

UFPI - Universidade Federal do Piauí
DC/CCN - Ciência da Computação
Matéria: Processamento Digital de Imagens
Prof.: Kelson Rômulo Teixeira Aires

RELATÓRIO DA ATIVIDADE PRÁTICA 3

Álvaro Ian de Andrade Sousa
Isaías Oliveira Carvalho

Teresina, 20 de Outubro de 2019

1. Introdução

Este relatório tem como objetivo apresentar o desenvolvimento e os resultados da segunda atividade prática proposta na disciplina de Processamento Digital de Imagens do período 2019.2. Segue uma descrição do que é pedido na atividade.

A prática consiste em implementar alguns algoritmos para resolver problemas utilizando operações morfológicas dada uma imagem fornecida, vista na Figura 1.



Figura 1

Foi pedido que se resolve-se os seguintes problemas na imagem:

1. Eliminar todos os pontos pretos;
2. Preencher os buracos dos objetos: azul, vermelho e verde;
3. Encontrar o fecho convexo dos objetos: azul, vermelho e verde;
4. Utilizando a transformada hit-or-miss, localize cada um dos objetos de cor magenta;
5. Encontre o esqueleto da imagem de cor vermelha;
6. Encontre o esqueleto do fecho convexo obtido a partir da imagem de cor vermelha.

Em seguida será apresentada a metodologia usada para realização do trabalho, como linguagem utilizada e estratégias para a solução do problema. E logo depois os resultados obtidos para cada implementação para resolução do problema.

2. Metodologia

Os programas para implementação das operações morfológicas foi desenvolvido na linguagem Python 3, com o auxílio das bibliotecas OpenCV para manipulação de imagens da linguagem e Numpy para oferecer mais ferramentas de manipulação de matrizes.

Para o problema de eliminar os pontos pretos na imagem, simplesmente fazemos um fechamento (dilatação seguida de uma erosão) na imagem, isso é feito utilizando as funções `dilate` e `erode` da biblioteca OpenCV, com significados óbvios. Vale ressaltar que essas operações morfológicas são feitas com relação ao branco da imagem.

Para o problema de preencher os buracos internos dos objetos, a metodologia utilizada foi limiarizar a imagem de forma inversa e após isso aplicar uma função de floodFill que preenchia tudo deixando somente os buracos na imagem. Após isso inverteu-se agora

a imagem somente com os buracos e fez-se uma união com a imagem limiarizada inicial, fazendo com que os buracos sumissem.

O problema de encontrar os fechos convexos das imagens foi resolvido usando as funções `findContours` e `convexHull` da biblioteca OpenCV. A primeira encontra os pontos de contorno de um elemento numa imagem, e a segunda traça o fecho convexo desse contorno. Dessa forma o procedimento foi eliminar os elementos que não queremos encontrar o fecho convexo, depois tratar a imagem para ficar no ponto de usar as funções, a imagem deve estar limiarizada com os elementos que queremos o fecho convexo de preto no fundo branco, e depois disso usamos as funções para delinear os fechos convexos dos elementos.

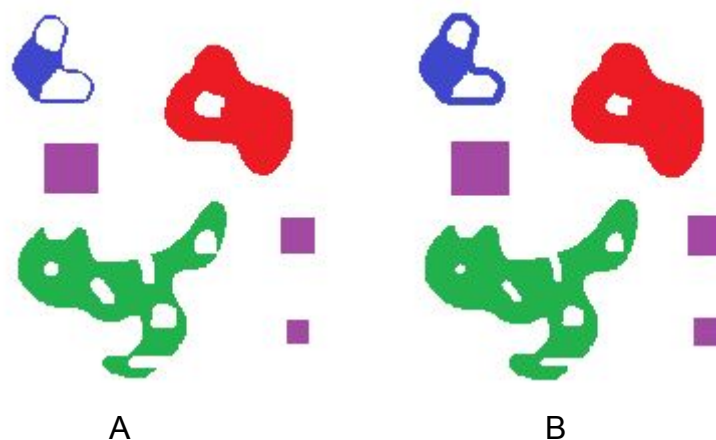
Para o problema de número 4 que pedia o uso da Transformata Hit-or-Miss, fez-se o uso de máscaras de três tamanhos diferentes, sendo eles do menor para o maior, 16x15, 22x21 e 29x31 pixels. A transformata foi aplicada sobre a imagem limiarizada inversamente. Cada máscara resultou em uma imagem diferente, gerando a necessidade de juntá-las novamente em uma nova imagem, gerando o resultado com a localização desejada. Após isso dilatou-se os pontos até que alcançassem o tamanho original e foi feita uma interseção com a imagem original.

