

# ESTUDO E IMPLEMENTAÇÃO DE TÉCNICAS DE SEGMENTAÇÃO DE IMAGENS

Samuel Clayton Maciel Neves (bolsista CNPq - PIBIC, [scmn@ufpa.br](mailto:scmn@ufpa.br))  
Prof. Dr. Evaldo Gonçalves Pelaes (DEEC/CT-UFPA, [pelaes@ufpa.br](mailto:pelaes@ufpa.br))  
Universidade Federal do Pará – Departamento de Engenharia Elétrica e de Computação –  
Laboratório de Processamento de Sinais (UFPA – DEEC – LaPS).  
Campus Universitário do Guamá – CEP: 66075-000 – Belém – Pará Tel.: +55 91 211 1674

**Resumo:** Devido o grande avanço tecnológico dos últimos tempos, verifica-se a utilização cada vez maior do computador nas tarefas de extração de informações de imagens. Uma dessas tarefas é a segmentação. Segmentar, em processamento digital de imagem, significa subdividir uma imagem em suas partes ou objetos constituintes. Esse trabalho mostra as diversas técnicas de segmentação de imagens em níveis de cinza e suas aplicações. As técnicas são baseadas em descontinuidade: detecção de pontos isolados, detecção de linhas e detecção de bordas, e similaridade: limiares (*Thresholding*), crescimento de regiões (*Region Growing*), junção e separação (*Split & Merge*) e aglomeração (*Clustering*).

**Abstract:** Due the great technological progress, the use every larger time of the computer is verified in the tasks of extraction of information of images. One of those tasks is the segmentation. To segment, in digital image processing, it means to subdivide an image in its parts or constituent objects. This paper shows the several techniques of image segmentation in gray levels and its applications. The techniques are set in discontinuity: detection of isolated points, detection of lines and detection of borders, and similarity: Thresholding, Region Growing, Split & Merge and Clustering.

## 1. INTRODUÇÃO

Devido o grande avanço tecnológico dos últimos tempos, verifica-se a utilização cada vez maior do computador nas tarefas de extração de informações de imagens para auxiliar na solução de diversos problemas práticos, tais como descoberta automática de fissuras em materiais, contagem de glóbulos brancos em amostras de sangue, cálculo do volume sangüíneo nas câmaras do coração, contagem de microorganismos em amostras orgânicas, e outras.

Para que seja possível a utilização de imagens na solução desses problemas, elas devem ser adquiridas e digitalizadas (amostradas e quantizadas). Após a digitalização, podem ser feitas diversas transformações na imagem até conseguir-se a solução do problema para o qual ela está sendo usada. Essas transformações podem ser agrupadas nos seguintes conjuntos: Filtragem, análise, compressão e síntese de imagens.

O presente trabalho concentra-se no estudo do grupo de análise de imagens, o qual lida com a extração de medidas, dados, ou informação de uma dada imagem por métodos automáticos ou semi-automáticos.

O primeiro passo na análise de imagem é, geralmente, segmentar a imagem. Segmentar, em processamento digital de imagem, significa subdividir uma imagem em suas partes ou objetos constituintes. Em geral a segmentação é uma das mais difíceis tarefas em processamento de imagem. Esse passo no processamento determina o eventual sucesso ou falha de toda análise [4].

Nesse trabalho são utilizadas imagens em níveis de cinza, ou seja, a imagem é representada por um conjunto finito de números inteiros que vai de 0 a 255, onde cada valor é um nível de cinza, que representará cada ponto da imagem (pixel). Os algoritmos de segmentação de imagens em níveis de cinza, geralmente são baseados em uma das duas propriedades dos valores de níveis de cinza: descontinuidade e similaridade. Na primeira categoria, a abordagem é particionar uma imagem baseando-se nas mudanças abruptas no nível de cinza. As principais áreas de interesse dentro dessa categoria são a detecção de pontos isolados, detecção de linhas e detecção de bordas numa imagem. As principais abordagens da segunda categoria são baseadas em limiares (*Thresholding*), crescimento de regiões (*Region Growing*), junção e separação (*Split & Merge*) e aglomeração (*Clustering*).

