

2

Fundamentos Teóricos

Neste capítulo são abordados alguns fundamentos teóricos importantes para o entendimento deste trabalho. Na seção 2.1 é apresentado o conceito básico de segmentação de imagens e as principais técnicas existentes. A seção 2.2 apresenta alguns conceitos sobre características de objetos em imagens digitais e na seção 2.3 são apresentados conceitos sobre avaliação da qualidade da segmentação. Na seção 2.4 são apresentadas métricas de discrepância para avaliação da qualidade da segmentação, consideradas neste trabalho. A seção 2.5 apresenta o conceito de correlação de ordem. Por fim, a seção 2.6 apresenta uma breve revisão teórica dos métodos de otimização utilizados nos experimentos realizados.

2.1. Técnicas de Segmentação

A segmentação é o processo de divisão de uma imagem em suas regiões ou objetos constituintes (Gonzalez07). Representa uma etapa crítica no processamento de imagens digitais e a sua qualidade é de vital importância para o sucesso das etapas subsequentes, como extração de atributos e classificação.

Os algoritmos de segmentação são geralmente baseados em uma de duas propriedades básicas: descontinuidades e similaridades (Gonzalez07). Na primeira categoria, a ideia é dividir a imagem levando em consideração variações abruptas na intensidade dos pixels. Na segunda, a abordagem é particionar a imagem em regiões que são similares de acordo com algum critério.

O conceito de segmentação por descontinuidades é apresentado na próxima seção e depois, os principais métodos de segmentação por similaridades são apresentados nas seções subsequentes.

2.1.1. Segmentação por Descontinuidades

A segmentação por descontinuidades identifica objetos na imagem com base em variações abruptas na intensidade dos pixels. Este método se aplica basicamente a imagens em escala de cinza e visa detectar três tipos básicos de descontinuidades: pontos, linhas e bordas. Enquanto a detecção de pontos e linhas se baseia na varredura da imagem por máscaras, a detecção de bordas leva em consideração as derivadas de primeira e segunda ordem aplicadas sobre a intensidade dos pixels (Gonzalez07).

Se as bordas são detectáveis em todo o contorno do objeto, essa abordagem se mostra bem interessante para uma diversidade de casos (Yu04, Yu-gian06, Zhang10). Este método, entretanto, possui algumas desvantagens. Já que esta abordagem busca por “estruturas primitivas” na imagem, se faz necessária uma etapa posterior para analisar essas estruturas e identificar corretamente os limites dos objetos ou regiões. Além disso, este método se mostra bastante sensível a condições de iluminação e ruído, o que o torna mais aplicável em imagens tomadas em condições controladas.

Quanto a esses dois aspectos, o método de segmentação por regiões (Seção 2.1.3) se mostra uma alternativa bem interessante, já que produz objetos com bordas bem definidas e se mostra mais robusto a imagens tomadas em condições diversas de iluminação, ruído, etc.

2.1.2. Limiarização

A limiarização é um dos métodos de segmentação mais simples e eficientes computacionalmente, tendo sido usado em um vasto conjunto de aplicações, como por exemplo: identificação de digitais (Baig08), reconhecimento de íris (Du09), inspeção de materiais (Sezgin00, Sezgin01), reconhecimento de texto em vídeos (Du02), etc. Este método atua sobre os pixels da imagem, classificando-os de acordo com um limiar fornecido. Primeiro, os pixels da imagem são agrupados em um histograma de acordo com a sua intensidade, depois os pixels cujos valores estão acima ou abaixo do limiar são atribuídos a classes diferentes, geralmente

uma representando o objeto (ou objetos) de interesse e a outra o fundo (parte da imagem sobre a qual não se tem interesse).

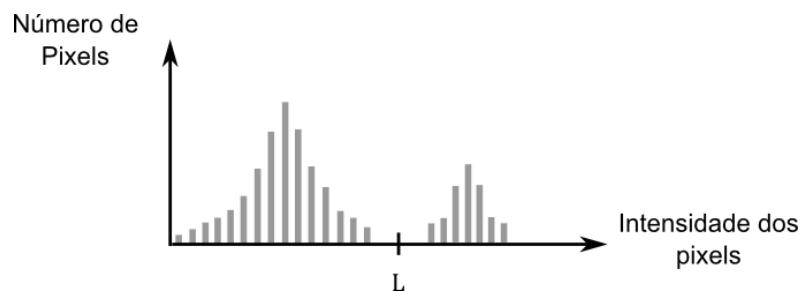


Figura 2.1: Limiar L definido sobre o histograma de uma imagem.

A limiarização é chamada global quando a definição do limiar L se baseia apenas na intensidade dos pixels. Se L depende da intensidade dos pixels e de características locais, como a intensidade dos pixels vizinhos por exemplo, a limiarização é chamada local. Se além destes dois aspectos L depender também da posição dos pixels na imagem, a limiarização é chamada dinâmica ou adaptativa (Gonzalez07).

Pela definição do método pode-se perceber que ele se aplica principalmente a imagens em escala de cinza, embora haja uma extensão do método para imagens coloridas, chamada *limiarização multiespectral*. Esta última baseia-se na clusterização dos pixels em um espaço multidimensional. A principal dificuldade é que a procura dos clusters se torna uma tarefa cada vez mais complexa à medida que o número de bandas espectrais aumenta.

Existem também limiarizações com mais de um limiar, chamadas *limiarizações multinível*, que permitem a busca por mais classes de interesse, identificadas pelos espaços entre os limiares no histograma. Em geral, os problemas que requerem mais de um limiar são mais bem resolvidos usando métodos por crescimento de regiões (Gonzalez07), discutidos na próxima seção.

Diversos fatores como ruído, iluminação ambiente, contraste inadequado e tamanho do objeto não equiparável ao tamanho da imagem dificultam a aplicação do método (Sezgin04).

2.1.3. Segmentação por Regiões

A segmentação por regiões consiste em dividir a imagem em regiões homogêneas de acordo com um determinado critério de homogeneidade (Gonzalez07).

