

## 1. Aula 2

### Leitura, Visualização e Gravação de imagens

**imread, imshow, imwrite, subplot, subimage, title, imixelinfo, size, numel**

- 1.1- Ler a imagem *Rx\_mao.jpg*, visualizá-la numa figura com o título “*Rx\_Mão*” e gravá-la em disco com o nome *Rx\_gravado.tif*.
- 1.2- Ler as imagens *Rx\_mao.jpg* , *Rx\_Torax.jpg*, *Rx\_bexiga.jpg* e *brain.jpg* e apresentar as 4 como subimagens de uma figura, com 2 linhas e 2 colunas. Coloque os respectivos nomes como título.
- 1.3- Ler a imagem *Retina.tif* e verifique os valores dos pixéis da imagem, deslocando o rato sobre a imagem. Apresente as dimensões da imagem (linhas e colunas) e apresente também o número total de pixéis da imagem.
- 1.4- Ler a imagem *Retina.tif* e apresenta-a numa figura, juntamente com as imagens das 3 bandas que compõem esta imagem RGB.
- 1.5- Construir, visualizar e gravar em formato *bmp* uma imagem 256 x 256, branca com um quadrado preto de dimensões 100 x 100, com origem no canto superior esquerdo de coordenadas (90,70).

## 2.

### Manipulação de imagens

**imcrop, imresize, imrotate**

- 2.1- Ler a imagem *Rx\_mao.jpg*, verificar o valor do píxel da posição (70,180) e alterar o seu NDC para 20.
- 2.2- Ler a imagem *Rx\_mao.jpg* e alterar o bloco de pixéis localizado entre as linhas 10 a 30 e as colunas 10 a 30 para o valor NDC=100.
- 2.3- Considere as imagens *Irina.jpg* e *Cristiano.jpg*. Aplicar a concatenação das duas imagens numa única para que fiquem lado a lado.
- 2.4- Ler a imagem *Rx\_bexiga.jpg* e recortar a subimagem de dimensão 90 x 80 com origem no canto superior esquerdo de coordenadas 72 x 70, linha e coluna respetivamente .

- 2.5- Ler a imagem *Rx\_mao.jpg* e redimensionar a imagem para duplicar a sua largura e altura.
- 2.6- Ler a imagem *brain.jpg* e aplicar uma transformação geométrica para inverter a imagem de baixo/cima.

### 3. Aula 3

#### Operadores relacionais, lógicos e aritméticos

*imadd, imabsdiff, and, operador >, ~ (not), imlincomb*

- 3.1- Adicionar o valor 50 à imagem *mamograma.tif*. Verifique o que acontece aos valores da imagem quando se soma à imagem o valor 150.
- 3.2- Somar as imagens *soma\_1.jpg* e *soma\_2.jpg*.
- 3.3- Apresentar as diferenças entre as imagens *dif1.bmp* e *dif2.bmp*. Em seguida use o operador relacional *>* para segmentar a região que corresponde ao rato.
- 3.4- Apresente o negativo (complemento) da imagem *xadrez.tif*.
- 3.5- Considere as imagens *And1.bmp* e *And2.bmp*. Determine os pontos negros da imagem *And2.bmp* que estão em posições correspondentes às zonas escuras na imagem *And1.bmp*.
- 3.6- Na imagem *hospital\_61.jpg*, use a imagem binária *masc.jpg* para destacar a região do sinal, sob um fundo escuro.
- 3.7- Considere as imagens *dif1.bmp* e *dif2.bmp*. Gerar uma sequência de imagens que crie o aparecimento suave do rato na cena.

### 4.

#### Propriedades estatísticas da imagem

*imhist, mean2, std2*

- 4.1- Apresentar o histograma simples e normalizado, a intensidade média e o desvio padrão de intensidades nas imagens *fce\_clara.bmp* e *fce\_escura.bmp*.
- 4.2- Apresentar o histograma da imagem *vertebra.jpg* e determinar o NDC mais frequente da imagem. Apresentar também o histograma da imagem *vertebra.jpg* com 25 classes de intensidade.