## 2. METODOLOGIA

### 2.1 A Linguagem Utilizada

O MATLAB é um sistema interativo e uma linguagem de programação para computação técnica e científica em geral. Ele integra a capacidade de fazer cálculos, visualização gráfica e programação em um ambiente fácil de usar, em que os problemas e as soluções são expressos em uma linguagem matemática familiar. O MATLAB é um sistema interativo cujo elemento de dados básico é uma matriz que não requer dimensionamento. Isso permite solucionar muitos problemas computacionais, principalmente os que envolvem formulações matriciais ou vetoriais, em uma fração do tempo que seria necessário para escrever um programa em uma linguagem como C ou FORTRAN. Utilizou-se, por isso, o software MATLAB para o desenvolvimento dos algoritmos de segmentação.

A imagem utilizada para a validação dos algoritmos implementados foi a Lena512, mostrada na figura 1, uma vez que a mesma é bastante utilizada na literatura, podendo-se desta forma comparar os resultados obtidos.



---

**Figura 1:** Imagem Lena 512x512 pixels utilizada para teste dos algoritmos de segmentação.

Os algoritmos desenvolvidos baseiam-se em similaridades: limiarização (*Thresholding*), crescimento de regiões (*Region Growing*), junção e separação (*Split & Merge*), aglomeração (*Clustering*), e em descontinuidades: detecção de pontos, retas e bordas através de máscaras de convolução. Estas técnicas são discutidas a seguir.

## 2.2 Técnicas Baseadas em Similaridades

### 2.2.1 Limiarização (Thresholding)

Matematicamente, a operação de limiarização pode ser descrita como uma técnica de processamento de imagens na qual uma imagem de entrada  $f(x, y)$  de  $N$  níveis de cinza produz à saída uma imagem  $g(x, y)$ , chamada de imagem limiarizada, cujo o número de níveis de cinza é menor que  $N$ . Normalmente,  $g(x, y)$  apresenta 2 (dois) níveis de cinza, sendo

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

onde os pixels rotulados com 1 correspondem aos objetos e os pixels rotulados com 0 correspondem ao fundo e  $T$  é um valor de tom de cinza predefinido denominado limiar. Neste caso tem-se a limiarização simples.

Caso se tenha dois limiares tem-se a limiarização múltipla [5], que admite dois limiares  $T_1$  e  $T_2$ , e é definida por:

$$g(x, y) = \begin{cases} 1, & \text{se } T_1 \leq f(x, y) \leq T_2 \\ 0, & \text{nos demais} \end{cases}$$

#### 2.2.1.1 Conceito de Histograma

O histograma de uma imagem é um conjunto de números indicando o percentual de pixels naquela imagem, que apresentam um determinado nível de cinza [6].

Através da visualização do histograma de uma imagem obtemos uma indicação de sua qualidade quanto ao nível de contraste e quanto ao seu brilho médio (se a imagem é predominantemente clara ou escura).

Cada elemento desse conjunto é calculado como:

$$p_r(r_k) = \frac{n_k}{n}$$

onde:

$$0 \leq r_k \leq 1$$

$k = 0, 1, 2, \dots, L-1$ , onde  $L$  é o número de níveis de cinza da imagem digitalizada;

$n$  = número total de pixels na imagem;

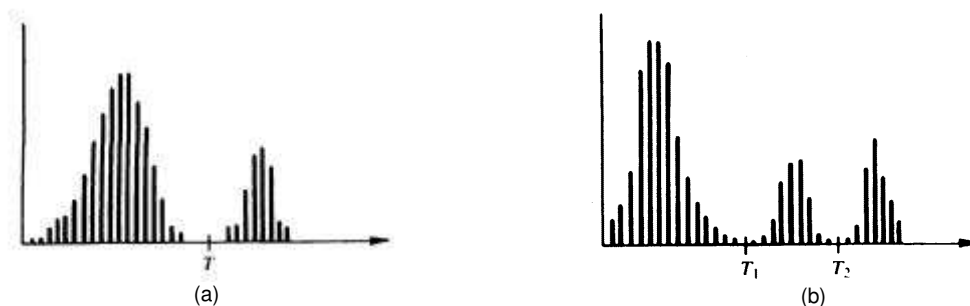
$p_r(r_k)$  = probabilidade do  $k$ -ésimo nível de cinza;

$n_k$  = número de pixels cujo nível de cinza corresponde a  $k$ .

#### 2.2.1.2 Critérios para o Estabelecimento de Limiares

Quando o histograma possui um vale como na figura 1a pode-se escolher o limiar como sendo o valor mínimo da curva do histograma. Caso haja mais de um vale tem-se o que a literatura chama de múltiplos limiares figura 1b.

