

ã03.328

Ueide dos Santos Souza

*ANÁLISE DE MÉTODOS DE
SEGMENTAÇÃO PARA EXTRAÇÃO DE
INFORMAÇÕES EM IMAGENS DE
MICROSCOPIA*

Feira de Santana – BA

Março / 2008

Ueide dos Santos Souza

*ANÁLISE DE MÉTODOS DE
SEGMENTAÇÃO PARA EXTRAÇÃO DE
INFORMAÇÕES EM IMAGENS DE
MICROSCOPIA*

Monografia apresentada à Banca de Graduação em Engenharia de Computação da Universidade Estadual de Feira de Santana para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador:

Prof. Dr. Carlos Alberto Rodrigues

DEPARTAMENTO DE TECNOLOGIA
UNIVERSIDADE ESTADUAL DE FEIRA DE SANTANA

Feira de Santana – BA

Março / 2008

Monografia de Projeto Final de Graduação sob o título “*ANÁLISE DE MÉTODOS DE SEGMENTAÇÃO PARA EXTRAÇÃO DE INFORMAÇÕES EM IMAGENS DE MICROSCOPIA*”, defendida por Ueide dos Santos Souza e aprovada em 04 de abril de 2008, em Feira de Santana Ba, pela banca examinadora constituída pelos professores:

Prof. Dra. Michele Fúlvia Angelo
Departamento de Tecnologia - UEFS
Examinador

Prof. Dr. Carlos Alberto Rodrigues
Departamento de Exatas - UEFS
Orientador

Prof. Dr. Haroldo Gonçalves Benatti
Departamento de Exatas - UEFS
Examinador

*Dedico esta monografia a meus pais,
cujo exemplo de honestidade e trabalho
tem sido um norteador para a minha vida,
e para minha namorada, que tem
me dado apoio nos momentos mais difíceis
e mostrado a simplicidade de ter esperança.*

Agradecimentos

A DEUS por iluminar meus caminhos.

Aos meus pais pelo incentivo e amor em todos os momentos.

Ao meu irmão pelo apoio.

A minha namorada Laiz pelo seu carinho, compreensão e incentivo.

Aos amigos Eder, Mateus, Ortega, Silas e Vitor pelo companheirismo e tempo que passamos juntos.

Aos meus companheiros de república pelo convívio e cumplicidade demonstrados ao longo da minha graduação.

Aos colegas da UEFS pelas contribuições neste trabalho.

Ao meu orientador Prof. Dr. Carlos Alberto Rodrigues pela paciência e generosidade em ensinar e me orientar, além da sua inestimável amizade.

Sumário

Lista de Figuras

Resumo

1	Introdução	p. 11
2	Definição do Projeto	p. 13
3	Revisão de Literatura	p. 14
3.1	Passos fundamentais em Processamento Digital de Imagens (PDI) . . .	p. 14
3.2	Conceitos básicos	p. 15
3.3	Filtros	p. 16
3.3.1	Equalização do histograma	p. 17
3.3.2	Filtros da Mediana	p. 17
3.3.3	Filtros da Média	p. 18
3.4	Segmentação da imagem	p. 18
3.4.1	Identificação de Segmentos	p. 19
3.4.2	Segmentação por limiarização	p. 19
3.4.3	Segmentação por textura	p. 21
3.4.4	Segmentação por regiões	p. 22
3.4.5	Segmentação por contornos ativos	p. 23
3.4.6	Dificuldades inerentes	p. 23
3.5	Algoritmos Básicos	p. 24

3.5.1	Histograma de Níveis de Intensidade	p. 24
3.5.2	Equalização do Histograma	p. 25
3.5.3	Limiarização	p. 27
4	Desenvolvimento do <i>software</i>	p. 29
4.1	Componentes Conexos	p. 29
4.1.1	Passo 01	p. 29
4.1.2	Passo 02	p. 30
4.1.3	Passo 03	p. 31
4.1.4	Passo 04	p. 31
4.2	Extração de informações do Objeto Segmentado	p. 32
5	Resultados	p. 34
5.1	Pré-processamento	p. 35
5.2	Segmentação	p. 36
5.3	Extração de Características	p. 39
6	Conclusão e Comentários Finais	p. 41
6.1	Conclusões	p. 41
6.2	Contribuições da monografia e desenvolvimentos futuros	p. 41
	Referências	p. 43

Listagens de Pseudo Código e de código-fonte

3.1	Algoritmo 1 - Histograma de uma imagem	p. 24
3.2	Algoritmo 2 - Equalização de uma imagem	p. 25
3.3	Algoritmo 3 - Limiarização de Imagens	p. 28
4.1	Método que encontra o primeiro pixels de cada objeto	p. 29
4.2	Captura Vizinhos de 8 de intensidade preta	p. 30
4.3	Método que captura o objeto	p. 31
4.4	Extração do Perímetro e Raio	p. 32

Lista de Figuras

1	Passos fundamentais PDI. (GONZALEZ; WOODS, 2000)	p. 14
2	Relações de vizinhança mais comuns. (MEYER, 1994)	p. 16
3	Pixel $P_{i,j}$ e os pixels 8-vizinhança.	p. 17
4	Vetor de pixels ordenados.	p. 18
5	Histograma de uma imagem em níveis de cinza com limiar global.	p. 21
6	(a) Imagem particonada; (b) "quadtree" correspondente. (GONZALEZ; WOODS, 2000)	p. 22
7	Imagem e seu Histograma.	p. 24
8	Imagem original, FDA e seu Histograma.	p. 26
9	Imagem equalizada, FDA e seu Histograma.	p. 27
10	Limiarização de Imagens.	p. 27
11	Console abrir imagem PNG, JPEG e GIF.	p. 34
12	Inteface MDI.	p. 35
13	Filtro da mediana.	p. 35
14	Entrada do limiar.	p. 36
15	Resultado da segmentação. Limiar $T = 30$	p. 37
16	Resultado da segmentação. Limiar $T = 85$	p. 37
17	Resultado da segmentação. Limiar $T = 127$	p. 38
18	Resultado da segmentação. Limiar $T = 200$	p. 38
19	Informações de contagem perímetro raio e area para um $T = 127$ para uma escala de 10cm de largura e altura	p. 39

20	Gráfico da polidispersividade para um $T = 127$ e escala de 10cm de largura e altura	p.40
----	--	------

Resumo

O problema de segmentação de imagens digitais tem gerado uma grande diversidade de abordagens para solucioná-lo. Em vista disso, inúmeras áreas do conhecimento vem se beneficiando desta tecnologia, tais como Biologia, Física, Engenharia de Matérias, Medicina, Astronomia, Microscopia entre outras. Uma abordagem bastante simples, a limiarização, ainda tem sido intensivamente utilizada, por viabilizar aplicações de visão em tempo-real.

Este trabalho sugere o desenvolvimento de um *software* para Processamento e Análise de Imagens para imagens de Microscopia, na linguagem Java. Além de desenvolver o *software*, serão concentrados estudos em uma etapa fundamental da análise de imagens: a segmentação. A segmentação da imagem consiste em separar da imagem o elemento que será analisado e medido.

Este programa é capaz de manipular os vetores bidimensionais de pixels que compõe uma imagem, com o objetivo de extrair diversos elementos de uma imagem de microscopia e posteriormente calcular medidas, tais como, raio, perímetro e área, que permite caracterizar esses elementos. A contagem de elementos de uma imagem também pode ser realizada através desse *software*.

1 *Introdução*

Análise e Processamento de Imagens é uma das áreas da Computação que mais se expandiu na última década, isto devido ao aumento da velocidade de processamento dos computadores. Ao processar uma imagem, há vários objetivos que podem ser atingidos, como, por exemplo, melhorar a qualidade de uma imagem ruidosa. Uma definição mais formal é dada por Russ (RUSS, 1998), onde ele afirma que processar uma imagem consiste em melhorar a aparência visual da imagem para a visão humana e/ou preparar a imagem para a análise e mensuração de características e estruturas presentes.

Muitas vezes, fala-se de processamento e análise de imagens como uma mesma área, mas convém que se faça uma distinção. A análise de imagens engloba todas as medidas que são feitas nas imagens, que de outra forma não seria possível, ou muito difícil para o homem fazer. Seguindo esta tendência, quando for mencionado processamento, também inclui-se análise de imagens.

Historicamente, o interesse em técnicas de Análise e Processamento de Imagens surgiu, principalmente, da necessidade de melhorar a qualidade da informação pictorial para interpretação humana. Uma das primeiras aplicações dessas técnicas foi a melhoria de ilustrações de jornais enviados por cabo submarino entre Londres e New York por volta de 1920. Mas, a evolução das técnicas de processamento digital de imagens se deu em meados dos anos 60 com o advento de computadores digitais e com o programa espacial norte-americano. Em 1964 as imagens da lua transmitidas pela sonda Ranger 7 foram processadas por um computador para corrigir vários tipos de distorções inerentes à câmara de televisão à bordo. As técnicas de processamento usadas nesta época serviram de base para o realce e restauração de imagens de outros programas espaciais posteriores, como as expedições tripuladas da série Apollo para a lua, por exemplo.

