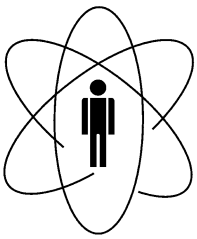
ISSN 2447-1127



CBPF - CENTRO BRASILEIRO DE PESQUISAS FÍSICAS

Rio de Janeiro

Monografias – CBPF

CBPF-MO-V.1,N.4

setembro 2015

Processamento digital de imagens: conceitos fundamentais

André Persechino & Márcio Portes de Albuquerque

GOVERNO FEDERAL
BRASIL
PÁTRIA EDUCADORA

Monografia, CBPF, v.1, n. 4, p. 1–41, 2015
dx.doi.org/10.7437/MO24471119/2015.04.001

Processamento digital de imagens: conceitos fundamentais

Digital image processing: fundamental concepts

André Persechino[1] e Márcio Portes de Albuquerque[1]

[1]Centro Brasileiro de Pesquisas Físicas - Coordenação de Atividades Técnicas, Rua Dr. Xavier Sigaud 150, Rio de Janeiro, RJ. CEP: 22290-180. Brasil.

15 de Outubro de 2015

Resumo

A proposta deste trabalho é fornecer um material conciso sobre processamento de imagens, já que o uso de técnicas de imageamento é uma realidade em vários ramos da física. Com essa proposta em mente, são apresentados os principais conceitos da área de processamento digital de imagens: representação digital de imagens, filtragem, segmentação *labeling* e *data clustering*. Adicionalmente, o conteúdo é finalizado com uma apresentação sucinta da morfologia matemática. Visando dar o mínimo de suporte prático, algumas aplicações em MATLAB® são expostas.

Palavras-chave: Processamento digital de imagens; instrumentação científica

Abstract

The aim of this work is to provide a concise introduction on image processing, once the use of imaging systems is consolidated in several branches of physics. With this purpose in mind, the main concepts of digital image processing are discussed: digital representation, filtering, segmentation, labeling and data clustering. In addition, a discussion on mathematical morphology is carried out. In order to give minimum support to practice, some MATLAB®applications are shown.

Key-words: Digital image processing; scientific instrumentation

Introdução

É frequente o uso de técnicas de imageamento em física: sistemas biológicos microscópicos podem ser monitorados via microscopia óptica; sólidos cristalinos podem ter sua estrutura determinada por análise de figuras de difração; materiais podem ser diferenciados via tomografia computadorizada, dentre muitos outros exemplos. Mais do que meros sistemas de visualização, sistemas experimentais baseados em imagens permitem a *quantificação* das informações contidas nos sinais analisados. Entretanto, para fazer uso de todo esse potencial, é importante que os envolvidos com a análise destes sinais possuam um mínimo de conhecimento em processamento digital de imagens.

Como o processamento de sinais/imagens não faz parte da formação básica nas gradua-

ções em física no Brasil, ocorre frequentemente que os envolvidos com medidas realizadas via imageamento não possuem os conhecimentos mínimos para analisar criticamente os sinais gerados, implicando em uma dependência perigosa do uso de pacotes comerciais fechados para a realização de tarefas diversas, mesmo as mais simples. Visando reduzir essa dependência, propomos nestas notas uma introdução robusta ao processamento de imagens. Obviamente, não foi possível cobrir todos os temas dessa área, de forma que direcionamos o enfoque para os tópicos mais fundamentais e corriqueiros, mas que frequentemente são negligenciados.

O material aqui contido é fortemente baseado na dissertação do autor A. Persechino [20], com algumas pequenas mudanças. O trabalho está estruturado da seguinte forma: os conceitos introdutórios são apresentados na Seção 1; os fundamentos do processo de filtragem são descritos na Seção 2; em sequência, são abordadas as idéias centrais em segmentação 3 e processamento de alto-nível (*labeling* e *clustering*, Seção 4). Adicionalmente, uma breve exposição sobre morfologia matemática é feita na Seção 5. Por fim, há um apêndice onde constam duas implementações em MATLAB® : uma para segmentação via limiarização e outra para filtragem no domínio das frequências. Para aqueles que desejarem aprofundamento nos tópicos discutidos, um curso completo em processamento de sinais pode ser encontrado em Diniz *et al.*[7] e um curso específico em processamento de imagens pode ser encontrado no livro de Pedrini e Schwartz [19].

1 Conceitos introdutórios

Uma imagem pode ser compreendida como um sinal representativo de alguma estrutura, seja ela de natureza física, artística, conceitual etc. De uma forma geral, uma imagem pode ser tomada como um sinal (normalmente multidimensional) que carrega informação sobre a estrutura retratada. Um primeiro modelo matemático para uma imagem seria a de uma função escalar que atribui a cada ponto do plano um valor dado por

$$\begin{cases} u(x,y) = I(x,y)R(x,y), \\ 0 \leq I(x,y) < \infty \\ 0 \leq R(x,y) < 1. \end{cases} \tag{1}$$

em que $I(x,y)$ e $R(x,y)$ referem-se à iluminância e à reflectância, respectivamente. A iluminância corresponde a uma medida da quantidade de luz incidente sobre a cena ou estrutura representada, ao passo que a reflectância está associada à quantidade de luz refletida pelas estruturas representadas [19].

O modelo apresentado na Equação (1) triunfa por sua simplicidade. Contudo, dentro de um cenário de processamento *digital* de imagens, ele deve ser adaptado. Uma imagem digital u com M linhas e N colunas pode ser compreendida como uma matriz (que pode ser multidimensional[1]), em que cada entrada u_{ij} está associada à intensidade luminosa na posição (i,j). Ou seja,

$$u = \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1\mathrm{N}} \\ u_{21} & u_{22} & \cdots & u_{2\mathrm{N}} \\ \vdots & \vdots & \ddots & \vdots \\ u_{\mathrm{M}1} & u_{\mathrm{M}2} & \cdots & u_{\mathrm{MN}} \end{bmatrix}. \tag{2}$$

Cada elemento u_{ij} da imagem digital é chamado de *pixel*, abreviação de *picture element*.

Amostragem e quantização

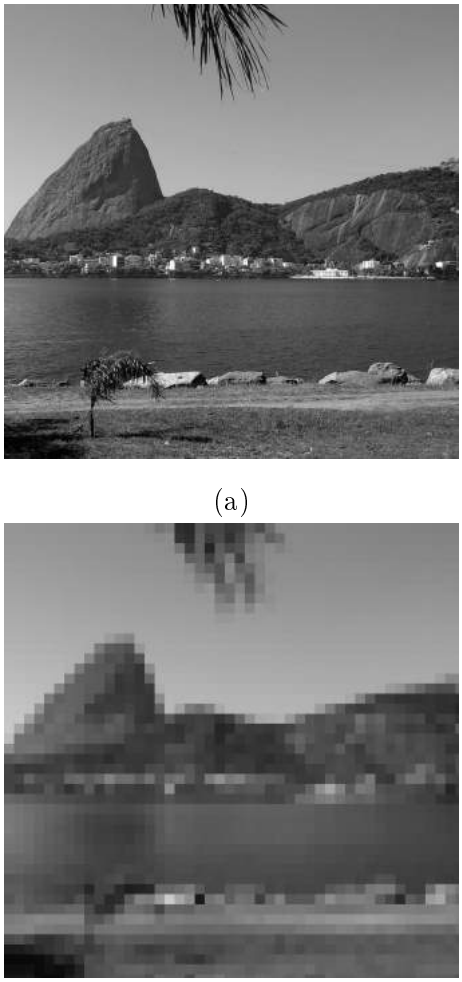
Deve-se ter em mente que o processo de digitalização não pode ser realizado de qualquer maneira: o domínio da função imagem deve ser corretamente *amostrado* e seus valores devem ser adequadamente *quantizados*. A amostragem do domínio da imagem deve ser tal que a *frequência de amostragem*, f_s, seja maior que

[1] Neste caso, à entrada u_{ij} associa-se um conjunto ordenado de valores $\{u_{ij1}, u_{ij2}..., u_{i,jk}\}$, ao invés de apenas um. Esse é o caso de uma imagem colorida.

o dobro da frequência máxima presente no sinal original (teorema da amostragem). Caso este critério não seja atendido, poderá ocorrer o fenômeno de *aliasing*, que corresponde ao surgimento de componentes espectrais que não existiam originalmente, causando a corrupção ou degradação do sinal [19, 7]. Uma unidade de medida típica para resolução espacial é o ppi: pixels por polegada (*pixels per inch*). Um exemplo de reamostragem de uma imagem pode ser visualizado na Figura 1.

Por sua vez, a quantização está associada à quantidade de bits usados para representar os valores de intensidade luminosa da imagem digital. Em geral, usam-se quantidades de bits que são potências de 2, dessa forma o intervalo de valores possíveis para uma imagem quantizada em N bits varia de 0 a $2^{\mathrm{N}} - 1$. Por exemplo, uma imagem quantizada em 8 bits tem entradas que podem assumir valores entre 0 e 255. O efeito de diferentes quantizações sobre a imagem mostrada na Figura 1 (a) pode ser visualizado nas Figuras 2, 3 e 4.

Podemos concluir das discussões sobre amostragem e quantização que quão mais refinada for a amostragem e mais largo o intervalo de bits usados na quantização, maior espaço será necessário para o armazenamento de uma imagem e mais ineficiente será sua eventual transmissão. Ainda sobre a questão da amostragem, há o fato de que uma imagem com um número elevado de entradas demandará maior esforço computacional em processamento. Questões como essas devem ser consideradas previamente ao estabelecimento do sistema de aquisição e processamento.



(a)

(b)

Figura 1: Imagem original (a) e imagem reamostrada com frequência $f_s = 1/32\mathrm{px}^{-1}$ em ambas direções espaciais (b). A imagem reamostrada foi mantida com as mesmas dimensões em pixels da imagem original para evidenciar os efeitos da amostragem.

Coordenadas espaciais e vizinhanças

Usualmente, o sistema de coordenadas espaciais segue a orientação sugerida pela representação matricial mostrada na Equação (2) e tal como mostrado na Figura 5.

Conforme será discutido adiante, há técnicas de processamento de imagens que se baseiam em relações entre pixels de uma determinada região. Nesse sentido, é importante definir *a priori* qual a *vizinhança* adotada no esquema de processamento. Dada uma imagem digital u, temos que – à exceção dos pixels de fronteira – cada pixel u_{ij} tem 8 vizinhos imediatos: dois na horizontal, dois na vertical e quatro nas diagonais. Se estes oito vizinhos se relacionam com o pixel u_{ij}, sua relação de vizinhança com os demais é chamada N_8. Por outro lado, se o pixel u_{ij} só se relaci-



Figura 2: Imagem mostrada na Figura 1 (a) quantizada em 4 bits. Os valores de suas entradas podem variar de 0 a 15.



Figura 3: Imagem mostrada na Figura 1 (a) quantizada em 2 bits. Os valores de suas entradas podem variar de 0 a 3.

onar com os vizinhos na vertical e horizontal, a vizinhança é dita ser N_4. O que define este relacionamento é a forma como as operações sobre vizinhança são executadas. A Figura 6 ilustra os dois sistemas de vizinhança.

É importante ressaltar que a definição da vizinhança tem importância fundamental quando se processa uma imagem via modelos derivados de equações diferenciais parciais [1, 20], em que a aproximação das derivadas parciais se dá por diferenças finitas sobre os



Figura 4: Imagem mostrada na Figura 1 (a) quantizada em 1 bit . Os valores de suas entradas podem variar entre 0 ou 1.



Figura 5: Sistema de coordenadas usual em processamento de imagens.

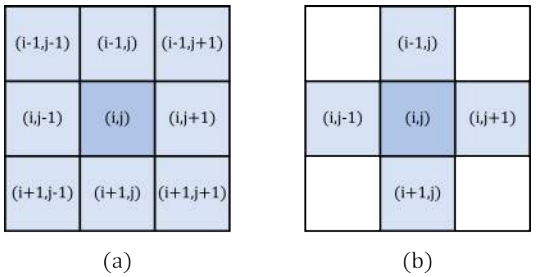


Figura 6: Vizinhanças (a) N_8 e (b) N_4 do pixel u_{ij}.

pixels da região processada.

Histograma de uma imagem

O histograma de uma imagem digital u de dimensões M×N é definido como sendo a sequência

$$h_k = \frac{p_k}{\mathrm{M} \times \mathrm{N}}. \tag{3}$$

em que $\frac{p_k}{\mathrm{M}\times\mathrm{N}}$ é a probabilidade de ocorrência do k-ésimo nível de intensidade.

Na forma apresentada na Equação (3), o histograma h constitui uma aproximação à distribuição de probabilidades associada aos níveis de intensidade de u. Nesse sentido, nenhuma informação espacial é mantida quando o histograma de uma imagem é obtido. Por esse motivo, um mesmo histograma pode estar associado a mais de uma imagem.

Do ponto de vista de *análise* de imagens, o histograma guarda informações sobre o contraste da imagem a ele associada. Imagens com boa relação de contraste apresentam histogramas mais distribuídos ao longo dos valores possíveis, ao passo que imagens de baixo contraste apresentam maior concentração em torno da média. As Figuras 7 e 8 apresentam uma ilustração deste fato.

Alterações na distribuição dos níveis de cinza de uma imagem podem ser executadas segundo as técnicas de filtragem – a serem discutidas mais adiante –, ou mesmo por meio de transformações sobre um único pixel, cuja forma geral é

$$w_{ij} = \mathrm{T}\left(u_{ij}\right), \tag{4}$$

em que $\mathrm{T}(\cdot)$ corresponde a uma transformação qualquer, linear ou não.

Um exemplo típico desse tipo de transformação consiste na operação de *alargamento linear de constraste*[2], $\mathrm{T_{ALC}}(\cdot)$. Essa operação redistribui – linearmente – os níveis de cinza da imagem, que estão dentro do intervalo $[u_{\min}, u_{\max}]$ original de forma a abranger todos os valores possíveis de uma nova escala, cujos limites são $v_{\min}$ e $v_{\max}$. A expressão para esta operação pode ser vista na Equação (5):

$$\mathrm{T_{ALC}}\left(u\right) = \frac{v_{\max} - v_{\min}}{u_{\max} - u_{\min}}(u - u_{\min}) + v_{\min}. \tag{5}$$

Para finalizar a discussão sobre histogramas, é importante notar que este sinal possibilita a extração de atributos estatísticos importantes, tais como valor médio, desvio padrão e variância, energia, entropia etc. Por isso, o histograma de uma imagem costuma ser utilizado frequentemente no processo de segmentação, a ser discutido na Seção 3.

2 Filtragem

Apresentados os conceitos básicos sobre representação de imagens digitais, faz-se necessário introduzir um conjunto de conceitos relativos às operações mais básicas definidas sobre estes objetos. Especificamente, a classe das operações aritméticas desempenha importante papel, pois leva naturalmente ao conceito de operação orientada à vizinhança. Esta, por sua vez, leva ao conceito de filtragem, sendo este um dos procedimentos mais fundamentais em processamento de imagens.

Operações aritméticas básicas

Uma primeira classe de operações sobre imagens consiste em manipularmos duas ou mais imagens [3] por meio de operações aritméticas básicas (adição, subtração, multiplicação, divisão). Ao efetuarmos operações desse tipo sobre imagens, devemos nos atentar ao fato de que os valores resultantes podem estar fora do intervalo de quantização. Por exemplo, ao somarmos duas imagens quantizadas a 8 bits, é possível que algum pixel da imagem resultante tenha valor maior que 255. Valores negativos podem ocorrer e deve-se atentar ao fato de que em divisão entre imagens

[2] Usualmente conhecido como *linear stretching*. Não confundir com *equalização do histograma*.

[3] Naturalmente, uma imagem pode ser manipulada com ela própria.

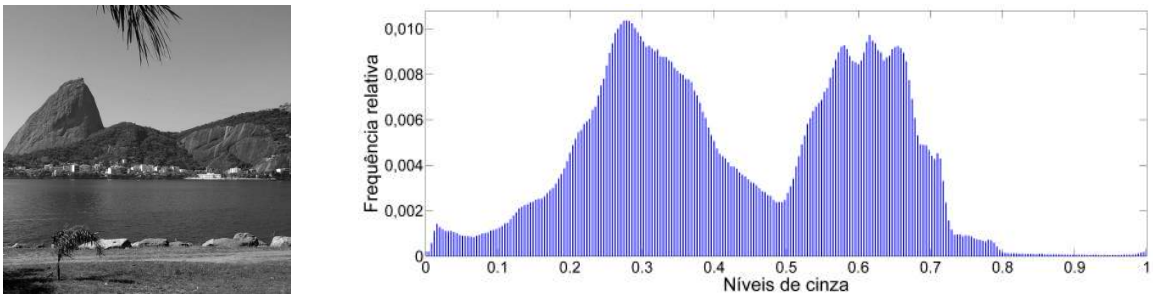


Figura 7: Figura original à esquerda e seu histograma, estimado com 256 divisões de níveis de cinza. O histograma fornece informações acerca da distribuição dos níveis de intensidade, mas não de sua localização. Os valores de intensidade (eixo horizontal) foram normalizados para o intervalo [0,1].

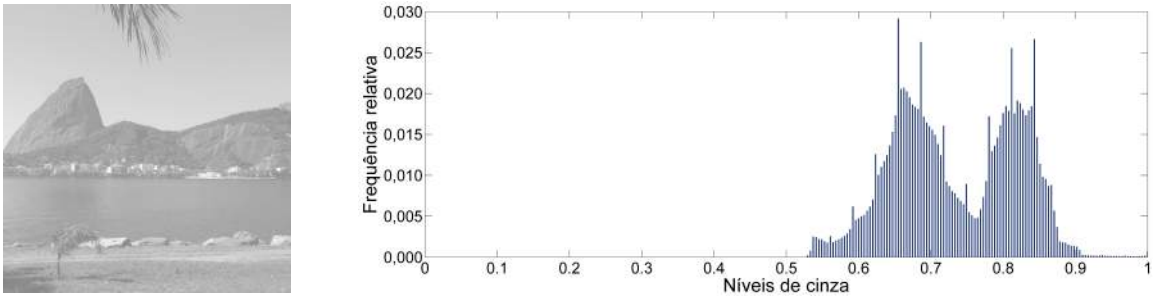


Figura 8: Figura com baixa relação de constraste (à esquerda) e seu histograma. Nota-se nesse caso, quando comparado com a Figura 7, que o perfil do histograma associado a uma imagem com pouco contraste apresenta maior concentração espacial, refletindo o fato de que os valores de intensidade estão menos dispersos em torno da média. Os valores de intensidade (eixo horizontal) foram normalizados para o intervalo [0,1].

pode ocorrer divisão por zero. Para contornar a questão do intervalo de quantização, normalmente submete-se a imagem resultante a um processo de normalização para que os valores estejam dentro do intervalo requerido. Quanto à divisão por zero, geralmente soma-se 1 no divisor.

