

	Universidade Federal do ABC Bacharelado em Ciência da Computação Disciplina: Processamento Digital de Imagens Prof.: Francisco Zampirolli Turma: Imagem 2020 Exame: Atividade 3	Sala: 123 Data: 10-05-2020
	Ass.: _____ Estudante: Marcelo Pena ID/RA: 11039314	



#138 - 2020-04-20 - 13:52:40

Instruções:

- (a) Esta é a Atividade 3 do ECE, para ser enviada pelo Moodle.
- (b) Esta é uma atividade individual, foram geradas mais de 50 variações desta questão e cada aluno vai receber uma questão distinta.
- (c) Sugestão: resolver o problema primeiro para dimensões pequenas, para facilitar a validação do seu código.
- (d) Antes de submeter, valide o seu código em IDE's com Jupyter Notebook, ou <https://repl.it/languages/python3>

Questões Dissertativas:

1. Implemente a **transformada de distância utilizando sucessivas erosões**, considerando o exemplo abaixo. Considere a erosão definida por:

$$ero(f, b) = (f \ominus b)(x) = \varepsilon_b(f)(x) = \min\{f(y) - b(x - y) : y \in \mathbb{B}_x \cap \mathbb{E}\}$$

onde $x \in \mathbb{E}$, $f \in K^{\mathbb{E}}$ ou $f \in [0, k]^{\mathbb{E}}$, k é um inteiro positivo representando os níveis de cinza da imagem digital com domínio \mathbb{E} , $b \in \mathbb{Z}^{\mathbb{B}}$. Considere também $b \in \mathbb{Z}^{\mathbb{B}}$ a função estruturantes (vizinhança/kernel), conforme exemplo abaixo, com origem sendo o seu centro (neste exemplo, a imagem f possui 32 pixels).

ATENÇÃO: Submeter o arquivo **Q1.py** (com a resposta).

SUGESTÃO: Fazer o código usando um vetor pequeno e depois de validado, submeter o código para correção.

OBSERVAÇÃO: Esta atividade foi gerada utilizando as seguintes versões. Em alguns testes, o número de erosões (iterações) necessárias para convergir para a transformada de distância foi diferente nos servidores que validam esta atividade no moodle. Ignore se isso ocorreu em algum dos seus casos de teste. Acredito que o erro esteja neste comando `np.array_equal` para comparar dois vetores do tipo `numpy`.

Python, versão: 3.7.3

Numpy, versão: 1.17.4

Exemplo (considerar somente os números como elementos de entrada/saída para os casos de teste):

```

b          : -3 -2 0 -2 -3
f          : 9 9 9 9 9 9 0 9 9 9 9 9 9 9 9 9 0 9 9 9 9 9 9 9 9 0 9 9 9 9 9
dist(f,b): 9 8 6 5 3 2 0 2 3 5 6 8 6 5 3 2 0 2 3 5 6 8 6 5 3 2 0 2 3 5 6 8
iteracoes: 4

```