Atualmente, inúmeras áreas do conhecimento vêm se beneficiando desta tecnologia, já que, onde quer que se usem imagens, alguma forma de processamento e análise eventualmente terá de ser feita. Por exemplo, em aplicações de automação industrial, o processa-

mento se dá na melhoria das imagens vindas de sensores visuais em robôs. Na medicina, o processamento é feito para melhorar o contraste ou o nível de intensidade das cores para uma melhor interpretação das imagens de Raios-X e outras imagens biomédicas. Na análise de recursos naturais e meteorologia, por meio de imagens de satélites, o processamento é feito para definir e analisar condições climáticas. Na arqueologia, métodos de processamento de imagens restauraram com sucesso imagens borradas que eram os únicos registros de artefatos raros perdidos ou danificados após serem fotografados. Na Física e em outros campos relacionados, as técnicas computacionais fazem contínuos melhoramentos em áreas tais como plasmas de alta energia e microscopia. Similarmente, técnicas bem sucedidas de análise e processamento podem ser encontradas em astronomia, biologia, medicina nuclear e defesa.

A interdisciplinaridade é uma característica marcante desta área, pois o processamento das imagens envolverá conhecimento do problema em que a aplicação está sendo feita. Além disso, o desenvolvimento dos algoritmos requer conhecimentos de linguagens de computação, física e matemática, para a implementação dos mesmos.

O foco de estudo desse trabalho está atrelado ao primeiro passo da análise de imagem, denominada segmentação de imagem. Existem autores que afirmam que o sucesso ou fracasso da extração de informação a partir de uma imagem, esta totalmente ligada à eficiência da segmentação da imagem. Daí surge a grande importância das técnicas de segmentação de imagens. Segmentação de imagem consiste em usar o computador para definir na imagem, recortes automáticos ao redor de objetos de interesse.

A segmentação subdivide uma imagem em suas partes ou objetos constituintes. O nível até o qual essa subdivisão deve se realizada, assim como a técnica utilizada, depende do problema que está sendo resolvido(GONZALEZ; WOODS, 2000). Na seção 3.4, serão dadas definições mais formais sobre segmentação.

Nas seções seguintes, serão detalhadas cada uma das fases do desenvolvimento do presente trabalho, desde a descrição do projeto (seção 2), passando pela revisão da literatura (seção 3) e desenvolvimento (seção 4) até os resultados(seção 5), e por fim, a conclusão e perspectivas(seção 6).

2 *Definição do Projeto*

Neste trabalho foi desenvolvido um *software* de interface gráfica utilizando a linguagem JAVA (SUN, 2007), para processamento e análise das imagens em Microscopia. Essas imagens foram fornecidas pelo grupo de polímeros "Bernhard Gross" do Instituto de Física da USP de São Carlos, cujo intercâmbio é bastante promissor com respeito a perspectiva de publicações de nossos trabalhos.

Além de desenvolver o *software*, foi dado enfoque no estudo de uma etapa fundamental em análise de imagens: a segmentação. Como foi dito na introdução, a eficiência da análise depende, em grande parte, da eficiência da segmentação, com isso, diversos métodos de segmentação podem ser testados a fim de saber qual é o mais eficiente.

Aplicou-se em imagens de polímeros métodos que permite separar e contar os glóbulos, e posteriormente, extrair informações de medida de raio, área e perímetro. Medir o raio é útil para avaliar o grau de polidispersividade destes glóbulos sobre a amostra do polímero, devido ao solvente utilizado (RODRIGUES; COSTA, 2003). A partir de uma técnica eficiente de segmentação, diversas imagens poderão ser caracterizadas com respeito ao grau de polidispersividade e a influência do solvente neste processo. Outra maneira de caracterizar e analisar um polímero é de acordo com a contagem dos glóbulos, daí a importância dessa informação.

O método de segmentação desenvolvido e aplicado neste trabalho foi o de limiarização (CASTLEMAN, 1996), que apesar da simplicidade se mostrou bem eficiente para o tipo de imagem utilizada neste trabalho. Para analisar a imagem e extrair as informações, foi desenvolvido um algoritmo baseado em componentes conexos, cuja implementação será detalhada na seção 4.

É comum em análise de imagens que cada imagem tenha suas particularidades, sendo que determinado método de segmentação seja eficiente em um problema e não seja em outro. Deste modo, torna-se necessário testar diversos métodos para se determinar qual é o mais apropriado para cada imagem.

3 Revisão de Literatura

Nesta seção são apresentados os principais conceitos que nortearam o desenvolvimento deste trabalho, contendo desde os princípios fundamentais do processamento de imagens até as técnicas avançadas de segmentação de imagens.

3.1 Passos fundamentais em Processamento Digital de Imagens (PDI)

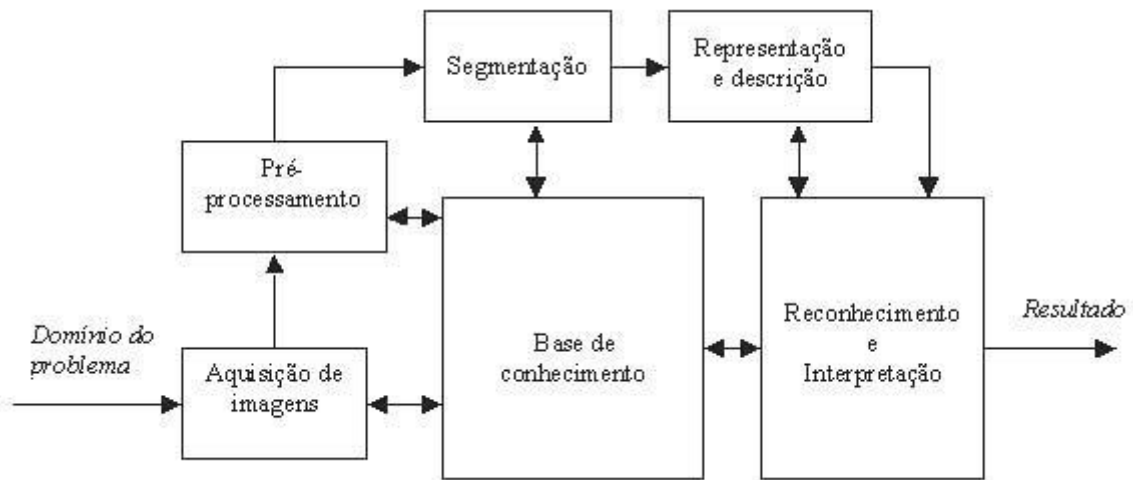


Figura 1: Passos fundamentais PDI. (GONZALEZ; WOODS, 2000)

A figura 1, retirada de Gonzalez (GONZALEZ; WOODS, 2000) ilustra os passos fundamentais em processamento de imagens. O primeiro passo no processo é a aquisição da imagem isto é, adquirir uma imagem digital. Para se fazer isso, é necessário um sensor para imageamento e a capacidade de digitalizar o sinal produzido pelo sensor. Neste trabalho, esse sensor será um microscópio de Força Atômica (GERBER, 1996), que é utilizado pelo grupo "Bernhard Gross" IFSC/USP, que será o fornecedor das imagens de polímeros.

Após a obtenção de uma imagem digital, o próximo passo trata de pré-processar

aquela imagem. A função chave no pré-processamento é melhorar a imagem de forma a aumentar as chances para o sucesso dos processos seguintes. O pré-processamento tipicamente envolve técnicas para o realce de contrastes, remoção de ruído, entre outras.

O próximo estágio trata da segmentação que divide uma imagem de entrada em partes ou objetos constituintes. Em geral, a segmentação automática é uma das tarefas mais difíceis no processamento de imagens digitais. Como já foi dito anteriormente, é o objetivo de estudo desse projeto e será mais detalhado na próxima seção.

A representação por fronteira é adequada quando o interesse se concentra nas características da forma externa, tais como cantos ou pontos de inflexão. A representação por região é adequada quando o interesse se concentra em propriedades internas, tais como textura ou a forma do esqueleto. Em algumas aplicações, entretanto, essas representações coexistem. Essa situação acontece em aplicações de reconhecimento de caracteres, que freqüentemente requer algoritmos baseados na forma da borda, bem como também esqueletos e outras propriedades internas.

O processo de descrição, também chamado seleção de características, procura extrair características que resultem em alguma informação quantitativa de interesse ou que sejam básicas para discriminação entre classes de objetos. Em se tratando de reconhecimento de caracteres, descritores tais como buracos e concavidades são características poderosas que auxiliam na diferenciação entre uma parte do alfabeto e outra.

O último estágio envolve reconhecimento e interpretação. Reconhecimento é o processo que atribui um rótulo a um objeto, baseado na informação fornecida pelo seu descritor. A interpretação envolve a atribuição de significado a um conjunto de objetos reconhecidos.

3.2 Conceitos básicos

O processamento digital de imagens atua sobre imagens digitais. Dessa forma, existe a necessidade de converter um campo de imagem contínuo para uma imagem digital equivalente (PRATT, 2001). Essa conversão ocorre através do mapeamento da imagem em uma matriz de pixels (*picture elements*). Cada pixel é composto por uma coordenada (x, y), que fornece a sua posição na imagem, e um modelo de cor, definido de acordo com o objetivo de uso da imagem.