Inicialmente, as operações aritméticas são aplicadas pixel a pixel, isto é, dada uma operação aritmética f a ser aplicada sobre duas imagens – u e v –, temos

$$w_{ij} = f(u_{ij}, v_{ij}).$$

Ou seja, cada pixel da imagem resultante w é dado pela aplicação de f sobre os pixels correspondentes de u e v.

Como exemplo de aplicação de operações aritméticas sobre imagens, são mostrados na Figura 10 os resultados de diversas manipulações de duas imagens-teste mostradas na Figura 9.

Operações orientadas a uma vizinhança

Conforme dito antes, as operações aritméticas relacionam diretamente um pixel de uma imagem ao pixel correspondente de outra imagem a ser operada. Contudo, estas operações podem ser estendidas para que sejam realizadas sobre uma vizinhança de certo pixel, daí a importância de se estabelecer *a priori* a estrutura de vizinhanças de uma imagem, tal como apontado ao final de Seção 1.

A dinâmica das operações orientadas a vizinhança pode ser compreendida como a seguinte sequência:

1. Fixe uma janela de tamanho fixo sobre o

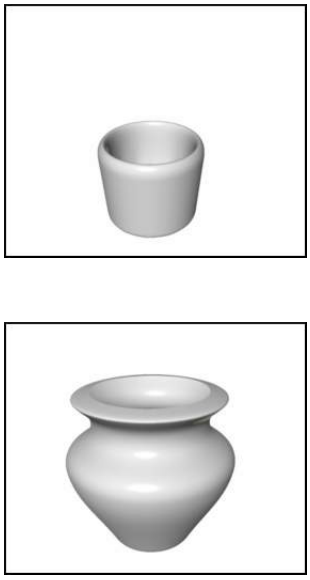


Figura 9: Imagens u (copo) e v (vaso) a serem usadas nas ilustrações de operações aritméticas sobre imagens.

pixel u_{ij}, estabelecendo uma vizinhança $\eta(i,j)$ a ser levada em consideração na operação;

2. atribua ao pixel resultante w_{ij} o resultado da operação descrita na Equação (6):

$$w_{ij} = \sum_{k,l \in \eta(i,j)} f_{kl} \cdot u_{ij}, \tag{6}$$

em que f é uma *máscara* cujos coeficientes definirão qual operação está sendo realizada [4]; e

3. passe ao pixel seguinte e reinicie o processo até varrer todos os pixels da imagem original.

Uma ilustração desta dinâmica pode ser visualizada na Figura 11.

Convém ressaltar que os pixels de fronteira não possuem a mesma estrutura de vizinhança de pixels internos. Esse fato demanda uma adaptação de vizinhos para pixels localizados nas fronteiras da imagem. Duas abordagens usuais são: complementar as posições ausentes com zeros (*zero padding*) ou replicar os pixels. Adicionalmente, os valores atualizados dos pixels processados devem ser armazenados em outra imagem, de forma que a vizinhança não seja alterada durante o processamento dela própria.

Um exemplo básico de operação orientada à vizinhança consiste em atribuir ao pixel u_{ij} a média aritmética dos pixels delimitados pela máscara. Um exemplo da aplicação da média sobre uma vizinhança de 400 pixels pode ser visualizado na Figura 12. Neste exemplo, a máscara a ser usada é da forma

$$f = \frac{1}{400} \cdot \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & 1 & 1 & \dots & 1 \\ 1 & 1 & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix}_{20\times 20}.$$

Resultados completamente distintos podem ser atingidos alterando-se os coeficientes de f na equação (6). Essa é a base da filtragem de imagens, a ser discutida a seguir.

Em primeiro lugar, é conveniente rescrever a Equação (6) de forma explícita [19]:

$$w_{ij} = \sum_{k=\lfloor -\frac{m}{2} \rfloor}^{\lfloor \frac{m}{2} \rfloor} \sum_{l=\lfloor -\frac{n}{2} \rfloor}^{\lfloor \frac{n}{2} \rfloor} f_{kl} \cdot u_{i+k,j+l}, \tag{7}$$

para uma máscara de dimensão $m \times n$.

Analisando a Equação (7), percebemos que uma operação orientada a vizinhança corresponde, de fato, à operação de *correlação discreta* entre a máscara f e a imagem u. De maneira análoga, é possível que a operação de correlação torne-se uma operação de *convolução discreta*:

$$w_{ij} = (\tilde{f} * u)_{ij} = \sum_{k=\lfloor -\frac{m}{2} \rfloor}^{\lfloor \frac{m}{2} \rfloor} \sum_{l=\lfloor -\frac{n}{2} \rfloor}^{\lfloor \frac{n}{2} \rfloor} \tilde{f}_{kl} \cdot u_{i-k,j-l}, \tag{8}$$

em que o asterisco representa a convolução entre os dois sinais e $\tilde{f}$ corresponde a f rotaci-

[4] Conforme será visto adiante, a máscara corresponde, de fato, ao *núcleo* de um certo operador

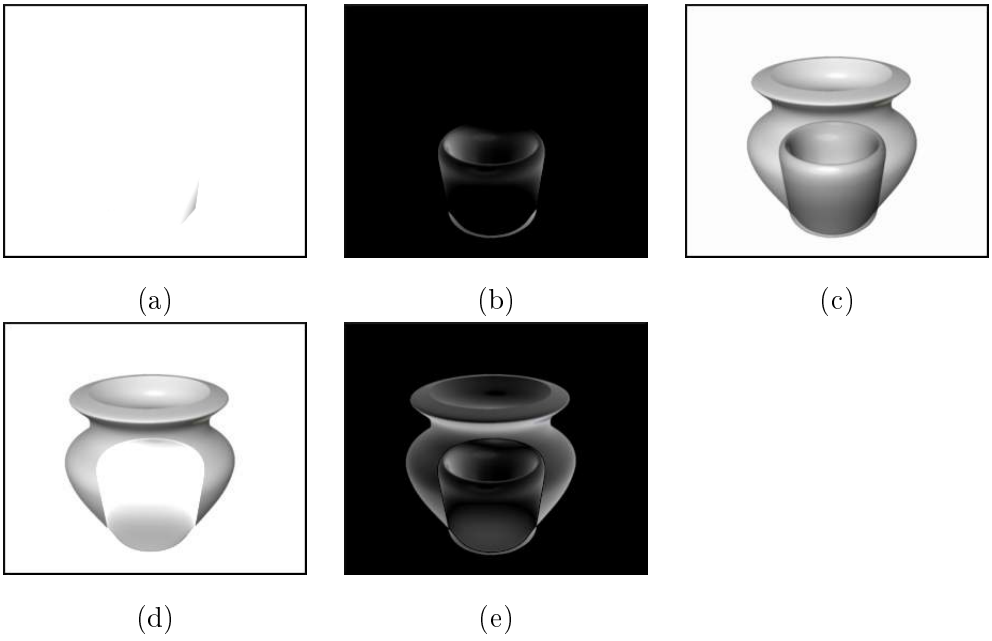


Figura 10: Resultados de operações aritméticas entre as imagens constantes na Figura 9. Em (a) é mostrado $w = u + v$, em (b) $w = u - v$, em (c) $w = u \times v$, em (d) $w = \frac{u}{v}$ e em (e) é mostrado $w = |u - v|$.

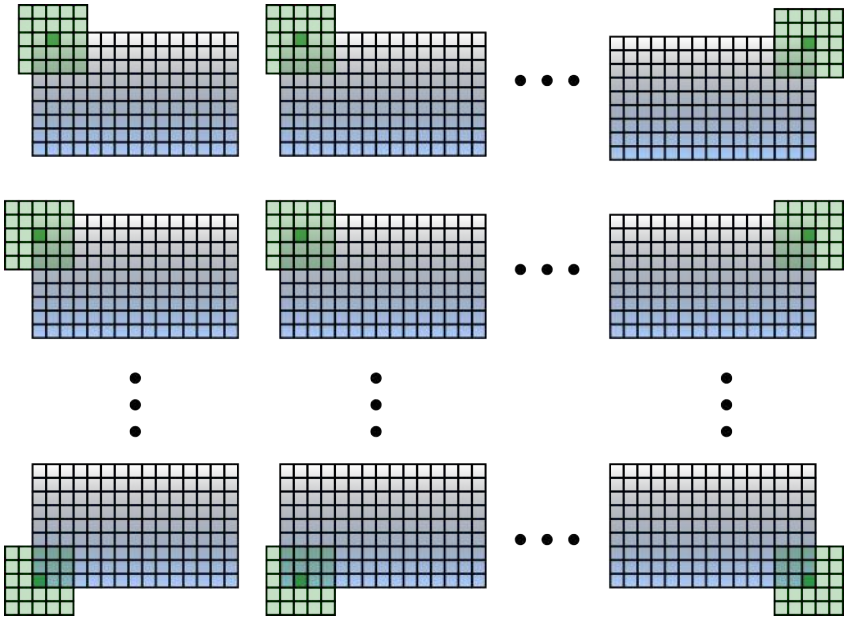


Figura 11: Ilustração da dinâmica de operação orientada a vizinhança: uma máscara (em verde) varre a imagem original (em azul). Uma vez centralizada sobre um determinado pixel, uma operação sobre os pixels delimitados é efetuada.

onada em 180°(caso f seja simétrica, as operações de correlação e convolução são equivalentes).

Do ponto de vista conceitual, o interesse em



Figura 12: Imagem mostrada na Figura 1 (a) submetida à operação de média aritmética sobre uma vizinhança de 400 pixels. Note-se a perda de detalhes finos.

representar uma operação por meio da convolução reside no fato de que esta está intimamente ligada ao produto das transformadas de Fourier de f e u, permitindo assim uma análise no domínio da frequência. Em uma dimensão e em domínio contínuo, temos:

$$\mathcal{F}(w(x)) = \mathcal{F}((f * u)(x)) = \mathcal{F}(f(x)) \cdot \mathcal{F}(u(x)).$$

Ou seja, a transformada de Fourier da convolução entre $f(x)$ e $u(x)$ é o produto das transformadas individuais:

$$\mathcal{W}(k) = \mathcal{F}(k) \cdot \mathcal{U}(k) \tag{9}$$

Este é um resultado bastante conhecido [7, 19, 16] e pode ser estendido para o caso bidimensional:

$$\mathcal{W}(\omega_1, \omega_2) = \mathcal{F}(\omega_1, \omega_2) \cdot \mathcal{U}(\omega_1, \omega_2), \tag{10}$$

em que ω_1 e ω_2 correspondem às frequências espaciais.

Levadas em conta as devidas questões sobre discretização dos sinais, todas as características da transformada de Fourier podem ser exploradas no contexto de imagens digitais. Ou seja, analogamente ao processamento de sinais unidimensionais, pode-se processar imagens no domínio da frequência. Em termos práticos, a análise no domínio das frequências é útil para se entender o princípio de filtros passa-baixa, passa-alta e passa-banda.

Uma descrição alternativa de uma imagem *digital* u consiste em tomarmos sua Transformada Discreta de Fourier (DFT[5]). Nesta representação, podemos avaliar a composição espectral do sinal original; isto é, como este é composto em termos de frequências. Transições abruptas - tais como bordas bem definidas - correspondem a componentes de alta frequência, ao passo que regiões de pouca - ou nenhuma - variação espacial estão associadas a componentes de baixa frequência.

2.1 Filtros passa-baixas

Como primeira abordagem para o processo de filtragem no domínio das frequências, tomemos o caso dos filtros passa-baixas. Esse tipo de filtro remove do sinal original as componentes de alta frequência, removendo do sinal ruídos impulsivos, mas sob a pena de remover também detalhes finos. A expressão de um filtro passa-baixas ideal no domínio das frequências é da forma

$$|H(\omega_1, \omega_2)| = \begin{cases} 1; & |r| \leq r_c \\ 0; & \text{caso contrário.} \end{cases} \tag{11}$$

Na Equação acima, $r = \sqrt{\omega_1^2 + \omega_2^2}$[6]. Percebe-se que a magnitude do filtro H consiste em um disco de raio r_c centrado na origem. Como ilustração, um filtro passa-baixas semelhante ao representado pela Equação (11) é aplicado sobre uma imagem teste, tal como mostrado nas Figuras 13 e 14.

Conforme ilustrado na Figura 14, pode-se verificar que a aplicação do filtro passa-baixas representado pela Equação (11) conserva as

[5] Do inglês, *Discrete Fourier Transform*.

[6] Note-se que o sistema de coordenadas usual em imagens digitais é retangular. A definição dos filtros ideais em coordenadas polares é realizada aqui por simplicidade, uma vez que seus efeitos são os mesmos do que os correspondentes retangulares – salvos os intervalos de frequências afetadas.

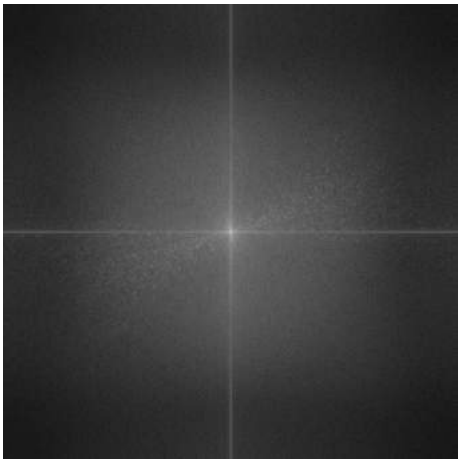



Figura 13: Imagem original (à esquerda) e seu espectro discreto de potência, dado pela magnitude da DFT. Para melhor visualização do espectro, este foi submetido a uma transformação logarítmica. A origem do sistema de coordenadas do espectro está localizada no centro da imagem.

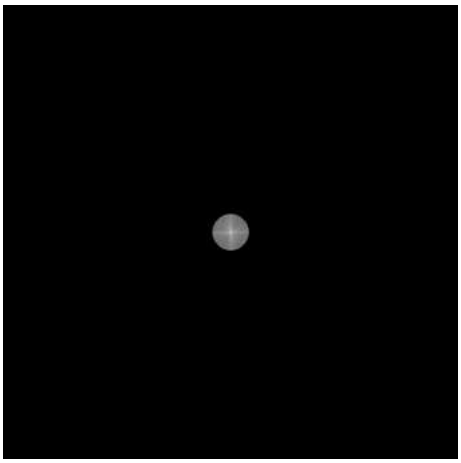



Figura 14: Espectro discreto de potências da imagem filtrada por um passa-baixas ideal (à esquerda) e imagem reconstruída por inversão da DFT (à direita). Nota-se a suavização das bordas e a consequente perda dos detalhes finos. As ondulações devem-se à transição abrupta do filtro (vide texto).

componentes espectrais de baixa frequência, levando a uma suavização generalizada da imagem. As ondulações presentes na imagem filtrada são devidas ao *fenômeno de Gibbs* [7, 16], que está associado, em parte, ao fato de descontinuidades completamente abruptas só poderem ser representadas com um número infinito de componentes espectrais. Como uma representação com infinitos termos é impossível na prática, o truncamento sobre o número de componentes espectrais leva ao surgimento das referidas ondulações. Uma maneira de minimizar esse fenômeno consiste em usar janelas nos sinais [7, 16].

2.2 Filtros passa-altas

O caso oposto ao filtro passa-baixas é o filtro passa-altas, cuja forma ideal tem a representação

$$|H(\omega_1, \omega_2)| = \begin{cases} 1; & |r| \geq r_c \\ 0; & \text{caso contrário.} \end{cases} \quad (12)$$

Nesse caso, o disco delimitado por r_c define quais frequências internas a ele serão anuladas. Isso acarreta a evidenciação das regiões de maior variação espacial (alta frequência). Devido ao fato de que bordas apresentam transições razoavelmente intensas entre tons de cinza distintos, estes detalhes são preservados, ao passo que regiões de maior uniformidade espacial são suprimidas. Uma ilustração da aplicação deste tipo de filtro pode ser vista na Figura 15. Analogamente ao exemplo anterior, o fenômeno de Gibbs pode ser observado na imagem filtrada, dada a variação abrupta do filtro em torno de r_c.

2.3 Filtros passa-banda

É possível selecionar um intervalo fechado específico de frequências, diferentemente dos filtros passa-altas e passa-baixas. Isto é, seleciona-se uma região em forma de anel no plano das frequências. Tal tipo de filtro é chamado de filtro passa-banda e seu caso ideal tem a forma

$$|H(\omega_1,\omega_2)| = \begin{cases} 1; & r_{c_1} \leq |r| \leq r_{c_2} \\ 0; & \text{caso contrário.} \end{cases} \quad (13)$$

Com essa configuração de anel de largura $|r_{c_1} - r_{c_2}|$, o filtro passa banda faz com que o sinal filtrado mantenha características intermediárias quando comparado com os dois filtros anteriores. Isto é, a supressão de ruídos não é extrema, bem como não o é a evidenciação de bordas. Uma ilustração da aplicação deste tipo de filtro pode ser visualizada na Figura 16.

2.4 Filtros Gaussiano e diferenciador

Uma vez explicitados os conceitos fundamentais de filtragem sobre imagens, convém apresentar dois tipos de filtro de especial interesse: Gaussiano e diferenciador. Um filtro Gaussiano corresponde a um passa-baixas cuja transição entre as frequências a serem mantidas e suprimidas não é abrupta. Por sua vez, o diferenciador constitui um filtro passa-altas, proporcionando aproximações para as derivadas direcionais da imagem original. A seguir, será feita uma discussão sobre os dois filtros listados acima, devido à sua utilização massiva neste trabalho.

