

# Conectividade

Alexandre Zaghetto

## 1 Relatório

### 1.1 Objetivos

Uma aplicação em biotecnologia resulta imagens binárias de manchas possuindo forma geral como mostrado na figura 1.1.1.

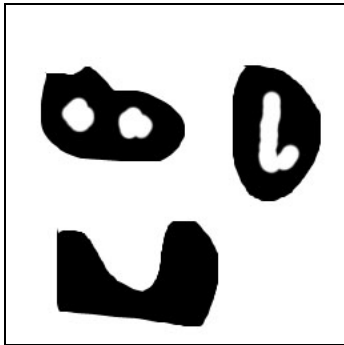


Figura 1.1.1 – Imagem binária (fig1.jpg)

O objetivo desse projeto é desenvolver um método para determinar se uma imagem contém alguma mancha e, se a resposta for positiva, classificar cada mancha como tipo A se ela não tem buracos ou como tipo B em caso contrário.

### 1.2 Introdução

Para que o projeto e a metodologia adotada sejam entendidos corretamente, é preciso abordar alguns conceitos que tratam do relacionamento entre os *pixels* de uma imagem. Assim, nas linhas que se seguem, serão definidos os termos vizinhança, adjacência, conectividade, região e contorno.

Cada *pixel*  $p$ , localizado em  $(x,y)$ , possui 4 vizinhos no sentido horizontal e vertical, identificados pelos pares ordenados  $(x+1, y)$ ,  $(x-1, y)$ ,  $(x, y+1)$ ,  $(x, y-1)$ . Esses quatro vizinhos, chamados de 4-vizinhos, são denotados por  $N_4(p)$ . Os vizinhos de  $p$  localizados nas diagonais são dados por  $(x+1, y+1)$ ,  $(x+1, y-1)$ ,  $(x-1, y+1)$ ,  $(x-1, y-1)$  e denotados por  $N_D(p)$ . Os pontos de  $N_D(p)$  unidos aos pontos de  $N_4(p)$  são chamados de 8-vizinhos e são denotados por  $N_8(p)$ . Está, então, definida a vizinhança entre *pixels*.

Seja considerado agora um conjunto  $V$  que contenha níveis de cinza tidos como similares. Dessa maneira, pode-se definir três tipos de adjacência:

- (a) 4-adjacência: dois *pixels*  $p$  e  $q$  são considerados 4-adjacentes se ambos possuem seus níveis de cinza em  $V$  e se  $q$  é 4-vizinho de  $p$ , ou seja, pertence ao conjunto  $N_4(p)$ .

- (b) 8-adjacência: dois *pixels*  $p$  e  $q$  são considerados 4-adjacentes se ambos possuem seus níveis de cinza em  $V$  e se  $q$  é 8-vizinho de  $p$ , ou seja, pertence ao conjunto  $N_8(p)$ .

- (c) m-adjacência: dois *pixels*  $p$  e  $q$  são considerados m-adjacentes se ambos possuem seus níveis de cinza em  $V$  e se:
  - a.  $q$  está em  $N_4(p)$ , ou
  - b.  $q$  está em  $N_D(p)$  e não há interseção entre  $N_4(p)$  e  $N_4(q)$ .

Dois *pixels* pertencentes a um subconjunto  $S$  são considerados conectados caso, seguindo um dos critérios de adjacência, exista um caminho entre eles também pertencente a  $S$ . Para um *pixel*  $p$ , o conjunto de *pixels* que o une a  $q$  é chamado de *conjunto conectado*. Caso  $q$  seja adjacente a  $p$ ,  $q$  é chamado de *componente conectado*.

Uma região  $R$  é definida com sendo um *conjunto conectado*.

E, por fim, um contorno  $C$  de uma região  $R$  é definido como sendo aqueles *pixels* que possuem vizinhos não pertencentes a  $R$ .