Em imagens coloridas o pixel assume o valor de 0 a 255, e a variação de intensidade

se dá em três canais, RGB(vermelho, verde e azul). Em imagens com níveis de cinza, o pixel assume o valor de 0 a 255, representando a variação das intensidades de cinza. Em imagem binária, o pixel assume valor 0 ou 255, sendo este preto ou branco.

Além do modelo de cor, o pixel possui outras propriedades determinantes no processamento e análise da imagem: as relações de vizinhança e a conectividade. As relações de vizinhança mais utilizadas são a 4-vizinhança e a 8-vizinhança, em grade quadrada, e a 6-vizinhança em uma grade hexagonal (MEYER, 1994) . A representação dessas relações de vizinhança pode ser vista na figura 2. Essas relações podem ser expandidas para formas mais complexas, como 16-vizinhança em grade quadrada e 12 ou 24-vizinhança em grade hexagonal, o que possibilita a obtenção de melhores resultados no processamento e análise, mas aumenta o custo computacional.

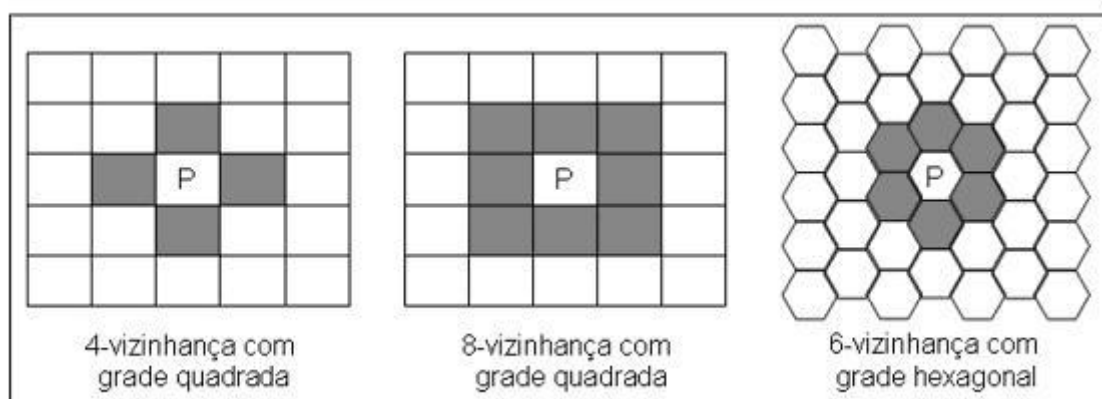


Figura 2: Relações de vizinhança mais comuns. (MEYER, 1994)

A conectividade está condicionada pela relação de vizinhança, pois para que dois pixels sejam conexos deve haver uma seqüência de pixels que os liga, de forma que dois pixels consecutivos desse caminho tenham características comuns e sejam vizinhos (FACON, 2001). Relações de vizinhança e conectividade são os critérios básicos para a formação de regiões no processo de segmentação (MEYER, 1994), que será mais especificada na seção 4.

3.3 Filtros

Filtragem digital é um conjunto de técnicas destinadas a corrigir e realçar uma imagem. A correção é a remoção de características indesejáveis, e a melhoria/realce é a acentuação de características. O objetivo principal das técnicas de realce é processar uma

imagem, de modo que o resultado seja mais apropriado para uma aplicação específica do que a imagem original (GONZALEZ; WOODS, 2000).

Existem duas grandes categorias básicas que classificam as técnicas de realce; Domínio Espacial: o processamento é efetuado pela transformação/manipulação direta dos pixels; Domínio da Frequência: o processamento é efetuado pela modificação da Transformada de Fourier da imagem (GONZALEZ; WOODS, 2000). Técnicas de realce envolvendo as duas categorias básicas não são incomuns, porém nesse trabalho optou-se apenas pela utilização de técnicas de Domínio Espacial, cujos funcionamentos serão descritos nas sub-seções seguintes.

3.3.1 Equalização do histograma

A opção Equalização do Histograma é uma maneira de manipulação de histograma que reduz automaticamente o contraste em áreas muito claras ou muito escuras numa imagem. Expande também os níveis de cinza ao longo do intervalo.

3.3.2 Filtros da Mediana

Para ajudar a melhorar a qualidade da imagem e ter maior sucesso no processo de segmentação, este projeto utilizará uma técnica de pré-processamento denominada filtro da mediana, que serve para eliminar ruídos na imagem com intensidades muito diferentes da vizinhança, mantendo assim os principais detalhes da imagem.

O filtro da mediana baseia-se em um algoritmo simples e eficiente, que sua idéia será mostrada na figura 3. Considere imagens em níveis de cinza representados por inteiros variando de 0 (preto) a 255 (branco). Dado um pixel $P_{i,j}$ e os pixels que o circundam (8-vizinhança).

$P_{i-1,j-1}$	$P_{i,j-1}$	$P_{i+1,j-1}$
$P_{i-1,j}$	$P_{i,j}$	$P_{i+1,j}$
$P_{i-1,j+1}$	$P_{i,j+1}$	$P_{i+1,j+1}$

Figura 3: Pixel $P_{i,j}$ e os pixels 8-vizinhança.

O valor do pixel $Q_{i,j}$ na nova imagem será definido pela mediana dos valores desses

9 pontos. A mediana é obtida ordenando-se os 9 pontos em um vetor e tomando o valor do ponto que está na posição central do mesmo. Na ilustração abaixo (figura 4), o valor do pixel na nova imagem será A5:

				Q_{ij}				
A1	A2	A3	A4	A5	A6	A7	A8	A9

Figura 4: Vetor de pixels ordenados.

3.3.3 Filtros da Média

O filtro da média é uma técnica que permite a redução de ruído em uma imagem. Considerando uma dada vizinhança, o método consiste em calcular a média de todos os pontos dentro desta vizinhança para cada pixel da imagem original.

O filtro da média é útil por ser uma técnica simples, relativamente rápida de ser aplicada, pois não necessita de nenhum processamento mais complexo do que somas e divisões. O filtro da média é mais indicado para ruído do tipo "Salt and Pepper", pois pode retirar pixels espúrios de uma imagem (RUSS, 1998).

3.4 Segmentação da imagem

Ao analisar uma imagem, é muito freqüente a necessidade de destacar uma determinada região, permitindo uma visualização mais precisa dos objetos que se busca analisar na imagem. Para tanto, utilizam-se técnicas de segmentação de imagens. Segundo Soille (1999), segmentar uma imagem consiste em dividí-la em diferentes regiões, de forma que os pixels de cada uma delas possuam características específicas, por exemplo, nível de cinza com valor semelhante. A partir da imagem segmentada, é possível extrair informações de cada objeto que a compõe, que em nosso problema, seria o perímetro, área e quantidade de glóbulos presente na imagem.

São várias as técnicas disponíveis para a segmentação de imagens, mas a grande maioria deriva de uma das seguintes abordagens que serão conceituadas na subseções abaixo:

3.4.1 Identificação de Segmentos

Existe uma certa dificuldade em se estabelecer procedimentos de segmentação em imagens nos computadores. A identificação dos segmentos deve obedecer a algumas características:

Os pixels devem possuir alguma propriedade em comum dentro da imagem. A propriedade em comum pode ser uma superfície que representa um osso dentro de uma radiografia, uma peça sendo submetida a um controle de qualidade ou um mapa ilustrando alguma característica de uma foto.

Dentre as propriedades desejáveis de uma imagem são destacadas algumas a seguir:

- A região representada pelos pixels deve ser homogênea.
- Os segmentos são regiões fechadas e devem ser delimitadas por bordas ou outros segmentos.
- As regiões adjacentes não devem possuir pixels em comum
- Os segmentos devem ser uniformes e homogêneos com relação a níveis de cinza e textura.
- As regiões devem ser simples e não apresentar buracos pequenos.
- Diferenças entre as regiões adjacentes devem ser significativas (níveis de cinza etc.).
- As bordas dos segmentos devem ser precisas.

Na prática todas as características mencionadas acima são utópicas porque superfícies homogêneas são geralmente cheias de furos e as bordas em geral são irregulares. Além das regiões adjacentes tenderem a se fundir e perder as bordas. Como regra geral a identificação de segmentos é específica e típica para cada aplicação. Em geral a separação da região de interesse é destacada buscando-se a descontinuidade e a similaridade nos diferentes tons da imagem. As descontinuidades são representadas pelas mudanças bruscas nos níveis de cinza como linhas e bordas. As similaridades baseiam-se nos limiares dos níveis de cinza, subdivisão da imagem em regiões homogêneas e crescimento de regiões.

3.4.2 Segmentação por limiarização

A limiarização de imagens é uma ferramenta de segmentação muito simples e computacionalmente rápida. É indicada para aplicações que exijam a identificação e extração

do objeto da imagem, cujo tempo computacional despendido seja um fator relevante.

As técnicas de segmentação estão divididas em dois conceitos distintos: similaridade (agrupamento de regiões homogêneas) ou descontinuidades (detecção de bordas). A limiarização é uma forma de agrupamento de regiões homogêneas (GONZALEZ; WOODS, 2000).

Alguns pesquisadores procuram classificar as técnicas de limiarização normalmente observando quais informações são usadas na escolha do limiar e o número de limiares utilizados. Weska (1978) classifica os métodos de limiarização em três grupos: global, local e dinâmico. O critério usado nesta definição é a dependência do operador de limiar. Se for determinado usando apenas as intensidades é classificado como global. Se usar as intensidades e alguma informação espacial da imagem é classificado como local. Se usar ambas as informações mais as coordenadas espaciais é classificado como dinâmico.