O problema de encontrar o esqueleto de um elemento foi resolvido eliminando os elementos desnecessários e deixando apenas o em questão que se quer extrair o esqueleto. Depois são aplicadas várias iterações na imagem limiarizada do elemento aplicando um elemento estruturante em forma de cruz e aplicando o resultado numa nova imagem que conterá o esqueleto do elemento.

Para se encontrar o esqueleto do fecho convexo do elemento vermelho apenas se repetiu os processos dos pontos 3 e 5 agora em conjunto.

3. Resultados

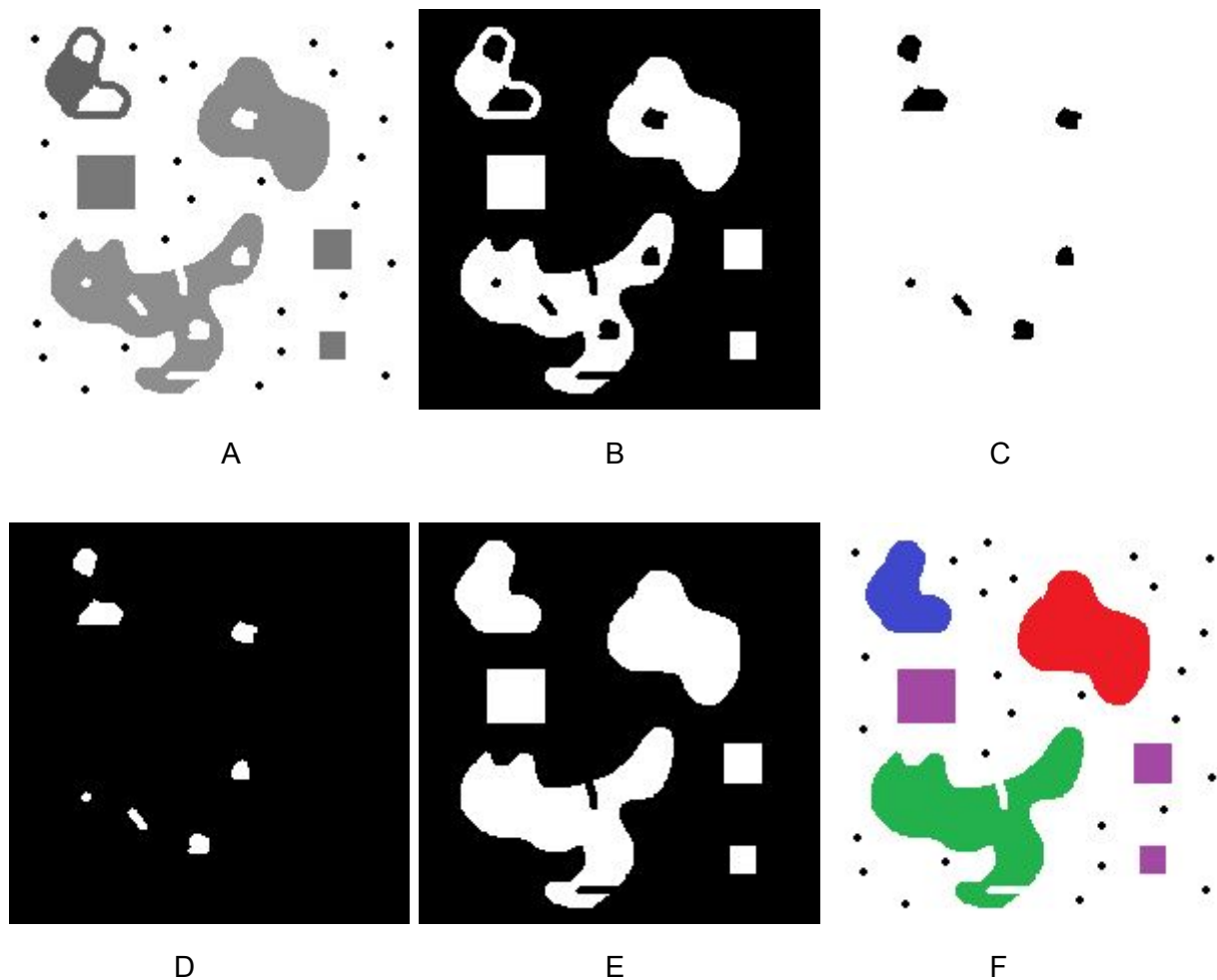
Para o item 1, obteve-se os seguintes resultados:



Em A temos a imagem dilatada por uma máscara 3x3; B) Erosão de A por máscara 3x3.