Formalmente, seja R a representação da extensão total de uma imagem, a segmentação é um processo que divide a imagem em n sub-regiões, $R_1, R_2, \dots, R_n$, de forma que:

- (a) $\bigcup_{i=1}^n R_i = R$.
- (b) R_i é uma região conectada, $i=1,2,\dots,n$.
- (c) $R_i \cap R_j = \emptyset$ para todo i e j , $i \neq j$.
- (d) $P(R_i) = \text{VERDADEIRO}$ para $i=1,2,\dots,n$.
- (e) $P(R_i \cup R_j) = \text{FALSO}$ para $i \neq j$.

O termo $P(R_i)$ representa um predicado lógico que corresponde a algum critério de homogeneidade. A condição (a) indica que a imagem é a soma de todas as sub-regiões, ou seja, cada pixel deve pertencer a uma região. A condição (b) estabelece que os pixels que pertencem a uma região devem ser contíguos. A condição (c) indica que as regiões precisam ser disjuntas. A condição (d) determina que todas as regiões devem atender ao critério de homogeneidade. Por fim, a condição (e) estabelece que a união de regiões não deve atender ao predicado P .

É possível perceber algumas potencialidades interessantes para um grande número de aplicações, em comparação aos métodos de segmentação apresentados nas seções anteriores. Além de não limitar o processo à análise das características de pixels isolados, a segmentação baseada em regiões permite a consideração de um conjunto maior de critérios como forma e textura, que caracterizam os objetos de imagem e não os seus elementos constituintes (pixels). Desta maneira, os métodos de segmentação por regiões se mostram uma alternativa poderosa e genérica, sendo aplicáveis a uma grande variedade de problemas (ambiente não controlado, condições diversas de iluminação, etc.).

A seguir, duas das técnicas de segmentação por regiões mais usadas serão apresentadas.

2.1.3.1.

Divisor de Águas (Watershed)

Neste método, a imagem é vista como uma superfície topográfica, em que as intensidades dos pixels correspondem à altitude dos pontos. Desta maneira, as regiões formadas por pixels com menor intensidade são representadas por vales e as regiões cujos pixels possuem maior intensidade por elevações.

Um processo então simula a inundação dessas superfícies a partir de “furos” realizados nas regiões “mais baixas”, chamadas mínimos regionais. À medida que a água vai inundando essas áreas, elas passam a formar bacias. Quando os limites de duas bacias vizinhas se encontram, uma linha de contenção é criada entre elas (Pedrini08).

O processo continua até que toda a superfície esteja inundada, restando apenas as linhas de contenção entre as bacias. Estas linhas então definem as regiões identificadas na imagem.

A segmentação quando realizada desta forma geralmente leva à supersegmentação, principalmente devido a ruídos e ao grande número de mínimos potenciais (Gonzalez07). Uma alternativa é limitar o número de bacias que podem se formar através do uso de marcadores (Gonzalez07).

Uma das principais vantagens do método é a sua eficiência computacional, sendo por isso usado em alguns casos para gerar regiões iniciais para um segmentador por crescimento de regiões (descrito na próxima seção). Pela natureza do algoritmo, ele não leva em consideração as cores da imagem, sendo aplicado a imagens em escala de cinza.

2.1.3.2.

Crescimento de Regiões

Este método consiste na agregação de pixels a regiões maiores em um processo iterativo. O passo inicial consiste na determinação de um conjunto de pixels denominados sementes, a partir das sementes as regiões crescem agregando

pixels ou regiões que atendam a algum critério de similaridade. O algoritmo para quando mais nenhuma agregação puder ser realizada.

A seleção das sementes pode ser feita com base em algum conhecimento a priori sobre a imagem a ser segmentada ou sobre as classes de objeto de interesse. Se este conhecimento não existir, uma possibilidade é definir todos os pixels da imagem como sementes.

A escolha do critério de similaridade está intimamente relacionada com o problema a ser solucionado, com o tipo de imagem analisada e com a classe de objetos-alvo que se deseja encontrar. Um exemplo bem simples é o critério de homogeneidade intra-regiões adotado pelo segmentador SPRING (Binis96). Neste segmentador, as regiões são agrupadas de acordo com a distância euclidiana calculada sobre um vetor de n dimensões, onde n representa o número de bandas espectrais.

Muitos outros critérios podem ser definidos, e na verdade existe uma gama razoável de diferentes implementações disponíveis. Tantas, que alguns pesquisadores tem se dedicado a avaliar a qualidade desses métodos (Meinel04). Um dos métodos de melhor avaliação nestes estudos surgiu há uma década (Baatz00). Este método, conhecido como Segmentação Multiresolução, além de outras inovações, propôs a introdução de características de forma dos objetos (regiões) em seu critério de homogeneidade intra-regiões. O método é hoje provavelmente o mais usado em aplicações de sensoriamento remoto e interpretação de imagens médicas. A próxima seção descreve o método em detalhes.

2.1.4. Segmentação Multiresolução

O método de Segmentação Multiresolução é um algoritmo de crescimento de regiões proposto por Baatz e Schäpe (Baatz00).

A segmentação começa com cada pixel da imagem como uma semente, representando um objeto. A cada iteração, cada objeto se funde a um objeto vizinho. Este vizinho é aquele para o qual o objeto resultante da fusão represente o menor acréscimo de heterogeneidade em relação à soma das medidas de heterogeneidade internas dos dois objetos candidato à fusão. A fusão, porém, só

acontece se o acréscimo de heterogeneidade for menor do que certo limiar. A medida de heterogeneidade possui uma componente espectral e uma morfológica.

A componente espectral é definida a partir dos valores dos pixels que compõem o objeto, sendo proporcional ao desvio-padrão desses valores, ponderados por pesos arbitrários definidos para cada banda espectral da imagem.

A componente morfológica é definida pelo desvio relativo da forma do objeto em relação a uma forma compacta e uma forma suave, ponderadas por pesos. A compacidade é definida como a razão entre o comprimento de borda do objeto e a raiz quadrada de sua área (seu tamanho em pixels), e a suavidade como a razão entre o comprimento de borda do objeto e o comprimento de borda do retângulo envolvente.

Para permitir um crescimento distribuído dos objetos, cada objeto é selecionado para fusão apenas uma vez a cada iteração e essa seleção é realizada de uma forma espacialmente distribuída.