### 1.3 Projeto

Uma rotina (projeto2.m) e 5 funções (assocbur.m, contem.m, numanchas.m, renum.m, rotula.m) foram desenvolvidas em MATLAB, visando atingir os objetivos estabelecidos no item 1.1 do projeto.

As entradas-teste para o programa estão retratadas nas figuras 1.1.1, 1.3.1 e 1.3.2.

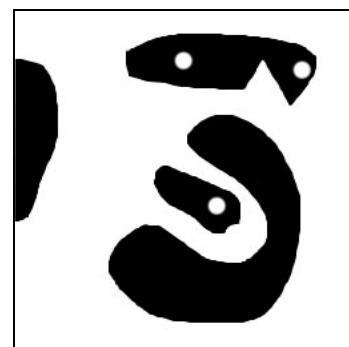


Figura 1.3.1 – Imagem binária (fig2.jpg)

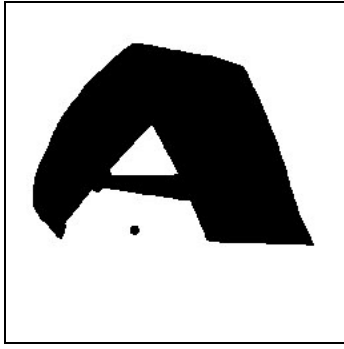


Figura 1.3.2 – Imagem binária (fig3.jpg)

O programa retorna, para cada imagem, o número de manchas (NM), o número de buracos (NB), o número de manchas do tipo A (NMTA - não contém buracos), o número de manchas do tipo B (NMTB - contém buracos) e, para as imagens tipo B, o número de buracos para cada mancha (NBPMB).

O resultado esperado para cada uma das imagens está sintetizado na tabela abaixo:

	NM	NB	NMTA	NMTB	NBPMB
fig1.jpg	3	3	1	2	2
fig2.jpg	4	3	2	2	2
fig3.jpg	2	1	1	1	1

## 1.4 Metodologia

Considerando que as imagens binárias são representadas pelos bits 1 (branco) e 0 (preto), a metodologia adotada resume-se basicamente no algoritmo descrito abaixo:

- Para a matriz binária que representa a figura foi criada uma matriz de rótulos:
  - Rotulação das manchas
    - Todos os *pixels*  $p = p(x,y)$  da imagem são analisados
    - Se  $p = 1$ , nada é feito.
    - Se  $p = 0$ , os vizinhos  $t = p(x-l,y)$  e  $r = p(x,y-l)$  são analisados.
      - Se  $r$  e  $t$  são iguais a 1, designa-se um novo rótulo para  $p$ .
      - Se apenas um deles é 0, designa-se o mesmo rótulo para  $p$ .
      - Se ambos são iguais a 0 e possuem o mesmo rótulo, aplicar o rótulo em  $p$ .
      - Se ambos são iguais a 0 e possuem rótulos diferentes, aplicar um dos rótulos em  $p$  e estabelecer

a equivalência entre os rótulos.

- Substituir os rótulos equivalentes por um mesmo rótulo.
- Reorganizar os rótulos de maneira que o fundo da imagem tenha rótulo 1, e as manchas tenham rótulos de 2 a  $N+1$ , sendo  $N$  o número de manchas.

### b. Rotulação dos buracos

- Inverter a imagem de maneira que os *pixels* que valem 1 passem a valer 0 e vice-versa.
- Rotular os buracos da mesma maneira que as manchas. Se o número total de rotulações for dado por  $L$ , o número de buracos será  $L-1$ , pois o fundo branco, também convertido para preto, acaba sendo considerado erroneamente como um buraco.
- Reorganizar os rótulos dos buracos de maneira que o rótulo 2, referente ao fundo da imagem erroneamente considerado como um buraco, seja ajustado para 1 e os valores dos demais rótulos variem de  $N+2$  (o valor inteiro imediatamente acima do último valor utilizado para a rotulação das manchas) a  $N+2+(L-2)=N+L$ .