4.3- Apresentar o histograma da imagem *retina.tif* das 3 bandas, RGB.

## 5. Aula 4

### Melhoramento de imagem baseado no histograma

**histeq, imhistmatch**

5.1- Aplicar uma equalização sobre a imagem *polen.tiff*. Visualizar os histogramas antes e após a equalização.

5.2- Aplique à imagem *babesia\_1.jpg* uma transformação para que o seu histograma “iguale” o da imagem de referência *babesia\_2.jpg*

## 6.

### Melhoramento de imagem baseado em funções de transformação de intensidades

**imcomplement(I), imadjust**

6.1- Aplicar o negativo à imagem *mamograma.tif*.

6.2- Visualizar o histograma da imagem *polen.tif* e identificar o problema que afeta a imagem. Aplicar à imagem uma transformação de intensidades com a função *imadjust*. Experimente vários valores do parâmetro *gamma* para ver qual o valor que produz melhores resultados.

6.3- Aplicar à imagem *rx\_torax.jpg* uma transformação de intensidades de modo a ganhar detalhe (maior nitidez) na região dos pulmões.

6.4- Aplicar à imagem *fce\_clara.bmp* uma transformação exponencial do tipo  $I^k$ , com  $k=2.9$ , de modo a aumentar o contraste das zonas mais claras da imagem. Use uma *look-up-table* para tornar a operação mais rápida e apresente o gráfico da curva de transformação.

## 7. Aula 5

### Melhoramento de imagem baseado em filtragem espacial

**imfilter, fspecial, medfilt2, imsharpen, imnoise**

7.1- Efetuar a filtragem espacial entre a imagem *face.jpg* e os filtros:

**Filtro 1:**  $1/9 \times [1,1,1; 1,1,1; 1,1,1]$ .

## EXERCÍCIOS DE PROCESSAMENTO DIGITAL DE IMAGEM EM MATLAB

**Filtro 2:**  $1/49 \times [1,1,1,1,1,1,1; 1,1,1,1,1,1,1; 1,1,1,1,1,1,1; 1,1,1,1,1,1,1; 1,1,1,1,1,1,1; 1,1,1,1,1,1,1; 1,1,1,1,1,1,1]$ .

Tire conclusões com base nos resultados.

**7.2-** Efetuar a filtragem espacial entre a imagem *face.jpg* e um filtro gaussiano,  $5 \times 5$  e  $\sigma = 1$ .

**7.3-** Compare os resultados da filtragem da imagem *face\_noise.jpg* com os filtros de média ( $3 \times 3$ ), ( $5 \times 5$ ), filtro de gaussiano ( $5 \times 5$ ),  $\sigma = 1$  e o filtro de mediana ( $3 \times 3$ ).

**7.4-** Implemente os vários passos do operador *unsharp masking* para realçar os vasos sanguíneos na imagem *retina.jpg*. Use o filtro de média  $27 \times 27$  para obter a versão suavizada da imagem e utilize um valor de  $k$  igual a 5.

**7.5-** Use a função Matlab para aplicar o operador *unsharp masking* sobre a imagem *moon.tif*.

**7.6-** Fazer o realce (*sharpenning*) da imagem *moon.tif* obtida pela subtração entre a imagem original e o respetivo laplaciano (*unsharp masking*).

**7.7-** A partir da imagem *edifício.tif* crie 2 novas imagens, uma com ruído gaussiano (parametrização por defeito) e outra com ruído sal e pimenta, com probabilidade de ocorrência igual a 0.15. Procure encontrar entre os filtros de média e de mediana aquele que mais eficazmente remove o ruído das imagens sem as degradar. Utilize a medida *root mean square error* (RMSE) para comparar os resultados dos filtros.

## 8. Aula 6

### Segmentação – Detecção de descontinuidades e arestas

**imfilter, fspecial, edge**

**8.1-** Efetuar a convolução discreta entre a imagem

$I =$

0	0	0.2	0.5	0.8	1	1
0	0	0.2	0.5	0.8	1	1
0	0	0.2	0.5	0.8	1	1
0	0	0.2	0.5	0.8	1	1

e a máscara  $W = [-1 \ 0 \ 1]$ . Efetue o exercício manualmente e computacionalmente.