Como mencionado anteriormente, o acréscimo de heterogeneidade resultante da fusão entre dois objetos é o que direciona o algoritmo através das iterações, e por esse motivo será chamado desse ponto em diante de fator de fusão. Quando um objeto é selecionado, o fator de fusão é calculado para cada um dos seus vizinhos. O vizinho para o qual esse fator é mínimo é selecionado para a fusão. Mas a fusão ocorre apenas se o fator de fusão for menor do que certo limiar, definido como o quadrado do parâmetro de escala p_e^2 , um dos parâmetros do algoritmo. A segmentação termina quando mais nenhuma fusão puder ser realizada. O parâmetro de escala é um termo abstrato que determina o máximo aumento de heterogeneidade permitido, resultante da fusão entre dois objetos, e que influencia indiretamente no tamanho médio dos objetos finais, produzidos pela segmentação (Brodský06).

Como mostra a Equação 2-1, o fator de fusão f contém uma componente espectral h_{cor} e uma componente morfológica h_{forma} . A importância relativa de cada componente é dada pelo peso da forma w_{forma} , outro parâmetro do algoritmo.

$$f = (1 - w_{forma}) \cdot h_{cor} + w_{forma} \cdot h_{forma} \quad 2-1$$

A Equação 2-2 mostra a formulação da componente espectral do fator de fusão. $Obj1$ é o objeto selecionado para fusão, $Obj2$ é um objeto vizinho e $Obj3$ é o objeto resultante da fusão de $Obj1$ com $Obj2$. Na equação, c e w_c são, respectivamente, o índice e o peso da banda espectral; σ_c é o desvio-padrão dos valores dos pixels que compõem cada objeto para a banda c , e n é o número de pixels de cada objeto.

$$h_{cor} = \sum_c w_c \left(n_{Obj3} \cdot \sigma_c^{Obj3} - (n_{Obj1} \cdot \sigma_c^{Obj1} + n_{Obj2} \cdot \sigma_c^{Obj2}) \right) \quad 2-2$$

A componente morfológica, por sua vez, possui outras duas componentes, uma componente de compacidade h_{compac} e outra de suavidade h_{suav} . A importância relativa de cada componente é dada pelo peso w_{compac} .

$$h_{forma} = (w_{compac}) \cdot h_{compac} + (1 - w_{compac}) \cdot h_{suav} \quad 2-3$$

As Equações 2-4 e 2-5 mostram como as componentes h_{compac} e h_{suav} são calculadas. Nas equações, l é o comprimento de borda do objeto e l_{re} o comprimento de borda do retângulo envolvente.

$$h_{compac} = n_{Obj3} \cdot \frac{l_{Obj3}}{\sqrt{n_{Obj3}}} - \left(n_{Obj1} \cdot \frac{l_{Obj1}}{\sqrt{n_{Obj1}}} + n_{Obj2} \cdot \frac{l_{Obj2}}{\sqrt{n_{Obj2}}} \right) \quad 2-4$$

$$h_{suav} = n_{Obj3} \cdot \frac{l_{Obj3}}{l_{re}} - \left(n_{Obj1} \cdot \frac{l_{Obj1}}{l_{re}} + n_{Obj2} \cdot \frac{l_{Obj2}}{l_{re}} \right) \quad 2-5$$

Pode-se observar que tanto a componente h_{cor} quanto h_{forma} são medidas do acréscimo da heterogeneidade resultante da fusão de dois objetos – heterogeneidades associadas à cor e à forma, respectivamente. Essas componentes são ponderadas pela área em pixels dos objetos, de forma que cada objeto/segmento tenha a mesma importância relativa, contribuindo de forma equivalente para a heterogeneidade média global da segmentação resultante (Baatz00).

O algoritmo possui flexibilidades com relação às chamadas heurísticas de decisão (Baatz00) que podem variar em cada implementação. Por exemplo, a forma como um objeto seleciona um vizinho para fusão pode variar. A forma apresentada neste texto, onde o vizinho escolhido é aquele que apresenta o menor fator de fusão, foi apresentada por (Baatz00) como a heurística do Melhor Vizinho. Entre outras propostas, cabe destacar a heurística do Melhor Vizinho Mútuo, onde não basta apenas que o vizinho apresente o menor fator de fusão para que a fusão aconteça, mas é necessário que essa condição seja mútua. Apesar de tornar o algoritmo um pouco mais caro em termos computacionais, esta abordagem se torna interessante ao passo que tende a gerar uma segmentação de melhor qualidade.

2.2. Características dos Objetos de Imagem

A medição e análise das características dos objetos presentes em imagens digitais são fundamentais para a interpretação, visual ou automática. Tais características podem tanto ser usadas durante o processo de segmentação quanto na etapa de classificação. Este trabalho, entretanto, se limita a analisar essas características no contexto da segmentação de imagens.

As medições que podem ser realizadas sobre características dos objetos de imagem podem ser agrupadas em quatro tipos: brilho, localização, tamanho e forma (Russ98). Para cada tipo, existe uma variedade de medidas e uma variedade de maneiras de realizar as medições. A maioria dos sistemas de análise de imagens oferece pelo menos algumas medidas diferentes de cada tipo. Frequentemente, usuários encontram dificuldades para lidar com essas medidas, no sentido de selecionar as medidas mais apropriadas, e os respectivos limiares, para resolver um problema específico.

Neste trabalho, o foco está na classe de medidas (atributos) de forma. Nas próximas seções serão apresentados alguns conceitos importantes para o cálculo desses atributos. Em alguns momentos, o termo fundo será empregado para representar a parte da imagem que não faz parte dos objetos de interesse.

2.2.1. Comprimento de borda

O comprimento de borda é definido como o número de arestas ou lados dos pixels do objeto que tocam o fundo da imagem. Veremos na próxima seção que o comprimento de borda é utilizado em algumas formas de cálculo do perímetro. Entretanto, neste trabalho preferiu-se definir este conceito em separado para evitar confusão. Desta forma, alguns atributos serão definidos em termos de perímetro e outros em termos de comprimento de borda.

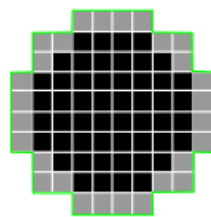
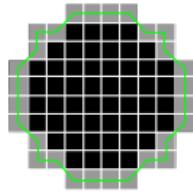


Figura 2.2: Comprimento de borda de um círculo mostrado em verde (pixels de borda em cinza).