A desvantagem dessa técnica, que combina o histograma com a limiarização, é que a mesma não resolve todos os problemas de segmentação, pois não leva em consideração, por exemplo, a forma dos objetos na imagem, isto é, dois objetos de formatos diferentes podem ser indistinguíveis usando-se esta técnica.



**Figura 2:** Histogramas de níveis de cinzas que podem ser particionados por (a) um único limiar e (b) por múltiplos limiares.

### 2.2.1.3 Limiarização pelas Propriedades Estatísticas da Imagem

Sabendo que o histograma é uma representação gráfica da distribuição de propriedade de ocorrência dos níveis de cinza em uma imagem, é lícito imaginar a possibilidade de uso de técnicas de cálculo do valor ótimo de limiar com base nas propriedades estatísticas da imagem.

Uma destas técnicas, denominadas limiarização ótima, parte de uma imagem da qual se conhecem as principais propriedades estatísticas (supondo que sua distribuição de probabilidades é normal ou gaussiana), a saber:

- $\mu_1$  : média dos tons de cinza da região de interesse;
- $\mu_2$  : média dos tons de cinza da região de fundo (background);
- $\sigma_1, \sigma_2$  : desvios padrão; e
- $P_1, P_2$ : probabilidade de ocorrência dos pixels.

Pode-se mostrar [4] que existe um valor ótimo de limiar  $T$ , dado por uma das raízes da equação

$$AT^2 + BT + C = 0,$$

onde:

$$A = \sigma_1^2 - \sigma_2^2$$

$$B = 2(\mu_1\sigma_2^2 - \mu_2\sigma_1^2)$$

$$C = \mu_2^2\sigma_1^2 - \mu_1^2\sigma_2^2 + 2\sigma_1^2\sigma_2^2 \ln\left(\frac{\sigma_2 P_1}{\sigma_1 P_2}\right)$$

Duas raízes reais e positivas indicam que a imagem pode requerer dois valores de limiar para obter uma solução ótima.

Se as variâncias forem iguais ( $\sigma^2 = \sigma_1^2 = \sigma_2^2$ ), um único valor  $T$  é necessário:

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 + \mu_2} \ln\left(\frac{P_2}{P_1}\right)$$

Se além disso, as duas classes forem equiprováveis:

$$T = \frac{\mu_1 + \mu_2}{2}$$

O que está de acordo com o conceito intuitivo de que o valor ótimo de limiar, quando as classes apresentam a mesma distribuição de probabilidade, (os lóbulos são exatamente iguais) é o ponto médio entre as médias das classes.

### 2.2.2 Aglomeração (Clustering)

O conceito de segmentação por aglomeração refere-se ao agrupamento de um dado conjunto de objetos em subconjuntos de acordo com as propriedades de cada objeto, no entanto é computacionalmente intenso. Essa técnica atua sobre um espaço denominado espaço de atributos [5], que representa um espaço de dimensão elevada no qual cada ponto da imagem é representado por um vetor de atributos. Atributo é o nome que se dá a uma medida feita sobre a imagem tais como tamanho de objetos na imagem, desvio padrão, componentes de um sistema de cores(RGB, HSB, etc) e outros.

O vetor de atributos é dado por  $\mathbf{x}=[x_1, x_2, \dots, x_n]^T$ , onde  $n$  representa a dimensão do espaço de atributos e  $x_i$  é o valor do  $i$ -ésimo atributo de numa dada localização do pixel. Um espaço de atributos é um sistema cartesiano em que cada eixo representa um atributo diferente.

A técnica abordada [2] é dividida em dois estágios onde primeiro são computados os vetores de atributos e segundo é determinado o número ótimo de aglomerados, segundo um critério de proximidade dos centros de aglomeração.

O algoritmo usado para isso tenta, primeiramente, aglomerar todos os  $M$  vetores em torno de 2 centros (que podem ser escolhidos ao acaso dentre os  $M$  vetores), segundo a distância do vetor ao centro mais próximo dele. Após isso, o número de centros vai aumentando de 1 e um fator de qualidade  $\beta$  vai sendo computado a cada iteração até que ele atinja um valor máximo, que igual a 1 (que fornecerá o número  $K$  ótimo de aglomerados). A cada próxima etapa, o centro do novo aglomerado a ser criado será o ponto que possui a maior distância ao centro do aglomerado, dentre os aglomerados atuais.