**8.2-** Visualize a resposta da convolução discreta entre a imagem *xadrezado.jpg* e as seguintes máscaras:  $W1 = [-2 \ 0 \ 2]$  e  $W2 = [-2 \ 0 \ 2]^T$ .

EXERCÍCIOS DE PROCESSAMENTO DIGITAL DE IMAGEM EM MATLAB

8.3- Aplicar à imagem *edificio.tif* o filtro *prewitt*. Visualize a resposta à deteção das arestas verticais, horizontais e a magnitude da resposta do filtro.

8.4- Repita o exercício anterior, mas usando o filtro *sobel*.

8.5- Visualize a imagem resultante da convolução da imagem *pulmao.jpg* com o filtro *Laplacian of Gaussian (LoG)*. Em seguida visualize as arestas da imagem resultantes do operador *zero crossing*.

## 9. Aula 7

### Segmentação – Transformada de Hough

[hough](#), [houghpeaks](#), [houghlines](#), [imfindcircles](#), [viscircles](#)

9.1- Na imagem *estrada.tif* detete com a Transformada de Hough as duas retas que delimitam a linha branca.

9.2- Na imagem *bolas.jpg* detete com a Transformada de Hough os círculos presentes na imagem. **Nota:** No uso da função **imfindcircles** tenha em consideração o parâmetro 'ObjectPolarity' que deve tomar o valor 'dark' (círculos são escuros).

## 10.

### Segmentação – Binarização

[im2bw](#), [imquantize](#), [VideoReader](#), [readFrame](#), [hasFrame](#), [graythresh](#), [adaptthresh](#), [imbinarize](#)

10.1- Visualizar a imagem *bolas.jpg* e o respetivo histograma. Após a análise visual, escolha um valor de binarização (manual) para segmentar os círculos escuros da imagem.

10.2- Visualizar a imagem *celulas.jpg* e o respetivo histograma. Após a análise visual, efetue a binarização multinível escolhendo (manualmente) dois valores de binarização para segmentar a imagem em três regiões: células, círculo grande no interior do qual se encontram as células e o fundo da cena (quatro cantos da imagem).

10.3- Visualizar o vídeo *Color\_Ball.avi*. Em cada frame do vídeo binarize a bola vermelha, escolhendo os valores de binarização adequados para cada uma das três bandas: R, G e B.

**10.4-** Considere a imagem *impressão.jpg*. Visualize o histograma da imagem e determine automaticamente o nível de binarização ótimo para separar o padrão das papilas (objeto) do fundo da imagem (*background*). Compare o resultado usando três métodos distintos: Otsu, iterativo e Pico/Vale.

**10.5-** Procure binarizar a imagem *text\_grad.bmp* para separar os caracteres do fundo. Estude a função *adaptivethreshold* que se encontra na pasta das imagens e utilize-a para obter uma melhor segmentação do texto. Em alternativa use a função do Matlab *adaptthresh*.

**Nota:** Na parametrização da função *adaptivethreshold* use uma vizinhança 11 x 11, uma constante  $C=7/255$  e o operador de média.

## 11. Aula 8

### Segmentação – Crescimento de regiões, Divisão e Reunião de regiões, Método das bacias hidrográficas

**regiongrow, splitmerge** - Implementadas no livro DIPUM – e **watershed**

**11.1-** Estude a função *regiongrow* do livro DIPUM – pág 408-411. Aplique o método de segmentação por crescimento de regiões na imagem *Img\_region\_growing.tif*. Parametrize a função com os seguintes valores:  $S=255$  e  $T=65$ .

