

PTC2892

Princípios da Formação e Processamento de Imagens Médicas

Sérgio S Furuie

S.Furuie 19/8/2010 - 1

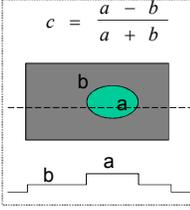
Plano da aula de hoje

- Características das imagens
 - ✓ Resposta impulsiva (PSF: Point Spread Function)
 - ✓ Contraste
 - ✓ SNR, CNR
- As aulas estão no site:
 - ✓ <http://moodle.stoa.usp.br>

S.Furuie 19/8/2010 - 2

Imagens: características

- Resolução espacial: resposta impulsiva (PSF)
- Contraste
- Resolução intens.: 1/256
- Relação sinal/ruído: SNR
- Relação contraste/ruído: CNR

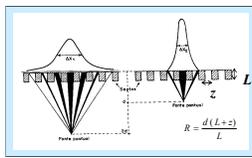
$$c = \frac{a - b}{a + b}$$




S.Furuie 19/8/2010 - 3

Resolução espacial

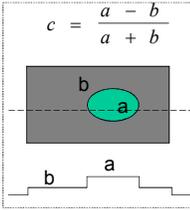
- Conceito intuitivo
 - ✓ Distinguir 2 pontos próximos
 - ✓ Por quê ocorre perda de resolução?
 - ✓ Variante com posição?
- Medida da resolução espacial
 - ✓ Resposta impulsiva (PSF: Point Spread Function)
 - ✓ FWHM: Full Width at Half Maximum
- Estimar FWHM usando o ImageJ
 - ✓ Usar imagem: phantom_IVUS + filtro 5x5

S.Furuie 19/8/2010 - 4

Contraste

- Adimensional
 - ✓ Pode ser em %
 - ✓ Estimar o contraste em diferentes regiões do phantom

$$c = \frac{a - b}{a + b}$$


S.Furuie 19/8/2010 - 5

Relação sinal/ruído e contraste/ruído

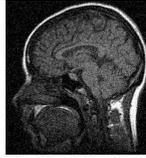


Imagem com ruído (gauss, dp=10)

Signal to noise ratio

$$SNR = 10 \log \left(\frac{\text{var}(\text{sinal})}{\text{var}(\text{ruído})} \right)$$

$$SNR(dB) = 10 \log \left(\frac{\sigma_s^2}{\sigma_r^2} \right)$$

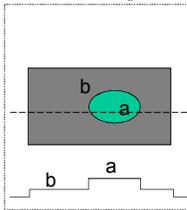
$$SNR(dB) = 20 \log \left(\frac{\sigma_s}{\sigma_r} \right)$$

Contrast to noise ratio

$$CNR = \frac{a-b}{\sigma_r}$$

$$CNR(dB) = 20 \log \left(\frac{a-b}{\sigma_r} \right)$$

Estimar no Phantom com ruído Gaussiano



S.Furuie 19/8/2010 - 6

Medida de qualidade universal

- ❑ Comparação entre 2 imagens (similaridade)
- ❑ **Q: correlação x distorção de luminância x distorção de contraste**

$$Q(f, g) = \frac{\sigma_{fg}}{\sigma_f \sigma_g} \cdot \frac{2\overline{fg}}{\overline{f}^2 + \overline{g}^2} \cdot \frac{2\sigma_f \sigma_g}{\sigma_f^2 + \sigma_g^2}$$

σ_f, σ_g : variancias de f e g
 σ_{fg} : co- variancias de f e g
 \overline{fg} : valores medios de f e g

$-1 \leq Q \leq 1$

S.Furue 19/8/2010 - 7

Implementando ROI: Region of Interest

```

// usando o ROI
Rectangle roi_s=ip.getRoi();
int r0 = roi_s.y,      hROI = roi_s.height;
int c0 = roi_s.x,      wROI = roi_s.width;
  
```

S.Furue 19/8/2010 - 8

```

149 // Convolucao_media.java S.Furue EPOSII
150 #import java.awt.Rectangle;
151 public class Convolucao_media implements PlugInFilter {
152     ImagePlus imp;
153     public int setup(String arg, ImagePlus imp) {
154         this.imp = imp;
155         return DOES_ALL;
156     }
157     public void run(ImageProcessor ip) {
158         // transformando no float p/ float independente do tipo
159         FloatProcessor fp_float=FloatProcessor.convertToFloat(ip);
160         ImagePlus imp_float=new ImagePlus("Float", fp_float);
161         // criando um kernel de 5x5
162         int w=ip_float.getWidth();
163         int h=ip_float.getHeight();
164         int wROI=5; //gera um ROI de 5x5
165         int hROI=5;
166         // usando o ROI
167         Rectangle roi_s=ip.getRoi();
168         int r0 = roi_s.y, hROI = roi_s.height;
169         int c0 = roi_s.x, wROI = roi_s.width;
170         ID.showMessage("Atenção", "Processa somente no ROI");
171         // criando novo array p/ não interferir no original
172         float[] pixels_temp=new float[w*h];
173         // criando as pesos para as medias
174         float[] pesos=new float[N*N];
175         for(int c=0; c<w; c++)
176             for(int r=0; r<h; r++) pesos[r*c] = (float)1.0/(N*N);
177         // realizando a convolucao com o kernel de media
178         for(int y=0; y<h-hROI; y++) // não considera bordas
179             for(int x=0; x<w-wROI; x++) // não considera bordas
180                 { int px = x + wROI; // posicao no array pixels[]
181                   float media=0;
182                   for(int i=0; i<wROI; i++)
183                       for(int c=0; c<hROI; c++)
184                           media=media + pesos[(c+hROI)*ip_float.getPixelValue(x-c,y-c)];
185                   pixels_temp[p]=media;
186                 }
187         // atualizando o processor
188         ip_float.setPixels(pixels_temp);
189         // refresh na imagem
190         imp_float.show();
191     }
192 }
  
```

S.Furue 19/8/2010 - 9

Exercício: plugin em ROI

- ❑ Dada a imagem f, obter a imagem filtrada g (media simples em janela 5x5)
- ❑ Para um dado ROI, estimar:
 - ✓ Média de f e g
 - ✓ variância de f e g (Implementação com apenas 1 loop?)
 - ✓ Correlação entre f e g
 - ✓ Índice de qualidade universal Q(f,g)
- ❑ O que acontece com o índice se a janela fosse 3x3 e 7x7?

S.Furue 19/8/2010 - 10

Resumo: qualidade

- ❑ Resolução Espacial
 - ✓ Define riqueza de detalhes da imagem
- ❑ Resolução de Profundidade
 - ✓ Define riqueza de meios-tons
 - ✓ Define cores
- ❑ Relação sinal/ruído
- ❑ Contrast to noise ratio

S.Furue 19/8/2010 - 11