O cálculo do fator de qualidade  $\beta$  é feito da seguinte forma. Sejam:

$M$  o número de pontos a serem aglomerados;

$K$  o número de aglomerados atual;

$M_k$  o número de pontos do aglomerado  $A_k$  ( $k$ -ésimo aglomerado);

$\mathbf{x}_i$  um vetor de características,  $1 \leq i \leq N$

$\mu_0$  o vetor médio de todos os vetores, dado por  $\mu_0 = \frac{1}{M} \sum_{i=1}^M \mathbf{x}_i$

$\mu_k$  a média dos vetores de  $A_k$

$S_k$  o conjunto dos elementos de  $A_k$

$\mathbf{S}_B$  e  $\mathbf{S}_W$ , duas matrizes esparsas (externa e interna, respectivamente) dadas

por:

$$\mathbf{S}_W = \frac{1}{K} \sum_{k=1}^K \frac{1}{M_k} \sum_{\mathbf{x}_i \in S_k} [\mathbf{x}_i - \mu_k][\mathbf{x}_i - \mu_k]^T$$
$$\mathbf{S}_B = \frac{1}{K} \sum_{k=1}^K [\mu_k - \mu_0][\mu_k - \mu_0]^T$$

onde  $u^T$  é o vetor transposto de  $u$ . Finalmente,

$$\beta = \text{tr}[\mathbf{S}_W] \text{tr}[\mathbf{S}_B];$$

onde  $\text{tr}$  é o traço da matriz ou a soma dos elementos da diagonal da matriz.

Depois de aglomerados, os diversos aglomerados formam segmentos da imagem.

### 2.2.3 Crescimento de Regiões (Region Growing)

Crescimento de regiões é um procedimento que agrupa pixels ou sub-regiões em regiões maiores.

A mais simples das abordagens é a agregação de pixels, a qual começa com um conjunto de pontos “sementes”, e desses crescem regiões pela junção de cada ponto semente aqueles pixels vizinhos que têm propriedades similares (tais como nível de cinza, textura, cor).

Os operadores simples, i.e., baseados apenas num critério de vizinhança, podem conduzir a resultados ruins, pois uma região A pode ligar-se a uma outra região B devido apenas a diferença existente entre um ponto da borda da região A e seu vizinho na região B. A abordagem utilizada aqui foi introduzida primeiramente por Brice e Fenema [5], que desenvolveram um conjunto de regras simples de crescimento.

Na primeira etapa do processo, pares de pontos vizinhos são ligados para formar um conjunto, que os autores chamaram de regiões atômicas, se eles têm o mesmo valor de cinza. Em outras palavras, se eles têm a mesma intensidade e são 4-conectados, então eles irão formando regiões conexas;

Na segunda etapa, que eles chamaram de heurística da fagocitose, regiões maiores devem absorver as regiões menores segundo o seguinte critério: Sejam  $P_1$  e  $P_2$  os comprimentos das bordas de duas regiões  $R_1$  e  $R_2$ , respectivamente. Seja  $|I|$  o comprimento da fronteira  $I$  entre  $R_1$  e  $R_2$ . Seja  $|W|$  o comprimento das partes “fracas” da fronteira, i.e., o comprimento de  $W \subset I$  onde a diferença absoluta entre os níveis de cinza de  $R_1$  e  $R_2$  através da fronteira é menor do que um certo  $\epsilon_1$ . Desta forma, duas regiões,  $R_1$  e  $R_2$ , são ligadas se:

$$\frac{|W|}{\min\{|P_1|, |P_2|\}} > \epsilon_2 \text{ onde } \epsilon_2 = \frac{1}{2}, \text{ usualmente.}$$

Na terceira etapa, regiões adjacentes cuja fronteira é fraca, i.e., têm vários pontos cuja diferença absoluta entre os níveis de cinza através da fronteira é menor do que  $\epsilon_1$ , são ligadas segundo o seguinte critério:

$$\frac{|W|}{|I|} > \epsilon_3, \text{ onde } \epsilon_3 = \frac{3}{4}, \text{ usualmente.}$$

O método utilizado produz uma segmentação razoavelmente aceitável para imagens simples com poucos objetos e pouca textura.