2.2.2. Perímetro

Existem várias maneiras de se calcular o perímetro de um objeto de imagem (Russ98). Alguns sistemas estimam o perímetro contando o número de pixels de borda (método 1), o que tende a gerar um valor subestimado, dependendo da orientação do objeto. Outros sistemas calculam o perímetro da mesma forma como que o comprimento de borda, somando os lados dos pixels do objeto que tocam o fundo da imagem (método 2). Este método, por outro lado, tende a superestimar o perímetro e também varia com a orientação do objeto.

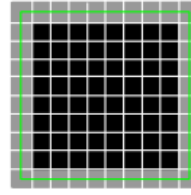
Existe ainda outro método (método 3) que tenta compensar os caminhamentos diagonais que existem na borda do objeto. Caminha-se pelos pixels de borda contabilizando 1 para um passo vertical ou horizontal e $\sqrt{2}$ para um passo diagonal.



Método 1 – 28

Método 2 – 40

Método 3 – 31,31



Método 1 – 36

Método 2 – 40

Método 3 – 36

Figura 2.3: Cálculo do perímetro para um círculo e um quadrado de 10x10 (pixels de borda em cinza).

Neste trabalho considera-se a vizinhança 4, ou seja, pixels que tocam o fundo pela sua quina não são contados como de borda. Destacado em verde, se encontra o perímetro como é calculado pelo método 3.

Apesar de ser mais preciso, o método 3 é o mais caro computacionalmente, o que em termos de um algoritmo de segmentação o torna praticamente inviável. Os outros dois métodos tem fácil implementação, e podem parecer semelhantes em uma primeira análise, porém ao investigar as suas definições podemos perceber que o método 2 tem uma limitação importante. Ele não consegue distinguir o perímetro de um quadrado de lado d do perímetro de um círculo de mesmo diâmetro (Figura 2.3). Isso se deve ao efeito da *pixelização* ao se representar um círculo em uma imagem digital.

Por estas razões, neste trabalho optou-se por calcular o perímetro como disposto no método 1, ou seja, contando o número de pixels de borda. O que torna o cálculo do atributo mais rápido e também permite diferenciar o perímetro de um círculo e de um quadrado de mesmas dimensões.

2.2.3. Ajuste de Elipse

Existem dois principais métodos para se calcular as dimensões e a orientação de um objeto. O primeiro é definido utilizando-se as informações da borda do objeto e o segundo utilizando as informações de todos os pixels do objeto. Segundo (Russ98), utilizar as informações de todos os pixels do objeto

tende a produzir resultados mais precisos, uma vez que se diminui a influência da presença ou ausência de pixels isolados na periferia do objeto, onde acidentes na aquisição da imagem ou ruído podem causar alterações significativas.

Este segundo método é conhecido como Ajuste de Elipse. Ao ajustar uma elipse aos pixels de um objeto é possível ter uma boa ideia das suas dimensões e orientação. O ajuste é baseado nas estatísticas da distribuição espacial dos pixels que pertencem ao objeto. Uma maneira simples de trabalhar com essas estatísticas é através da matriz de covariância:

$$S = \begin{bmatrix} Var(X) & Cov(X,Y) \\ Cov(X,Y) & Var(Y) \end{bmatrix} \quad 2-6$$

Sendo:

- a) X , vetor com todas as coordenadas x dos pixels do objeto;
- b) Y , vetor com todas as coordenadas y dos pixels do objeto;
- c) n , o número de pixels do objeto (tamanho);
- d) $\bar{x}$, média das coordenadas x dos pixels do objeto. Também representa a coordenada x do centroide;
- e) $\bar{y}$, média das coordenadas y dos pixels do objeto. Também representa a coordenada y do centroide.

$$Var(X) = \frac{1}{n} \sum_x (\bar{x} - x)^2 \quad 2-7$$

$$Var(Y) = \frac{1}{n} \sum_y (\bar{y} - y)^2 \quad 2-8$$

$$Cov(X,Y) = \frac{1}{n} \sum_x \sum_y (\bar{x} - x)(\bar{y} - y) \quad 2-9$$

Os autovalores da matriz de covariância correspondem aos dois eixos da elipse ajustada: $2a$ (maior) e $2b$ (menor). Sendo a e b os semieixos.

$$2a = \sqrt{8 \cdot (Var(X) + Var(Y) + \sqrt{(Var(X) - Var(Y))^2 + 4 \cdot Cov(X, Y)^2})} \quad 2-10$$

$$2b = \sqrt{8 \cdot (Var(X) + Var(Y) - \sqrt{(Var(X) - Var(Y))^2 + 4 \cdot Cov(X, Y)^2})} \quad 2-11$$

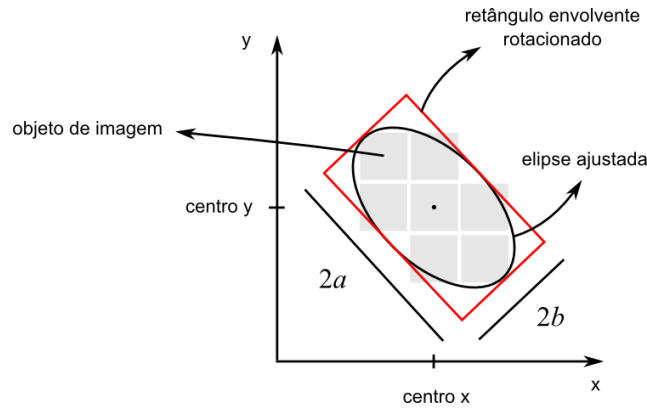


Figura 2.4: Elipse ajustada a um objeto de imagem.

A partir desses eixos é possível obter as dimensões do retângulo envolvente mínimo *rem*, utilizadas no cálculo de outros atributos:

a) Comprimento de borda:

$$l_{rem} = 4a + 4b \quad 2-12$$

b) Área:

$$n_{rem} = 2a * 2b \quad 2-13$$

No texto, o termo retângulo envolvente *re* irá se referir ao retângulo que envolve o objeto paralelo aos eixos da imagem, ficando o termo retângulo envolvente mínimo para se referir ao retângulo envolvente calculado através da elipse ajustada, que é rotacionado, possuindo a mesma orientação da elipse.