Sahoo(1988), por sua vez, divide as técnicas em dois grandes grupos: globais e locais. Os limiares globais são aqueles que dividem toda a imagem usando apenas um limiar. Os locais dividem a imagem em subimagens e para cada subimagem é atribuído um limiar. Esta técnica também é conhecida como limiar multiníveis. Esta classificação usa como critério a quantidade de limiares selecionados para a imagem.

Ainda em seu trabalho, Sahoo classifica os limiares globais e locais como: ponto-dependente ou região-dependente. Se o limiar é determinado apenas pelas intensidades de cada pixel, então o método é chamado ponto-dependente. Se o limiar é determinado por alguma propriedade local como, por exemplo, a distribuição de intensidades da vizinhança de cada pixel, então é região-dependente.

Neste trabalho é adotada a classificação proposta por Sahoo, a qual apresenta mais critérios de classificação sendo conseqüentemente mais precisa. Sem desprezar a simplicidade, também é a mais comumente usada pelos pesquisadores desta área.

Definidos os limiares globais e multiníveis, a aplicação do limiar global em uma imagem bidimensional $I(x, y)$ com níveis de intensidade $N_c = [0, 1, 2, \dots, 255]$ consiste em determinar um único limiar $T \in N_c$ que separe os pixels em duas classes distintas: objeto e fundo. Esta técnica, normalmente é aplicada ao histograma da imagem $h(N_c)$, o qual pode ser visto como uma descrição da distribuição das intensidades pela imagem.

Na figura 05 se vê um histograma com as intensidades variando entre 0 (preto) e 255 (branco). Vê-se também dois picos (histograma bimodal): o primeiro representa o objeto (escuro) e o segundo o fundo da imagem (claro). Eles estão separados por um vale, que

neste exemplo seria o local mais apropriado para estabelecer o limiar.

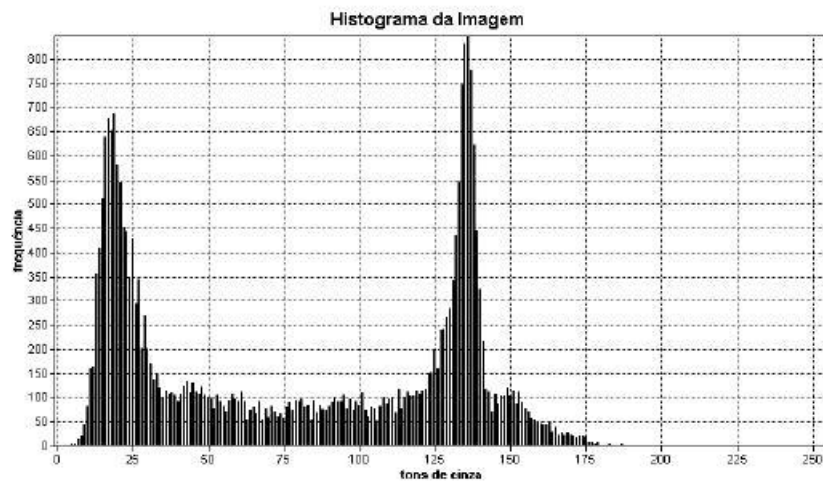


Figura 5: Histograma de uma imagem em níveis de cinza com limiar global.

Após determinar um limiar T , a imagem inteira fica dividida em duas regiões: o objeto, que recebe a cor preta (0) e o fundo da imagem, que recebe a cor branca (255), ou vice-versa. Desta forma, uma imagem limiarizada $L(x, y)$ pode ser definida pela equação (3.1), a qual é representada na forma de um algoritmo na seção 3.5.3.

$$L(x,y)=\begin{cases} 0 & \text{se } I(x,y) \leq T \\ 255 & \text{se } I(x,y) > T \end{cases} \quad (3.1)$$

3.4.3 Segmentação por textura

A segmentação por textura (GONZALEZ; WOODS, 2000; RUSS, 1998) deve permitir a colocação em evidência dos motivos da imagem considerando-se as propriedades de regularidade e de repetição. Os três métodos principais para segmentar uma imagem pela textura são: estatísticos, estruturais e espectrais. As técnicas estatísticas permitem caracterizar uma textura do tipo suave, irregular, granulada, etc. As técnicas estruturais tratam da disposição das primitivas numa imagem, por exemplo, textura baseada sobre linhas paralelas dispostas regularmente. As técnicas espectrais levam às propriedades do espectro de Fourier (FACON, 2001) e permitem detectar a periodicidade global, identificando altas energias.

3.4.4 Segmentação por regiões

Esse tipo de segmentação é mais utilizado quando existem várias regiões nas imagens, podendo ser classificadas em dois tipos.

Técnicas baseadas no crescimento de regiões (COSTA, 2000) realizam o particionamento da imagem em regiões conexas pelo agrupamento de pixels vizinhos com intensidades semelhantes. O processo é iniciado com um conjunto de pixels semente, aos quais vão sendo adicionados mais pixels que obedecem a um critério de aumento da região, isto dependendo de cada problema, podendo ser baseado no tom ou cor dos pixels, e considerar a "história" da região.

Pode ocorrer sobre-segmentação, que produz a fragmentação da imagem, e há uma maior dificuldade para obtenção das regiões quando as bordas estão borradas. Têm por objetivos produzir regiões coerentes (permitindo alguma flexibilidade de variação dentro da região) com o maior tamanho possível (número pequeno de regiões). Os dois principais métodos de segmentação por crescimento de regiões são o Watersheds (divisor de águas) (SERRA, 2000; BAXES, 1994), que baseia-se no princípio de "inundação de relevos topográficos" (RUSS, 1998) e o Algoritmo do Funcional de Mumford e Shah.

Na técnica de divisão e fusão de regiões (GONZALEZ; WOODS, 2000), figura 06, a imagem é recursivamente dividida em regiões até estas verificarem uma determinada condição. Posteriormente, as sub-regiões adjacentes, que verificarem uma condição, são reunidas numa só.

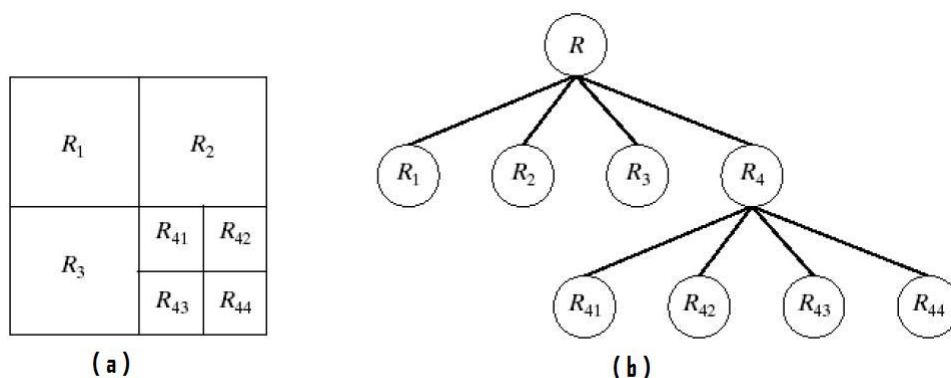


Figura 6: (a) Imagem particionada; (b) "quadtree" correspondente. (GONZALEZ; WOODS, 2000)

3.4.5 Segmentação por contornos ativos

Métodos baseados em contornos ativos fazem uso de informações locais sobre um contorno. É aplicado um contorno inicial sobre o objeto, sob a forma de curvas *spline*, que vai sendo iterativamente modificado, expandido e reduzido de acordo com uma função de energia, para tomar a forma do objeto que se deseja segmentar. Um dos modelos de contorno mais utilizados é o modelo Snake, que é um tipo particular de modelo deformável no qual uma curva inicial é deformada em direção à borda desejada pela ação de forças internas e externas a ele (RUSS, 1998).

É preciso ressaltar que nem sempre o método de segmentação de maior complexidade é o melhor. O sucesso de uma técnica segmentação dependerá da aplicação em questão.

3.4.6 Dificuldades inerentes

Existem algumas dificuldades inerentes ao processo de segmentação de imagens. Alguns fatores devem ser considerados antes de executar um processo de segmentação. Dentre eles podem destacar as seguintes:

1. Segmentação de forma autônoma (larga escala). Quando a segmentação de imagens envolver processos automáticos, existe uma grande necessidade em controlar o ambiente aonde se retira a imagem. Ambientes bem controlados (grandes contrastes) tendem a facilitar a interpretação de imagens. Ambientes externos apresentam certas dificuldades pois dependem do clima, iluminação etc.
2. Controle da luminosidade: Conforme a aplicação envolvida a existência de sombras tendem a dar uma falsa impressão acerca do tamanho real da região a ser segmentada.
3. Bordas das regiões são muitas vezes irregulares e imprecisas.
4. Precisão da técnica aplicada depende da qualidade da distinção entre os diferentes elementos da imagem.
5. Escolha da melhor estratégia e adequação à aplicação que se deseja.