A dilatação em A com a máscara 3x3 fez com que os pontos pretos sumissem, mas por consequência aumentou o tamanho dos buracos internos dos objetos. Para tentar dar uma amenizada nesse efeito colateral fez-se uma erosão com a mesma máscara 3x3 sobre a imagem obtida em A, resultando a imagem representada em B, com uma visível melhora.

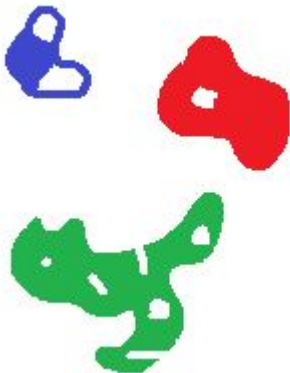
Os resultados alcançados na eliminação dos buracos contidos nas formas é retratado abaixo. Não considerou-se os golfos apresentados por alguns dos mesmos como um buraco.



Da esquerda para direita, de cima para baixo temos os seguintes passos de processamento: A) Imagem em escala de cinza; B) Imagem limiarizada inversamente; C) Floodfill; D) Floodfill invertido; E) Interseção D e B; 6) Interseção E com C e preenchendo os buracos com as cores das bordas da imagem original;

O processamento para preenchimento colorido baseou-se no item 5, verificando onde na imagem tinha-se o valor 255 de intensidade e na imagem 3 tinha-se 0. Caso isso fosse verdade era então identificado uma região de buraco. Como a análise foi feita pixel a pixel e linha por linha, era garantido que o pixel anterior ao que tal evento fosse verdadeiro ainda era o corpo do objeto, portanto coletava-se a intensidade dos três canais RGB do pixel anterior e seguia-se trocando na imagem original colorida todos os pixels que fossem brancos pelo valor coletado até que o valor do pixel analisado da vez fosse diferente de branco, significando que alcançou-se o outro lado do buraco. Então o processo se repetia para todas as linhas.

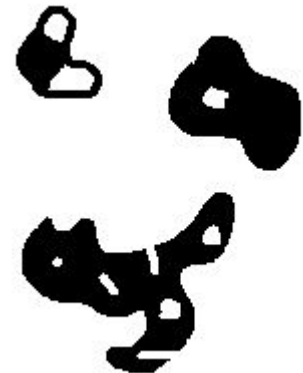
Seguem os resultados da implementação do problema 3, de encontrar o fecho convexo dos elementos vermelho, azul e verde da imagem.



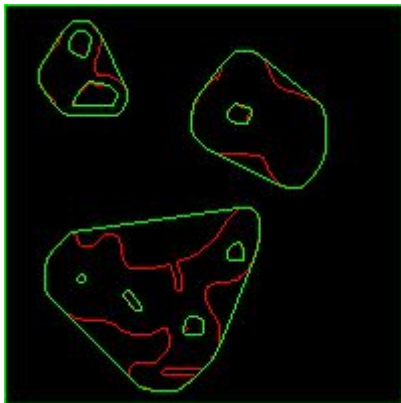
A



B



C



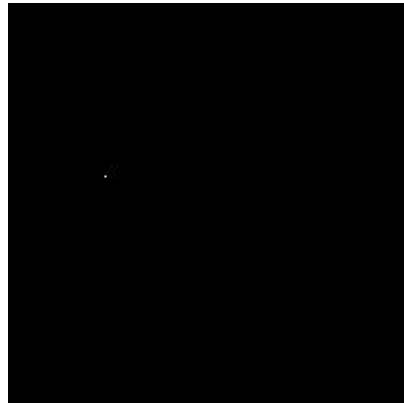
D

Em ordem alfabética: A) Imagem com os elementos desnecessários para a aplicação eliminados; B) Imagem convertida para graus de cinza; C) Imagem limiarizada para utilização das funções; e D) Imagem resultante, onde encontramos em vermelho o contorno dos elementos em suas formas originais e em verde o seu fecho convexo.

