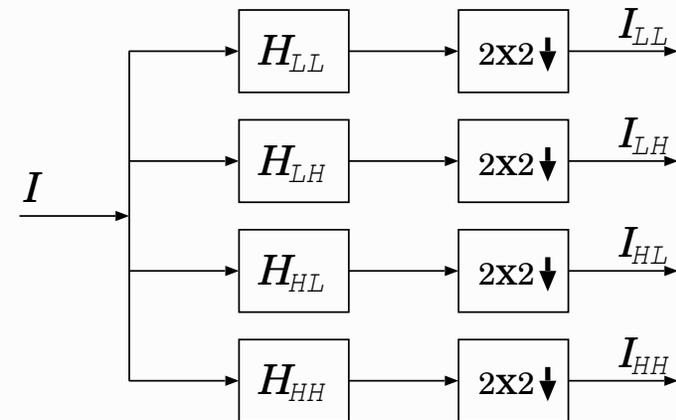


Ilustração da Transformada
“Wavelet” de 2×2 .

Este processo leva a uma subamostragem da imagem, de que resultam quatro subimagens, cada uma resultante de sua filtragem.

A grande vantagem em relação há **DCT** é que lida com a imagem como um todo, em vez de lidar em blocos

Transformadas de Imagem

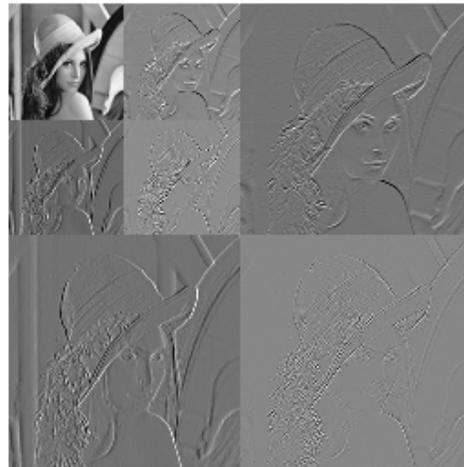
DWT - “Discrete Wavelet Transform” - Nível 1



LL	LH
HL	HH

Transformadas de Imagem

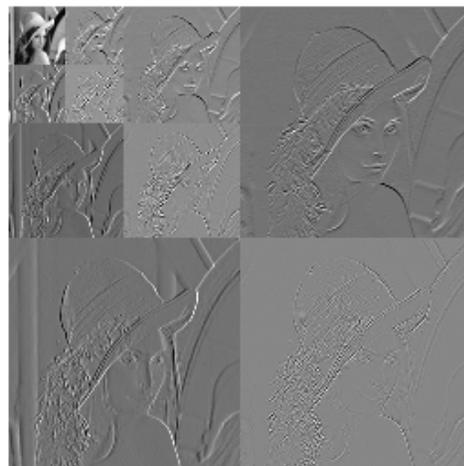
DWT - “Discrete Wavelet Transform” - Nível 2



LLL	LLH	LH
LLH	LLH	
HL		HH

Transformadas de Imagem

DWT - “Discrete Wavelet Transform” - Nível 3



LLL	LLL	LLLH	LH
LLL	LLH		
LLL	LLL		
LHL	LHH	LLHH	HH
LLHL	LLHH		
HL			

Filtros de Estimação Linear

As técnicas de **Estimação Linear** aplicam-se para estimar uma imagem que é corrompida com ruído.

O objectivo destas técnicas é a de estimar a imagem, de forma a que ela possa ser visualizada com redução de ruído.

Dois Filtros são especialmente considerados:

- Filtro de Wiener
- Filtro de Kalman

Filtros de Estimação Linear

Filtro de “Wiener”

Pretende obter uma estimativa, $\hat{I}[n, m]$, de uma imagem, $I[n, m]$ que foi corrompida por ruído aditivo, $I_r[n, m] = I[n, m] + v[n, m]$

$$\hat{I}[n, m] = h_W[n, m] \oplus I_r[n, m]$$

Sendo $I_r[n, m]$ a imagem com ruído, $\hat{I}[n, m]$ a estimativa da imagem e $h_W[n, m]$ o filtro de Wiener.

A transformada de Fourier do Filtro de Wiener é dada por:

$$H_W(U, V) = \frac{P_I(U, V)}{P_I(U, V) + P_v(U, V)}$$

sendo $P_I(U, V)$ é o Espectro de Potência da imagem, obtida pelo cálculo da transformada de Fourier autocorrelação da imagem e $P_v(U, V)$ é o espectro de Potência do ruído.

Filtros de Estimação Linear

Filtro de “Wiener”



Imagem Original



Imagem com Ruído e “blurring”



Estimativa obtida pelo Filtro de Wiener

Filtros de Estimação Linear

Filtro de “Wiener”