3.5 Algoritmos Básicos

Os atributos apresentados na seção anterior dependem de algoritmos para serem extraídos das imagens. Nesta seção são apresentados resumidamente alguns dos algoritmos utilizados neste trabalho.

3.5.1 Histograma de Níveis de Intensidade

O histograma de uma imagem digital é a distribuição das intensidades, ou seja, o número de pixels para cada um dos níveis de intensidade presentes na imagem. A figura 07 mostra um exemplo de uma imagem e seu respectivo histograma.

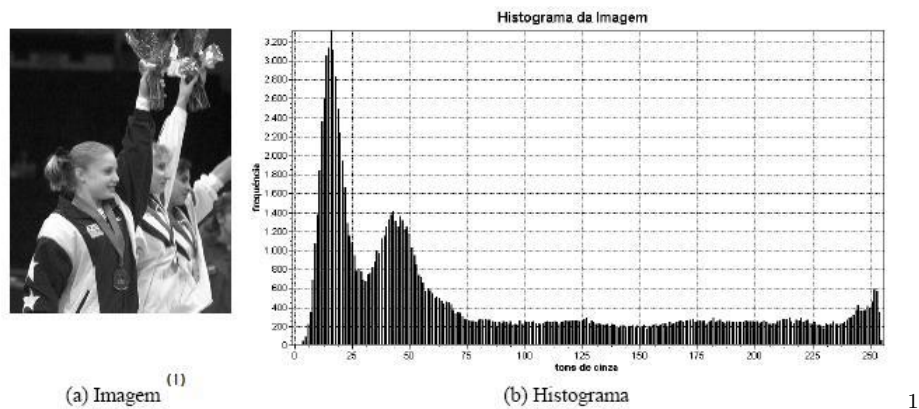


Figura 7: Imagem e seu Histograma.

Um algoritmo típico, em pseudo-código, para o cálculo do histograma de uma imagem digital é apresentado no Algoritmo 1.

Listagem 3.1: Algoritmo 1 - Histograma de uma imagem

Tipo histograma = vetor [0..255] de inteiros;

Função ghistograma (Img : imagem) : histograma;

Variáveis

5 x,y, cinza : inteiro;

h : histograma;

Início

Para y = 1 Até Img.altura Faça

Para x = 1 Até Img.largura Faça

10 cinza = Img(x,y);

h[cinza] = h[cinza]+1;

Fim-Para

```

    Fim-Para
  Retorne (h) ;
15 Fim .

```

3.5.2 Equalização do Histograma

A equalização é aplicada ao histograma da imagem com o objetivo de tornar a distribuição de intensidades de cinza mais uniforme ao longo do histograma. A consequência dessa uniformidade é o aumento de contraste. A equalização é obtida mapeando-se cada nível de intensidade presente na imagem original em um outro nível. Esse mapeamento é resultante da aplicação da função de distribuição acumulada (FDA) na frequência de ocorrência de um determinado nível. A FDA é obtida acumulando-se as frequências de ocorrência dos níveis de intensidade menores que um determinado nível i e dividindo-se este total acumulado pelo número total de pixels na imagem.

$$Ce_k = \sum_{i=0}^n \frac{f(i)}{N} \quad (3.2)$$

Sendo:

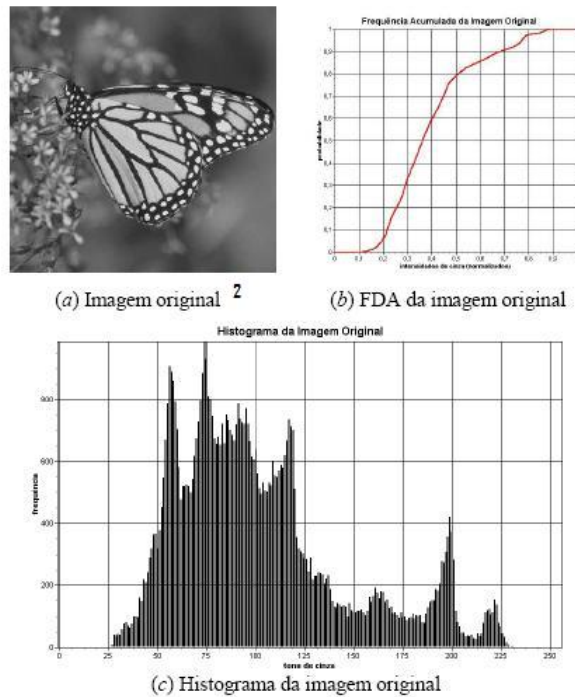
Ce_k = representa a intensidade k equalizada pela FDA.

$f(i)$ = representa o número de ocorrências da intensidade i na imagem original.

N = representa o número de pixels da imagem.

Na verdade, a equalização é uma tentativa de transformar a distribuição das intensidades dos pixels em uma distribuição uniforme. O Algoritmo 2 apresenta a implementação da equalização de uma imagem. Um exemplo da aplicação desse algoritmo pode ser visto na Figura 09, a qual apresenta a imagem equalizada (a), a FDA (b) e o histograma equalizado (c). Esse exemplo teve como entrada as informações apresentadas pela Figura 08.

A equalização é normalmente utilizada por métodos automáticos de limiarização. Nesses métodos a equalização impede a ocorrência de platôs ou mínimos/máximos locais nas funções que definem o critério de limiarização. O efeito da alteração no contraste de uma imagem equalizada pode ser observado na figura 08 e figura 09.



2

Figura 8: Imagem original, FDA e seu Histograma.

```

Tipo histograma = vetor [0..255] de inteiros;
Função equaliza (Img : imagem; h : histograma) : imagem;
variáveis ImgE : imagem;
        N, x,y, cinza : inteiro;
5         freqacumulada, he : histograma;

Início
    N = Img.altura * Img.largura;
    Para x = 0 até 255 faça
        acum = acum + h[x] / N;
10     freqacumulada[x] = acum;
    Fim-Para
    Para y = 1 até Img.altura faça
        Para x = 1 até Img.largura faça
            cinza = Img(x,y);
15     ImgE(x,y) = freqacumulada[cinza] * 255;
        Fim-Para
    Fim-Para
    Retorne(ImgE);
Fim.

```

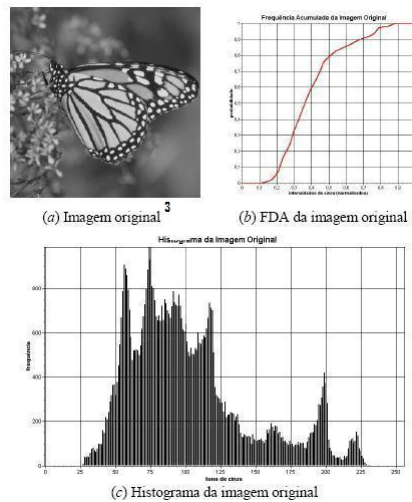


Figura 9: Imagem equalizada, FDA e seu Histograma.

3

3.5.3 Limiarização

A limiarização de imagens pode ser global ou multi-níveis. Nesta seção será apresentado o algoritmo de limiarização global. A limiarização multi-níveis pode ser vista como uma extensão do limiar global.

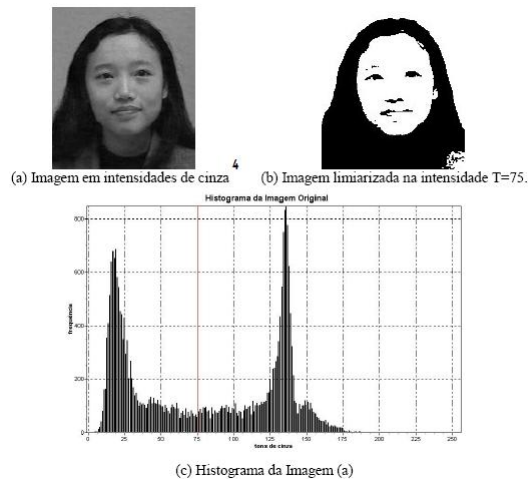


Figura 10: Limiarização de Imagens.

O resultado de uma limiarização global é a intensidade que classificará cada pixel como pertencente a um fundo ou a um objeto. Um algoritmo típico de limiarização global é implementado no Algoritmo 3. Tomando como entrada a imagem da figura 10.a e o limiar $T = 75$, a função implementada terá como saída a imagem exibida na figura 10.b.

Listagem 3.3: Algoritmo 3 - Limiarização de Imagens

```
Função Limiarização (Im : Imagem; T : inteiro): Imt : Imagem;  
variáveis  
  x,y : inteiro;  
Início  
5   Para y = 1 até Im.altura faça  
    Para x = 1 até Im.largura faça  
      Se Im(x,y) > T então  
        Imt(x,y) = 255 {branco}  
      Senão  
10      Imt(x,y) = 0 {preto}  
    Fim-Para  
  Fim-Para  
Fim.
```

4 *Desenvolvimento do software*

Este capítulo tem o objetivo de explicar a idéia dos algoritmos que foram desenvolvidos e mostrar como os mesmos foram implementados. Um desses algoritmos tem o nome de segmentação por componentes conexos, que tem o foco de capturar os pixels que compõe cada objeto na imagem e armazenar em uma estrutura de dados, para que possam ser acessados e manipulados. Um segundo algoritmo serve para acessar esses objetos e extrair informações destes, tais como, contagem, perímetro, raio e área. Esses algoritmos são baseados na técnica de segmentação por Limiarização.

