

Relatório do Trabalho 5

Raissa Cavalcante Correia - RA: 150619

Disciplina: MC920 - 1s2019 - Turma A

24 de Junho de 2019

Introdução

Esse trabalho diferente do #4 foi possível ser realizado com o CV 4.1, a mesma versão do trabalho 3, não houve a demanda de nenhuma instalação específica nem nenhuma outra biblioteca apenas o CV4, apesar do tutorial seguido na referência [1] ser o mesmo para a versão 3, juntamente com o numpy e o scipy.

Decisões no Código

Conforme pedido fiz a geração da imagem para $K = 16, 32, 64$ e 128 através de um loop, iniciando em 16 parando em 128 e multiplicando por 2 a cada passo.

Foi também implementado 2 técnicas de agrupamento obtidas aplicando-se diferentes flags na função `cv2.kmeans`, o último parâmetro.

Conforme descrito na referência [2] as flags podem ser `KMEANS_PP_CENTERS` ou `KMEANS_RANDOM_CENTERS`. Nesta referências também descreve-se suas diferenças:

“Flag that can take the following values:

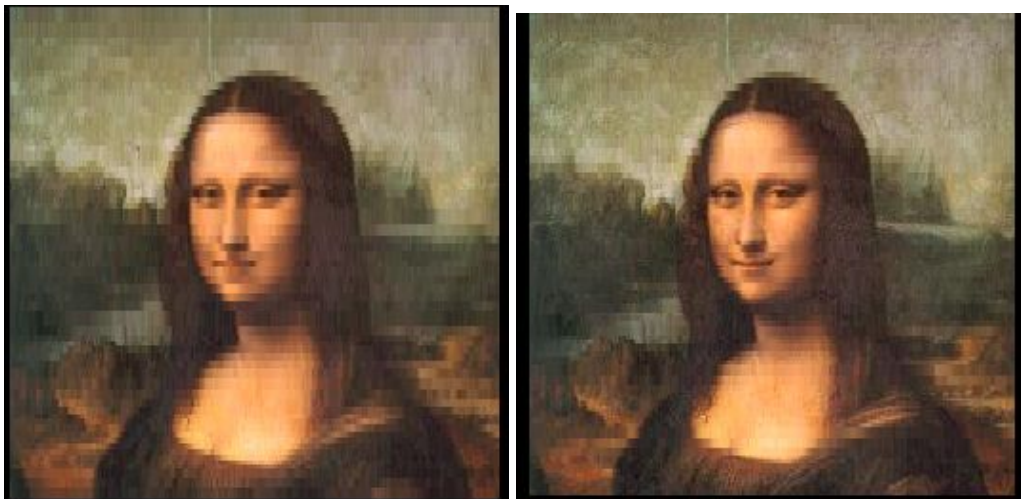
- **KMEANS_RANDOM_CENTERS** Select random initial centers in each attempt.
- **KMEANS_PP_CENTERS** Use kmeans++ center initialization by Arthur and Vassilvitskii [Arthur2007].
- **KMEANS_USE_INITIAL_LABELS** During the first (and possibly the only) attempt, use the user-supplied labels instead of computing them from the initial centers. For the second and further attempts, use the random or semi-random centers. Use one of `KMEANS_*_CENTERS` flag to specify the exact method.”

Ao rodar o algoritmo, o usuário escolhe uma das imagens pelo seu nome, e se deseja fazer pelo `RANDOM` ou pelo `PP`, e serão geradas as 4 imagens (com K igual a 16, 32, 64 e 128) com esse critério.

Resultados



Monalisa PP K=16, na esquerda. E na direita Monalisa RANDOM k=32



Monalisa PP K=64, na esquerda. E na direita Monalisa RANDOM k=128

Conclusões

Pelo fato do algoritmo ser muito bem especificado, assim como suas variações os resultados foram totalmente de acordo com o esperado pela literatura.

Referências

[1]

https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_ml/py_kmeans/py_kmeans_opencv/py_kmeans_opencv.html

[2]

<https://docs.opencv.org/3.0-beta/modules/core/doc/clustering.html?highlight=kmeans>