### 2.2.4 Separação e Junção (Split & Merge)

Supondo que  $R$  representa a região da imagem inteira. O processo de segmentação pode ser visto como um processo que particiona  $R$  em  $n$  sub-regiões,  $R_1, R_2, \dots, R_n$ , tal que:

- (a)  $\bigcup_{i=1}^n R_i = R$ ;
- (b)  $R_i$  é uma região conectada,  $i=1, 2, 3, \dots, n$ ;
- (c)  $R_i \cap R_j = \emptyset$  para todo  $i$  e  $j$ ,  $i \neq j$ ;
- (d)  $P(R_i) = \text{VERDADEIRO}$  para  $i=1, 2, 3, \dots, n$ ; e
- (e)  $P(R_i \cup R_j) = \text{FALSO}$  para  $i \neq j$ .

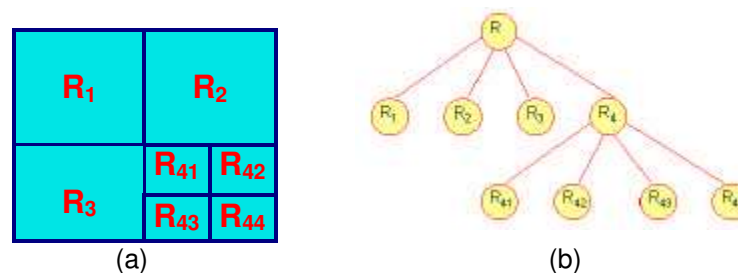


Onde  $P(R_i)$  é um predicado lógico sobre os pontos no conjunto  $R_i$  e  $\phi$  é o conjunto vazio.

A condição (a) indica que a segmentação deve ser completa, isto é, todo pixel deve estar na região. A segunda condição requer que os pontos na região devem ser conectados. A condição (c) indica que as regiões devem ser distintas. A condição (d) lida com as propriedades que devem ser satisfeitas pelos pixels numa região segmentada – por exemplo  $P(R_i) = \text{VERDADEIRO}$  se todos os pixels em  $R_i$  tiverem a mesma intensidade. Finalmente, a condição (e) indica que as regiões  $R_i$  e  $R_j$  são diferentes no sentido do predicado  $P$ .

Para a segmentação de uma imagem uma alternativa é subdividi-la em um conjunto arbitrário de regiões distintas e então juntar ou separar essas regiões numa tentativa de satisfazer as condições acima discutidas. Um algoritmo que iterativamente trabalha no sentido de satisfazer essas condições é como se segue.

Suponha que  $R$  representa a imagem inteira e um determinado predicado  $P$ . Para uma imagem quadrada, uma maneira de segmentar  $R$  é subdividi-la sucessivamente em porções de quadrantes mais e mais pequenos. Tais que para alguma região  $R_i$ ,  $P(R_i) = \text{VERDADEIRO}$ . Isto é, se  $P(R) = \text{FALSO}$ , divide-se a imagem em quadrantes. Se  $P$  é FALSO para algum quadrante, subdivide-se aquele quadrante em um subquadrante, e assim por diante. Esta técnica particular de separação tem uma conveniente representação na forma “quadtree”, como ilustrado na figura 5. Nota-se que a raiz da árvore corresponde à imagem inteira e que cada nó corresponde a uma subdivisão. Neste caso, somente  $R_4$  foi subdividida posteriormente.



**Figura 3:** (a) Imagem particionada; (b) Quadtree correspondente.

As técnicas de S&M podem ser computacionalmente “pesadas” se o processo de separação for iniciado pela imagem inteira. Por outro lado, se o processo começar com a árvore excessivamente separada (ramificada), tal que as folhas são os pontos da imagem, podem ocorrer erros devido às medidas da uniformidade de uma região estarem baseadas, inicialmente, apenas em pontos vizinhos. Assim, o S&M inicia, normalmente, num estágio intermediário entre estes dois extremos [5].

### 2.3 Técnicas Baseadas em Descontinuidades

Os objetos que podem ser localizados por técnicas baseadas em descontinuidades são de três tipos: pontos, linhas e bordas. Essas técnicas são definidas por uma operação orientada a vizinhança.