### c. A matriz contendo os rótulos dos buracos é sobreposta a matriz contendo os rótulos das manchas, formando uma matriz de rotulação única.

- Por último, é analisada a vizinhança entre rótulos para se estabelecer que buraco pertence a que mancha.

Para exemplificar, consideremos a figura 1.4.2

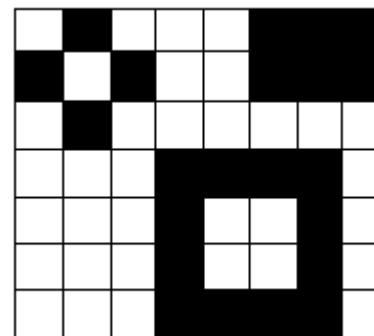


Figura 1.4.1 – Representação pixel a pixel de uma imagem binária

A matriz que contém os valores dos *pixels* da imagem está retratada na tabela abaixo:

1	0	1	1	1	0	0	0
0	1	0	1	1	0	0	0
1	0	1	1	1	1	1	1
1	1	1	0	0	0	0	1
1	1	1	0	1	1	0	1
1	1	1	0	1	1	0	1
1	1	1	0	0	0	0	1

A matriz de rotulação das manchas seria dada por:

1	2	1	1	1	3	3	3
2	1	2	1	1	3	3	3
1	2	1	1	1	1	1	1
1	1	1	4	4	4	4	1
1	1	1	4	1	1	4	1
1	1	1	4	1	1	4	1
1	1	1	4	4	4	4	1

Para determinar os buracos é preciso calcular o negativo da imagem original, o que resultaria em:

0	1	0	0	0	1	1	1
1	0	1	0	0	1	1	1
0	1	0	0	0	0	0	0
1	0	0	1	1	1	1	0
0	0	0	1	0	0	1	0
0	0	0	1	0	0	1	0
0	0	0	1	1	1	1	0

Rotulando o negativo da imagem:

2	1	2	2	2	1	1	1
1	3	1	2	2	1	1	1
2	1	2	2	2	2	2	2
2	2	2	1	1	1	1	2
2	2	2	1	4	4	1	2
2	2	2	1	4	4	1	2
2	2	2	1	1	1	1	2

Ajustando a rotulação do negativo da imagem, segundo o item b do algoritmo, obtemos a matriz de rotulação dos buracos:

1	1	1	1	1	1	1	1
1	5	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	6	6	1	1
1	1	1	1	6	6	1	1
1	1	1	1	1	1	1	1

Sobrepondo as duas matrizes de rotulação obtemos:

1	2	1	1	1	3	3	3
2	5	2	1	1	3	3	3
1	2	1	1	1	1	1	1
1	1	1	4	4	4	4	1
1	1	1	4	6	6	4	1
1	1	1	4	6	6	4	1
1	1	1	4	4	4	4	1

Através da matriz de rotulação acima, facilmente concluímos pelo critério de vizinhança que o buraco rotulado como 5 pertence à mancha rotulada como 2, e que o buraco rotulado como 6 pertence à mancha rotulada como 4.

## 1.5 Resultados obtidos

Os resultados obtidos foram exatamente os esperados e estão sintetizados na tabela abaixo:

	NM	NB	NMTA	NMTB	NBPMB
fig1.jpg	3	3	1	2	B1 - 2 B2 - 1
fig2.jpg	4	3	2	2	B1 - 1 B2 - 2
fig3.jpg	2	1	1	1	B1 - 1

## 1.6 Bibliografia

1. Wanderley, Juliana F. C., **Processamento de Imagens - Notas de Aula**. Brasília: UnB, 2002.
2. Gonzalez, Rafael C. e Woods, Richard E., **Digital Image Processing**, 2o ed, 2001, Addison Wesley Pub. - ISBN: 0201180758.