

Introdução ao Processamento de Imagem

Trabalho 2

João Victor Pinheiro de Souza
 dept. Ciência da Computação
 Ciência da Computação - UNB
 Matrícula: 180103407
 email: joaovictorps28@gmail.com

Abstract—Trabalho de Introdução ao Processamento de imagem onde vai trabalhar em temas como filtragem, morfologia matemática, segmentação.

Index Terms—Octave, Matlab, processamento de Imagem.

I. INTRODUCTION

O Trabalho tem como objetivo explorar algumas técnicas de processamento de imagens e aplicá-las em diferentes contextos. Em particular, vamos abordar o tema como morfologia matemática, segmentação de imagem e binarização, além de filtragem.

A ferramenta utilizada no trabalho foi o Octave/Matlab, sendo um software desenvolvida para computação matemática, para a solução de problemas numéricos, lineares e não-lineares. Onde foi utilizado o pacote de imagens que possibilita ler, manipulação e realizar operações nas imagens.

Com relação ao que vai ser abordado é importante explicar alguns conceitos tais como: Binarização, Morfologia e Segmentação.

A binarização de imagem é um processo de segmentação em que uma imagem em tons de cinza é convertida em uma imagem binária. A binarização é uma técnica comumente usada no processamento de imagens para separar os objetos de interesse do restante da imagem.

A binarização é baseada em um limiar, que é um valor de intensidade escolhido para dividir a imagem em pixels considerados como objeto (branco) e pixels considerados como fundo (preto). Podendo ser escrita como:

$$B(x, y) = \begin{cases} 1, & \text{se } I(x, y) \geq T \\ 0, & \text{se } I(x, y) < T \end{cases} \quad (1)$$

Sendo o "T" o limiar escolhido, a binarização simplifica a imagem, reduzindo a complexidade para análises e operações posteriores, além de destacar os objetos de interesse com mais clareza.

A morfologia matemática é uma técnica de processamento de imagens que lida com a forma, a estrutura e as propriedades geométricas dos objetos em uma imagem. Ela é baseada em conceitos e operações da teoria dos conjuntos

A morfologia matemática em processamento de imagens permite a análise e a manipulação de objetos e suas características, como tamanho, forma, conectividade e orientação. Ela é amplamente utilizada em diversas aplicações, como segmentação de imagens, remoção de ruído, detecção de bordas, preenchimento de buracos, extração de características e reconhecimento de padrões. Este processo é considerado de nível médio onde a saída do processo é um atributo da imagem. Porém, como estes atributos são resultados de processamentos na estrutura geométrica dos objetos. São representados em formato de imagens digitais.

A segmentação de imagem é o processo de dividir uma imagem em regiões ou objetos distintos com base em suas propriedades ou características. O objetivo da segmentação é separar automaticamente as áreas de interesse dos fundos ou outras regiões indesejadas na imagem. A segmentação desempenha um papel fundamental em várias aplicações de processamento de imagens, como análise de cena, reconhecimento de objetos, rastreamento de objetos, medição de características entre outros.

II. METODOLOGIA

A. Problema 1

Mediante operações morfológicas conte o número de buracos e indica seu diâmetro em pixels da imagem 'pcb.jpg'.

Para que todos os buracos fossem preenchidos foi aplicado o operação morfológica de fechamento definido por:

$$A \bullet B = (A \oplus B) \ominus B \quad (2)$$

O fechamento é especialmente útil para melhorar a qualidade da segmentação de objetos em uma imagem, eliminando pequenos buracos e unindo regiões adjacentes que devem ser consideradas como um único objeto. Na imagem foi feita para conectando as bordas próximas, completando o contorno do círculo.

Mas para a operação de fechamento ter um bom desempenho foi necessário binarizar a imagem com um limiar baixo, podendo ser pego pixels de baixas intensidade.

Com os buracos fechados foi feito o preenchimento utilizando a função `imfill`, onde usa um algoritmo baseado na reconstrução morfológica. Para que fossem pego os buracos e contar quantos existe foi feito a operação "AND" entre o

inverso da imagem original e na imagem que foi preenchido os buracos, para restringir as regiões de interesse.