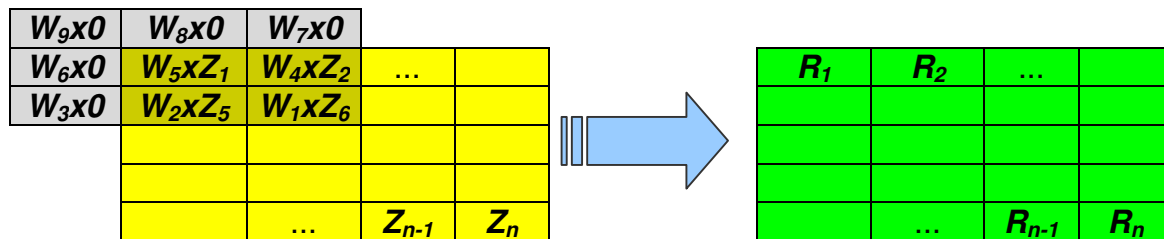
Operações lógicas e aritméticas orientadas a vizinhança utilizam o conceito de convolução com máscaras (ou janelas ou templates ou kernel ou masks) [1, 3, 9].

A cada posição relativa da máscara sobre a imagem, o pixel central da subimagem em questão será substituído, em uma matriz denominada imagem destino, pela soma dos produtos dos coeficientes com os níveis de cinza contidos na sub-região

envolvida pela máscara. Isto é, a resposta da máscara em algum ponto na imagem é:

$$R_n = W_1Z_1 + W_2Z_2 + \dots + W_9Z_9 = \sum_{i=1}^9 W_iZ_i \quad (n = 1, \dots, \text{tamanho da imagem}) \quad [6]$$

A figura 3 ilustra o cálculo da resposta percorrendo-se uma imagem com uma dada máscara.



**Figura 4:** Processo de convolução com máscara e o resultado. Deve-se notar o espelhamento da máscara.

### 2.3.1 Detecção de Pontos Isolados

A detecção de pontos isolados em uma imagem é direta. Usando a máscara mostrada na figura 4.

-1	-1	-1
-1	8	-1
-1	-1	-1

**Figura 5:** Máscara usada para detecção de diferentes pontos isolados.

Diz-se que um ponto tem sido detectado na localização em que a máscara está centrada se

$$|R| > T$$

onde T é um limiar não negativo, e R é como definido anteriormente. Basicamente, tudo que essa formulação faz é medir as diferenças de pesos entre o ponto central e seus vizinhos. A idéia é que o nível de cinza de um ponto será totalmente diferente dos níveis de cinza de seus vizinhos.

### 2.3.2 Detecção de Linhas

Da mesma forma que um ponto isolado, também pode ser segmentada uma linha, desde que a máscara de peso usado seja adequado para realçar as partes de uma reta. Utilizam-se as máscaras da figura 5.

-1	-1	-1	-1	-1	-1	-1	-1	-1
2	2	2	-1	-1	-1	2	2	2
-1	-1	-1	-1	-1	-1	-1	-1	-1
Horizontal			+45°			Vertical		
-1	-1	2	-1	-1	2	-1	2	-1
-1	2	-1	-1	2	-1	-1	2	-1
-1	2	-1	-1	2	-1	-1	2	-1
2	-1	-1	-1	-1	-1	-1	-1	-1
-1	2	-1	-1	-1	-1	-1	-1	-1
-1	-1	2	-1	-1	-1	-1	-1	-1
			-45°					

**Figura 6:** Máscaras de linhas.



### 2.3.3 Detecção de Bordas

Para a detecção de bordas, aplicam-se filtros espaciais lineares de dois tipos: baseados no gradiente da função luminosidade da imagem, e baseados no laplaciano [6].

#### 2.3.3.1 O Operador Gradiente

O gradiente de uma imagem  $f(x, y)$  na localização  $(x, y)$  é o vetor

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

Sabe-se da análise vetorial que o vetor gradiente aponta na direção da máxima taxa de variação de  $f$  em  $(x, y)$ . Na detecção de borda uma quantidade importante é o módulo desse vetor, geralmente referido simplesmente como gradiente e denotado  $\nabla f$ , onde:

$$\nabla f = \text{mag}(\nabla f) = [G_x^2 + G_y^2]^{1/2}$$

Essa quantidade é igual a máxima taxa de crescimento de  $f(x, y)$  por unidade de distância na direção de  $\nabla f$  [4]. Uma prática comum é aproximar o gradiente do valor absoluto:

$$\nabla f \approx |G_x| + |G_y|$$

o qual é muito mais simples de se implementar.