Filtro Gaussiano

Uma função Gaussiana, unidimensional, centrada na média μ e com variância σ^2 tem a forma

$$G(x,\mu,\sigma^2) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right), \quad (14)$$

em que σ corresponde ao *desvio-padrão* da função. O desvio padrão σ está associado à largura da Gaussiana, tal como ilustrado na Figura 17.

Em sua versão bidimensional, uma Gaussiana centrada na posição $\vec{r}_0 = (x_0, y_0)$ com variância σ^2 tem a forma

$$G(\vec{r},\vec{r}_0,\sigma^2) = \frac{1}{2\pi\sigma^2} \exp\left(\frac{-\|\vec{r}-\vec{r}_0\|^2}{2\sigma^2}\right), \quad (15)$$

em que $\vec{r} = (x, y)$ e o operador $\|.\|$ remete à norma Euclidiana, isto é, $\|\vec{r} - \vec{r}_0\| = \sqrt{(x-x_0)^2 + (y-y_0)^2}$. Na Figura 18 podem ser visualizadas projeções no plano de três Gaussianas com diferentes valores de σ.

A implementação digital de um filtro Gaussiano – isto é, a obtenção dos coeficientes de seu operador – pode ocorrer por amostragem de valores em sua forma contínua, ou seja:

$$\mathbf{G}_n = \frac{1}{2^{2n}} \cdot [g]_{n\times n},$$

em que

$$g_{ij} = \exp\left(\frac{-(i^2+j^2)}{2\sigma^2}\right), \quad \text{com } i,j \in \left[-\frac{n-1}{2}, \frac{n-1}{2}\right]. \quad (16)$$

Outra possibilidade consiste em representar o filtro Gaussiano 2-D como um produto de filtros Gaussianos 1-D, refletindo o fato de que

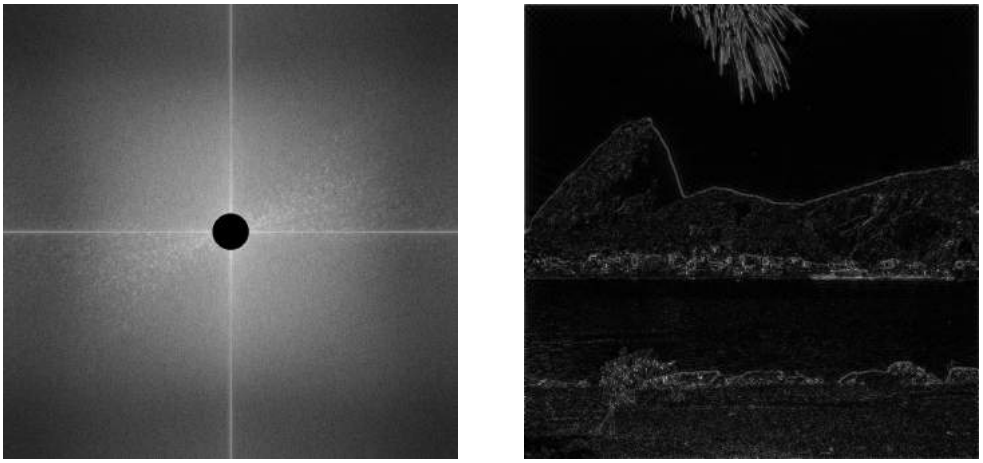


Figura 15: Espectro discreto de potências da imagem filtrada por um passa-altas ideal (à esquerda) e imagem reconstruída por inversão da DFT (à direita). Nota-se o realce dos detalhes finos e a evidenciação de bordas.



Figura 16: Espectro discreto de potência do produto entre a DFT da imagem original e o filtro passa-banda (esquerda) e a imagem reconstruída (à direita). Nesse caso, há a preservação de um intervalo fechado de frequências.

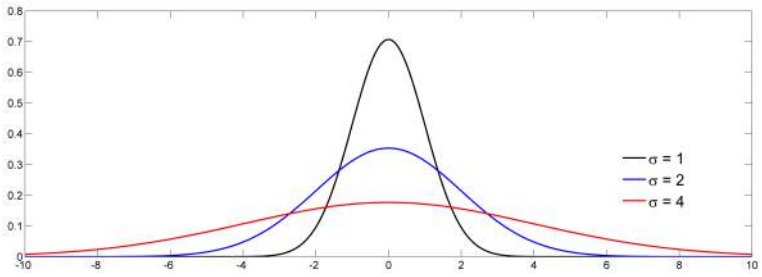
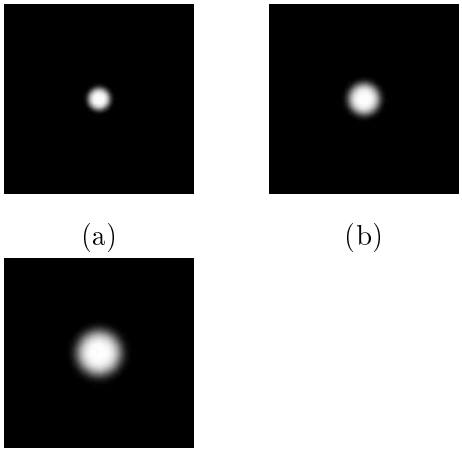


Figura 17: Três Gaussianas centradas na origem com diferentes valores de desvio padrão.

este é um operador separável [19][7]. Definindo

[7] Um operador separável é aquele que pode ser aplicado sobre linhas e colunas (ou vice-versa) da imagem em sequência, levando ao mesmo resultado que a aplicação simultânea.



(a) (b)

(a)

Figura 18: Projeções no plano de três Gaussianas bidimensionais centradas na origem com diferentes valores de desvio padrão: $\sigma = 1; 2$ e 4 para (a), (b) e (c), respectivamente. Para melhor visualização as Gaussianas tiveram suas amplitudes normalizadas.

$$\mathbf{B} = \begin{bmatrix} \frac{n!}{0!(n-0)!} \\ \frac{n!}{1!(n-1)!} \\ \vdots \\ \frac{n!}{n!(n-n)!} \end{bmatrix}, \tag{17}$$

temos que

$$\mathbf{G}_n = \frac{1}{2^{2n}} \cdot \mathbf{B}\mathbf{B}^{\mathrm{T}}, \tag{18}$$

em que o índice T indica transposição da matriz. Em ambas Equações (16) e (18), o fator $\frac{1}{2^{2n}}$ é uma constante de normalização correspondente à soma dos elementos das matrizes em questão.

Uma característica fundamental dos filtros Gaussianos é a sua *isotropia* [19]. Isto é, o filtro opera igualmente em todas as direções, sem privilegiar alguma direção em particular.

Devido ao fato de a transformada de Fourier de uma função Gaussiana também ser uma Gaussiana, constata-se que esse filtro pertence à classe dos filtros passa-baixas. Diferentemente do filtro ideal, a transição entre as componentes espectrais toleradas e suprimidas é suave. Na Figura 19 pode ser visualizada a aplicação de um filtro Gaussiano com $n = 60$ e com $\sigma = 8$ e $\sigma = 16$ sobre a imagem-teste apresentada na Figura 1 (a).

Para finalizar a discussão sobre filtros Gaussianos, cabe ressaltar que há uma outra maneira de se implementar esse tipo de filtro: ele pode ser implementado iterativamente devido à sua relação com processos de difusão lineares. Esta relação é estudada em detalhes em [20].

Diferenciador

Uma operação típica em sinais consiste em obtermos suas derivadas. No caso de sinais multidimensionais, é necessária a especificação da direção em que a derivada será tomada. Uma primeira aproximação das derivadas de um sinal contínuo $f(x)$ consiste em truncar sua série de Taylor:

$$f(x+h) = \sum_{n=0}^{\infty} \frac{h^n}{n!} f^{(n)}(x). \tag{19}$$

Na Equação (19) o termo $f^{(n)}(x)$ corresponde à n-ésima derivada. O termo em que a série será truncada definirá o erro associado a este procedimento. Duas formas imediatas de aproximação da derivada do sinal $f(x)$ se dão pelas diferenças *progressivas*, $\Delta_+[f(x)]$, e *regressivas*, $\Delta_-[f(x)]$:

$$\begin{cases} \Delta_+[f(x)] &= f(x+h) - f(x); \text{ e} \\ \\ \Delta_-[f(x)] &= f(x) - f(x-h). \end{cases}$$

Com isso, as aproximações das derivadas são obtidas por

$$\begin{cases} f'(x) &\approx \frac{1}{h}\Delta_+[f(x)]\,; \text{ ou} \\ \\ f'(x) &\approx \frac{1}{h}\Delta_-[f(x)]\,. \end{cases} \tag{20}$$

Pode-se mostrar [5] que a *diferença centrada*, $\delta[f(x)] = f(x+h) - f(x-h)$, constitui melhor aproximação para a derivada $f'(x)$:

$$f'(x) \approx \frac{1}{2h}\delta\cdot[f(x)] \tag{21}$$



Figura 19: Aplicação de filtros Gaussianos de tamanho $n = 60$, $\sigma = 8$ (imagem à esquerda) e $\sigma = 16$ sobre a imagem-teste apresentada na Figura 1 (a). Nota-se em ambos os casos a supressão de detalhes finos, característica de filtros passa-baixa. Contudo, devido ao perfil espectral suave do filtro, o fenômeno de Gibbs é praticamente inexistente.

Fazendo uso recursivamente destes operadores de diferença é possível obter aproximações para derivadas de ordem superior. Por exemplo, a estimativa da derivada segunda $f''(x)$ é dada por

$$f''(x) \approx \frac{f(x+2h) - 2f(x) + f(x-2h)}{(2h)^2}.$$

Uma forma mais frequente para a aproximação de $f''(x)$ usa o incremento $\frac{h}{2}$. Com isso, chega-se em

$$f''(x) \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}. \quad (22)$$

A generalização para funções de várias variáveis é imediata, bastando operar as diferenças em função da variável de interesse (x, y, z etc.). Por exemplo, o operador *Laplaciano*, $\nabla^2(\cdot) \equiv \partial_{xx} + \partial_{yy}$, pode ser aproximado por

$$\begin{aligned} &\frac{f(x+h_x, y) - 2f(x,y) + f(x-h_x, y)}{h_x^2} + \\ &\frac{f(x, y+h_y) - 2f(x,y) + f(x, y-h_y)}{h_y^2} \\ \Rightarrow \nabla^2 f(x,y) &\approx \frac{\delta_x^2 [f(x,y)]}{h_x^2} + \frac{\delta_y^2 [f(x,y)]}{h_y^2}. \end{aligned} \quad (23)$$

Os conceitos de diferenças finitas aqui apresentados podem ser aplicados a sinais discretos, tal como imagens digitais. A única diferença fundamental remete ao fato de que o refinamento da malha é limitado. Ou seja, o incremento h não pode tender a zero indefinidamente. Tipicamente, usa-se $h = 1$. Dessa forma, a vizinhança de um pixel segue a notação discutida na Seção 1 (vide Figura 6). Com base nessas informações, podemos definir uma série de filtros diferenciadores para imagens.

A diferença progressiva direcional pode ser aproximada por uma matriz cujas entradas correspondem aos coeficientes das aproximações via diferenças finitas:

$$\mathbf{D}_{x_+} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \text{ e } \mathbf{D}_{y_+} = \mathbf{D}_{x_+}^{\mathrm{T}}. \quad (24)$$

De maneira análoga, é possível definir filtros de diferenças regressivas e centradas:

$$\mathbf{D}_{x_-} = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \text{ e } \mathbf{D}_{y_-} = \mathbf{D}_{x_-}^{\mathrm{T}}. \quad (25)$$

$$\mathbf{D}_{x_c} = \frac{1}{2}\begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \text{ e } \mathbf{D}_{y_c} = \mathbf{D}_{x_c}^{\mathrm{T}}. \quad (26)$$

Por sua vez, o filtro Laplaciano tem a forma

$$\mathbf{L} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}. \tag{27}$$

Os coeficientes nulos nos operadores foram inseridos para proporcionar uma matriz quadrada de dimensão ímpar, visando facilitar a centralização do pixel a ser operado.

Cabe aqui uma observação importante: os filtros diferenciadores mostrados nas Equações (24), (25), (26) e (27) foram definidos segundo um esquema de vizinhança N_4 (vide Seção 1). Essa constatação surge do fato que os pixels nas diagonais do pixel central não são levados em consideração no cômputo das diferenças. Ao se adotar um esquema de vizinhança N_8, as diagonais devem ser consideradas.

Dois diferenciadores bastante populares que adotam vizinhança N_8 são os filtros de Prewitt [19, 21], $\mathbf{P}$, e Sobel [19], $\mathbf{S}$:

$$\mathbf{P}_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{e} \quad \mathbf{P}_y = \mathbf{P}_x^{\mathrm{T}}. \tag{28}$$

$$\mathbf{S}_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{e} \quad \mathbf{S}_y = \mathbf{S}_x^{\mathrm{T}}. \tag{29}$$

Para ilustrar a aplicação dos filtros diferenciadores, usaremos a imagem sintética retratada na Figura 20. Este tipo de figura, com padrões espaciais bem definidos, é ideal para ilustrar a ação de filtros passa-alta, dado que as bordas constituem regiões de variação espacial abrupta. Adicionalmente ao padrão espacial, foi inserido na imagem-teste um gradiente de transição suave no fundo que, conforme será visto, não influi na identificação das bordas. Nota-se pela Figura 21 que os filtros discutidos nesta seção constituem boa aproximação para operadores diferenciais em ambas direções espaciais. A aplicação destes filtros é padrão na detecção de bordas e contornos, sendo esta uma das principais abordagens em

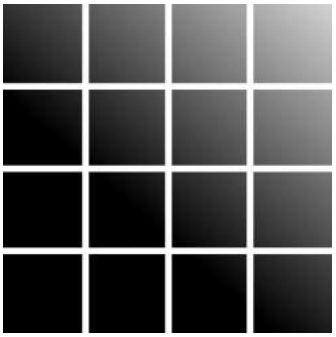
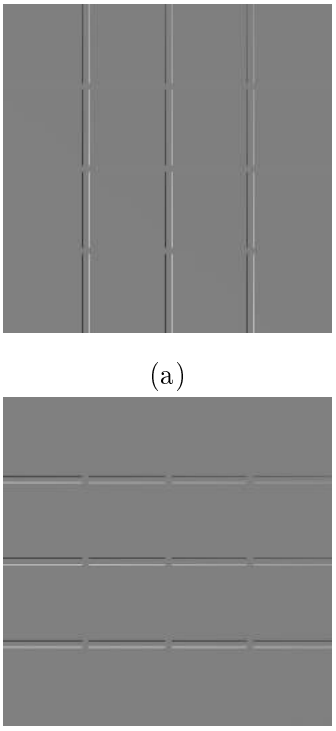


Figura 20: Imagem sintética usada para ilustrar a aplicação dos filtros diferenciadores. Esta imagem é constituída por uma padrão espacial retangular ideal para supressão segundo a filtragem por diferenciadores.



(a)

(b)

Figura 21: Aplicação do filtro de Prewitt sobre a imagem-teste na direções x (a) e y (b). Nota-se que a evidenciação das bordas verticais e horizontais ocorre, de fato. Observa-se também que o gradiente de fundo da imagem original tem pouca influência no resultado pois sua variação é suave.

segmentação de imagens, a ser discutida em detalhes na Seção 3.

Convém notar que é possível obtermos uma estimativa da *magnitude do gradiente* da imagem, $\|\vec{\nabla} f(x,y)\|$:

$$\|\vec{\nabla} f\| = \sqrt{\mathbf{D}_x^2[f] + \mathbf{D}_y^2[f]}, \tag{30}$$

em que $\mathbf{D}$ representa algum operador diferencial direcional de primeira ordem. Este recurso é usado tipicamente na evidenciação de bordas da imagem original. Além disso, conhecido $\|\vec{\nabla} f(x,y)\|$, é possível determinar a direção em que os níveis de cinza crescem localmente. A Figura 22 ilustra a determinação de $\|\vec{\nabla} f\|$ para a imagem-teste apresentada na Figura 20. As derivadas direcionais foram estimadas pelo operador de Prewitt.

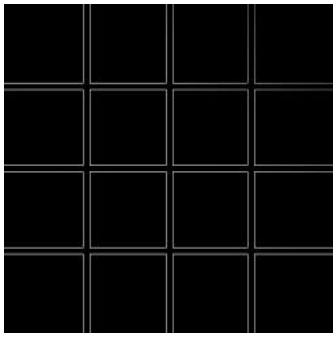


Figura 22: Estimação da magnitude do gradiente da imagem-teste apresentada na Figura 20. Nota-se o destaque eficiente das bordas da imagem, quase não sendo perceptível o efeito do gradiente suave de fundo, a menos do enfraquecimento do constraste no setor direito superior da imagem. Para melhor visualização, a imagem resultante foi submetida a um alargamento linear de constrate.

Para finalizar a discussão sobre filtros diferenciadores, é importante ressaltar que estes filtros constituem passa-altas. Portanto, são bastante sensíveis a ruídos (principalmente pontuais). Isso deve ser levado em consideração quando da utilização destes operadores. A Figura 23 traz a estimativa para o gradiente da Figura 1 (a) como exemplo da magnitude do gradiente de imagens reais, mais susceptíveis a ruídos diversos.

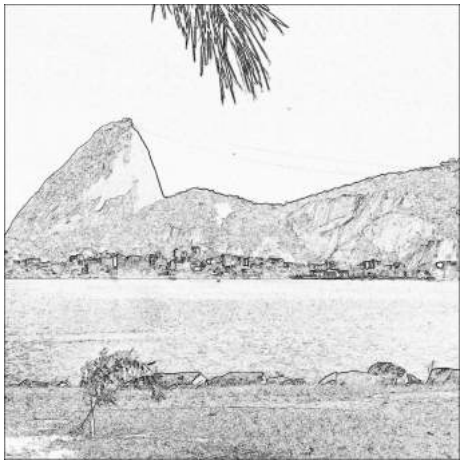


Figura 23: Estimação do gradiente da imagem-teste apresentada na Figura 1 (a). As derivadas foram aproximadas pelo operador Sobel. Nota-se o destaque eficiente das bordas, mas com a presença de ruídos relativos aos detalhes naturais da imagem, tais como a água ou a grama. Para melhor visualização, a imagem resultante foi submetida a inversão (negativo) e a um alargamento linear de constrate.