Tendo os buracos segmentados, foi utilizado mais uma etapa para remover área indesejadas que não são os buracos, e para isso foi feito a operação morfológica de abertura, frequentemente utilizada em processamento de imagem para remover pequenos objetos, eliminar ruídos e separar regiões conectadas. tendo a formula:

$$A \circ B = (A \ominus B) \oplus B \quad (3)$$

Com todos os buracos segmentados, foi usado a a função regionprops para contar os numeros de buracos e indicar seu diâmetros em pixels.

B. Problema 2

Com a imagem "morf_test" o programa deve entregar uma imagem binária como saída, com o fundo branco e as imagens pretas.

Para a primeira parte foi pedido criar uma imagem que seja somente o fundo (mediante operações morfológicas), e subtrair essa imagem da original. A solução do problema foi feita parti da operação morfológica de fechamento, conseguindo a imagem de fundo, e em seguida subtraindo ela com a imagem original. Obtendo uma imagem com fundo preto.

Depois de obter ela, é realizado a binarização, pegando o limiar com a função graythresh, que calcula um limite global da imagem em tons de cinza usando o método de Otsu. Quando é obtido a imagem binarizada, pega o inverso dela.

Foi pedido filtros prévios para tentar melhorar o resultado, e que aplique operações morfológicas para tentar evitar símbolos desconectados ou ruídos.

Para essa parte foi feito um filtro de mediana antes de passar para reduzir o ruído, preservando melhor as bordas e os detalhes da imagem. E foi utilizada o operação morfológica de dilatação para restaurar o tamanho original dos objetos e a melhorar a conectividade entre eles. tendo sua representação:

$$A \oplus B = \{Z | (\hat{B})_Z \cap A \neq \emptyset\} \quad (4)$$

C. Problema 3

Faça um programa que segmente a imagem "img_cells". Primeiro é realizado a binarização da imagem utilizando o mesmo método do Problema 2. em seguida é preenchida as células, utilizado a função imfill. Para que a função preencha todas as células foi preciso criar bordas laterais, depois de criadas, preenchia as células e no final retirava as bordas criadas anteriormente.

Com as células preenchidas utilizou a função bwdist para calcular a transformada de distância do complemento da imagem. O valor de cada pixel na imagem de saída é a distância entre esse pixel e o pixel diferente de zero mais próximo da imagem. Em seguida pega o complemento da imagem transformada de distância de modo que os pixels claros representem altas elevações e os pixels escuros representem baixas elevações para a transformação de Watershade. e finalmente é aplicada a transformada de Watershed.

A transformação de Watershade é uma técnica de segmentação de imagem baseada em regiões que visa identificar regiões de interesse e delimitar os contornos dos objetos presentes na imagem.

A ideia principal por trás da transformada de watershed é simular uma inundação em uma paisagem topográfica. Nesse contexto, a imagem é tratada como um mapa topográfico, onde os níveis de cinza representam diferentes altitudes. A inundação começa a partir de marcadores definidos pelo usuário, que indicam as áreas conhecidas de interesse ou os pontos de partida para a segmentação.

A transformada de watershed é amplamente utilizada em várias aplicações de processamento de imagens, como segmentação de objetos, análise de texturas, segmentação de regiões de interesse e detecção de bordas. Ela é especialmente útil quando existem objetos com sobreposição, toques ou limites mal definidos na imagem.

III. RESULTADOS

Como foi mostrado a resolução dos problemas, agora vai mostrar a parte dos resultados.

A. Problema 1

primeiramente mostrando a imagem original, depois aplicando a operação de fechamento, com as bordas fechadas possibilitou criar a imagem com o buracos preenchimento, logo depois obteve a imagem resultante que realiza o "AND" da imagem original com a dos buracos preenchidos, e por fim a que realiza a operação de abertura.