**11.2-** Estude a função *splitmerge* do livro DIPUM – pág 412-417. Aplique o método de segmentação divisão e reunião de regiões na imagem *Img\_spli\_merge.tif*. Utilize a seguinte parametrização na chamada da função:

$g = \text{splitmerge}(img, 16, @predicate)$

**11.3-** Sobre a imagem *steel.tif* aplique o método de segmentação de bacias hidrográficas.

**Nota:** use o negativo da imagem e selecione marcadores na imagem com o seguinte código:  $im3 = imhmin(im2, 20)$ .

## 12. Aula 9 e 10

### Morfologia matemática

**strel, imerode, imdilate, imopen, imclose, bwmorph, bwhitmiss,**

**12.1-** Aplicar sobre a imagem *texto\_partido.tif* o operador morfológico de dilatação com um elemento estruturante em forma de cruz, para conseguir ligar os caracteres quebrados.

**EXERCÍCIOS DE PROCESSAMENTO DIGITAL DE IMAGEM EM MATLAB**

- 12.2- Aplicar sobre a imagem *wire\_bond.tif* operadores morfológicos de erosão com elementos estruturantes em forma de disco com raio igual a 10, 5 e 20, respetivamente.
- 12.3- Aplicar sobre a imagem *shape.tif* os operadores morfológicos de abertura e fecho, com um elemento estruturante quadrado 20 x 20.
- 12.4- Aplicar sobre a imagem *shape.tif* uma abertura seguida de um fecho, com um elemento estruturante quadrado 20 x 20, de modo a obter as 2 formas perfeitas, quadrado e retângulo.
- 12.5- Sobre a imagem resultante do exercício anterior, aplique a transformação *hit-or-miss* para encontrar o canto superior esquerdo das 2 formas.
- 12.6- Considere a imagem *fingerprint.tif*. Aplique a esta imagem operadores morfológicos para eliminar o ruído (use um elemento estruturante quadrado 3 x 3). Sobre a imagem resultante do processamento anterior (sem ruído) aplique o operador morfológico de “emagrecimento” (*thinning*) de modo a que os sulcos da impressão fiquem com a espessura de 1 pixel.
- 12.7- Aplique sobre a imagem *bone.tif* o operador morfológico de “esqueletização” (*skeletonization*) de modo a reduzir a representação do objeto a um conjunto de traços finos.
- 12.8- Converta a imagem *hospital\_61.jpg* para níveis de cinzento e de seguida aplique o operador gradiente morfológico com uma máscara quadrada 3x3.

### 13. Aula 11

#### Representação e descrição de regiões

***bwboundaries*, *bwperim*, *minperpoly*, *connectpoly*, *bound2im*, *AP\_recurativa*, *regionprops***

- 13.1- Considere a imagem *folha.jpg*. Desenvolva código para:
- Converter a imagem para níveis de cinzento.
  - Binarizar a imagem pelo método de Otsu, para a região da folha.
  - Extrair o contorno da folha.
  - Calcular o perímetro, largura máxima e comprimento máximo da folha.
- 13.2- Sobre a imagem *Folha\_aprox\_poli.tiff* aplique o operador de aproximação poligonal MPP (*Minimum Polygonal Perimeter*), conforme se descreve no livro DIPUM.

- 13.3-** Considere as imagens *quad\_aprox\_polig.jpg* e *triangulo\_aprox\_polig.jpg*. Classifique as formas presentes nas imagens em triângulo ou quadrado, de acordo com o número de vértices obtidos com o algoritmo de aproximação poligonal fornecido pelo docente (*AP\_recurativa.m*)
- 13.4-** A partir do exemplo da técnica assinatura  $R(\Theta)$  que está no documento pratica -10.pdf, na página 3, implemente o código para classificar as duas formas presentes na imagem *assinaturas.jpg*, analisando a assinatura  $R(\Theta)$  de cada uma.
- 13.5-** Considere a imagem *rice\_1.jpg*. Processe devidamente a imagem para determinar a área do maior grão de arroz e de seguida visualize todos os grãos que têm um tamanho maior ou igual que 95% da área do maior grão.
- 13.6-** Considere as imagens *holes1.jpg* e *holes2.jpg*, representativas de um processo de fabrico em que se produzem peças com um ou 2 furos. Aplique uma sequência apropriada de operadores de forma a determinar para cada uma das imagens se a peça tem um ou dois buracos.