3 Segmentação de imagens

Todo o ferramental discutido anteriormente sobre operações em imagens é comumente direcionado à *segmentação* de imagens. De uma maneira geral, segmentar uma imagem consiste em particioná-la em suas componentes constituintes. Geralmente, o resultado obtido é o destaque dos *objetos* da imagem em relação ao *fundo*. De um ponto de vista formal, uma imagem u *idealmente* segmentada pode ser compreendida como a união de N regiões R_i disjuntas entre si, ou seja

$$u = \bigcup_{i=1}^{\mathrm{N}} R_\mathrm{i} \quad \text{e} \quad R_\mathrm{i} \cap R_\mathrm{j} = \emptyset, \text{ se } i \neq j.$$

Há uma subjetividade intrínseca na questão da segmentação de imagens: a decomposição da imagem em componentes depende da definição destes. Decorre disto que processos de segmentação que atendem requisitos para certa classe de imagens podem funcionar muito mal na segmentação de outra classe. Em síntese: não há, até hoje, uma *teoria da*

segmentação de imagens. Isto que dizer que os processos desenvolvidos para segmentar imagens guardam relações profundas com a estrutura destas, não permitindo generalizações.

Conforme dito, a subjetividade acerca das definições de objeto e fundo representa uma dificuldade natural do processo de segmentação. Entretanto, este ponto é mais profundo: podemos dizer que o processo de segmentação é a etapa mais crítica do processamento de imagens, pois é o resultado da segmentação quer será processado e avaliado por etapas posteriores. O diagrama retratado na Figura 24 ilustra a organização básica de um sistema de processamento de imagens.

Há duas abordagens básicas para segmentação de imagens: técnicas baseadas em *similaridade* entre pixels e técnicas baseadas em *descontinuidade* entre pixels. Estas duas abordagens podem ser combinadas. A seguir será feita discussão sobre duas técnicas básicas destas duas classes. A primeira delas, limiarização, faz uso da semelhança entre os valores dos pixels; a segunda, detecção de bordas, faz uso de descontinuidades entre pixels.

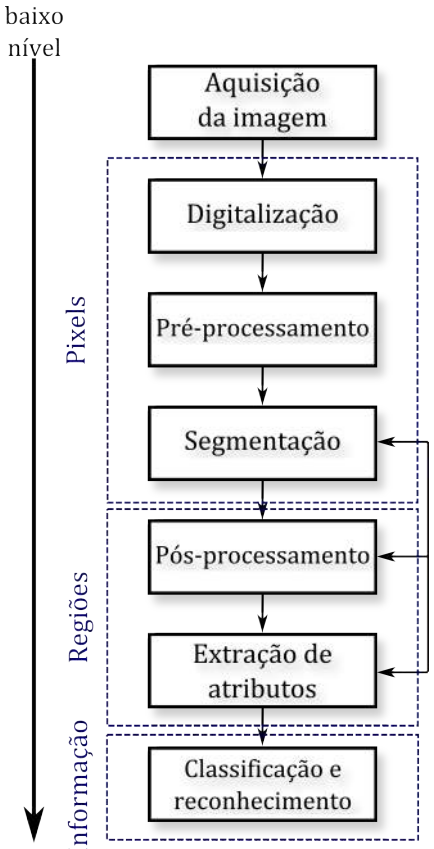


Figura 24: Diagrama representativo das etapas de um sistema de processamento de imagens. As etapas de pré-processamento e segmentação operam sobre os pixels da imagem. A transformação de pixels (dados) em informação se dá ao longo das etapas de pós-processamento e extração de atributos, que podem envolver realimentação, isto é, estas etapas são interdependentes juntamente com a segmentação.

3.1 Limiarização

Conforme discutido na Seção 1, o histograma de uma imagem apresenta informações sobre a distribuição de seus valores de intensidade. Este perfil de distribuição pode ser utilizado para definir duas classes na imagem: *objeto* e *fundo*. Para tanto, é aplicada sobre a imagem uma transformação da forma

$$\mathrm{T}\left(u_{ij}\right) = \begin{cases} 0, & \text{se } u_{ij} \leq \mathrm{L} \\ 1, & \text{caso contrário} \end{cases} . \quad (31)$$

Na Equação (31), o parâmetro L corresponde ao *limiar* de corte [8], daí o nome da técnica. Devido à forma da transformação T, esta operação é frequentemente chamada de binarização.

A técnica de binarização funciona bem desde que o histograma da imagem apresente um perfil aproximadamente *bimodal*, pois isto representa o fato de que os valores de intensidade da imagem estão estatisticamente separados em, pelo menos, duas classes distintas. A Figura 25 ilustra essa característica. Naturalmente, um histograma perfeitamente bimodal constitui uma idealização. Contudo, há situações em que essa aproximação é viável, tal como pode ser visualizado na Figura 7 (Seção 1). Entretanto, não raro lidamos com

[8] Usualmente denominado por sua forma inglesa, *threshold*.

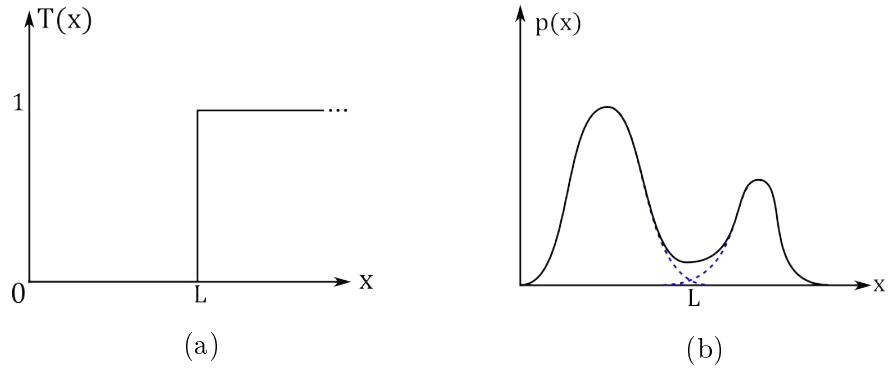


Figura 25: Transformação global para binarização (a). Esta transformação anula os pixels cujos valores estão abaixo do limiar L, enquanto conserva os demais. O limiar L é facilmente obtido quando o histograma apresenta perfil bimodal (b), pois corresponde ao valor de separação entre os dois pontos de máximo no histograma. Em pontilhado, é mostrado que um histograma bimodal pode ser interpretado como a composição de duas distribuições.

imagens cujo histograma em nada se aproxima a uma distribuição bimodal, tal como a imagem apresentada na Figura 26. Justamente pelo fato de lidar-se frequentemente com histogramas sem perfil bimodal, a escolha de um ou mais limiares constitui tarefa não trivial.

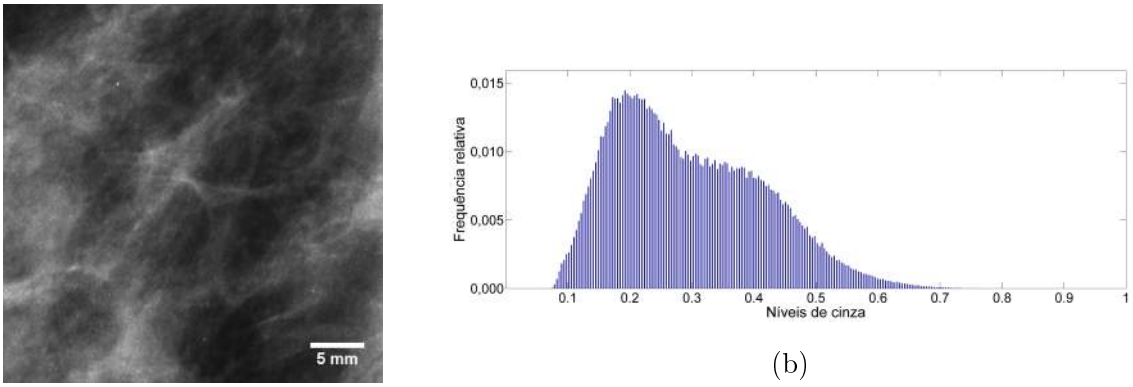
Dentre as mais diversas abordagens para determinação do limiar ótimo, uma característica bastante frequente destaca-se: a busca por um limiar L_0 que otimize (maximize ou minimize) alguma função critério. A função critério pode corresponder, por exemplo, a uma distância entre duas distribuições (método do erro mínimo), à entropia do histograma (métodos entrópicos) ou à razão entre variâncias (método de Otsu). Revisões sobre técnicas de limiarização podem ser encontradas nos trabalhos de Sahoo *et al.* [24], Weszka [27] e Chang *et al.* [4]. No Apêndice A, Seção A.1 constam descrições dos métodos utilizados neste trabalho. Na Figura 27 pode ser vista a aplicação de três binarizações sobre a imagem apresentada na Figura 26. Os limiares foram calculados pelo métodos do valor médio, Otsu e da máxima entropia.

Para finalizar a discussão sobre a técnica de limiarização, é importante ressaltar que a abordagem usada neste trabalho consiste em uma limiarização *global*, no sentido que o histograma é obtido sobre o conjunto de todos os pixels da imagem em questão. Embora isso simplifique o processo, atributos ou características locais não são considerados. Isso pode acarretar a segmentação inadequada de estruturas na imagem devido a diferenças locais. Nesse sentido, é possível que sejam aplicadas técnicas de limiarização *local*, que consistem basicamente em analisar atributos locais dos pixels para a definição de limiares para cada região delimitada.

3.2 Detecção de bordas

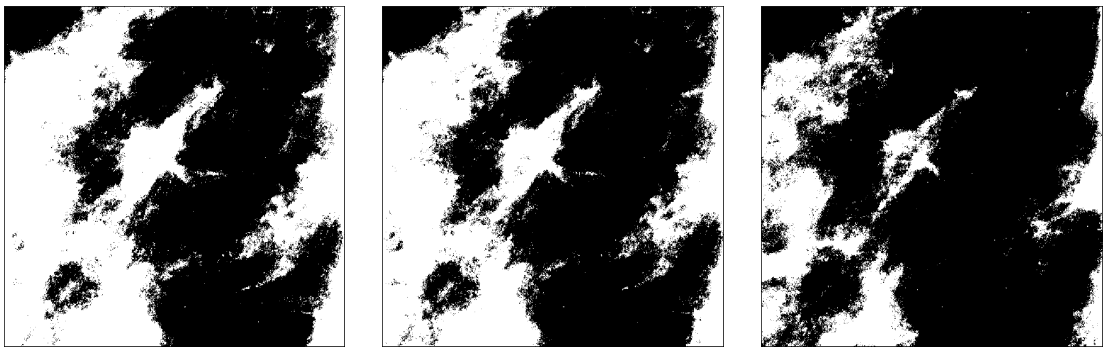
Diferentemente da limiarização, que opera via semelhança entre pixels, a detecção de bordas funciona com base em *descontinuidades* entre pixels. Idealmente, uma borda corresponde a uma transição completamente abrupta ao longo de certa direção da imagem. A aproximação para a borda ideal é a função degrau, tal como ilustrado na Figura 28. O princípio de detecção de bordas já foi discutido na Seção 2: descontinuidades são evidenciadas por meio de filtros passa-alta. Nesse sentido, os filtros diferenciadores desempenham papel importante, pois permitem a evidenciação das bordas, e quando combinados com a operação de limiarização, permitem sua segmentação.

Uma dificuldade intrínseca em detectar bor-



(a) (b)

Figura 26: Fragmento de uma mamografia (b) e seu histograma (b). Nota-se que o histograma não possui perfil bimodal, dificultando a escolha inicial do limiar para segmentação.



(a) (b) (c)

Figura 27: Aplicação de limiarização global sobre o fragmento de imagem mamográfica da Figura 26. Em (a), o limiar foi estimado pelo valor médio da distribuição ($L_0 = 0,2955$); em (b), pelo método de Otsu (($L_0 = 0,3137$) e em (c) pelo método da máxima entropia (($L_0 = 0,3843$). Nota-se que, para esta imagem em particular, os métodos da média e de Otsu proporcionaram resultados semelhantes. Os valores de intensidade e de limiar foram normalizados para o intervalo [0,1].

das via diferenciação está associada ao fato de que bordas abruptas não constumam ser frequentes em imagens naturais[9]. Nesse sentido, quão mais suave for a transição entre os níveis de intensidade (vide Figura 28 (b)), menos eficaz será um filtro diferenciador. No caso limite de não haver transição alguma (níveis de intensidade constantes ao longo de certa direção), o resultado da filtragem via diferenciador será nulo.

[9]Entenda-se *natural* no sentido de não-sintética, de forma que padrões espaciais regulares tais como os representados na Figura 20 não sejam predominantes.

Para melhor ilustrar a segmentação por detecção de bordas, tomemos o fragmento de imagem mamográfica mostrado na Figura 29. Este fragmento pode ser considerado um caso emblemático das dificuldades em se detectar estruturas na mama: além da baixa dimensionalidade das estruturas suspeitas, a imagem possui baixo contraste. Um estudo detalhado sobre a questão da detecção em imagens mamográficas pode ser consultado em [20].

Em primeiro lugar, é interessante expandir o histograma desta imagem, isto é, aumentar o contraste da imagem. Isso pode ser feito atra-

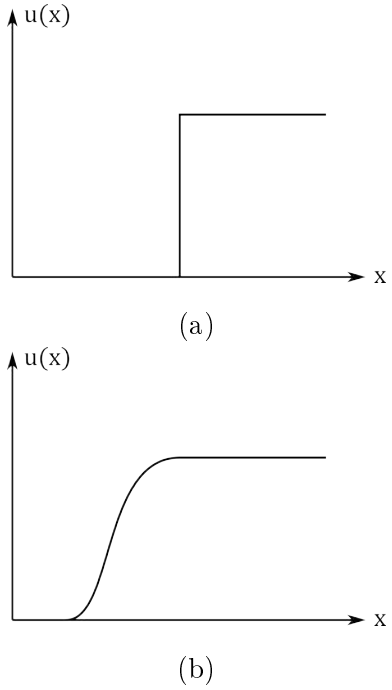


Figura 28: Modelos para borda em uma dimensão. Em (a) é mostrado a transição entre níveis de intensidade uma borda ideal, completamente descontínua. Em (b) é mostrado um modelo mais realista, que leva em consideração a possibilidade de a transição entre níveis de intensidade não ocorrer de forma abrupta, diminuindo a efetividade dos detectores.

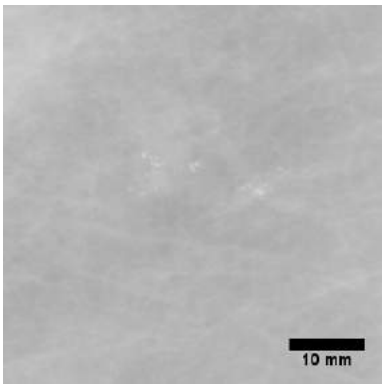
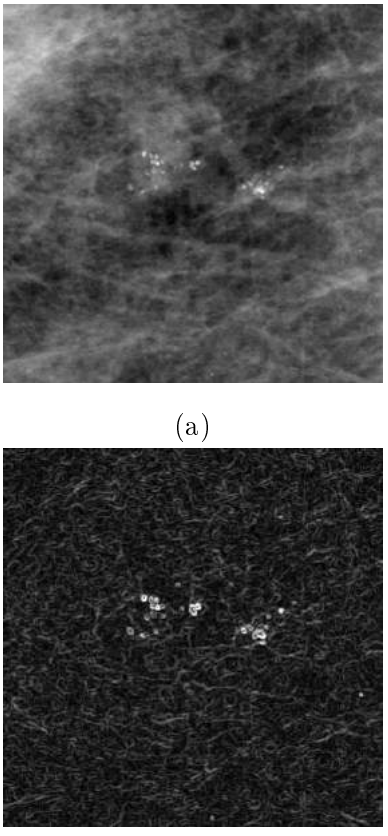


Figura 29: Fragmento de imagem mamográfica contendo um agrupamento de microcalcificações nas imediações do centro da imagem. Note-se que esta imagem possui baixa relação de contraste, motivando a aplicação da operação de alargamento linear.

vés do alargamento linear de contraste, definido na Equação (5). Feito isso, submete-se a imagem resultante a duas filtragens – uma em cada direção – segundo um filtro diferenciador. Isto é feito visando a obtenção da magnitude do gradiente, tal como definido na Equação (30). A Figura 30 ilustra esta ação. O filtro diferenciador usado foi o de Prewitt, tal como definido na Equação (28).



(a)

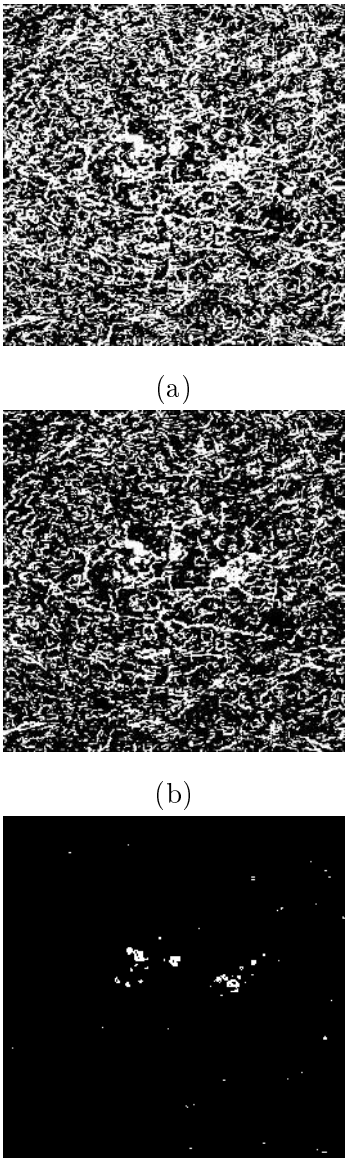
(b)

Figura 30: Etapas da segmentação por detecção de bordas da imagem retratada na Figura 29. Em (a) é mostrado o resultado da operação de alargamento linear de contraste da imagem original e em (b) é mostrada a magnitude do gradiente, calculada usando-se o filtro de Prewitt. Nota-se que, ao final destas duas etapas, as microcalcificações estão significativamente mais evidenciadas do que na imagem original.