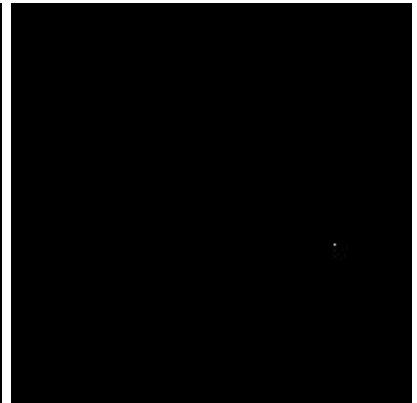
Os resultados da implementação da Transformata de Hit-or-Miss para detecção dos objetos da cor Magenta, que por conveniência eram todos quadriláteros obteve o seguinte resultado:



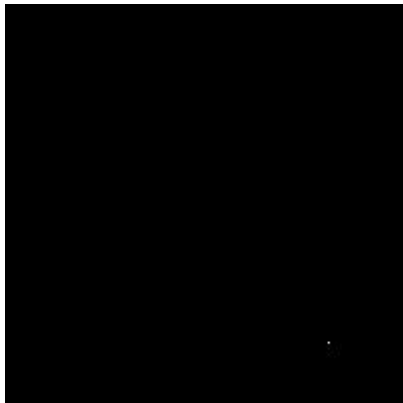
A



B



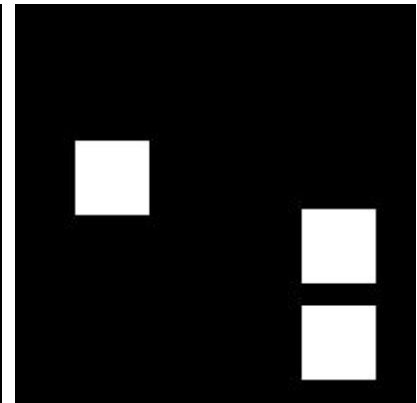
C



D



E



F



G

Em ordem alfabética temos: A) Imagem limiarizada inversamente; B) Hit-or-miss com máscara 29x31; C) Hit-or-miss com máscara 22x21; D) Hit-or-miss com máscara 16x15; E) Interseção de B, C e D; F) E dilatado 18 vezes com ES 3x3; G) Interseção F com imagem original;

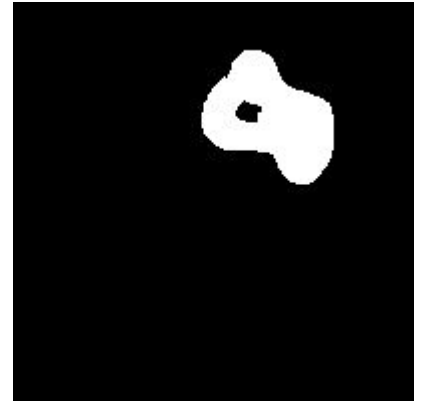
Seguem os resultados da aplicação do problema 5, de encontrar o esqueleto do elemento vermelho da imagem original:



