

Segunda tarefa – Filtros de imagens e Segmentação

1 – Escolher uma imagem em escala de cinza (ou canal de imagem colorida) do Scikit-Image e aplicar filtros para remoção de ruído com variação de parâmetro e dois tipos de ruído.

Ruídos a inserir:

- ruído gaussiano com 2 valores de variância
- ruídos salt-and-pepper com 2 valores de densidade

Filtros: Gaussiano com kernels de 2 tamanhos diferentes e filtro de mediana

Pode gerar o PDF do jupyter-notebook para apresentar a solução

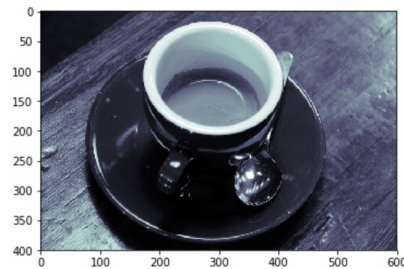
Funções principais para utilizar:

`Matplotlib.pyplot.imshow / Matplotlib.pyplot.show`

`Skimage.util.random_noise`

`Skimage.filters.median`

`Skimage.filters.gaussian`



2 – Reduzir a quantidade de cores de uma imagem utilizando o K-means.

Selecione uma imagem colorida e carregue numa variável.

Escolha um número de cores pequeno (máximo 16).

Utilize o algoritmo de KMeans da biblioteca Sklearn (ou faça o seu próprio) para estimar os centros (no RGB ou LUV) e os rótulos dos pixels.

Gere uma nova imagem substituindo a cor original de cada pixel pela cor do centro mais próximo.

Função principal para utilizar:

`Sklearn.cluster.KMeans`

Atenção para utilizar valores de 0 a 1 no `Matplotlib.pyplot.imshow`.

Tipicamente dividir por 255: `plt.imshow(img/255)`