4.1 Componentes Conexos

4.1.1 Passo 01

Aplica-se o limiar na imagem e a transforma em uma imagem binária, onde os objetos segmentados ficam destacados de cor preto e o fundo branco. Faz-se uma varredura na imagem no sentido horizontal com o objetivo de encontrar o primeiro pixel preto, que pertence ao primeiro objeto na imagem. No programa, o método encontraObjeto() se encarrega por esta tarefa. Abaixo, segue a ilustração da implementação deste método:

Listagem 4.1: Método que encontra o primeiro pixels de cada objeto

```
public void encontraObjeto() {
//          int v=1;

    String larguraS = JOptionPane.showInputDialog(null, "Entre
        com o a largura real da imagem em cm : ");
5   double largura = Double.parseDouble(larguraS);
    this.larguraSS = largura;

    String alturaS = JOptionPane.showInputDialog(null, "Entre
        com o a altura real da imagem em cm : ");
    double altura = Double.parseDouble(alturaS);
```

```

10
    for (int i=0;i<imagem1.getAltura();i++){
        for (int j=0;j<imagem1.getLargura();j++){
            if (gray[i][j]==0){
15
                gray[i][j]= 255;
                Objeto objeto = new Objeto();//
                    Esse objetos contera os objetos
                    segmentados
                pegaObjeto( new Pixel(j,i,0) ,
                    objeto);

20 //
                v++;
                this.extraiInformacoesObjeto(
                    objeto , largura , altura );
                objetos.add( objeto );
                //ArrayList oito = getVizinhoDeOito
                    (pixel);

25
            }

        }

30
    }
}

```

OBS: Nota-se que nesse método tem a opção de entrada da escala da imagem, que serve para posteriores calculo de áreas, raios e perímetro.

4.1.2 Passo 02

O próximo passo é capturar os 8-vizinhança do primeiro pixel do objeto, que possui intensidade de cor preta e guardar esses pixels vizinhos em uma estrutura de dados denominada ArrayList (SUN, 2007), para que possam ser reutilizados no passo 04. O código fonte referente a essa funcionalidade esta presente no método `getVizinhoDeOito()`, cuja listagem segue abaixo:

Listagem 4.2: Captura Vizinhos de 8 de intensidade preta

```

private ArrayList getVizinhoDeOito(Pixel pixel){

```



```
        ArrayList oito = new ArrayList();

        int l= pixel.getX() - 1;
5       int a= pixel.getY() - 1;

        for( int i=0 ; i<3; i++ ) {
            for( int j=0; j<3; j++ ) {
10
                if( !(i==1 && j==1) ) {

                    try {
15
                        if( gray[a+i][l+j] == 0 ) {
                            gray[a+i][l+j]=255;
                            oito.add(new Pixel(
                                l+j , a+i ,0));
                        }
                    } catch( Exception ex ) {}
                }
20
            }
        }

        return oito;
    }
}
```

4.1.3 Passo 03

Atribui o valor de intensidade branco(255) ao pixel em análise e adiciona esse pixel à uma outra estrutura de dados, do mesmo tipo da presente no passo 02, que formará um objeto segmentado.

4.1.4 Passo 04

A partir de agora, começa a recursividade, onde o primeiro pixel vizinho de 8 é lançado pra o método pegaObjeto, que é responsável por pegar todos os vizinhos pretos de todos os vizinhos pretos encontrados. A cada pixel que este método processa, executa-se o passo 02 e 03 novamente. Quando o método pegaObjeto() retorna NULL significa que não existe mais vizinho de intensidade preto, ou seja, já foram capturados todos os pixels que compõe o primeiro objeto.

Listagem 4.3: Método que captura o objeto

```

//— metodo especifico para achar os vizinho e acrescentar no vetor
public void pegaObjeto(Pixel pixel , Objeto objeto){

    objeto.addPixel( pixel );
5    ArrayList vizinhoPreto = getVizinhoDeOito(pixel); //Acha os
        vizinhos de Oito e adiciona a estrutura de dados

    for( int i=0; i<vizinhoPreto.size(); i++ ) {

        Pixel pixeltemp = (Pixel) vizinhoPreto.get( i );
10
        pegaObjeto( pixeltemp , objeto);
    }

15
}

```

Nota-se que à medida que o pixel é processado, é atribuído o valor de intensidade branca e este pixel. Isto permite que ao termino do execução do primeiro objeto, o mesmo algoritmo, a partir do passo 01 seja invocado, porque todos os pixels do primeiro objeto segmentado está com intensidade branca, e quando a varredura do passo 01 encontrar o primeiro pixel preto, este é um dos pixels que compõe o segundo objeto, e assim sucessivamente.

4.2 Extração de informações do Objeto Segmentado

Essa implementação é baseada na captura da borda através da técnica de comparações de vizinhos brancos, ou seja, se o pixel possuir um vizinho branco significa que este faz fronteira com o ambiente, logo é borda. Capturado a borda, o próximo passo é a contagem dos pixels que são bordas e então multiplica-se pela escala, e se obtém o perímetro do objeto. Abaixo segue a parte do código referente a captura dos pixels borda, calculo do perímetro e do raio, que é calculado através do perímetro, utilizando a função $r = \text{Perímetro}/2*\pi$.

Listagem 4.4: Extração do Perímetro e Raio

```

for( int w=0;w<size;w++){
    Pixel p = (Pixel) pixels.get(w);

```

```

        flag=false;
        for( int i=p.getX()-1; i<p.getX()+1; i++ )
            {
5                for( int j=p.getY()-1; j<p.getY()
                    +1; j++ ) {
                        if(i!=p.getX() && j!=p.getY()
                            ()){
                                try{
                                    if(imagem2[
                                        i][j]
                                        ]==255
                                        && flag
                                        ==false)
                                        flag=
                                        true;
                                        perim++;
                                }catch(Exception e)
                                    {}
10                                }
                            }
                        }
                    }
                }
15        double raio = 0;
        objeto.setPerimetro(new Double(perim).doubleValue()
            *escalaX);
        objeto.setRaio(objeto.getPerimetro()/(2*Math.PI));

        raio = perim/(2*Math.PI);
20        System.out.println("=====");
        System.out.println("Perim "+ perim);
        System.out.println("escalax "+ escalaX);
        System.out.println("Raio "+ raio);
        System.out.println("=====");

```

O calculo da área não depende da borda, mais sim da quantidade de pixels que contém o objeto, onde é efetuado a multiplicação dessa quantidade de pixels pela escala inserida pelo usuário.

5 Resultados

Essa seção ilustrará as funcionalidades que foram desenvolvidas e incorporadas ao *software* proposto neste trabalho. Nas ilustrações extraídas do *software*, está sendo utilizada uma imagem de teste extraída de um microscópio de força atômica, fornecida pelo grupo de polímeros "Bernhard Gross".

Para o desenvolvimento do *software*, foi utilizado um ambiente de programação bastante poderoso e livre, denominado Eclipse. Conforme, já dito anteriormente, optou-se pela linguagem Java para desenvolver o *software* de PDI, que utiliza o paradigma de orientação a objetos.

Apesar da existência de algumas bibliotecas em Java, que dão suporte a implementações voltadas para processamento de imagens, como por exemplo, API JAI (SUN, 2007), o presente trabalho não fará uso destas, e as implementações de técnicas de segmentação será feita manipulando as matrizes da imagem.

Este *software* segue uma estrutura padrão em *softwares* de processamento de imagens e consegue abrir qualquer imagem existente no computador em questão, que esteja no formatos JPEG,GIF e PNG. Segue abaixo(figura 11) a ilustração dessa funcionalidade.

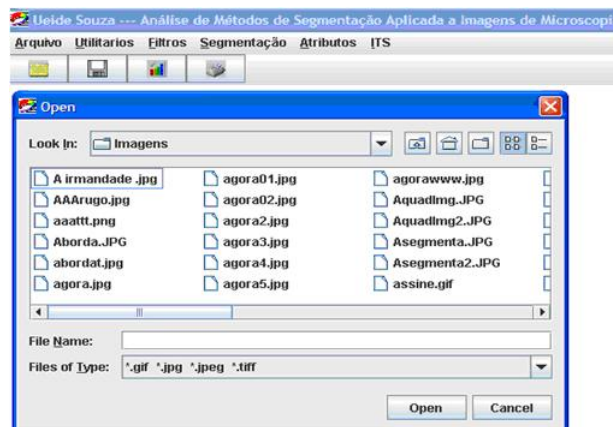


Figura 11: Console abrir imagem PNG, JPEG e GIF.

O protótipo inicial desse *software* foi desenvolvido utilizando uma interface MDI (*Multiple Document Interface*), onde se pode abrir diversas janelas de uma só vez. Essa interface pode ser observada na figura 12.