2.3. Qualidade da Segmentação

Vários algoritmos de segmentação foram propostos nas últimas décadas (Haralick85, Baatz00, Comaniciu02, Tilton89, Meinel04, Neubert06, Neubert08).

Como algoritmos diferentes não são igualmente adequados para uma aplicação específica, a avaliação da qualidade da segmentação se torna um assunto importante e indispensável em estudos sobre segmentação (Zhang96).

Apesar de muitos esforços terem sido aplicados na pesquisa e desenvolvimento de novos algoritmos de segmentação, poucos foram dirigidos à avaliação dos mesmos. Dentre as abordagens para a avaliação de segmentação destacam-se: (Zhang96, Zhang01), que classificou os algoritmos de avaliação da segmentação em três grandes categorias: analítico, qualidade empírica (EG) e discrepância empírica (ED); (Neubert04, Neubert06, Neubert08) que realizou uma análise qualitativa e visual da qualidade de programas de segmentação voltados para a área de sensoriamento remoto e (Feitosa06, Fredrich08, Janssen95, Gerig01) que propuseram métricas para a avaliação quantitativa da segmentação. Uma adaptação do esquema proposto por (Zhang96) é apresentada na Figura 2.5.

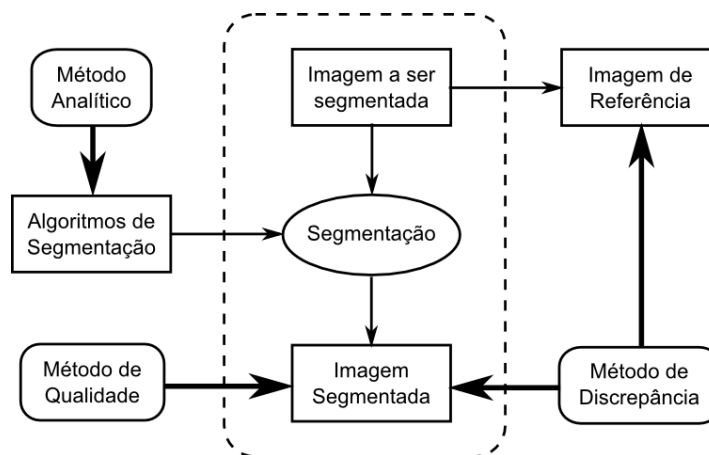


Figura 2.5: Esquema de segmentação e avaliação.

A seguir são apresentados os conceitos das três grandes categorias propostos por (Zhang96). A definição do método de avaliação utilizado neste trabalho se baseou nos resultados apresentados naquele trabalho.

2.3.1.1. Métodos Analíticos

Estes métodos não atuam sobre a saída do algoritmo de segmentação (imagem segmentada), mas sobre o algoritmo de segmentação em si (Figura 2.5). Os métodos analíticos avaliam os algoritmos de segmentação segundo os seus

princípios, requerimentos, utilidades, complexidade, etc. (Zhang96) argumenta que o estudo analítico é incapaz de obter todas as propriedades do algoritmo de segmentação, principalmente pela falta de uma teoria geral para a segmentação de imagens. Sendo assim, estes métodos são utilizados, na prática, em conjunto com outros métodos, e raramente sozinhos.

2.3.1.2. Métodos de Qualidade Empírica (EG)

Os métodos de qualidade empírica julgam a qualidade da segmentação indiretamente através de métricas de qualidade estabelecidas segundo a intuição humana.

Várias medidas de qualidade foram propostas para cobrir diferentes aspectos de uma segmentação “boa” ou “ideal”, sendo classificadas em três grupos como mostra a Tabela 2.1.

Medidas
Uniformidade intra-regiões
Contraste inter-regiões
Forma da região

Tabela 2.1: Medidas usadas em avaliações empíricas.

Os estudos realizados por (Meinel04, Neubert06, Neubert08), citados na introdução da seção, se enquadram nesse tipo de método, embora uma análise visual também seja realizada.

2.3.1.3. Métodos de Discrepância Empírica (ED)

Os métodos de discrepância empírica quantificam a diferença entre a segmentação produzida pelo algoritmo de segmentação e uma segmentação de referência (também chamada de “padrão ouro” ou “verdade de campo”). Em outras palavras, estes métodos procuram determinar o quão distante a segmentação automática está da referência. Estes métodos são classificados em cinco grupos como mostra a Tabela 2.2.

Os estudos de (Janssen95, Gerig01, Feitosa06, Fredrich08), se enquadram nesse tipo de método, uma vez que as métricas propostas levam em consideração segmentações de referência.

Medidas
Número de pixels subsegmentados
Posição de pixels subsegmentados
Número de objetos na imagem
Valor de características dos objetos segmentados
Quantidades diversas

Tabela 2.2: Medidas usadas em avaliações empíricas.

(Zhang96) conclui que a falta de uma teoria geral da segmentação torna os métodos empíricos mais adequados do que os métodos analíticos para a avaliação da qualidade dos algoritmos de segmentação. E dentre os métodos empíricos, os métodos de discrepância se mostraram melhores para avaliar objetivamente a qualidade da segmentação. Embora os métodos de discrepância sejam mais complexos que os de qualidade, considerando-se o fato de que uma segmentação de referência precisa ser produzida, os métodos de discrepância se mostraram mais efetivos. Neste trabalho, optou-se por utilizar um método de discrepância empírica para a avaliação da qualidade das segmentações.

2.4. Métricas de Discrepância

As métricas de discrepância são utilizadas para avaliar a diferença entre a segmentação de referência e a saída do algoritmo de segmentação. De modo geral, elas são definidas de forma que um valor igual a zero represente a similaridade máxima ou ideal entre referência e segmentação.

Antes de uma descrição detalhada de cada métrica, cabe definir alguns conceitos importantes, alguns deles representados na Figura 2.6.

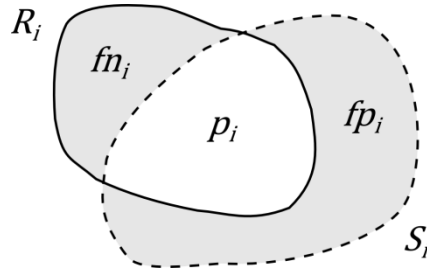


Figura 2.6: Conceitos utilizados nas métricas de discrepância.

