

# Roteiro

## Operadores morfológicos

### 1. Entendendo do operador "Dilatação"

Entenda o código de Dilatação que está disponível no método doDilation do arquivo Morpho8.java.

Teste várias repetições do operador dilatação ("Max") com elementos estruturantes 4-conecto e em 8-conecto com as imagens lena256.tif e figure.tif usando o plugin Morpho. Compare os resultados.

Todos os operadores em 4-conecto só estão disponíveis no modo executável.

### 2. Programando dos Operadores Morfológicos 8-conectos

Todos os operadores em 8-conecto estão no arquivo Morpho8.java.

#### Nota: Lista de rotinas úteis da classe ImageAccess

<code>imagem = im1.duplicate ();</code>	<code>imagem &lt; - im1</code>
<code>image.subtract(im1, im2);</code>	<code>imagem &lt; - im1 - im2</code>
<code>image.add(im1, im2);</code>	<code>imagem &lt; - im1 + im2</code>
<code>image.normalizeContrast ();</code>	Estende o contraste dentro de uma imagem de forma que os níveis de cinzas fiquem no escopo [0 ..255].
<code>double arr[] = new double[9]; image.getPattern(x, y, arr, ImageAccess.PATTERN_SQUARE_3x3);</code>	Retorna um bloco 8-conecto de pixels vizinhos centrado em (x,y) arr[0] contém o pixel (x-1, y-1) arr[1] contém o pixel (x, y-1) ... arr[7] contém o pixel (x, y+1) arr[8] contém o pixel (x+1, y+1)

### 2.1 codificação do operador "Erosão"

Programa a operador Erosão no método doErosion e teste com a imagem figure.tif.

### 2.2 codificação dos operadores "Abertura" e "Fechamento"

Programa os operadores Abertura e Fechamento que usa Erosão e Dilatação nos métodos doOpen e doClose, respectivamente. Teste seu código com a imagem

keys.tif.

## 2.3 codificação do operador "Gradiente"

Programe o operador de gradiente morfológico no doGradient de método. Chame a rotina

img.normalizeContrast(); ao fim para obter resultados melhores. A imagem de saída é obtida subtraindo a versão erodida da entrada da versão dilatada da entrada.

Teste este operador com a imagem cells.tif.

## 2.4 codificação dos operadores "TopHat Bright" e "TopHat Dark"

Codifique os operadores "TopHat Bright" e "TopHat Dark" nos métodos doTopHatBright e doTopHatDark, respectivamente. Chame a rotina img.normalizeContrast(); ao fim para obter resultados melhores.

Teste o seu código com a imagem figure.tif.

Obs:

TopHat Bright – subtração da imagem original pelo operador de abertura.

TopHat Dark - subtração da imagem original pelo operador de fechamento.

## 2.5 codificação do operador "Median"

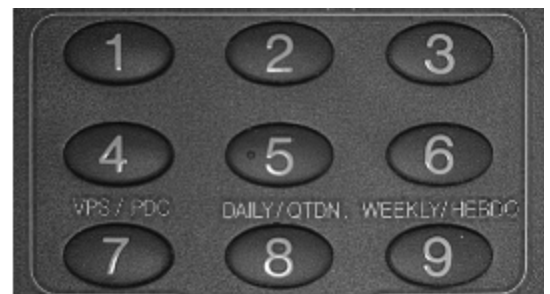
Codifique o operador "Median" no método doMedian. Ao fim do código, você pode achar

uma função útil sortArray () que ordena uma array 1D. Teste seu código com a imagem LenaSaltAndPepper.tif.

# 3. Aplicações

## 3.1 supressão de estruturas pequenas

Um sistema de processamento de imagens deveria conferir os dígitos de um teclado numérico (keyboard.tif). Proponha um algoritmo baseado em operadores morfológicos para preprocessar a imagem. O algoritmo deveria remover os elementos brancos pequenos (linhas, pontos) e manter só os dígitos.

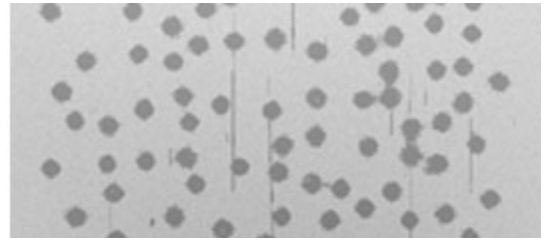


O último passo é o comando de "Threshold" (menu Image->Ajust do ImageJ).

Salve a imagem resultante binária e descreva os algoritmos no arquivo relatório.doc.

## 3.2 conta manchas

Proponha um algoritmo baseado em operadores morfológicos contar o número de manchas neste tipo de imagem (spots.tif). Use o analisador de partículas (menu Analyze de ImageJ, selecione ShowOutlines) para contar as partículas com uma imagem binária. Para gerar a imagem binária, use o comando de "Threshod" do ImageJ, escolha um limiar e clique em "Set".



Salve a imagem de esboço e descreva os algoritmos no arquivo no relatório.doc.

### 3.3 navegação de robô

Na imagem robot.tif, extraia as duas luzes que estão no robô usando uma combinação de operadores morfológicos e finalmente um limiar binário.



Introduza a imagem (robot.tif)



Imagem resultante

Sugestões:

- Primeiro, tente fazer as manchas luminosas mais finas. Então, use uma função para descobrir manchas brancas pequenas (rejeite manchas grandes).
- Resultados empíricos melhores são obtidos se usar os operadores 4-conectos.

Insira a imagem de resultado no relatório.doc e descreva a sucessão de funções no arquivo relatório.doc.