O resultado da detecção de bordas mostrado na Figura 30 (b) mostra que, de fato, filtros

diferenciadores são eficientes em evidenciação de descontinuidades. Contudo, a *segmentação* completa somente será atingida após a limiarização da imagem processada. Isto se faz necessário de forma a proporcionar a separação objeto/fundo, tal como discutido no início desta Seção.

A criticalidade do processo de limiarização fica evidente quando comparamos as imagens resultantes de diferentes formas de cálculo do limiar, tal como ilustrado na Figura 31. Nela fica evidente o papel desempenhado pelo limiar de separação. Vê-se que o cômputo deste limiar pode variar bastante segundo a técnica escolhida, e isso traz consequências importantes sobre o sinal resultante. Conforme será visto adiante, o resultado da segmentação é utilizado para as etapas de extração de informação (tal como esquematizado na Figura 24). Portanto, a obtenção de um sinal ruidoso ou inadequado na etapa de segmentação prejudica a eficiência das etapas posteriores. Pode-se, sem sombra de dúvida, afirmar que a segmentação é a etapa mais importante da cadeia de processamento de uma imagem.



(a) (b) (c)

Figura 31: Resultados da binarização da imagem com bordas realçadas mostrada na Figura 30: em (a), o *threshold* foi calculado segundo o valor médio da distribuição. Em (b), foi calculado pelo método de Otsu e em (c) pelo método da máxima entropia. Nota-se claramente os efeitos de uma escolha inadequada na forma de cômputo do limiar: enquanto os métodos do valor médio e de Otsu evidenciam uma quantidade significativa de ruído, o método da máxima entropia permite uma segmentação muito mais eficiente dos objetos de interesse.

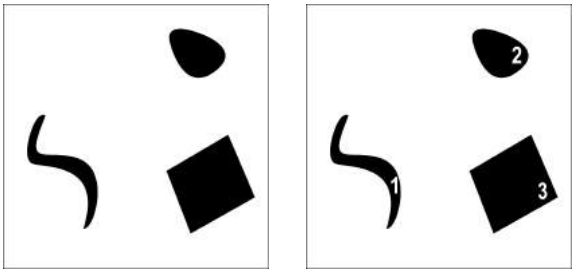
4 Operações de alto nível: labeling e clustering

Os tópicos discutidos até agora enquadram-se como processamento de baixo nível (vide esquema da Figura 24), pois as técnicas discutidas fazem uso da informação contida nos próprios pixels da imagem. As operações a serem discutidas nesta seção devem ser encaradas como operações de *alto-nível*, pois fazem uso de informação em nível mais abstrato do que aquela proporcionada pelos pixels da imagem. Estas operações, *labeling* e *clustering*, operam sobre a imagem segmentada a fim de proporcionar informações sobre as formas presentes na imagem, e como estas se agrupam. A seguir, serão apresentados os conceitos básicos destas classes de operações. Referências sobre clustering podem ser acessadas nos trabalhos

de Pedrini e Schwartz [19] e principalmente nos trabalhos de Jain *et al.* [13] e Jain e Dubes [14]. Referências sobre labeling podem ser encontradas nos trabalhos de Rosenfeld e Kak [23], Gonzalez *et al.*[9], He *et al.* [12] e Gotoh *et al.* [10].

4.1 Labeling

De posse de uma imagem segmentada, o passo natural a ser seguido é a obtenção de informações sobre as formas presentes nos objetos constantes nesta imagem. Medidas diversas podem ser realizadas, tais como cômputo de áreas, rugosidade, perímetro etc. Estas medidas fazem parte do conjunto dos *descritores* de forma dos objetos estudados e serão discutidas mais adiante. O que se deve compreender é que, para que qualquer medida seja realizada sobre objetos internos à imagem, estes devem ser reconhecidos. Isto é, os objetos presentes na imagem devem ser inicialmente *rotulados*. Usualmente, usamos o termo em inglês, *labeling*. Uma ilustração deste conceito é mostrada na Figura 32. A seguir, será feita breve discussão sobre o processo de labeling segundo a técnica básica de *two-raster scan* [23, 12], devido à sua simplicidade didática.

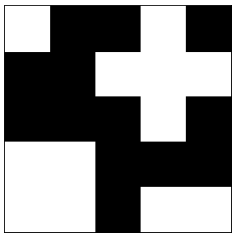


(a) (b)

Figura 32: Exemplificação do conceito de *labeling*: em (a) é apresentada uma imagem segmentada contendo três formas arbitrárias. Em (b) é dado um *rótulo* para cada uma destas formas, de modo que seja possível em etapas posteriores do processamento identificar estas estruturas e extrair informações descritivas diversas.

Para exemplificar o processo de labeling, tomemos a imagem apresentada na Figura 33. Nesta figura temos, além da imagem propriamente dita, sua representação matricial, em que pixels em branco correspondem ao valor 1 e pixels pretos correspondem a 0. Nota-se que há quatro formas distintas na imagem. Conforme discutido nos parágrafos anteriores, o objetivo é rotular estas formas.

O processo de labeling pode ser entendido como uma varredura sobre as linhas da imagem com uma máscara previamente definida, cujo objetivo é delimitar regiões da imagem e permitir a rotulação dos elementos evidenciados. Contudo, pode ocorrer de um pixel ser atribuído a mais de um rótulo, ou *label*. Por isso, uma segunda varredura é feita, corrigindo estas ambiguidades. Convém ressaltar que a forma da máscara varia conforme a vizinhança é definida: se a vizinhança for do tipo N_4 a máscara usada para varredura abrange o pixel em questão, seu vizinho imediato à esquerda e seu vizinho imediato superior; ao passo que em um esquema de vizinhança N_8, são abrangidos –adicionalmente aos pixels à esquerda e acima– os pixels das diagonais superiores. Estas máscaras estão ilustradas na Figura 34.

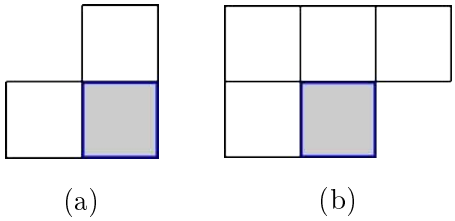


$$u = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \end{bmatrix}$$

(a) (b)

Figura 33: Em (a): Imagem a ser usada para exemplificação de labeling sobre vizinhanças N_4 e N_8. Em (b) é apresentada sua forma matricial, em que 1 equivale a branco e 0 a preto.

Conforme discutido, o processo de labeling consiste em realizar uma varredura na imagem atribuindo labels aos pixels. Na Figura 35 são mostrados os resultados da primeira var-



(a) (b)

Figura 34: Máscaras para varredura no processo de labeling. Em (a), é mostrada a máscara usada para varreduras em um esquema de vizinhança N_4, em que apenas os pixels superior e à esquerda do pixel atual são abrangidos. Em (b), é mostrada a máscara para um esquema N_8, em que – adicionalmente aos pixels abrangidos no esquema anterior– são abrangidos os pixels diagonais superiores. As posições associadas ao pixel principal estão destacadas em ambas as máscaras.

$$\mathcal{L}_{N_4} = \begin{bmatrix} a & & & b & \\ & & c & b,c & b,c \\ & & & b,c & \\ d & d & & & \\ d & d & & e & e \end{bmatrix}$$

(a)

$$\mathcal{L}_{N_8} = \begin{bmatrix} a & & & b & \\ & & b & b & b \\ & & & b & \\ c & c & & & \\ c & c & & d & d \end{bmatrix}$$

(b)

Figura 35: Primeira varredura no processo de labeling para a imagem retratada na Figura 33: em (a) é mostrada a varredura para um esquema de vizinhança N_4, ao passo que em (b) é mostrado o resultado da varredura para um esquema N_8. Nota-se que há uma equivalência entre os rótulos b e c no esquema N_4, pois estes dizem respeito aos mesmos elementos conectados. Já no caso do esquema N_8, não há equivalência entre labels, e o processo pode ser finalizado.

redura, em ambos esquemas N_4 e N_8, para labeling da imagem retratada na Figura 33. Cada label é representado por um caractere que o difere de outros rótulos.

Pode ser verificado na Figura 35 que ocorre uma equivalência entre os labels b e c no esquema N_4, mostrando a necessidade da segunda varredura: é nesta etapa que as equivalências são resolvidas e o processo é finalizado. A forma como a segunda varredura ocorre determina a eficiência do algoritmo envolvido [10, 12, 23].

Uma vez rotulados, os objetos segmentados da imagem podem ser usados para extração de informação quantitativa, proporcionada pelo uso de descritores, a serem discutidos a seguir.

4.2 Descritores

Conforme discutido previamente, para que informações quantitativas possam ser extraídas das formas segmentadas previamente, é necessário que estas formas sejam identificadas por meio de rótulos. O passo seguinte consiste na extração de informação propriamente dita. Ou seja, conhecida uma forma qualquer dentro da imagem, que atributos podemos levantar sobre este objeto?

Um *descritor* de uma forma pode ser compreendido como um atributo representativo da forma analisada. Usualmente, divide-se os descritores em duas grandes classes:

- Descritores de borda;
- Descritores de região.

A primeira classe diz respeito a descrição da *forma* do objeto analisado, levando-se em conta seu *contorno*. Já na segunda, a *área* da forma também é levada em consideração no cômputo das grandezas de interesse. Nos ateremos a discutir alguns descritores de região; isso é feito pois existe uma variedade enorme de descritores e seria impossível abordá-la completamente em um trabalho compacto. Para aprofundamento, recomenda-se os trabalhos de Zhang e Lu [29], Pedrini e Schwarz [19] e Pavlidis [18].

Para exemplificar o uso de descritores, será usada a imagem sintética mostrada na Figura 36. Esta imagem contém 3 objetos: um círculo de raio igual a 200 px; um quadrado de lado igual a 270 px e uma forma arbitrária. A imagem possui resolução de 20px/mm, o que permite a conversão de medidas em pixels para medidas em milímetros. A seguir serão definidos três descritores básicos a serem usados no exemplo.

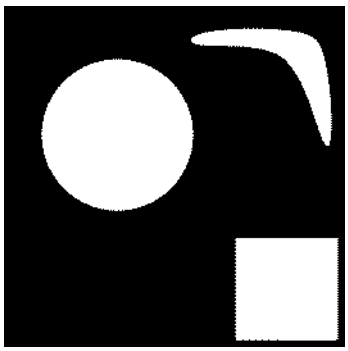


Figura 36: Imagem sintética a ser usada para exemplificar a utilização de descritores baseados em região.

Área: Dada uma imagem *binária*, u, com dimensões M × N, sua área é definida por:

$$\mathrm{A}(u) = \sum_{i=1}^{\mathrm{M}} \sum_{j=1}^{\mathrm{N}} u_{ij} \tag{32}$$

No caso da imagem mostrada na Figura 36, a área total é igual a 229328 px[10] . A Equação (32) anterior pode ser particularizada para regiões internas à imagem, dado que a área total da imagem é a soma das áreas que a compõem. Sendo assim, cada forma presente em uma imagem pode ter sua área estimada, bastando aplicar a expressão (32) sobre os índices que compreendem a estrutura de interesse. Para fins de tabulação, rotulemos o círculo como forma I, o quadrado como forma II e a forma arbitrária como III. Na tabela 1 são mostradas as áreas calculadas para as formas presentes na imagem retratada na Figura 36. Como a resolução da imagem é conhecida, é possível estimar a área real dos objetos na imagem. As formas I e II podem ter suas áreas estimadas facilmente, devido à sua simplicidade. Quando comparamos os valores calculados via Equação (32) com os previstos matematicamente, vemos que - de fato - a estimativa está dentro de uma faixa pequena de erro. O valor previsto para a área da forma I (círculo) é de aproximadamente 125663 px, que difere cerca de 0,19% do valor estimado via descritor de área. A área da forma II (quadrado) é de 72900 px, valor idêntico ao estimado pelo descritor de área.

ID forma	**Área (px)**	**Área (mm²)**
I	125420	313,55
II	72900	182,25
III	31008	77,52

Tabela 1: Cômputo das áreas dos objetos constantes na imagem da Figura 36.

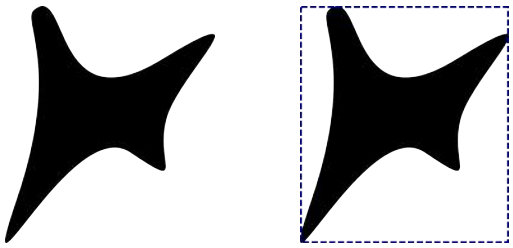
Uma aplicação típica do descritor área é a chamada *filtragem por área*. Este procedimento mantém (ou elimina) formas na imagem que estejam dentro de um certo intervalo. Por exemplo, é possível remover de maneira simples a maior forma presente em uma imagem.

Para finalizar as considerações sobre o descritor área, deve-se ressaltar que este descritor é invariante a translações e rotações das formas[11], mas não é invariante sobre transformações de escala (ampliação, redução, deformação etc.).

[10] Deve-se notar que, tal como definida na Equação (32), a área de uma região qualquer possui unidades de pixel, ao invés de pixel quadrado (px^2). Isso é devido ao fato de que o pixel, em uma imagem binária, pode ser tomado como um elemento de área. Entretanto, quando conversões entre unidades reais e pixels ocorrerem, deve-se atentar para a potência omitida. Por exemplo, um quadrado de lado 200 px tem sua área igual a 40 000 px; em uma imagem com resolução de 2px/mm, ele teria uma área real de 10 000 mm^2.

[11] Os valores do descritor área podem variar um pouco em função de rotações devido à natureza discreta dos dados considerados.

Retângulo envolvente: Dada uma forma $\mathcal{S}$ qualquer pertencente à imagem u, definimos o retângulo envolvente, ou *bounding box*, de $\mathcal{S}$ como sendo o menor retângulo que a contém. Na Figura 37 pode ser vista uma ilustração do uso deste descritor. Na Figura 38 pode



(a) (b)

Figura 37: Ilustração do uso do descritor retângulo envolvente: em (a) é mostrada uma forma $\mathcal{S}$ arbitrária, e em (b) é mostrado também seu retângulo envolvente. De um modo geral, esse descritor pode ser caracterizado pelas magnitudes de sua altura e largura.

ser vista a superposição dos retângulos envolventes às formas apresentadas na imagem da Figura 36

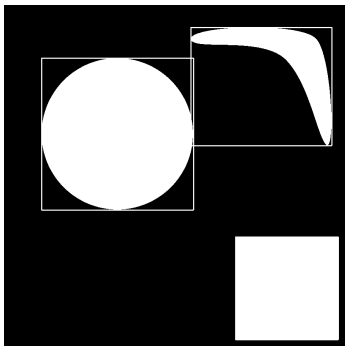


Figura 38: Superposição dos retângulos envolventes sobre as formas apresentadas na Figura 36. Naturalmente, o retângulo envolvente à forma II (quadrado) coincide com as suas bordas.

O descritor retângulo envolvente possui como quantidades de interesse as suas coordenadas [12] e suas largura e altura. O que este descritor proporciona é a transformação de uma geometria qualquer (tal como a forma III na Figura 36) em uma geometria retangular. Quando a forma do objeto não é relevante para a análise a ser realizada, esta simplificação facilita o processo. Isto é particularmente útil quando tratamos da *localização* de objetos na cena. Um exemplo desse tipo de transformação pode ser visualizado na Figura 39. Como o descritor retângulo envolvente leva em

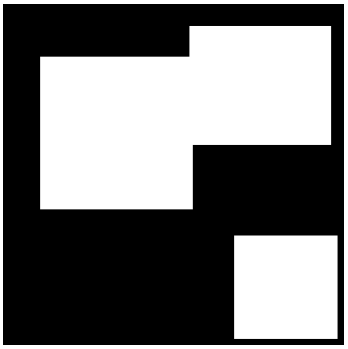


Figura 39: Exemplo de transformação de formas: é mostrada a transformação das formas constantes em na Figura 36 para uma geometria mais simples. Este tipo de aplicação é particularmente útil em detecção de objetos em uma cena.

consideração as dimensões da forma associada, bem como sua localização, é fácil perceber que ele não é invariante sobre quaisquer transformações de escala, translação, rotação ou deformação.

Projeções: Seja u uma imagem com dimensões $\mathrm{M} \times \mathrm{N}$. Definimos suas projeções nas direção x e y, respectivamente, por:

$$\begin{cases} \mathrm{P_x}\left(u\right) = \left(\sum\limits_{i=1}^{\mathrm{M}} u_{i1}, \sum\limits_{i=1}^{\mathrm{M}} u_{i2}, \ldots, \sum\limits_{i=1}^{\mathrm{M}} u_{i\mathrm{N}}\right); \text{ e} \\ \mathrm{P_y}\left(u\right) = \left(\sum\limits_{j=1}^{\mathrm{N}} u_{1j}, \sum\limits_{i=1}^{\mathrm{N}} u_{2j}, \ldots, \sum\limits_{i=1}^{\mathrm{N}} u_{\mathrm{Mj}}\right). \end{cases} \tag{33}$$

O que a Equação (33) mostra é que a projeção da imagem em uma direção consiste em um sinal 1-D cujas entradas carregam informação

[12] As coordenadas podem ser de um dos vértices do retângulo ou de seu centróide.

sobre as variações da imagem naquela direção. A descrição de uma forma por suas projeções difere de outras técnicas pois gera um sinal a ser analisado, e não um simples escalar (tal como o descritor de área visto anteriormente). Sua utilidade reside no fato de que o sinal resultante da projeção pode ser compreendido como um tipo de assinatura unidimensional de uma forma plana[13]. Na Figura 40 podem ser visualizadas as projeções em x e em y das formas constantes na imagem apresentada na Figura 36.

4.3 Clustering

As duas seções anteriores tiveram por objetivo explicitar como podemos *identificar* formas em uma imagem e como podemos *extrair/medir* atributos particulares destas formas. O passo seguinte consiste em entendermos como as formas constantes em um imagem podem ser *agrupadas*.

O primeiro passo para se estabelecer um processo de clustering sobre um conjunto de dados consiste em se definir quais *atributos* da forma serão comparados. Por exemplo, é preciso definir se as formas serão agrupadas por semelhança de área, por proximidade de seus centros de massa etc. Chamamos de *vetor de características* a grandeza $x = (x_1, x_2, \ldots, x_k)$ composta por k medidas sobre os objetos de interesse. É importante notar que cada amostra a ser avaliada terá um vetor de características associado.

