

Imagens Biomédicas – Lab01-Python

Tutores: André Arruda / Maíra Suzuka Kudo / Eric Rocha Santos

Professor: Matheus Cardoso Moraes

1 EXERCÍCIOS:

1. Criar em seu pen-drive ou computador uma pasta chamada “*ImagensBiomedicas/imagens*” . Baixe as imagens do Classroom, nesta atividade serão usadas as imagens *raioXTorax.pgm* e *Lamina-biopsia.jpg*.
2. Dentro da Pasta “*ImagensBiomedicas*” crie uma pasta com o nome da aula.
Ex.: Aula01
3. Abra o Spyder e use a opção Ver:

→ Layouts da Janela → MatLab Layout.

Abra e deixe na mesma área de trabalho os painéis janelas:

- a. Current Folder
- b. Console
- c. Explorador de Variáveis / Figuras
- d. Editor: “Arquivo”
- e. Histórico
- f. Explorador de Código

4. Dê um nome e salve o arquivo do Editor na pasta Aula01. Pode chamar:
fundamentos.py

.....Aula01\fundamentos.py

5. → No Editor(fundamentos.py), importe as bibliotecas abaixo:

```
import numpy as np
import matplotlib.pyplot as plt
import cv2 # OpenCV
import skimage
import skimage.exposure
```

Obs.: Talvez tenha que instalar as bibliotecas acima, para tanto, digite no console:

```
pip install numpy
pip install matplotlib
pip install opencv-python Ou pip install opencv-contrib-python
pip install scikit-image
```

Caso não consigam usar o pip install se refiram ao material do link abaixo:

<https://dicasdepython.com.br/resolvido-pip-nao-e-reconhecido-como-um-comando-interno/#:~:text=Uma%20situa%C3%A7%C3%A3o%20que%20pode%20acontecer,gente%20usa%20o%20pr%C3%B3prio%20python.>

6. (Ler Imagens) No Editor use a função “*imread*” para fazer a leitura dessas imagens, atribuindo seus valores a duas matrizes i0.

Ex.:

```
i0 = cv2.imread('raioXTorax.pgm', 0) # Gray
```

Obs.: colocar imagens na pasta Aula01 ou indicar corretamente o endereço do diretórios onde estão as imagens.

7. (Normalizar Intensidades) No Editor implemente a normalização abaixo para a matriz de imagem ficar normalizada entre 0 e 1 com resolução de intensidade ‘float’ .

Ex.:

Convert to normalized floating point como se fosse im2double do MatLab.

```
in0 = skimage.img_as_float(i0)
```

8. → Dado que uma imagem pode ser representada por uma matriz com M linhas e N colunas, use a função “np.shape” para encontrar o valor M e N da imagem representada pela matriz *in0*.

```
(M,N) = np.shape(in0)
```

Obs.: função da biblioteca → import numpy as np
Verificar Outras operações e funções no endereço
<http://mathesaurus.sourceforge.net/matlab-numpy.html>

9. Pelo Consolo, identifique o valor do pixel na posição *in0[50,50]*? Confira este valor no Explorador de Variáveis.

10. Para a imagem normalizadas *in0*, identifique e armazene os valores abaixo.

```
maximo = np.max(in0)  
minimo = np.min(in0)  
media = np.mean(in0)  
dP = np.std(in0)
```

Obs.: operações e/ou funções da biblioteca → import numpy as np
Verificar Outras operações e funções no endereço
<http://mathesaurus.sourceforge.net/matlab-numpy.html>

11. → Use as funções abaixo para exibir a imagem *In0*.

```
# Exibe Imagens
```

```
plt.figure  
plt.ylabel('linhas - M')  
plt.xlabel('colunas - N')  
plt.title('image0')  
plt.imshow(in0, cmap='gray') #--> Colormaps alternativo, verificar outros→  
cmap='jet'  
plt.colorbar()
```

Na janela Figuras, verifique se o tamanho da imagem condiz com o encontrado anteriormente?

12. Transforme a imagem em binário com diferentes limiares.

Ibnario1 = in0>0.2

Ibnario1 = in0>0.5

Ibnario1 = in0>0.8

Explique em comentário no código o que aconteceu? Para qual tipo a imagem se transformou, etc.

13. Refazer as questões usando a imagem colorida, *com as 3 dimensões In1 ou In1[:, :, :]*.

a. i1 = cv2.imread(' Lamina-biopsia.jpg', 1) # Color In1

b. Verifique qual o tamanho desta imagem? Quantas dimensões?

c. Trabalhar / exibir também uma banda de cada vez. In1[:, :, 0], In1[:, :, 1], In1[:, :, 2]. Indique em quais posições estão ás bandas RGB para o Open CV?

Para mais Informações exclusivamente sobre Python

<https://www.youtube.com/watch?v=IJjR906426o&list=PLfCKf0-awunOu2WyLe2pSD2fXUo795xRe>

<https://www.youtube.com/watch?v=S9uPNppGsGo&list=PLvE-ZAFRgX8hnECDn1v9HNTI71veL3oW0>