Fig. 1. Imagens original

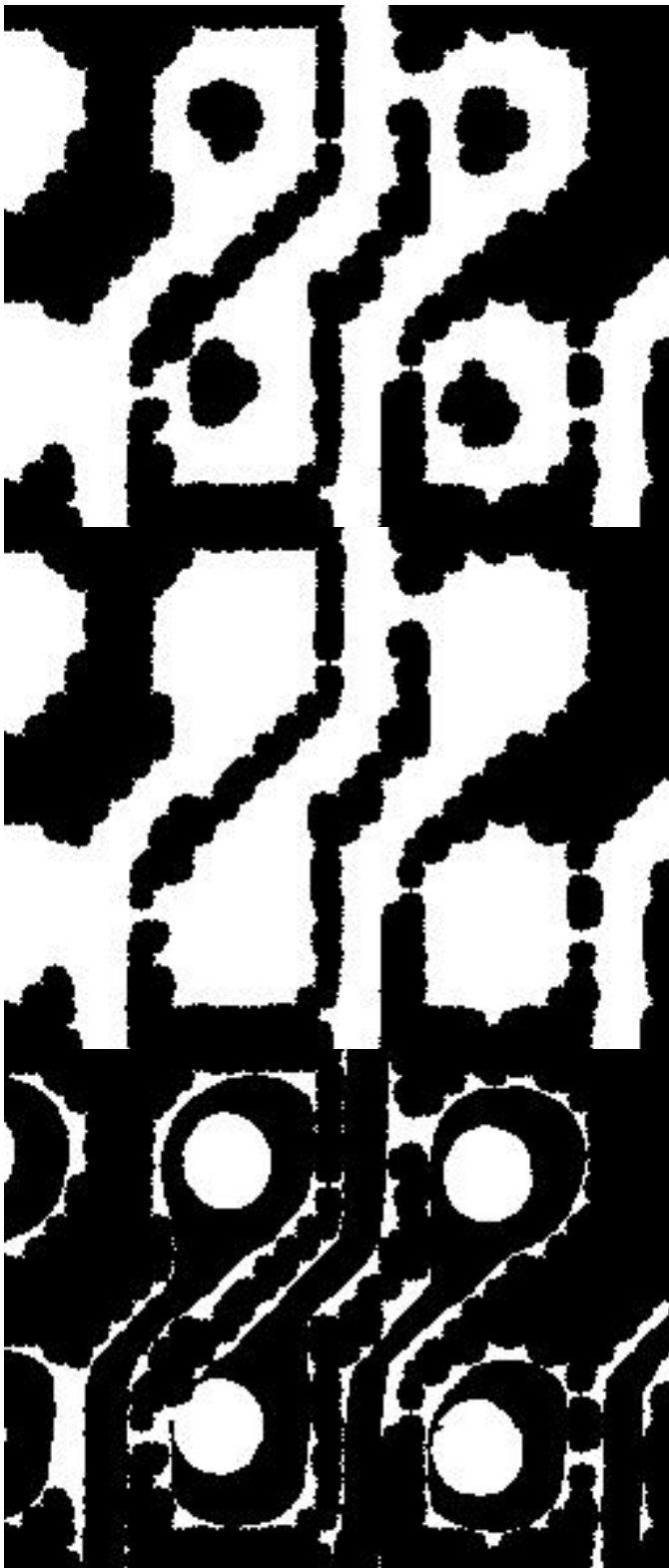


Fig. 2. Imagens com operação de fechamento, de preenchimento dos buracos, e seleccionando os buracos, respectivamente

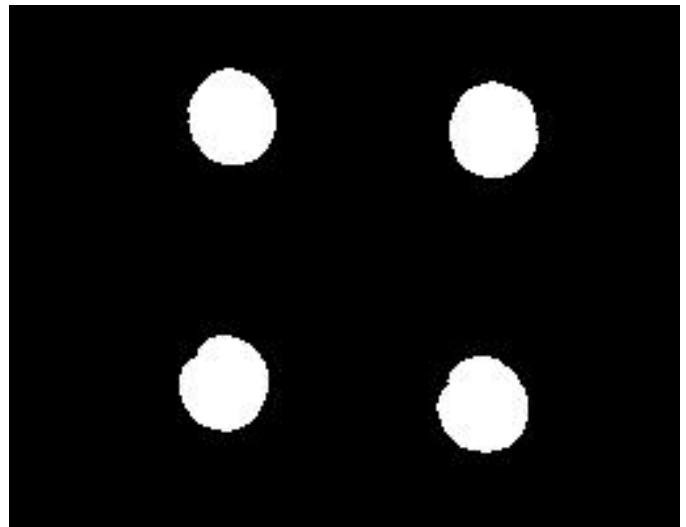


Fig. 3. Imagens final depois realizar a operação de abertura

Cada um dos buracos tendo seu diâmetro em pixels em 36.437, 36.849, 36.817 e 36.705.