Vamos assumir que existam N segmentos de referência delineados por um especialista. Seja R_i ($i=1,2,\dots,N$) o i -ésimo segmento de referência. Seja S_i o segmento produzido pelo algoritmo de segmentação com maior intercessão com R_i . Seja também CS_i (não mostrado na Figura 2.6) o conjunto dos segmentos resultantes do algoritmo de segmentação que tenham pelo menos 50% de intercessão com R_i . Se nenhum segmento atender a essa condição CS_i será um conjunto vazio. Vamos também definir:

- a) fn_i como o número de pixels de R_i que não pertencem a S_i , então chamados falso-negativos;
- b) fp_i como o número de pixels de S_i que não pertencem a R_i , então chamados falso-positivos;
- c) p_i como o número de pixels que pertencem à intercessão de S_i e R_i , então chamados positivos;
- d) B_i como a soma de todos os pixels de borda de S_i e R_i ;
- e) b_i como o número de pixels de borda em CS_i que intersectam R_i ;
- f) NS como o número de CS_i não-vazios;
- g) $\#()$ como um operador de área (em pixels).

As próximas subseções descrevem as métricas de discrepância.

2.4.1. Reference Bounded Segments Booster (RBSB)

Proposta por Feitosa et al. (Feitosa06) essa métrica corresponde à divisão da área da não-interseção pela área da referência. Tem-se $F_{RBSB}=0$ para um ajuste perfeito entre referência e segmentação e $F_{RBSB}>0$, caso contrário.

$$F_{RBSB} = \frac{1}{N} \sum_{i=1}^N \frac{(fn_i + fp_i)}{\#(R_i)} \quad 2-14$$

2.4.2. Larger Segments Booster (LSB)

Esta métrica foi proposta por Fredrich e Feitosa (Fredrich08). Os termos fp_i e fn_i na equação 2-15 favorecem soluções com uma sobreposição justa com os segmentos de referência, o que favorece soluções constituídas de inúmeros pequenos segmentos (no limite do pixel). O termo b_i contrabalança este efeito concedendo melhores avaliações às segmentações com menos segmentos, e consequentemente, maiores. Tem-se $F_{LSB}=0$ para um ajuste perfeito.

$$\begin{aligned} &\text{Se } NS < N, \\ &F_{LSB} = \frac{1}{N} \left[NS + \sum_{CS_i \neq \emptyset} \frac{fp_i + fn_i + b_i}{\#(R_i)} \right] \quad 2-15 \\ &\text{Senão, } F_{LSB} = \infty \end{aligned}$$

2.4.3. Janssen (JAN)

Esta métrica introduzida por Janssen e Molenaar (Janssen95) corresponde à raiz quadrada da razão entre o quadrado da área de interseção e o produto das áreas da segmentação e da referência. $F_{JAN}=1$ corresponde a um ajuste perfeito entre a referência e a segmentação e $F_{JAN}=0$ corresponde ao pior caso, quando não há interseção.

$$F_{JAN} = \frac{1}{N} \sum_{i=1}^N \sqrt{\frac{p_i^2}{\#(R_i) \cdot \#(S_i)}} \quad 2-16$$

2.4.4. Sobreposição Espacial (SE)

Esta métrica (Gerig01) corresponde à divisão da área da interseção entre segmentação e referência pela área da união. $F_{SE}=1$ corresponde a um ajuste perfeito e $F_{SE}=0$ a quando não há interseção.

$$F_{SE} = \frac{1}{N} \sum_{i=1}^N \frac{p_i}{(fn_i + fp_i + p_i)} \quad 2-17$$

2.4.5. Diferença de Área Relativa Absoluta (DARA)

Esta métrica (Gerig01) corresponde ao valor absoluto da razão entre as áreas da segmentação e da referência subtraída de 1. $F_{DARA}=0$ corresponde a um ajuste perfeito e $F_{DARA}>0$, caso contrário. É importante notar que $F_{DARA}=0$ pode também ser obtido para um ajuste não perfeito, contanto que a referência e a segmentação possuam a mesma área.

$$F_{DARA} = \frac{1}{N} \sum_{i=1}^N \left| \frac{\#(S_i)}{\#(R_i)} - 1 \right| \cdot 100 \quad 2-18$$

2.4.6. Distância de Perímetro Absoluta Simétrica Média (DPASM)

Para esta métrica (Gerig01) é definido um operador $d(R_i, S_i)$ que calcula a soma das distâncias entre cada pixel de borda do segmento de maior intercessão e o pixel de borda mais próximo da respectiva referência e vice-versa. A métrica é dada pela média dessas distâncias. Tem-se $F_{DPASM}=0$ para um ajuste perfeito.

$$F_{DPASM} = \frac{1}{N} \sum_{i=1}^N \frac{d(R_i, S_i) + d(S_i, R_i)}{B_i} \quad 2-19$$

2.4.7.**Distância de Perímetro Simétrica RMS (DPSRMS)**

Esta métrica é semelhante à anterior (Gerig01), como mostra a equação 2-20. Novamente, $F_{DPSRMS}=0$ corresponde a um ajuste perfeito.

$$F_{DPSRMS} = \frac{1}{N} \sum_{i=1}^N \sqrt{\frac{d(R_i, S_i)^2 + d(S_i, R_i)^2}{B_i}} \quad 2-20$$

2.4.8.**Distância de Perímetro Absoluta Simétrica Máxima (DPASMx)**

Esta métrica é semelhante às duas anteriores (Gerig01). Porém, em vez da média, a distância máxima é calculada.

$$F_{DPASMx} = \frac{1}{N} \sum_{i=1}^N \max(d(R_i, S_i) + d(S_i, R_i)) \quad 2-21$$

2.5.**Correlação de Ordem**

A correlação de ordem mede a tendência do valor de duas variáveis de aumentar ou diminuir juntas. O método de correlação mais usado, conhecido como correlação de Pearson (Neave88), mede o quão bem a relação entre duas variáveis pode ser descrita por uma função linear. A correlação de ordem é menos restritiva e se refere à evidência de que duas variáveis estão relacionadas por uma função monotônica. No contexto deste estudo, a correlação de ordem expressa a tendência de duas métricas de qualidade em concordarem com relação à melhor segmentação entre duas possíveis.