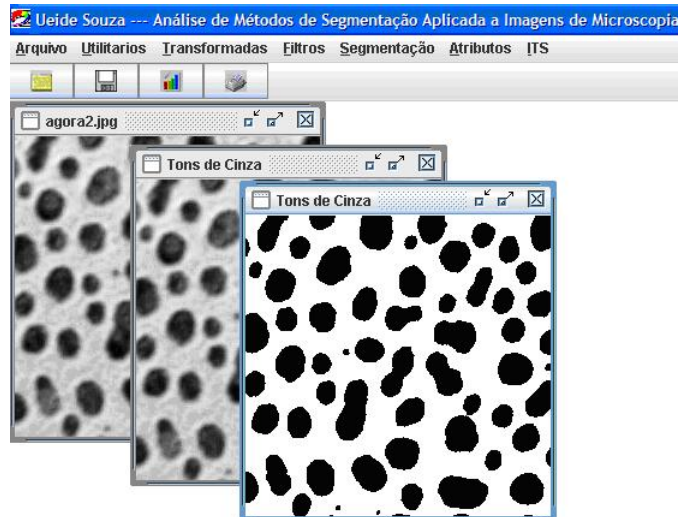


Figura 12: Inteface MDI.

5.1 Pré-processamento

Para o pré-processamento da imagem, afim de eliminar alguns ruídos, foi implementado os diversos métodos de filtragens e realce de imagens apresentados na seção 3.3 do capítulo anterior. O usuário tem a opção de escolher o que melhor realçar o objeto que será segmentado. Abaixo(figura 13), segue a representação funcional de uma dessas técnica, o filtro da mediana.



Figura 13: Filtro da mediana.

Este filtro por apresentar uma características que serve para eliminar ruídos na imagem com intensidades muito diferentes da vizinhança, mantendo assim os principais detalhes da imagem foi o que melhor se adaptou as imagens de microscopia estudadas e testadas nesse trabalho. O filtro da média, se mostrou bastante eficiente em cerca 15 por cento das imagens testadas, sendo que nos outros 85 por cento, as imagens estudadas pouco ou quase não se alteraram. A técnica de equalização de histograma não se mostrou muito eficiente, visto que as imagens testadas já apresentavam um contraste considerável. É de fundamental importância que esta etapa do PDI seja tratada com o maior cuidado possível, visto que o sucesso de alguns desses filtros acarretará menos problemas na segmentação da imagem. Por exemplo, uma pequena impureza na imagem de microscopia, seja esta ocasionada por lente suja ou outra qualquer situação, poderá ser considerada como um objeto da segmentação se esta imagem não for devidamente e corretamente filtrada.

5.2 Segmentação

Por ser computacionalmente mais rápido e por possuir algoritmos mais simples, decidiu-se primeiramente, implementar uma técnica de segmentação baseada em limiarização, do tipo limiarização global. Conforme ilustrado na figura 14, o usuário tem a opção de entrar com o limiar desejado de 0 a 255, e caracterizar de forma visual o que melhor representou os objetos.

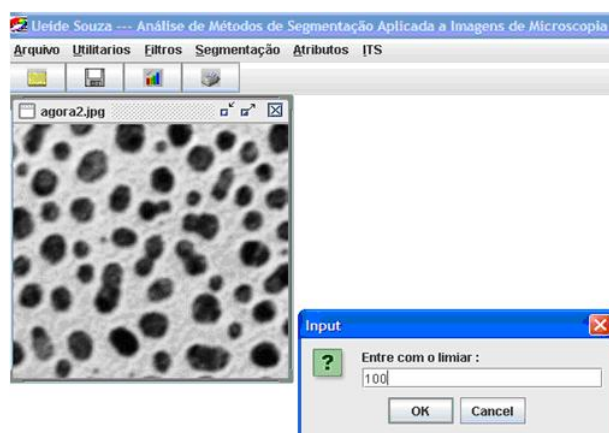


Figura 14: Entrada do limiar.

Nas figuras 15, 16, 17 e 18 serão apresentados os resultados da segmentação de uma imagem de microscopia com diversos limiares de valores iguais a 30, 85, 127 e 200, respectivamente. Nota-se que o resultado da segmentação por limiar é uma imagem binária,

cujo resultado em cor preta são os objetos, que se refere aos glóbulos dos polímeros que deseja-se separar, contá-los e extrair informações.



Figura 15: Resultado da segmentação. Limiar $T = 30$

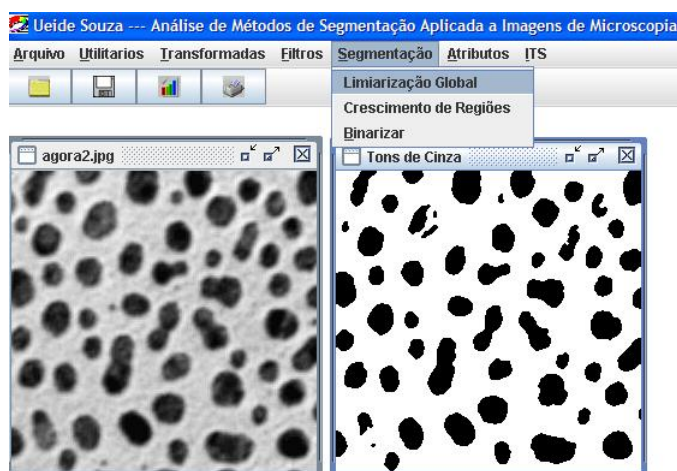
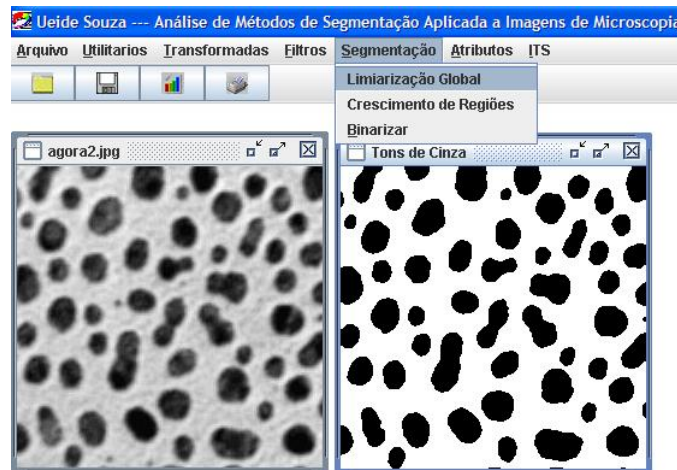
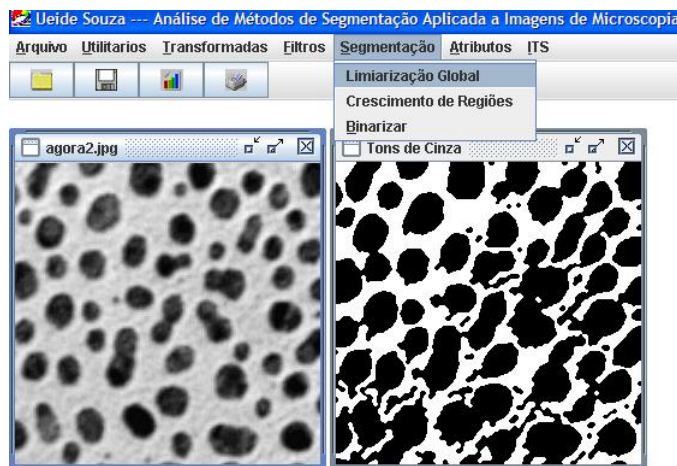


Figura 16: Resultado da segmentação. Limiar $T = 85$

O método de segmentação por limiarização global verifica se o usuário adicionou a imagem algum tipo de filtro. Se o usuário não adicionou nenhum, ele automaticamente executa o filtro da mediana.

Nota-se que na figura 15, com o limiar $T = 30$, o resultado da limiarização são os pixels que possuem os tons de cinza de maior intensidade. Esse tipo de análise é interessante, visto que, a depender da aplicação o que pode caracterizar um objeto pode ser a intensidade do ton de cinza presente nele. No caso da imagens de microscopia estudadas e testadas nesse trabalho, o limiar igual a 30 não é capaz de caracterizar um segmento, ou seja, esse limiar não garante o objetivo esperado da segmentação.

Figura 17: Resultado da segmentação. Limiar $T = 127$ Figura 18: Resultado da segmentação. Limiar $T = 200$

Na figura 16, aplicou-se um limiar $t = 85$, que é exponencialmente melhor do que o limiar $T = 30$, porque já consegue caracterizar 80 por cento dos objetos de uma imagem de microscopia. Esse limiar ainda não se aproxima do ideal visto que pode haver uma falsa impressão de um objeto, ou seja, partes de um mesmo objeto pode ser confundido como vários objetos. Em aplicações que se conhece a média dos tamanhos de cada objeto, esse limiar pode não ser tão ruim.

Um limiar que se aproximou do ideal, para a imagem da figura 17, foi o limiar $T = 127$. Observou-se que os objetos segmentados ficaram bem definidos em forma e tamanho do que era esperado visualmente. Esse limiar comprova que o simples algoritmo de limiarização global se mostrou eficiente para segmentação de imagens de microscopia.

Limiars acima de $T = 200$, se mostram inviáveis, devido a confusão entre os objetos de fato e a superfície onde ele se encontra, conforme é ilustrado na figura 18.

Observa-se que apesar da análise está sendo feita visualmente e com a imagem de teste, nota-se que o comportamento dos limiars em outras imagens da mesma natureza foram praticamente uniformes, com erros de 10 unidades nos limiars, tanto para cima quanto para baixo.