A representação espacial das características de cada amostra se dá pela formação do *espaço de características*. Se cada vetor de características possuir k coordenadas, o espaço gerado será k-dimensional. Um caso simples e que permite visualização imediata é o caso 2-dimensional, i.e., em que cada vetor de características é da forma $x = (x_1, x_2)$. A Figura 41 ilustra este conceito. Deve-se ter em mente que os vetores de características podem nada ter a ver com a representação física da imagem. Nesse sentido, o espaço de características não guarda informação da configuração espacial das formas nas imagens, exceto quando esse tipo de informação é usado como característica. É evidente que o processo de clustering ocorre via medidas de semelhança. Portanto, é necessário que se defina, *a priori*, qual métrica será usada. Dentre as diversas opções, a distância Euclidiana surge como opção mais natural. Dados dois vetores de características, $x_\alpha = (x_{\alpha,1}, \ldots, x_{\alpha,k})$ e $x_\beta = (x_{\beta,1}, \ldots, x_{\beta,k})$, a distância Euclidiana entre eles é definida por

$$d(x_\alpha, x_\beta) = \sqrt{\sum_{l=1}^{k} (x_{\alpha,l} - x_{\beta,l})^2} = \|x_\alpha - x_\beta\|_2 . \tag{34}$$

4.3.1 Formas de agrupamento

Estabelecido o espaço de características, é necessário definir a forma – ou classe de algoritmo – a ser usada para proporcionar o agrupamento dos dados. Duas abordagens básicas para clustering são os algoritmos hierárquicos e os particionais [13]. Neste trabalho apenas algoritmos hierárquicos serão discutidos.

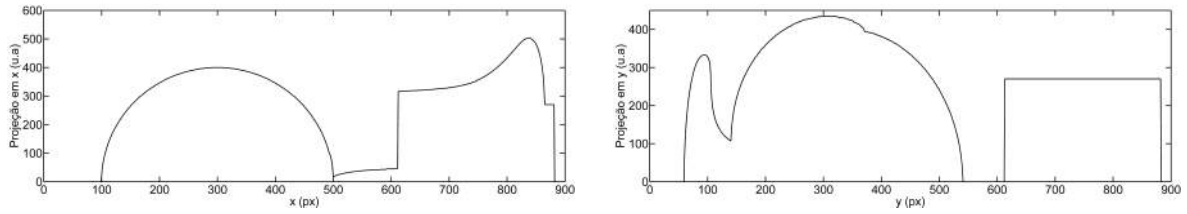
Algoritmos hierárquicos

Para discutir esta classe de algoritmos, é necessária a apresentação do conceito de *matriz de distâncias* associada a um espaço de características. Dado um conjunto com N pontos nos espaço de características, definimos a matriz de distâncias $\mathcal{D}$ como sendo

$$\mathcal{D} = \begin{bmatrix} d(1,1) & d(1,2) & \ldots & d(1,N) \\ d(2,1) & d(2,2) & \ldots & d(2,N) \\ \vdots & \vdots & \ddots & \vdots \\ d(N,1) & d(N,2) & \ldots & d(N,N) \end{bmatrix},$$

em que $d(a,b)$ é a distância entre os pontos a e b. Em função das propriedades de uma norma, temos que $d(a,a) = 0$ e $d(a,b) = d(b,a)$. Estas

[13] Existem outros descritores específicos para obtenção de *assinaturas* de uma forma. Para maior aprofundamento, vide Pedrini e Schwartz [19].



(a) (b)

Figura 40: Exemplo de obtenção das projeções da imagem mostrada na Figura 36. Em (a) é mostrada sua projeção sobre a direção x (horizontal) e em (b) é mostrada a projeção em y (vertical). Diferentemente de outros descritores, a projeção de uma imagem gera um sinal unidimensional, cujas variações correspondem a assinaturas das formas constantes na imagem. Contudo, como pode ser visto pelo exemplo, superposições podem ocorrer, de forma que as assinaturas carregam contribuições de mais de um objeto na cena.

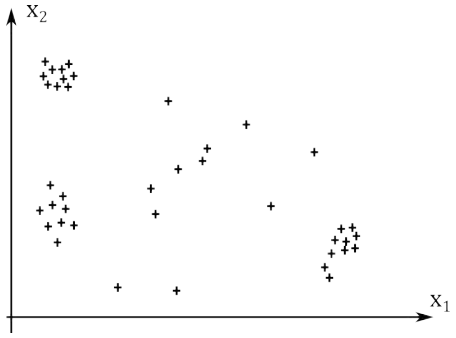


Figura 41: Espaço de característica bidimensional. Esse tipo de representação permite a análise que leva os elementos analisados a serem agrupados. Nota-se que há regiões mais compactadas, sugerindo um agrupamento inicial.

duas características fazem com que a diagonal principal de $\mathcal{D}$ seja nula e que esta matriz seja simétrica. Logo, podemos reescrever a matriz de distâncias como sendo

$$\mathcal{D} = \begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & \ddots & \\ d(N,1) & d(N,2) & d(N,3) & \dots & 0 \end{bmatrix}. \tag{35}$$

Um agrupamento hierárquico *aglomerativo* opera uma matriz de distâncias, atualizando sucessivamente uma estrutura de componentes aninhadas. Isto é, as estruturas mais próximas vão sendo agrupadas, até que todo o conjunto que compõe o espaço de características seja representado por um único cluster. Este conceito é ilustrado na Figura 42. Uma outra abordagem básica consiste em considerar inicialmente todas as amostras como componentes de um mesmo cluster e submetê-lo sucessivamente a fragmentações. Nesse caso, o algoritmo é chamado *divisivo*.

O processo de clustering hierárquico aglomerativo pode ser compreendido segundo o algoritmo mostrado na Figura 43, adaptado de Jain e Dubes [14].

Uma vez definido o espaço de características, é mandatório definir o critério de escolha do *representante de grupo*, que é o dado quantitativo que definirá a atualização da matriz de distâncias (passo 4 do algoritmo). Para algoritmos hierárquicos, há três abordagens básicas para determinação do representante de grupo [13, 14, 19], a serem descritas a seguir.

Ligação simples: A distância entre dois grupos, $\mathcal{A}$ e $\mathcal{B}$ é tomada como sendo a menor distância entre os pares formados por elementos dos dois grupos, i.e.,

$$d(\mathcal{A},\mathcal{B}) = \min_{i,j\in\mathcal{A}\times\mathcal{B}}\{d(i,j)\}, \tag{36}$$

em que $\mathcal{A}\times\mathcal{B}$ representa o produto cartesiano entre os conjuntos $\mathcal{A}$ e $\mathcal{B}$.

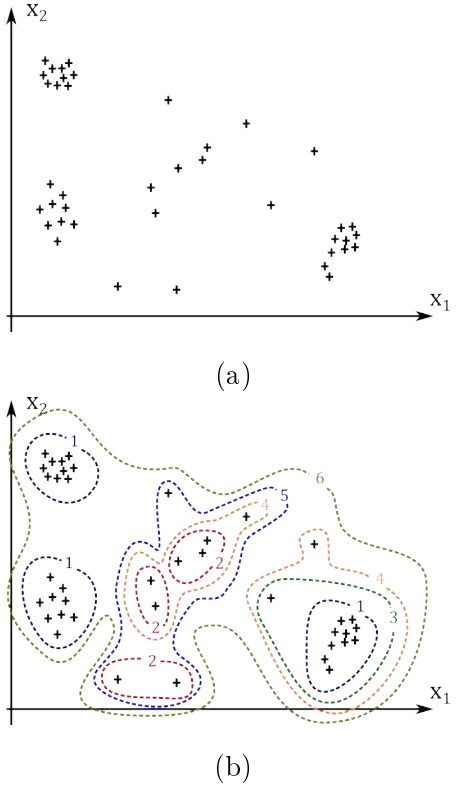


Figura 42: Ilustração do conceito de clustering hierárquico aglomerativo: em (a) é mostrado o espaço de características apresentado na Figura 41 e em (b) é mostrado como os objetos submetidos a agrupamento são sucessivamente agrupados. As linhas pontilhadas delimitam um cluster e os números ao longo destas linhas representam em qual etapa o conjunto de dados foi agrupado. Note-se que o produto final do algoritmo é um único cluster que comporta todos os objetos.

Ligação completa: A distância entre dois grupos, $\mathcal{A}$ e $\mathcal{B}$, é tomada como sendo a maior distância entre os pares formados por elementos dos dois grupos, i.e.,

$$d(\mathcal{A},\mathcal{B}) = \max_{i,j\in\mathcal{A}\times\mathcal{B}}\{d(i,j)\}, \tag{37}$$

Centróide: A distância entre dois grupos, $\mathcal{A}$ e $\mathcal{B}$, é definida como sendo a distância Euclidiana entre seus centros de massa, tal como apresentado na Equação (34) :

$$d(\mathcal{A},\mathcal{B}) = \left\|\mathrm{C}(\mathcal{A}) - \mathrm{C}(\mathcal{B})\right\|_2, \tag{38}$$

1. Inicialize a matriz $\mathcal{D}_{\mathrm{N}\times\mathrm{N}}$;
2. Encontre o par de clusters (a,b) tal que $d(a,b) = \min\{d(i,j)\}$ sobre todas as entradas ij de $\mathcal{D}$;
3. Agrupe o par de clusters $(a,b) : (a,b) \to \{a,b\}$;
4. Atualize a matriz $\mathcal{D}$: remova a linha a e a coluna b e calcule $d(\{a,b\},j)$ segundo a norma adotada;
5. Se todos os objetos estiverem agrupados no mesmo cluster, pare. Caso contrário, vá para o passo 2.

Figura 43: Algoritmo básico para agrupamento hierárquico via atualização da matriz de distâncias.

em que

$$\mathrm{C}(\mathcal{X}) = \frac{1}{n}\cdot\left(\sum_{i=1}^{n} x_{i,1}, \sum_{i=1}^{n} x_{i,2}, \ldots, \sum_{i=1}^{n} x_{i,k}\right)$$

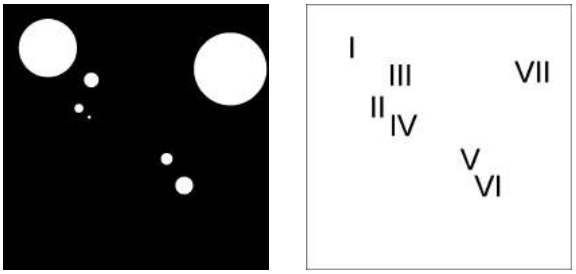
é o centro de massa de um dado cluster $\mathcal{X}$ contendo n elementos.

Para fins de exemplificação, submetamos a imagem retratada na Figura 44 a um processo de clustering hierárquico aglomerativo. Agruparemos as formas presentes na imagem usando como vetores de características as coordenadas dos centros de massa dos objetos na cena.

Na Tabela 2 são mostradas algumas grandezas de interesse para as formas constantes na imagem apresentada na Figura 44. Os vetores de característica usados correspondem aos pares $(x_{\mathrm{CM}}, y_{\mathrm{CM}})$.

Com base na definição de matriz de distâncias e dos dados constantes nos vetores de características, temos que a matriz de distâncias $\mathcal{D}$ para este espaço é dada por

$$\begin{bmatrix} 229,44 & & & & & \\ 182,41 & 104,79 & & & & \\ 272,68 & 46,10 & 126,19 & & & \\ 549,75 & 342,71 & 369,21 & 297,53 & & \\ 654,79 & 441,43 & 475,44 & 395,48 & 107,62 & \\ 621,07 & 528,99 & 471,45 & 504,08 & 372,35 & 423,76 \end{bmatrix}, \tag{39}$$



(a) (b)

Figura 44: Imagem a ser analisada no processo de clustering: em (a) constam 7 formas circulares com raios distintos e em posições tais que não há superposição entre elas. Em (b) são mostrados os rótulos de cada forma. A imagem possui 900x900 pixels e resolução de 20px/mm.

Label	**Área (px)**	$x_{\mathbf{CM}}$ **(px)**	$y_{\mathbf{CM}}$ **(px)**
I	30405	152	149
II	717	257	353
III	1941	299	257
IV	89	292	383
V	1237	554	524
VI	2801	613	614
VII	47409	769	220

Tabela 2: Atributos de área e posição dos centros de massa das formas constantes na imagem apresentada na Figura 44.

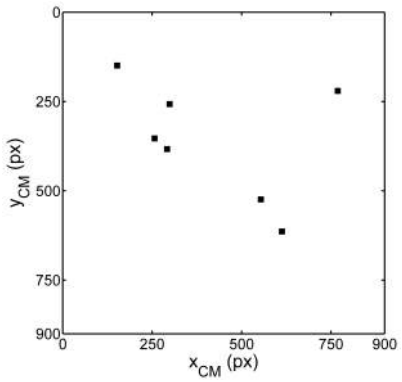


Figura 45: Espaço de características da imagem mostrada na Figura 44, cujos vetores de característica são as coordenadas dos centros de massa das formas presentes na imagem. Os valores destes vetores podem ser consultados na Tabela 2.

em que tanto a diagonal principal quanto as entradas repetidas não são mostradas. Seguindo o algoritmo mostrado na Figura 43, devemos localizar o par de clusters no espaço de características cuja distância é mínima. Analisando a matriz de distâncias (39), vemos que os clusters $\{2\}$ e $\{4\}$ são os mais semelhantes nesta primeira etapa. Portanto, temos a criação de um novo cluster: $\{2\},\{4\} \rightarrow \{2,4\}$. O próximo passo é recalcular a distância entre os clusters restantes e o centróide do cluster $\{2,4\}$. A matriz de distâncias atualizada é dada por

$$\mathcal{D} = \begin{bmatrix} 255,45 & & & & \\ 182,41 & 112,08 & & & \\ 549,75 & 312,30 & 369,21 & & \\ 654,79 & 411,20 & 475,44 & 107,62 & \\ 621,07 & 507,56 & 471,45 & 372,35 & 423,76 \end{bmatrix}.$$

Nessa versão atualizada de $\mathcal{D}$, vemos que os clusters $\{5\}$ e $\{6\}$ devem ser agrupados, ou seja, $\{5\},\{6\} \rightarrow \{5,6\}$. Repetindo a ação anterior, a matriz $\mathcal{D}$ é recalculada em função das distâncias dos clusters restantes ao centróide do cluster recém-criado:

$$\mathcal{D} = \begin{bmatrix} 255,45 & & & \\ 182,41 & 112,08 & & \\ 602,16 & 361,11 & 422,24 & \\ 621,07 & 507,56 & 471,75 & 395,24 \end{bmatrix}.$$

Nesta versão, temos $\{2,4\},\{3\} \rightarrow \{2,3,4\}$. A atualização de $\mathcal{D}$ fica na forma

$$\mathcal{D} = \begin{bmatrix} 227,60 & & \\ 602,16 & 378,91 & \\ 621,07 & 480,33 & 395,24 \end{bmatrix}.$$

A atualização seguinte agrupa os clusters $\{1\}$ e $\{2,3,4\}$. A matriz $\mathcal{D}$ fica

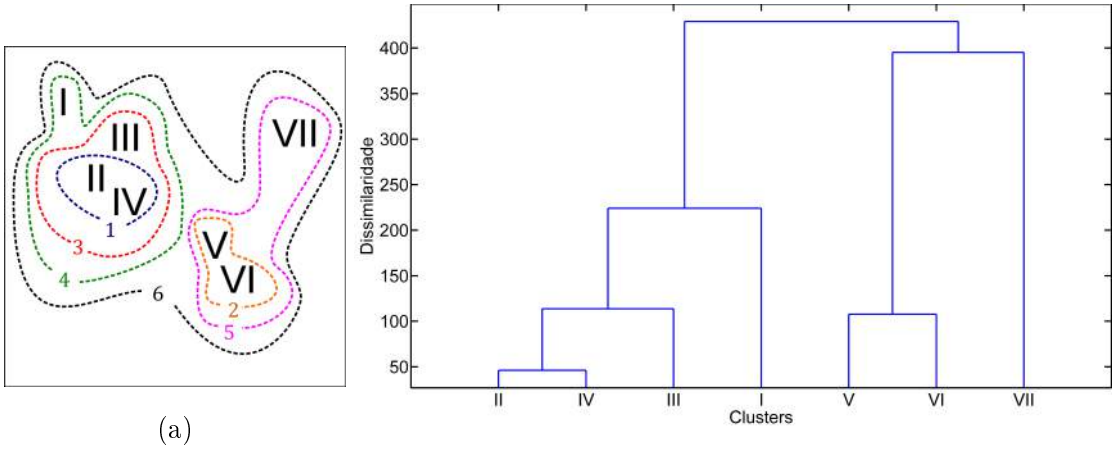
$$\mathcal{D} = \begin{bmatrix} 434,30 & \\ 518,65 & 395,24 \end{bmatrix}.$$

Esta atualização leva ao agrupamento de $\{5,6\}$ e $\{7\}$.

A última atualização leva ao agrupamento de todos os elementos e a forma mínima da matriz de distâncias é atingida:

$$\mathcal{D} = \begin{bmatrix} 425,28 \end{bmatrix}.$$

O processo de clustering pode ser melhor visu-



(a) (b)

Figura 46: Resultado do processo de clustering sobre as formas da imagem mostrada na Figura 44, cujos vetores de características são compostos pelos centros de massa das formas em questão. Em (a) é mostrada uma ilustração sobre os rótulos de cada forma, indicando os sucessivos agrupamentos sofridos pelos objetos. Os números indicam a etapa em que ocorreu o agrupamento. Em (b) é mostrado o dendrograma do processo de clustering em questão. O eixo vertical representa a dissimilaridade (não-semelhança) entre os clusters. Nota-se que quão mais agrupada se torna a coleção de dados, maior a dissimilaridade entre seus constituintes. Isto indica a necessidade do estabelecimento de um critério de parada para evitar que objetos discrepantes entre si sejam agrupados.

alizado por meio de um *dendrograma*, que consiste em uma representação gráfica em forma de árvore. Para este exemplo em particular, o dendrograma tem a forma apresentada na Figura 46. Conforme pode ser observado pelas discussões anteriores e pelo exemplo estudado, fica evidente que é necessário o estabelecimento de um *critério de parada para o algoritmo hierárquico aglomerativo*. Caso contrário, amostras que não guardam relação significativa serão agrupadas. Uma possibilidade intuitiva consiste em fazer com que o processo de clustering ceda quando a dissimilaridade for maior que um certo limiar. Ou seja, caso a dissimilaridade – ou distância – entre dois objetos seja maior ou igual que certo valor, estes objetos não são mais agrupados.