Os dois métodos de correlação de ordem mais utilizados são os coeficientes de Spearman e Kendall (Neave88). Nos experimentos relatados neste trabalho (Seção 4.1), os dois coeficientes foram computados e os resultados foram estritamente consistentes. Dessa maneira, para simplificar, apenas o coeficiente de Kendall será referido no texto que segue.

O coeficiente de correlação de ordem de Kendall pode ser facilmente calculado a partir de um processo gráfico. Vamos supor que $n=5$ segmentações chamadas de a, b, c, d e e foram ordenadas de acordo com um valor de qualidade dado por dois observadores distintos, como mostrado na Figura 2.7.

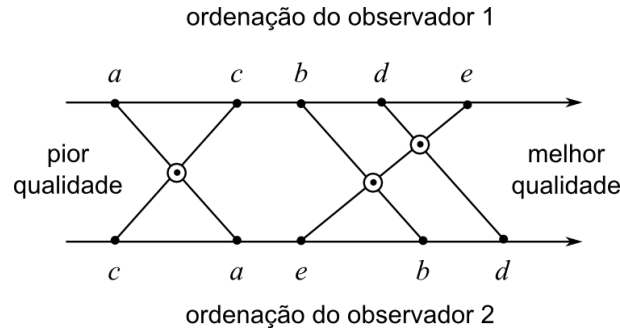


Figura 2.7: Exemplo do cálculo gráfico da correlação de Kendall.

É possível notar que ambos os observadores concordam sobre b ser melhor que a e b ser melhor que d , porém discordam em relação a a e c e em relação a b e e . Se nós definirmos N_C e N_D como sendo o número de pares concordantes e discordantes respectivamente, o coeficiente de correlação de ordem de Kendall (τ) pode ser definido como:

$$\tau = \frac{2(N_C - N_D)}{n(n-1)} \quad 2-22$$

Onde n é o comprimento da sequência e $n(n-1)/2$ é o número total de pares dentro da sequência. Como $N_C + N_D = n(n-1)/2$ pode ser facilmente demonstrado que:

$$\tau = 1 - \frac{4N_D}{n(n-1)} \quad 2-23$$

Esta é uma formulação particularmente conveniente dado que N_D é simplesmente o número de cruzamentos no desenho da Figura 2.7.

2.6. Técnicas de Otimização

A seguir serão descritas brevemente as técnicas de otimização utilizadas neste trabalho.

2.6.1. Algoritmos Genéticos (AG)

Algoritmos genéticos são uma ferramenta de busca automática para solução de problemas de otimização. São algoritmos estocásticos que se baseiam na teoria da evolução das espécies proposta por Charles Darwin (Darwin03).

O princípio básico dessa teoria é que características individuais são passadas de pais para filhos através das gerações e que os indivíduos mais adaptados ao ambiente tem maiores chances de sobreviver e se reproduzir, perpetuando assim, as suas características.

Um indivíduo, nesse contexto, representa uma solução potencial para o problema e os genes desse indivíduo representam as suas características relevantes para o problema. Neste estudo, os genes representam parâmetros de métodos de segmentação (Seção 4.1).

Uma população representa o conjunto de indivíduos de uma geração. Cada indivíduo é avaliado de acordo com a sua capacidade de resolver o problema proposto, sendo essa avaliação feita através de uma função de aptidão (Michalewicz96). Neste trabalho, tais funções representam métricas de disparidade, que avaliam a discrepância ou diferença entre a segmentação de referência e a segmentação produzida pelos parâmetros contidos nos genes do indivíduo (Seção 4.1).

AG propõem um processo evolucionário para encontrar soluções que minimizem (ou maximizem) a função de aptidão. Essa busca é realizada iterativamente através das gerações, onde os indivíduos menos aptos são descartados (seleção natural) e os mais aptos geram novos indivíduos, através de operadores genéticos que imitam os conceitos de mutação e da passagem de características entre gerações através da reprodução sexuada (crossover).

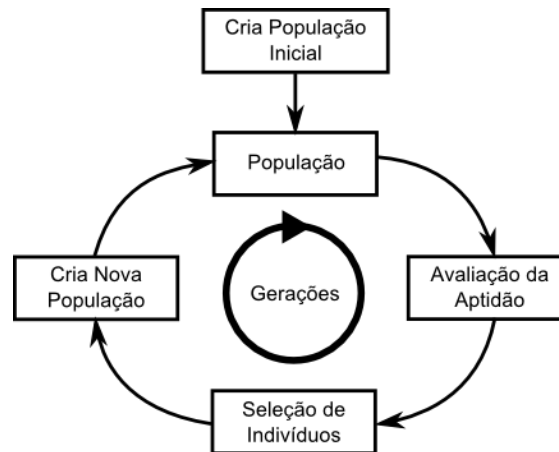


Figura 2.8: Estrutura básica dos AG.

Ao final do processo evolucionário, o indivíduo com melhor avaliação representa a solução final do problema de otimização.

2.6.2. Generalized Pattern Search (GPS)

Embora métodos de otimização mais tradicionais, que usam informações sobre o gradiente na busca por uma solução ótima, possam ser bastante eficientes, eles não são aplicáveis em diversos casos, por exemplo: quando a função objetivo é não-diferenciável; quando a informação de gradiente não está disponível ou não é confiável; ou quando os valores da função são imprecisos (Dennis94).

Métodos de otimização por busca direta têm se mostrado uma alternativa poderosa e segura para tratar esse tipo de problema já que não requerem o cálculo de nenhum gradiente ou informação de curvatura, sendo especialmente aplicados a problemas onde a função objetivo é não-diferenciável, estocástica ou mesmo descontínua (Song09).

Métodos GPS (*Generalized Pattern Search*) pertencem a um subconjunto dos métodos de busca direta chamados métodos *pattern search* e foram propostos por Torczon (Torczon97) como uma generalização de métodos anteriores como o algoritmo *pattern search* original de Hook e Jeeves (Hook61) e o algoritmo de busca multidirecional de Dennis e Torczon (Dennis91).

Algoritmos GPS calculam uma sequência de soluções que se aproximam iterativamente do ótimo global. A cada iteração o algoritmo seleciona um

conjunto de soluções (*mesh*) ao redor da solução atual à procura de uma solução para a qual o valor da função objetivo seja menor do que o da solução corrente. Se tal solução é encontrada, a seleção (*poll*) é considerada bem-sucedida e a solução encontrada se torna a solução atual na próxima iteração. Se o algoritmo não encontra uma solução com melhor avaliação, a seleção é considerada malsucedida e a solução corrente permanece a mesma para a próxima iteração.