5.3 Extração de Características

Para capturar cada objeto presente na imagem binária segmentada e posteriormente calcular informações de que permitem caracterizar os objetos, tais como, contagem, perímetro, área e raio; utilizou-se, respectivamente, o algoritmo de componentes conexos e de extração de informação do objeto. As informações de cada objeto ou segmento não foram incorporadas à interface gráfica do *software*. No entanto, podem ser visualizadas de duas maneiras distintas: em um arquivo denominado InformaçõesImagem.txt, situado no workspace do projeto; ou ainda, no próprio console do ambiente de desenvolvimento Eclipse, conforme ilustra a figura 19.

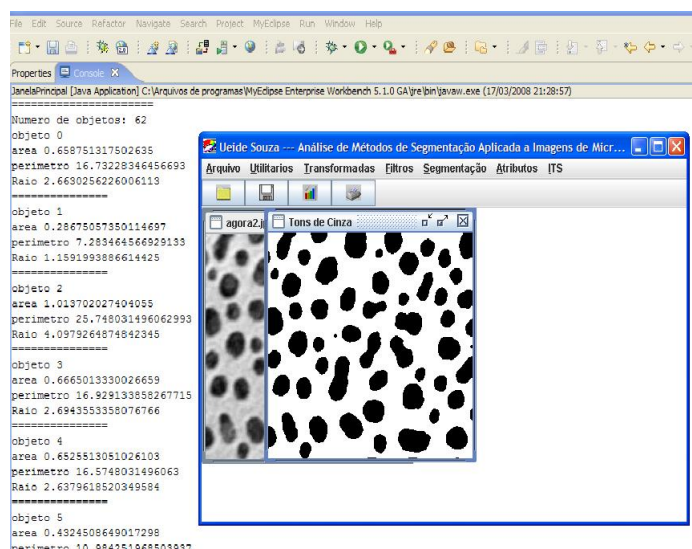


Figura 19: Informações de contagem perímetro raio e area para um $T = 127$ para uma escala de 10cm de largura e altura

Esta funcionalidade, apesar de esta em perfeito funcionamento, é acionada quando o usuário do *software* clica no Menu Segmentação - Limiarização Global, ou seja, o *software* automaticamente faz a limiarização e mostra as informações dos objetos. Na figura 13, o objeto 0 equivale ao primeiro objeto.

Para uma maior visualização, interpretação das informações da imagem e análise da influência do solvente, optou-se também por implementar uma gráfico da polidispersividade que tem a função de exibir o histograma (frequência) dos valores dos raios dos objetos segmentado, conforme mostra a figura abaixo(figura 20).

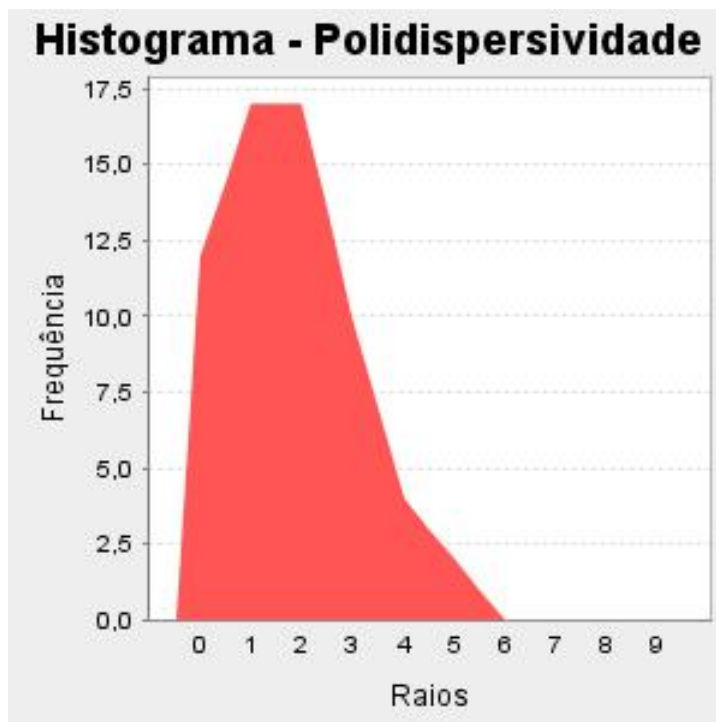


Figura 20: Gráfico da polidispersividade para um $T = 127$ e escala de 10cm de largura e altura

O cálculo do perímetro, da área e do raio se dá em função da escala da imagem, que é informada pelo usuário do sistema. Para cálculo da área de um objeto, multiplica-se as escalas das posições x e y pela quantidade de pixel de cada segmento. Nota-se que se as escalas não forem iguais, a área calculada é uma aproximação baseada na média. Para o perímetro, é efetuado a multiplicação dos pixels existentes na borda de cada segmento pela escala; observa-se que o artifício utilizado em imagens com as escalas das posições x e y diferentes, se dá retirando a média entre as escalas. Já o raio, é calculado em função do perímetro: $r = \text{Perímetro}/2*\pi$.

6 Conclusão e Comentários Finais

6.1 Conclusões

A escolha do método a ser utilizado está intrinsecamente ligada ao problema em foco e aos recursos computacionais disponíveis para a realização da operação. Portanto, a análise prévia do problema é de importância fundamental para que os resultados da segmentação sejam satisfatórios.

Nos resultados obtidos, utilizando o método de segmentação por Limiarização foram considerados bastante satisfatórios e condizentes, levando em consideração a simplicidade e o pequeno grau de complexidade dessa técnica. Vale ressaltar que a implementação e aplicação de métodos mais sofisticados e complexos do que a Limiarização na imagem de microscopia, não necessariamente dará melhores resultados. O sucesso de um método em um problema não significa que terá o mesmo êxito aplicado em outro problema.

Além da segmentação, o *software* também já é capaz de contar os polímeros existente na imagem, medir a área e o perímetro de cada um desses glóbulos, e apresentar o histograma das frequências dos raios. Essas informações são de fundamental importância, visto que o usuário será capaz de caracterizar a sua imagem de microscopia.

6.2 Contribuições da monografia e desenvolvimentos futuros

Uma das maiores contribuições está no *software* que reúne um conjunto considerável de funcionalidades e diversos algoritmos desenvolvidos. Este *software* será de grande utilidade para a continuidade da pesquisa e também para outros grupos interessados em manter colaborações conosco, tanto nesta Instituição como em outras. Há diversas perspectivas e desenvolvimentos futuros. Dentre estas podemos citar:

1. Pretende-se desenvolver outros métodos de filtragem de imagem, baseados no domínio da frequência, tais como transformada de Fourier, afim de obter melhoras mais consideráveis na imagem.
2. Métodos de segmentação baseados em crescimento de regiões, visto que já existe uma boa parte desenvolvida, inclusive a identificação das sementes.
3. Desenvolvimento para web, visto a linguagem JAVA da todo o suporte.
4. Testar os limiares em uma conjunto grande de imagens, com a finalidade de determinar um limiar ótimo para esse conjunto de imagem. Um outra boa opção seria a aplicação de limiares automáticos, onde no final da análise da imagem, seria dado ao usuário um relatório contendo a análise da imagem com diversos limiares, e também a sugestão do melhor limiar para uma dada imagem.
5. Para facilitar a navegabilidade e dinâmica na identificação do melhor limiar para uma dada imagem de microscopia, poderia ser implementado um slidebar.
6. O sistema poderá extrair mais informações da imagem original, tais como escala e tonalidade. Isso tornaria o software mais dinâmico e independente, evitando a necessidade do usuário entrar com a escala.

Referências

- BAXES, G. A. *Digital Image Processing: Principles and Applications*. [S.l.]: John Wiley Sons, 1994.
- CASTLEMAN, K. R. *Digital Image Processing*. [S.l.]: Prentice-Hall, Englewood Cliffs, 1996.
- COSTA, L. da F. *Shape Analysis and Classification*. [S.l.]: CRC Press, 2000.
- FACON, J. *Processamento e Análise de Imagens*. [S.l.]: Pontifícia Universidade Católica do Paraná, 2001.
- GERBER. *Atomic Force Microscope*. [S.l.]: Phys. Rev. Lett, 1996.
- GONZALEZ, R. C.; WOODS. *Processamento de Imagens Digitais*. [S.l.]: Livraria Editora Ltda, 2000.
- MEYER, F. *Topographic distance and watershed lines*. *Signal Processing*. [S.l.]: Springer, 1994.
- PRATT, W. K. *Digital Image Processing : PIKS Inside*. [S.l.]: PIKS Inside, 2001.
- RODRIGUES, C. A.; COSTA, L. F. *Statistical characterization of morphological features of layer-by-layer polymer films using image analysis*. [S.l.], 2003.
- RUSS, J. C. *The Image Processing Handbook*. [S.l.]: CRC Press, 1998.
- SAHOO, P. *Computer Vision, Graphics, and Image Processing*. [S.l.: s.n.], 1988.
- SERRA, J. *Image Analysis and Mathematical Morphology*. Florianópolis – SC: Academic Press, 2000.
- SOILLE, P. *Morphological Image Analysis - Principles and Applications*. [S.l.]: Springer, 1999.
- SUN. 10 2007. Disponível em: <http://www.sun.com>.
- WESKA, J. *Computer Graphics and Image Processing*. [S.l.: s.n.], 1978.