Para uma imagem digital  $f(x, y)$ , o módulo do gradiente pode ser aproximado por operadores 3x3 ou máscaras de convolução. Na tabela 1 são mostrados alguns importantes exemplos dessas máscaras.

Operador	Vertical ( $ G_y $ )	Horizontal ( $ G_x $ )
Roberts	$\begin{bmatrix} 0 & 0 & -1 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
Sobel	$\frac{1}{4} \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{4} \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$

**Tabela 1:** Operadores 3x3 utilizados para estimar a amplitude do gradiente através de uma borda.

#### 2.3.3.2 Operador Laplaciano

O laplaciano de uma função 2-D  $f(x, y)$  é uma derivada de segunda ordem definida como

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Como no caso do gradiente pode ser aproximado por máscaras 3x3 como mostrado na figura 6.

0	-1	0
-1	-4	-1
0	-1	0

**Figura 7:** Máscara utilizada para computar o Laplaciano.

Embora o laplaciano seja insensível à rotação, e portanto capaz de realçar ou detectar bordas em qualquer direção, seu uso é restrito devido a sua grande suscetibilidade a ruído [6].

### 3. RESULTADOS



**Figura 8** – Imagem binária obtida por limiarização. Limiar 111.



**Figura 9** – Imagem binária obtida por limiarização. Limiar 187



**Figura 10** – Detecção de pontos isolados



**Figura 11** – Detecção de retas horizontais.



**Figura 12** – Detecção de retas verticais.



**Figura 13** - Segmentação utilizando gradiente com máscaras de Sobel vertical.



**Figura 14** - A soma dos módulos do gradiente horizontal e vertical de Sobel.



**Figura 15** – Segmentação utilizando o Laplaciano

#### 4. CONCLUSÕES

Dentro da área de Processamento de Imagens, relacionada à segmentação, discutiu-se as seguintes técnicas baseadas em semelhanças entre os objetos ou padrões da imagem: limiarização (inspeção visual, baseada em histogramas e nas características estatísticas da imagem), crescimento de regiões, junção & separação e aglomeração. Fez-se o mesmo para as técnicas que envolvem operação de convolução com máscaras, ou seja, as que se baseiam em dissimilaridades ou descontinuidades. Essas técnicas, embora tenham sido aplicadas em âmbito geral, podem ser aplicadas em casos específicos, uma vez que, o uso de uma ou outra técnica apresenta melhores resultados, dependendo da imagem que se quer extrair as informações.

As técnicas baseadas em regiões podem ser aplicadas a imagens de múltiplo espectros ou imagem em cores, obtendo também um resultado satisfatório.

A partir dos resultados obtidos observou-se que os operadores que fazem uso do gradiente não apresentam visualmente diferenças que comprometam a qualidade da segmentação, o que mostra a eficiência de cada operador.

Pode-se, finalmente, notar que não existe um método universal de segmentar uma imagem, para cada caso uma técnica mostra-se mais eficiente que outra e a cada dia uma nova técnica pode ser implementada sendo este um campo ainda aberto para a pesquisa.

#### 5. BIBLIOGRAFIA

- [1] CASTLEMAN, Kenneth R. - Digital Image Processing. Prentice-Hall, 1996.
- [2] COLEMAN, Guy B. & ANDREWS, Harry C. - Image Segmentation by Clustering. Proceedings of the IEEE, vol. 67, no. 5, mai 1979.
- [3] DAWSON, Benjamin M. - Introduction to Image Processing Algorithms. Byte, pp-169-186, mar 1987.
- [4] GONZALEZ, Rafael C. & WOODS, Richard E. - Digital Image Processing. Addison - Wesley, 1992.
- [5] HIRATA Jr., Roberto - Segmentação de Imagens por Morfologia Matemática. Tese de Mestrado. IME-USP, 1997.
- [6] MARQUES Filho, Ogê & VIEIRA Neto, Hugo - Processamento Digital de Imagens. Brasport, 1999.
- [7] MATLAB Image Processing Toolbox, User's Guide, The Math Works, mai 1997.
- [8] PÄRT-ENANDER, Eva & SJÖBERG, Anders - The Matlab 5 Handbook. Addison-Wesley, 1999.
- [9] PRATT, William K. - Digital Image Processing. Wiley Interscience, 1991.