Para selecionar as soluções a serem avaliadas, o algoritmo soma um conjunto de vetores à solução atual, multiplicados por um fator de escala, chamado tamanho do *mesh*. Em uma seleção bem-sucedida, o tamanho do *mesh* é aumentado e em uma seleção malsucedida, ele é reduzido. O algoritmo para quando o tamanho do *mesh* for reduzido a um valor/limiar mínimo.

2.6.2.1. Algoritmo GPS

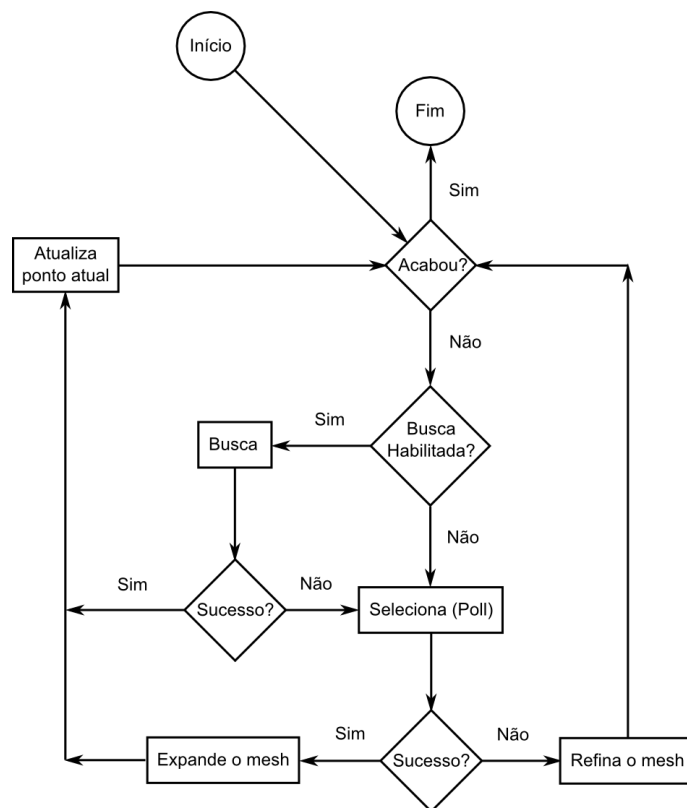


Figura 2.9: Fluxograma do algoritmo GPS (Mathworks11, adaptado).

Ao iniciar, o algoritmo calcula o valor da função objetivo para um ponto inicial $x_0 \in \mathcal{R}^n$ fornecido, que passa a ser o ponto corrente x_k , sendo $(k=0,1,...)$ o

número de iterações. Após isso, ele verifica se a busca está habilitada. Se estiver, o algoritmo realiza uma busca procurando por pontos que melhorem o valor da função objetivo. Essa busca é uma etapa de otimização em si mesma, e qualquer outro método pode ser utilizado, como outro padrão de busca direta ou um algoritmo genético. Se nessa busca é encontrado um ponto melhor, então esse ponto passa a ser o ponto corrente, nenhuma seleção é realizada e o algoritmo vai para a próxima iteração. Se um ponto melhor não é encontrado, então a seleção é realizada.

Na seleção, o algoritmo calcula um conjunto de pontos (*mesh*) ao redor do ponto corrente. Esses pontos são calculados a partir de um padrão, isto é, um conjunto de vetores $c_j \in \mathcal{R}^n$ geralmente representados como as colunas de uma matriz C . O número de linhas dessa matriz corresponde ao número de variáveis do problema n e o número de colunas p depende do tipo de padrão utilizado.

$$C = \begin{bmatrix} c_{11} & \cdots & c_{1p} \\ \vdots & \ddots & \vdots \\ c_{n1} & \cdots & c_{np} \end{bmatrix}$$

Existem vários padrões de vetores possíveis. Para o padrão Base Positiva $2n$, sendo o número de variáveis $n=2$, a matriz consistiria de 2 linhas e 4 vetores.

$$C_{Base\ Positiva\ 2n} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}$$

Cada coluna c_j é então multiplicada por um fator de escala, chamado tamanho do *mesh* $\Delta_k \in \mathcal{R}$, e depois o resultado é somado ao ponto corrente, gerando os pontos para a seleção:

$$\beta_j = x_k + (\Delta_k c_j) \quad 2-24$$

Novamente, para o exemplo do padrão Base Positiva $2n$, o *mesh* ao redor do ponto x_k ficaria assim:

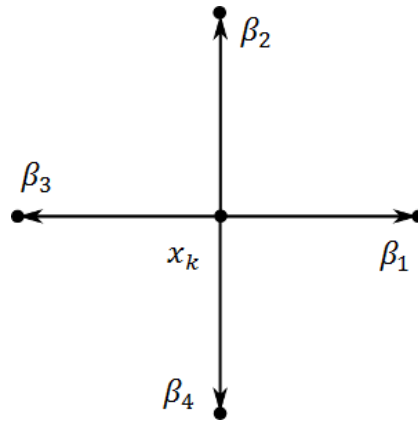


Figura 2.10: *mesh* ao redor do ponto x_k .

Para cada ponto é então verificada a variação na função objetivo $\rho_i = f(x_k) - f(\beta_i)$. Se $\rho_i > 0$, a seleção é considerada bem-sucedida, o ponto corrente é atualizado $x_{k+1} = \beta_i$ e o tamanho do *mesh* é expandido $\Delta_{k+1} = \lambda \Delta_k$, sendo $\lambda \geq 1$. Se $\rho_i \leq 0$, então a seleção é considerada malsucedida, o ponto corrente permanece o mesmo $x_{k+1} = x_k$ e o tamanho do *mesh* é contraído $\Delta_{k+1} = \theta \Delta_k$, sendo $0 < \theta < 1$.

Ao final da iteração, o algoritmo verifica se o tamanho do *mesh* é menor do que o limiar definido $\Delta_k < \varepsilon$. Neste caso, o algoritmo é terminado, e o ponto atual passa a ser a solução aproximada do problema. Caso contrário, uma nova iteração é iniciada.