B. Problema 2

Primeiramente mostrando a imagem original:

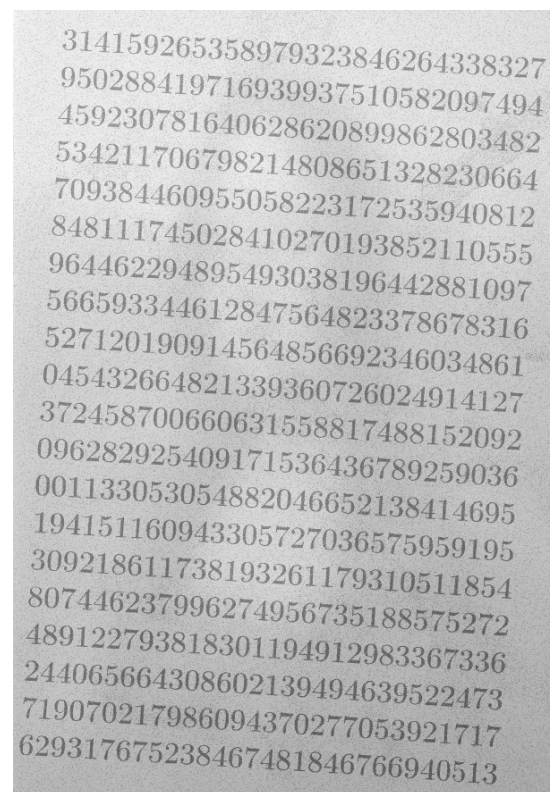


Fig. 4. Imagem original

Em seguida mostrando o resultado da imagem depois que retira o fundo e subtrai com o original. Nela contam ruídos, mas ainda tem algumas parte não estão legível.

314159265358979323846264338327
 950288419716939937510582097494
 459230781640628620899862803482
 534211706798214808651328230664
 709384460955058223172535940812
 848111745028410270193852110555
 964462294895493038196442881097
 566593344612847564823378678316
 527120190914564856692346034861
 045432664821339360726024914127
 372458700660631558817488152092
 096282925409171536436789259036
 001133053054882046652138414695
 194151160943305727036575959195
 309218611738193261179310511854
 807446237996274956735188575272
 489122793818301194912983367336
 244065664308602139494639522473
 719070217986094370277053921717
 629317675238467481846766940513

Fig. 5. Imagem sem o fundo

Em sequencia mostrando o resultado usando a operação de dilatação, as parte desconectadas foram corrigidas e conseguindo ter mais visibilidade.

314159265358979323846264338327
 950288419716939937510582097494
 459230781640628620899862803482
 534211706798214808651328230664
 709384460955058223172535940812
 848111745028410270193852110555
 964462294895493038196442881097
 566593344612847564823378678316
 527120190914564856692346034861
 045432664821339360726024914127
 372458700660631558817488152092
 096282925409171536436789259036
 001133053054882046652138414695
 194151160943305727036575959195
 309218611738193261179310511854
 807446237996274956735188575272
 489122793818301194912983367336
 244065664308602139494639522473
 719070217986094370277053921717
 629317675238467481846766940513

Fig. 6. Imagem Usando a operação de dilatação

E por fim mostrando a imagem usando o filtro de mediana, nela contem resultados parecidos quanto aplica sem o filtro.

314159265358979323846264338327
 950288419716939937510582097494
 459230781640628620899862803482
 534211706798214808651328230664
 709384460955058223172535940812
 848111745028410270193852110555
 964462294895493038196442881097
 566593344612847564823378678316
 527120190914564856692346034861
 045432664821339360726024914127
 372458700660631558817488152092
 096282925409171536436789259036
 001133053054882046652138414695
 194151160943305727036575959195
 309218611738193261179310511854
 807446237996274956735188575272
 489122793818301194912983367336
 244065664308602139494639522473
 719070217986094370277053921717
 629317675238467481846766940513

Fig. 7. Imagem com filtro

C. Problema 3

Inicializando com a imagem original:

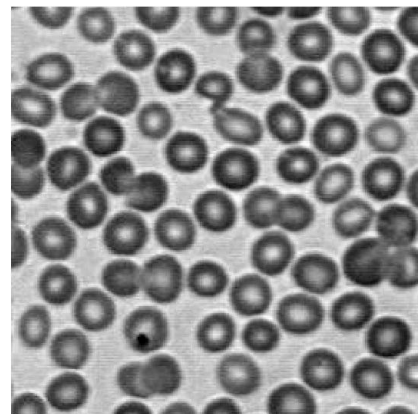


Fig. 8. Imagem original

mostrando a imagem binarizada com as células preenchidas, nota que depois do tratamento foram preenchidos todas as células.