5 Pós-processamento: morfologia matemática

Para finalizar, apresentaremos algumas informações básicas sobre técnicas de processamento derivadas da morfologia matemática. Podemos dizer, de maneira bastante imprecisa, que a morfologia matemática consiste em um conjunto de técnicas que proporcionam um processamento voltado à forma das estruturas analisadas. Em particular, daremos ênfase ao caráter de pós-processamento das técnicas descritas[14]. Isto é, discutiremos algumas técnicas básicas apropriadas para extração de pontos, dilatação de bordas, preenchimento de regiões não-simplesmente conexas etc. Por simplicidade, consideraremos as operações morfológicas como ocorrendo sobre imagens binárias. Ao final da seção, será apre-

[14] Embora a abordagem aqui seja em pós-processamento, é importante notar que a morfologia matemática coompreende toda uma área de estudo, abrangendo muito mais do que mostrado aqui.

sentada a generalização dos conceitos apresentados para imagens monocromáticas.

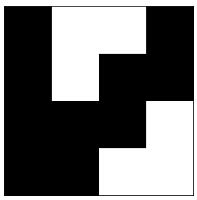
5.1 Conceitos fundamentais

Uma imagem binária u pode ser representada como um subconjunto de $\mathbb{Z}^2$ composto pelas coordenadas em que ela assume o valor 1, isto é,

$$u = \left\{(x, y) \in \mathbb{Z}^2 | I(x, y) = 1\right\}, \tag{40}$$

em que $I(x, y)$ corresponde à intensidade luminosa na posição (x, y). Para fins de exemplificação, são mostradas na Figura 47 uma imagem binária de dimensões 4x4 e sua representação matricial juntamente com os eixos coordenados de $\mathbb{Z}^2$. A representação para essa imagem, em termos de um conjunto seguindo a definição dada na Equação (40) é dada por

$$u = \{(0,0), (0,1), (0,2), (0,3), (1,2), (1,3), (2,1), (2,2), (3,0), (3,1)\}.$$



	0	**1**	**2**	**3**
0	1	0	0	1
1	1	0	1	1
2	1	1	1	0
3	1	1	0	0

(a) (b)

Figura 47: Imagem binária com dimensões 4x4 pixels (a) e sua representação matricial (b). As entradas em negrito separadas por linhas representam as coordenadas no plano $\mathbb{Z}^2$ sobre o qual a imagem está definida.

Conforme pode ser verificado pela definição dada por (40) e visualizado na representação (b) da Figura 47, a representação da imagem depende, necessariamente, do arranjos do sistema de coordenadas. Em particular, a escolha da origem do sistema de referência tem influência nas operações básicas em morfologia.

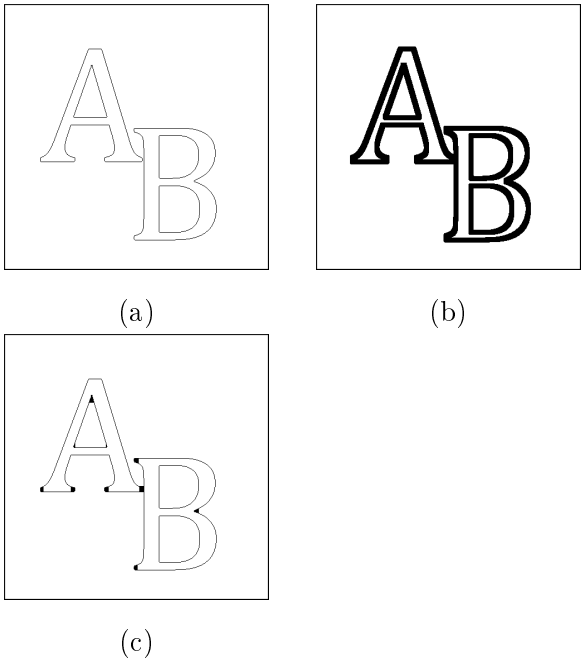
Podemos manipular imagens binárias sem recorrer à abordagem apresentada na Seção 2. Sendo definidas agora como conjuntos, imagens binárias podem ser operadas segundo operações básicas derivadas deste arcabouço teórico. Sejam duas imagens binárias, u e v e $\beta \in \mathbb{Z}$. Derivam diretamente da teoria de conjuntos as operações mostrada na Tabela 3. Em geral, se operamos u com base em v, chamamos esta última imagem de *elemento estruturante*. As duas operações mais básicas em morfologia são a *dilatação* e a *erosão*. Definimos [19, 11] a dilatação e erosão de u por v, respectivamente, como

União	$u \cup v = \{w \| w \in u \text{ ou } w \in v\}$
Interseção	$u \cap v = \{w \| w \in u \text{ e } w \in v\}$
Translação	$u + \beta = \{w + \beta \| w \in u\}$
Reflexão	$\hat{u} = \{-w \| w \in u\}$
Complemento	$u^{\mathrm{C}} = \{w \| w \notin u\}$

Tabela 3: Operações básicas em morfologia matemática derivadas da teoria de conjuntos.

$$\begin{cases} \mathrm{D}(u, v) = u \oplus v = \bigcup\limits_{w \in v} (u + w); \\ \\ \mathrm{E}(u, v) = u \ominus v = \bigcap\limits_{w \in v} (u - w) = \bigcap\limits_{w \in \hat{v}} (u + w). \end{cases} \tag{41}$$

Em termos práticos, a dilatação $u \oplus v$ possibilita o espessamento de bordas de u, segundo a forma do elemento estruturante v. Por outro lado, a erosão $u \ominus v$ causa um afinamento das bordas de u segundo a forma de v. Uma ilustração destas operações pode ser visualizada na Figura 48. Neste exemplo, o elemento estruturante foi tomado como sendo o produto cartesiano $[-4, 4] \times [-4, 4]$, ou seja, $v = \{(i, j) \in \mathbb{Z}^2 | i, j \in [-4, 4]\}$. As operações de dilatação e erosão gozam de diversas propriedades algébricas, tais como associatividade, comutatividade (não-comutatividade e não-associatividade no caso da erosão), dualidade etc. Para maior aprofundamento sugere-se o livro de Pedrini e Schwartz [19] ou o trabalho de Haralick *et al.* [11].



(a) (b)

(c)

Figura 48: Ilustração das operações de dilatação e erosão. Em (a) é mostrada uma imagem sintética cujas bordas foram detectadas e segmentadas. Em (b) é mostrado o resultado da dilatação da imagem binária (a) segundo o elemento estruturante $v = [-4, 4] \times [-4, 4]$. Em (c) é mostrada a erosão da imagem dilatada (b) segundo o mesmo elemento estruturante v. Nota-se claramente o espessamento de bordas quando da aplicação da dilatação, assim como o refino dos contornos quando da aplicação da erosão. Para fins de visualização, as imagens mostradas correspondem aos negativos dos resultados das operações.

5.2 Operações derivadas

5.2.1 Abertura e fechamento

Uma classe grande de operações morfológicas pode ser derivada das diferentes combinações entre dilatação e erosão, tal como definidas na Equação (41). Nos ateremos a expor três operações derivadas – *abertura*, *fechamento* e *preenchimento de regiões*.

Dados uma imagem u e um elemento estruturante v, definimos a abertura e fechamento, respectivamente, por

$$\begin{cases} u \circ v = (u \ominus v) \oplus v \\ u \bullet v = (u \oplus v) \ominus v \end{cases} \tag{42}$$

Tais como definidas na Equação (42), as operações de abertura e fechamento podem ser entendidas como removedores de ruídos, dependendo da convenção adotada (0 para preto e 1 para branco ou 1 para preto e 0 para branco). A remoção de ruídos pode ser entendida analisando-se a operação de abertura: uma vez erodida por v, a imagem u perderá pequenos detalhes que não surgirão quando se dilatar o resultado. A interpretação para o fechamento é análoga. O que permite analisar as operações de abertura e fechamento como quase-equivalentes é a sua relação de dualidade, dada por [19, 11]

$$\begin{cases} (u \circ v)^{\mathrm{C}} = u^{\mathrm{C}} \bullet \hat{v}; \\ (u \bullet v)^{\mathrm{C}} = u^{\mathrm{C}} \circ \hat{v}. \end{cases} \tag{43}$$

Um exemplo de aplicação de abertura e fechamento com vistas à remoção de ruídos é mostrado na Figura 49

5.2.2 Preenchimento de regiões

A morfologia matemática permite que regiões não-simplesmente conexas – isto é, com "buracos" – sejam preenchidas através de um processo iterativo. Uma realização possível [19] é tomar a imagem u como sendo o resultado do processo iterativo

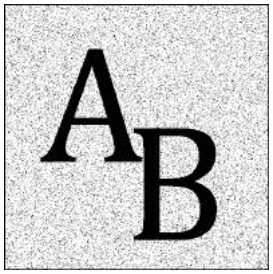
$$u_{t+1} = (u_t \oplus v) \cap u_0^{\mathrm{C}}, \tag{44}$$

em que u_0 é a imagem inicial.

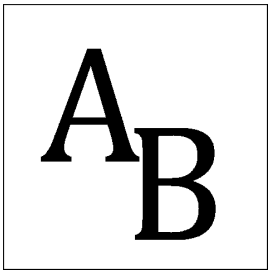
A interpretação do processo iterativo descrito em (44) é bastante simples: dilata-se a imagem e obtém-se a interseção com o complemento da imagem inicial para que não ocorra deformação das bordas. O processo é encerrado quando $u_{t+1} = u_t$. Na Figura 50 pode ser visualizada a aplicação do preenchimento morfológico sobre uma imagem sintética.

5.3 Extensão a imagens monocromáticas

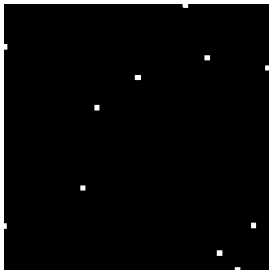
As operações de dilatação e erosão podem ser estendidas para imagens monocromáticas.



(a)



(b)



(c)

Figura 49: Ilustração do conceito de abertura e fechamento morfológicos. Em (a) é mostrada uma imagem binária corrompida por ruído. Em (b) é mostrado o resultado da abertura de (a) segundo o elemento estruturante $v = [-4, 4] \times [-4, 4]$, e em (c), o resultado do fechamento de (a) segundo o mesmo elemento estruturante. Nota-se a efetividade da abertura quanto à remoção de ruídos. Nesse exemplo, o fechamento privilegiou a junção dos pontos de ruído, mas operaria igualmente bem na remoção de ruído caso a imagem original fosse invertida (dualidade). Para melhor visualização, as imagens mostradas são apresentadas como os negativos dos resultados.

Definimos [19, 11] a dilatação e erosão, respectivamente, de u por v por

$$(u \oplus v)(x, y) = \max_{(m,n)\in v} \{u(x - m, y - n) + v(m, n)\}$$

e

$$(u \ominus v)(x, y) = \min_{(m,n)\in v} \{u(x - m, y - n) + v(m, n)\}$$

Uma vez redefinidas as operações básicas de dilatação e erosão, as operações de abertura, fechamento e preenchimento são redefinidas automaticamente, pois seguem a mesma definição que suas versões para imagens binárias. Uma ilustração da aplicação de operações morfológicas sobre imagens monocromáticas pode ser vista na Figura 51.



(a)



(b)

Figura 50: Ilustração do preenchimento morfológico. Em (a) é mostrada uma imagem sintética de dois caracteres com regiões a serem preenchidas. Em (b) é mostrado o resultado do preenchimento, ilustrando que o processo iterativo descrito na Equação (44) opera segundo o esperado, fechando as regiões. Para fins de visualização, as imagens mostradas são tomadas como sendo o negativo das originais.

Agradecimentos

Os autores gostariam de registrar seus agradecimentos a Marcelo Lobato Martins do Departamento de Física da Universidade Federal de Viçosa e a Sérgio José Barbosa Duarte, do Centro Brasileiro de Pesquisas Físicas; seus comentários e opiniões nos motivaram a dar sequência na realização destas notas.

Referências

[1] AUBERT, G.; KORNPROBST, P. *Mathematical problems in image processing: partial differential equations and the calculus of variations.* (Applied mathematical sciences . v. 147). New York: Springer, 2001. 286 p.

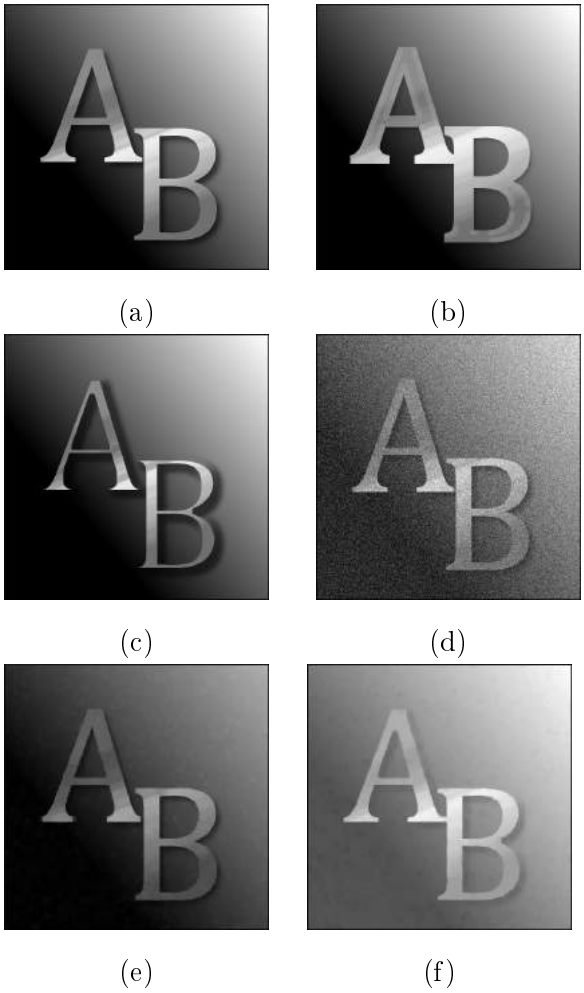


Figura 51: Ilustração da extensão da morfologia matemática a imagens monocromáticas. Em (a) é mostrada uma imagem sintética a ser processada. Em (b) e (c) são mostradas sua dilatação e erosão, respectivamente, pelo elemento estruturante $v = \{v(x, y) = 1 \forall (x, y) \in [-4, 4] \times [-4, 4]\}$. Em (d) é mostrada uma versão de (a) corrompida com ruído, e em (e) e (f) são mostradas sua abertura e fechamento, respectivamente, com o mesmo elemento estruturante anterior.

[2] BARCELOS, C.A.Z. *Restauração e análise de imagens via equações diferenciais parciais*. **Tendências em matemática aplicada e computacional**. v.3. n. 2. 1–13. 2002.

[3] CANNY, J. *A computational approach to edge detection*. **IEEE Transactions on Pattern Analysis and Machine Intelligence**. v. 8. n. 6. 679–698. 1986.

[4] CHANG, C. I.; DU, Y.; WANG, J.; GUO, S. M.; THOUIN, P. D. *Survey and comparative analysis of entropy and relative entropy thresholding techniques*. **IEEE proceedings on vision, image and signal processing**. v. 153. n. 6. 837–850. 2006.

[5] CUMINATO, J.A.; MENEGUETE JR., M. *Discretização de equações diferenciais parciais*. 1 ed. Rio de Janeiro: SBM, 2013. 379p.

[6] DeGROOT, M.H.; SCHERVISH, M. J. *Probability and statistics*. 4 ed. Boston: Pearson, 2012. 893 p.

[7] DINIZ, P.S.R.; SILVA, E.A.B. da; NETTO, S.L. *Digital signal processing - system analysis and design*. 1 ed. UK: Cambridge University Press. 2002.

[8] GONZALEZ, R. C.; WOODS, R. E. *Digital image processing*. 2 ed. New Jersey: Prentice Hall, 2002. 793 p.

[9] GONZALEZ, R. C.; WOODS, R. E.; EDDINS, S. L. *Digital image processing using MATLAB®*. 2 ed. USA: Gatesmark Publishing, 2009. 827 p.

[10] GOTOH, T.; OHTA, Y.; YOSHIDA, M.; SHIRAI, Y. *Component labeling algorithm for video rate processing*. **Proceedings SPIE, Advances in Image Processing**. v. 0804. 217–224. 1987.

[11] HARALICK, R. M.; STERNBERG, S. R.; ZHUANG, X. *Image analysis using mathematical morphology*. **IEEE Transactions on Pattern Analysis and Machine Intelligence**. v. 9. n. 4. 532–550. 1987.

[12] HE, L.; CHAO, Y.; SUSUKI, K.; WU, K. *Fast connected-component labeling*.

Pattern Recognition. v. 42. n. 9. 1977–1987. 2009.

[13] JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. *Data Clustering: A Review.* **ACM Computing Surveys**. v. 31. n. 3. 264–323. 1999.

[14] JAIN, A.K.; DUBES, R.C. *Algorithms for clustering data*(Prentice-Hall advanced reference series). 1 ed. New Jersey: Prentice-Hall. 1988.

[15] KAPUR, J. N.; SAHOO, P. K.; WONG, A. K. C. *A new method for gray-level picture thresholding using the entropy of the histogram.* **Computer vision, graphics and image processing**. v. 29. 273-285. 1985.

[16] LYONS, R. G. *Understanding digital signal processing.* 1 ed. Massachusetts: Pearson Education, 1997. 517p.

[17] OTSU, N. *A threshold selection method from gray-level histograms.* **IEEE transactions on systems, man, and cybernetics**. v. 9. n. 1. 62–66. 1979.

[18] PAVLIDIS, T. *Algorithms for shape analysis of contours and waveforms.* **IEEE Transactions on Pattern Analysis and Machine Intelligence**. v. 2. n. 4. 301–312. 1980.