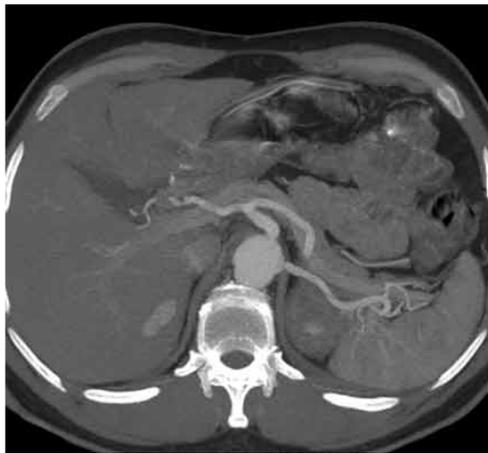


Imagem original



Imagem com aditivo ruído
gaussiano

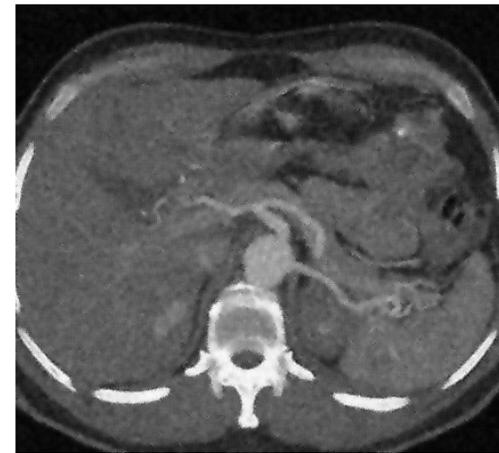


Imagem filtrada

Filtros de Estimação Linear

Filtro de Kalman

Considerando a imagem $x[n, m] = \sum_{[j,i] \in \mathcal{R}_{\oplus+}} c_{j,i} x[n-j, m-i] + w[n, m]$ em que

$w[n, m]$ é ruído aditivo.

A versão da imagem observável, corrompida com ruído

$$y[n, m] = \sum_{[j,i] \in \mathcal{R}_{\oplus+}} h[j, i] x[m-j, n-i] + v[m, n]$$

em que $v[n, m]$ é ruído aditivo.

Filtros de Estimação Linear

Filtro de Kalman

O filtro de Kalman é dado pela seguinte solução iterativa:

Preditor

$$\hat{x}_b^{[n,m]}[n, m] = \sum_{[s,r] \in \mathcal{R}_{\oplus+}} c_{s,r} \hat{x}_a^{[m,n-1]}[m - s, n - r],$$

$$\hat{x}_b^{[n,m]}[j, i] = \hat{x}_a^{[m,n-1]}[j, i], \quad [j, i] \in S_{\oplus+}[n, m]$$

“Update”

$$\hat{x}_a^{[n,m]}[j, i] = \hat{x}_b^{[n,m]}[j, m] + K^{[n,m]}[n - j, m - i] \times$$

$$\left[y[n, m] - \sum_{[s,r] \in \mathcal{R}_{\oplus+}} h[s, r] \hat{x}_b^{[n,m]}[n - s, m - r] \right]$$

para

$$[j, i] \in S_{\oplus+}[n, m]$$

Filtros de Estimação Linear

Filtro de Kalman

Antes da “Update”

$$R_b^{[n,m]}[n, m; j, i] = \sum_{[s,r] \in \mathfrak{R}_{\oplus+}} c_{s,r} R_a^{[n,m]}[n - s, m - r; j, i]$$

para

$$[j, i] \in S_{\oplus+}[n, m]$$

$$R_b^{[n,m]}[n, m; n, m] = \sum_{[s,r] \in \mathfrak{R}_{\oplus+}} c_{s,r} R_a^{[n,m]}[n, m; n - s, m - r] + \sigma_w^2$$

Filtros de Estimação Linear

Filtro de Kalman

Depois da “Update”

$$R_a^{[n,m]}[j, i; s, r] = R_b^{[n,m]}[j, i; s, r] - K^{[n,m]}[n - j, m - i]R_b^{[n,m]}[n, m; s, r]$$

para

$$[j, i] \in S_{\oplus+}[n, m]$$

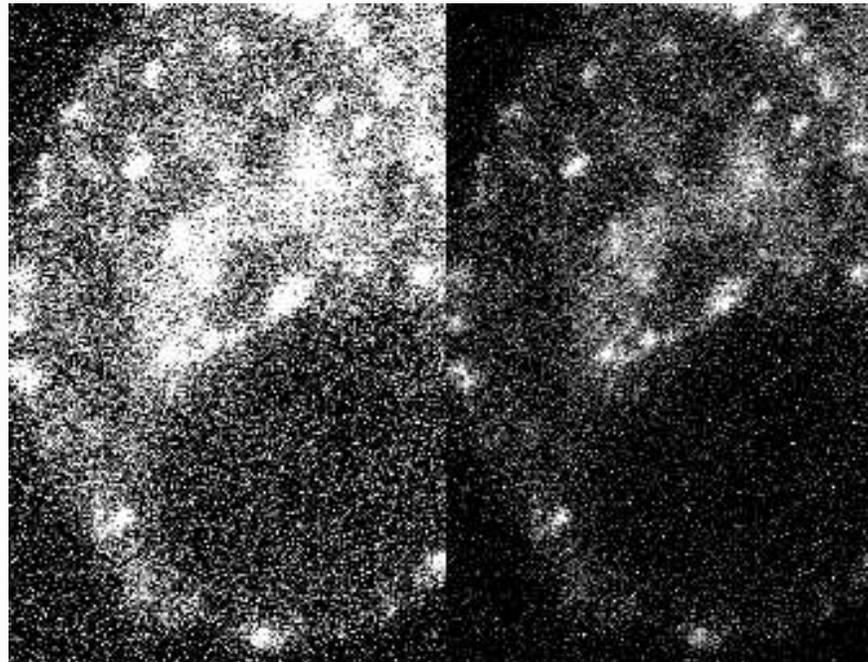
Ganho de Kalman

$$K^{[n,m]}[s, r] = R_b^{[n,m]}[n, m; n - s, m - r] / \left(R_b^{[n,m]}[n, m; n, m] + \sigma_v^2 \right)$$

Filtros de Estimação Linear

Filtro de Kalman

Exemplo



Estimativa obtida pelo Filtro de Kalman