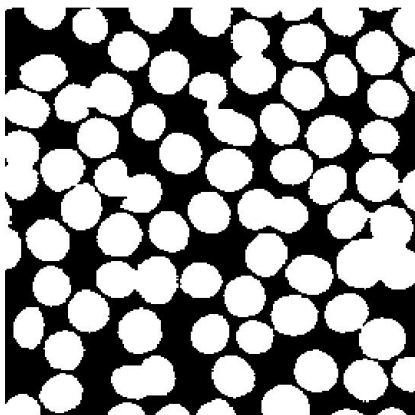


Fig. 9. Imagem célula preenchida

Em seguida com a imagem da distância calculada:

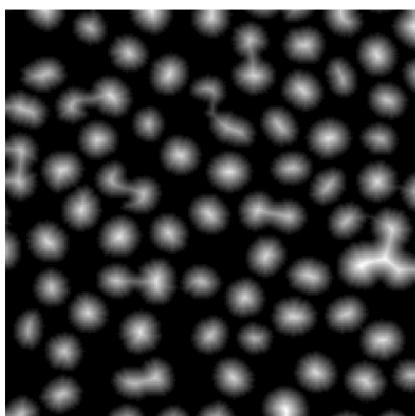


Fig. 10. Imagem da distância calculada

Colocando como ficou a segmentação da imagem:

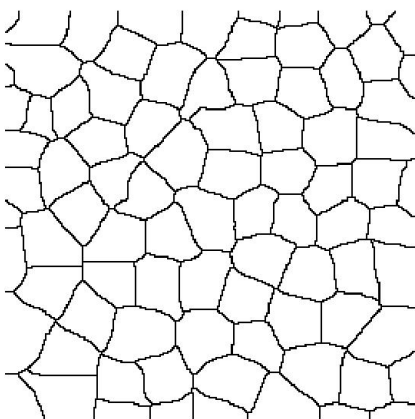


Fig. 11. Imagem segmentada

Colocando a imagem segmentada com a imagem binarizada, acrescentando com core para ficar melhor a visualização da segmentação.

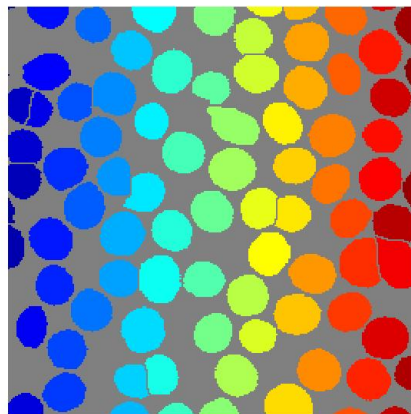


Fig. 12. Imagem das células segmentada

E por fim a imagem original com a segmentação:

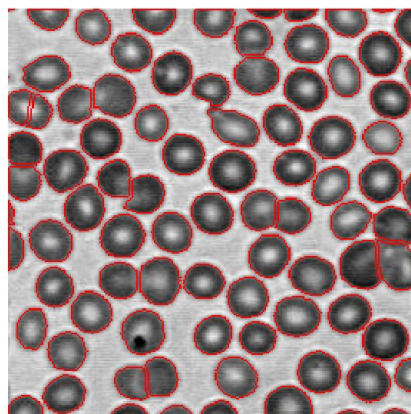


Fig. 13. Imagem original com as células segmentada

IV. CONCLUSÃO

No problema 1 a segmentação dos buracos teve algumas parte imperfeitas, principalmente no buraco que tinha uma parte da borda faltando, mas apesar disso ficou bastante próximo de como seria ela, obtendo a quantidade de buracos e seus respectivos diâmetros.

No problema 2 comparando a imagem que não foi aplicado a operação de dilatação com a que foi aplica teve uma grade diferença, na parte visualização e legibilidade. Enquanto a que foi aplicada o filtro de mediana, por mais que os resultados sejam parecidos com a que foi aplicada a dilatação, nela contem umas partes desconectadas nos dígitos, pois quando foi tirado os ruídos e suavizado as bordas, foram apagados uma partes dos dígitos, dificultando para conectar as partes descontinuas.

Por fim no problema 3 foi obtido a segmentação das células, apesar que algumas partes não foram muito preciso, pois as células não são uma esfera perfeita, apesar desse problema teve uma boa segmentação e bem definidas.