[19] PEDRINI, H.; SCHWARTZ, W.R. *Análise de imagens digitais.* São Paulo: Thomson Learning, 2008. 508 p.

[20] PERSECHINO, A. *Influência da difusão não-linear na detecção de calcificações em imagens mamográficas.* Dissertação (Mestrado em física com ênfase em instrumentação científica). Centro Brasileiro de Pesquisas Físicas. 2015.

[21] PREWITT, J. M. S. *Object enhancement and extraction.* In: LIPKIN, B. S.; ROSENFELD, A. (Edits.). **Picture processing and psichopictorics**. New York: Academic, 1970. p. 75–149.

[22] RASBAND, W. S. *ImageJ.* National Institutes of Health, Bethesda, Maryland, USA, `http://imagej.nih.gov/ij/`, 1997-2014. Acesso em Novembro/2014.

[23] ROSENFELD, A.; KAK, A.C. *Digital picture processing*(Computer science and applied mathematics series). vols. 1 e 2. New York: Academic Press. 1982.

[24] SAHOO, P. K.; SOLTANI, S.; WONG, A. K. C. *A survey of thresholding techniques.* **Computer vision, graphics, and image processing**. v. 41. 2 ed. 233–260. 1988.

[25] SANTORO, A.; MAHON, J. R.; OLIVEIRA, J. U. C. L. de; MUNDIM FILHO, L. M.; SILVA, W. L. P. da; OGURI, V. (org.). *Estimativas e erros em experimentos de física.* 2 ed. Rio de Janeiro: EDUERJ, 2008. 131 p.

[26] THE MATHWORKS, INC. *MATLAB®. The language of technical computing.* Natick, Massachusets, USA. Disponível em `http://mathworks.com/products/matlab`, 1994-2015. Acesso em Junho/2015.

[27] WESZKA, J. S. *Survey of Threshold Selection Techniques.* **Computer Graphics and Image Processing** . v. 7. n. 2. 259–265. 1978.

[28] WITKIN, A.P. *Scale-space filtering: a new approach to multi-scale description.* **Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84**, v. 9. 150–153. 1984.

[29] ZHANG, D.; GUOJUN, L. *Review of shape representation and description techniques.* **Pattern Recognition**. v. 37. 1 ed. 1–19. 2004

[30] ZIOU, D.; TABBONE, S. *Edge Detection Techniques - an overview*. **International Journal of Pattern Recognition and Image Analysis**. v. 8. 537–559. 1998.

A Implementações

Neste apêndice são apresentadas implementações em MATLAB®[26] de dois tópicos discutidos ao longo do texto:

- limiarização; e
- filtragem no domínio das frequências.

Isso é feito para que o estudante possa ter um referencial inicial de como trabalhar com a programação em imagens. Naturalmente, existem variações no modo de programar, de forma que tais como expostas aqui, as implementações são apresentadas de maneira mais didática, explorando o caráter *vetorial* da linguagem MATLAB®. Ressalte-se que o MATLAB® é proprietário, mas seu uso é difundido em processamento de imagens há anos em função de suas sintaxe e funcionalidades específicas. Alternativas gratuitas ao MATLAB® são:

1. SCILAB[15]; e
2. ImageJ [22].

A linguagem de programação SCILAB se assemelha muito ao MATLAB®, mas com a diferença que é um projeto livre e *open source*. Por sua vez, o ImageJ, também livre e open source, corresponde a um aplicativo consagrado na área de *análise de imagens*, possuindo uma gama gigantesca de funcionalidades.

A.1 Obtenção do threshold para binarização

O ponto de partida é o histograma de uma imagem, tal como discutido na Seção 1. Recordando a Equação (3), temos:

$$h(k) = \frac{p_k}{\mathrm{M} \times \mathrm{N}}.$$

em que $\frac{p_k}{\mathrm{M}\times\mathrm{N}}$ é a probabilidade de ocorrência do k-ésimo nível de intensidade. Ou seja, o histograma da imagem é uma aproximação à função de densidade de probabilidade (discreta) dos seus níveis de intensidade. Dito isto, temos que todos os parâmetros estatísticos de interesse podem ser obtidos a partir do histograma.

Valor médio

Dado o histograma h de uma imagem com N níveis de cinza, o *valor médio* para os níveis de intensidade é definido como

$$\bar{x} = \sum_{i=1}^{\mathrm{N}} i h(i). \tag{45}$$

Tal como definido na Equação (45), percebemos que $\bar{x}$ corresponde, de fato, ao centro de massa da distribuição aproximada por h. Nesse sentido, se o histograma for simétrico em relação ao ponto i_0 – isto é, $h(i_0 - k) = h(i_0 + k)$ –, teremos $\bar{x} = i_0$. Portanto, o uso do valor médio como threshold é apropriado para distribuições simétricas de níveis de cinza.

Método de Otsu

Em 1979, N. Otsu propôs uma técnica de limiarização global erguida sobre a hipótese de que o histograma da imagem pode ser dividido em duas classes distintas [17]. Este método calcula o threshold L como sendo o nível de intensidade que maximiza uma função critério $\eta(\mathrm{L})$ definida como sendo a razão entre a variância entres classes e a variância global, isto é:

$$\eta(\mathrm{L}) = \frac{\sigma_{\mathrm{B}}^2}{\sigma_{\mathrm{T}}^2}, \tag{46}$$

[15] `http://www.scilab.org`

em que as variâncias entre-classes (σ_{B}^2) e total (σ_{T}^2) têm a forma

$$\begin{cases} \sigma_{\mathrm{B}}^2 &= \omega_0\omega_1(\mu_0\mu_1)^2; \quad \text{e} \\ \\ \sigma_{\mathrm{T}}^2 &= \sum_{i=1}^{\mathrm{L}} (i-\mu_{\mathrm{T}})^2 h_i. \end{cases} \tag{47}$$

Na Equação (47), ω_0 e ω_1 são as probabilidades de ocorrência das classes 0 e 1, e são dadas por

$$\omega_0 = \sum_{i=1}^{\mathrm{L}} h_i \qquad \text{e} \qquad \omega_1 = 1 - \omega_0.$$

As médias das classes 0 (μ_0) e 1 (μ_1) são dadas por

$$\mu_0 = \frac{\mu_{\mathrm{S}}}{\omega_0} \qquad \text{e} \qquad \mu_1 = \frac{\mu_{\mathrm{T}} - \mu_{\mathrm{S}}}{\omega_1},$$

em que

$$\mu_{\mathrm{T}} = \sum_{i=1}^{\mathrm{N}} i h_i \qquad \text{e} \qquad \mu_{\mathrm{S}} = \sum_{i=1}^{\mathrm{L}} i h_i.$$

O threshold deve ser escolhido como sendo o valor L^* que maximiza a função critério dada pela Equação (46), ou seja,

$$\mathrm{L}^* = \operatorname{argmax}\{\eta(\mathrm{L})\}. \tag{48}$$

Conforme dito antes, este método parte da premissa que a imagem é formada por duas classes distintas (0 e 1, segundo a notação adotada). Contudo, histogramas puramente bimodais não são frequentes e este fato deve ser observado, pois influenciará o resultado da limiarização.

Método da máxima entropia

A entropia[16] do histograma h é definida como

$$S = -\sum_{i=1}^{\mathrm{N}} h_i \ln(h_i). \tag{49}$$

[16] Essa definição é conhecida como entropia de Shannon, que é a equivalente em teoria da informação à entropia de Boltzmann na termodinâmica. Há outras entropias, tais como Entropia Generalizada, Entropia de Tsallis etc.

A entropia de uma fonte fornece uma medida da taxa com que informação é produzida por essa fonte. As abordagens entrópicas para limiarização operam no sentido de estimar um threshold que maximize a entropia, promovendo assim a máxima transferência de informação possível da fonte em questão (a imagem).

Uma maneira simples de se obter o limiar pela maximização da entropia foi proposto por Kapur *et al.* [15]: deve-se maximizar uma função critério dada pela soma das entropias das duas classes (A e B, segundo a notação original adotada):

$$S_{\mathrm{T}}(\mathrm{L}) = S_{\mathrm{A}} + S_{\mathrm{B}}, \tag{50}$$

em que, segundo a Equação (49), temos

$$\begin{cases} S_{\mathrm{A}} = -\sum_{i=1}^{\mathrm{L}} \frac{h_i}{P_{\mathrm{A}}} \ln\left(\frac{h_i}{P_{\mathrm{A}}}\right); \quad \text{e} \\ \\ S_{\mathrm{B}} = -\sum_{i=L+1}^{\mathrm{N}} \frac{h_i}{P_{\mathrm{B}}} \ln\left(\frac{h_i}{P_{\mathrm{B}}}\right). \end{cases} \tag{51}$$

Na Equação (51), P_{A} e P_{B} são as probabilidades de ocorrência das classes A e B, respectivamente:

$$P_{\mathrm{A}} = \sum_{i=1}^{\mathrm{L}} h_i \qquad \text{e} \qquad P_{\mathrm{B}} = 1 - P_{\mathrm{A}}.$$

De maneira análoga ao método de Otsu, o threshold L^* é escolhido como sendo aquele que maximiza a função critério S_{T}, tal como definida na Equação (50):

$$\mathrm{L}^* = \operatorname{argmax}\{S_{\mathrm{T}}(\mathrm{L})\}. \tag{52}$$

A.2 Filtragem no domínio das frequências

O ponto de partida para filtragem no domínio das frequências é o produto entrada-a-entrada das transformadas de Fourier da imagem e do filtro, expresso pela Equação (10), repetida aqui:

$$\mathcal{W}(\omega_1, \omega_2) = \mathcal{F}(\omega_1, \omega_2) \cdot \mathcal{U}(\omega_1, \omega_2),$$

em que k_1, k_2 são as frequências espaciais e $\mathcal{F}$,$\mathcal{U}$ são as transformadas dos sinais f e u, respectivamente.

Naturalmente, a transformada de Fourier discreta (DFT) de imagens difere da contínua, principalmente por sua periodicidade:

$$\mathcal{F}(\omega_1, \omega_2) = \mathcal{F}(\omega_1 + 2\pi, \omega_2 + 2\pi). \quad (53)$$

O que a Equação (53) mostra é que a análise de sinais no domínio das frequências pode ocorrer sem perda alguma de generalidade sobre qualquer intervalo de largura 2π, tal como $[0, 2\pi], [-\pi, \pi]$ etc. Usualmente usa-se o intervalo $[-\pi, \pi]$, pois dentro desse intervalo simétrico a magnitude de $\mathcal{F}$, dada por

$$|\mathcal{F}| = \sqrt{\mathrm{Re}\,\{\mathcal{F}\}^2 + \mathrm{Im}\,\{\mathcal{F}\}^2} \quad (54)$$

é simétrica em relação à origem, *desde que o sinal f seja real.*

Deve-se atentar que a classe de algoritmos rápidos usados para o cômputo das DFTs – chamados de FFT[17] – operam mais eficientemente sinais cujas dimensões sejam potências de dois. Para isso, podemos realizar completamento com zeros (*zero padding*) nos sinais até que as dimensões sejam potências inteiras de dois.

Por fim, vale recordar que a transformada de Fourier é, em geral, complexa, de forma que podemos expressar sua fase, dada por

$$\varphi(\mathcal{F}) = \arctan\left(\frac{\mathrm{Im}\,\{\mathcal{F}\}}{\mathrm{Re}\,\{\mathcal{F}\}}\right). \quad (55)$$

[17] Do inglês *Fast Fourier Transform.*

A.3 Códigos

A.3.1 Obtenção do limiar

```
%   |--------------------------------------------------------------------|
%   |           Centro Brasileiro de Pesquisas Fisicas - CBPF            |
%   |           Coordenacao de Atividades Tecnicas - CAT                 |
%   |                                                                    |
%   |                   limiar.m                                         |
%   |                                                                    |
%   |   Descricao:      Retorna o valor de limiar para binarizacao       |
%   |                   segundo o metodo escolhido. Os parametros de     |
%   |                   entrada sao a imagem x (double) e a string str   |
%   |                   com o metodo de obtencao do threshold:           |
%   |                   - 'med' calcula o valor medio da imagem;         |
%   |                   - 'ent' calcula o valor que maximiza a entropia  |
%   |                     de Shannon da distribuicao da imagem;          |
%   |                   - 'ots' calcula o limiar segundo o metodo de     |
%   |                     Otsu.                                          |
%   |--------------------------------------------------------------------|

function    y   =   limiar(x,str)
    s   =   size(x);
    N_bins  =   256;

    h   =   imhist(x,N_bins);
    h   =   h/(s(1)*s(2));

    switch  lower(str)
        case 'med'
            y   =   Media(h);
        case 'ent'
            y   =   MaxEntropia(h);
        case 'ots'
            y   =   Otsu(h);
        otherwise
           y   =    Media(h);
    end

end

function    y   =   Media(h)
    N_bins  =   length(h);
    y       =   linspace(0,1,N_bins)*h;
end

function    y   =   MaxEntropia(h)
    N_bins  =   length(h);
    S       =   zeros(1,N_bins);   % funcao criterio

    % os proximos passos garantem que nao ocorra entrada nula no histograma
    % avaliado, para evitar divergencias. As entradas nulas sao trocadas
    % pelo menor valor nao-nulo da distribuicao.
        hMin    =   min(h(h~=0));

        hEnt            =   h;
        hEnt(hEnt==0)   =   hMin;
        hEnt            =   hEnt/sum(hEnt);

    for k=1:N_bins
        PA  =   sum(hEnt(1:k));    % probab. da classe A
        PB  =   1-PA;              % probab. da classe B

        % entropia classe A
        SA  =   -hEnt(1:k)'*log(hEnt(1:k)/PA)/PA;
```

```
        % entropia classe B
        SB  =   -hEnt(k+1:N_bins)'*log(hEnt(k+1:N_bins)/PB)/PB;

        S(k)=   SA+SB;
    end

    arg =   find(S==max(S));
    y  =   (arg-1)/(N_bins-1);
end

function    y   =   Otsu(h)
    N_bins  =   length(h);
    C       =   zeros(1,256);                   % funcao criterio

    for k=1:N_bins
        w0  =   sum(h(1:k));                        % probab. da classe 0
        w1  =   1-w0;                               % probab. da classe 1
        u0  =   ((0:k-1)*h(1:k))/w0;                % media da classe 0
        u1  =   ((k:N_bins-1)*h(k+1:N_bins))/w1;    % media da classe 1
        ut  =   linspace(0,1,N_bins)*h;             % media total

        desv_med_quad    =   ((0:N_bins-1)-ut).^2;

        sb  =   w0*(u0-ut)^2+w1*(u1-ut)^2;      % variancia entre-classes
        st  =   desv_med_quad*h;                % varianca total

        C(k)   =   sb/st;
    end
    arg =   find(C==max(C));
    y   =   (arg-1)/(N_bins-1);
end
```

A.3.2 Filtragem no domínio das frequências

```
%   |--------------------------------------------------------------------|
%   |           Centro Brasileiro de Pesquisas Fisicas - CBPF            |
%   |           Coordenacao de Atividades Tecnicas - CAT                 |
%   |                                                                    |
%   |                   filtragem_dominio_frequencias.m                  |
%   |                                                                    |
%   |   Descricao:      Realiza a filtragem da imagem de entrada (x)     |
%   |                   pelo filtro dado (h). O procedimento e realizado|
%   |                   por meio das DFTs dos sinais dados. A imagem     |
%   |                   filtrada (y) e retornada e os espectros e fases  |
%   |                   sao exibidos.                                    |
%   |--------------------------------------------------------------------|

function    y   =   filtragem_dominio_frequencias(x,h)
    % OBTENCAO DA MAIOR DIMENSAO DA IMAGEM
    s = size(x);                    DimMax= max(s);

    % ESTIMATIVA DA MENOR POTENCIA DE 2 MAIOR QUE DimMax
    M = log2(DimMax);               M = 2^ceil(M);

    % OBTENCAO DAS DFTs, SEPARADAS EM ESPECTROS DE POTENCIA E FASE
    X = fft2(x,M,M);
    H = fft2(h,M,M);

    % os espectros e fases sao obtidos sobre as versoes centralizadas das
    %DFTs dos sinais
    specX   =   abs(fftshift(X));
    specH   =   abs(fftshift(H));
    phaseX  =   angle(fftshift(X));
    phaseH  =   angle(fftshift(H));

    % FILTRAGEM REALIZADA PELO PRODUTO PONTO-A-PONTO DAS TRANSFORMADAS
    Y   =   X.*H;
    % DFT inversa para reconstrucao do sinal filtrado
    y   =   ifft2(Y);
    % transformacao da matriz iDFT em uma imagem
    y   =   mat2gray(y(1:s(1),1:s(2)));

    Y   =   ifftshift(Y);

    % VISUALIZACAO
    figure;
    subplot(1,2,1);
        imshow(log(specX+1),[]);    title('Espectro da imagem');
    subplot(1,2,2);
        imshow(phaseX,[]);          title('Fase da imagem');

    figure;
    subplot(1,2,1);
        imshow(log(specH+1),[]);    title('Espectro do filtro');
    subplot(1,2,2);
        imshow(phaseH,[]);          title('Fase do filtro');

    figure;
    subplot(1,2,1);
        imshow(log(Y+1),[]);        title('Espectro da imagem filtrada');
    subplot(1,2,2);
        imshow(y);                  title('Imagem filtrada');
end
```

Pedidos de cópias desta publicação devem ser enviados aos autores ou ao:

Centro Brasileiro de Pesquisas Físicas
Área de Publicações
Rua Dr. Xavier Sigaud, 150 – 4º andar
22290-180 – Rio de Janeiro, RJ
Brasil
E-mail: alinecd@cbpf.br/valeria@cbpf.br
http://portal.cbpf.br/publicacoes-do-cbpf
http://revistas.cbpf.br/index.php/MO/index

Requests for copies of these reports should be addressed to:

Centro Brasileiro de Pesquisas Físicas
Área de Publicações
Rua Dr. Xavier Sigaud, 150 – 4º andar
22290-180 – Rio de Janeiro, RJ
Brazil
E-mail: alinecd@cbpf.br/valeria@cbpf.br
http://portal.cbpf.br/publicacoes-do-cbpf
http://revistas.cbpf.br/index.php/MO/index