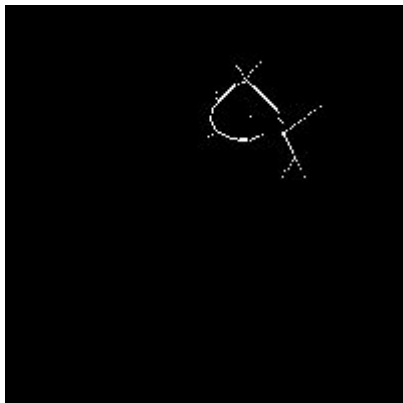
A



B



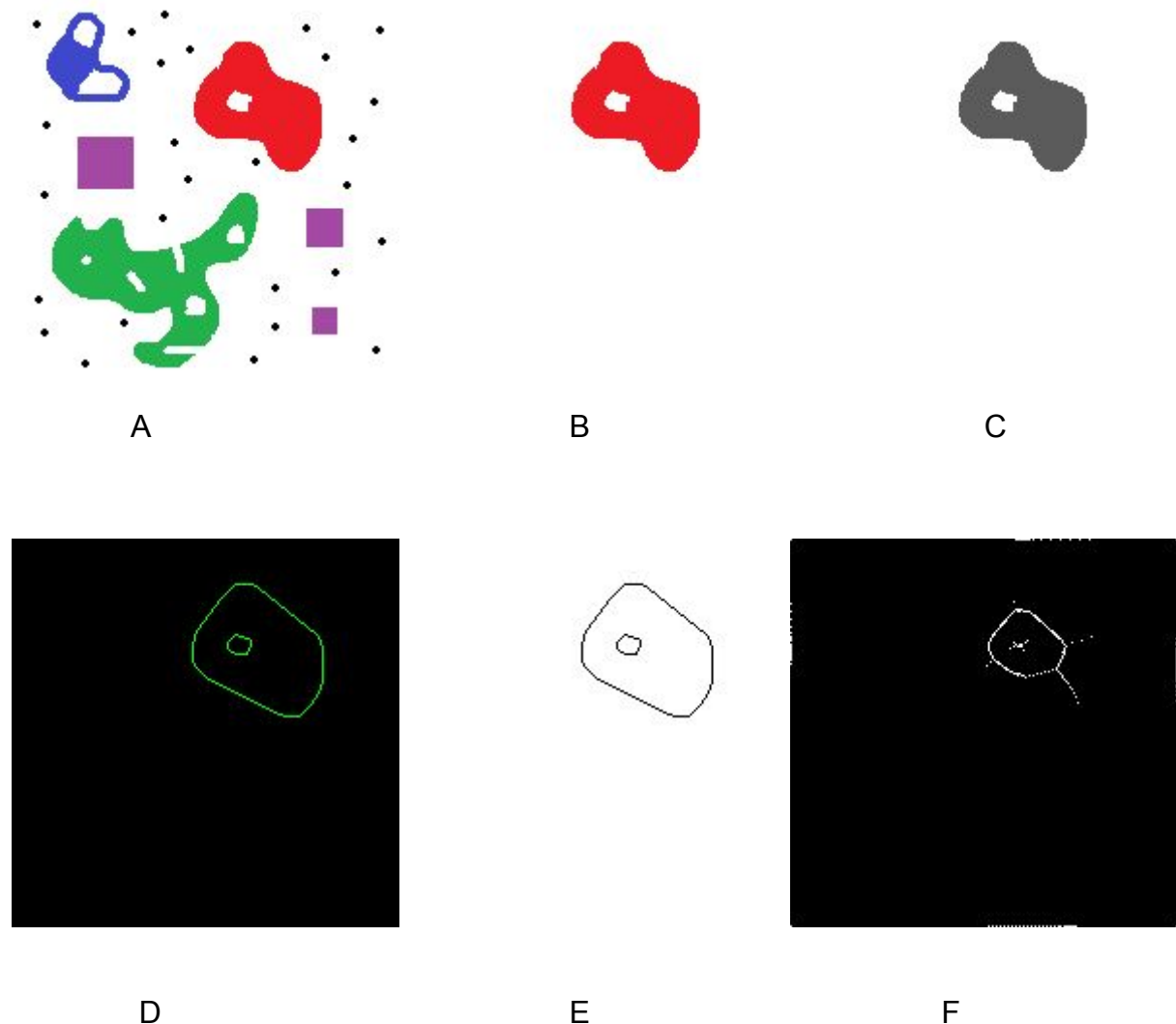
C



D

Em ordem alfabética: A) Eliminados os elementos desnecessários para a aplicação; B) Conversão da imagem para graus de cinza; C) Limiarização inversa da imagem para adaptar aos procedimentos seguintes; e D) Resultado, onde o esqueleto do elemento está em branco.

A resolução do problema 6 consistia na junção dos itens 3 e 5. Os seguintes resultados foram obtidos:



A) Imagem em escala de cinza; B) Exclusão dos elementos de outras cores; C) Elemento alvo em escala de cinza; D) Fecho convexo de C; E) Complemento de D; F) Esqueleto de E;

4. Conclusões

Concluiu-se com os operadores morfológicos é possível realizar uma gama de aplicações para as mais variadas finalidades de uma forma simples e com resultados de qualidade satisfatória.

A combinação das operações morfológicas com o auxílio de alguns processamentos adicionais para uma melhor aplicação resolveu de forma suficiente os problemas propostos como foi demonstrado no decorrer deste documento.