

Uma aplicação de CBIR à análise de imagens médicas de imuno-histoquímica utilizando Morfologia Matemática e espectro de padrões

Trabalho de Conclusão de Curso

Engenharia da Computação

Rodolfo Charamba de Santana
Orientador: Prof. Wellington Pinheiro dos Santos

Rodolfo Charamba de Santana

**Uma aplicação de CBIR à análise de imagens
médicas de imuno-histoquímica utilizando
Morfologia Matemática e espectro de padrões**

Monografia apresentada como requisito parcial para
obtenção do diploma de Bacharel em Engenharia da
Computação pela Escola Politécnica de Pernambuco
– Universidade de Pernambuco.

Recife, novembro de 2008.

Resumo

Os sistemas de recuperação de imagens baseada em conteúdo (CBIR – *Content-based Image Retrieval*) possuem a capacidade de retornar imagens utilizando como chave de busca outras imagens. Essa técnica, que utiliza características visuais para procurar imagens em um grande banco de dados de imagens de acordo com os interesses do usuário, vem se tornando uma área de pesquisa bastante ativa desde 1990. Quando uma imagem de consulta é passada, o sistema CBIR tem como função pesquisar em um banco de dados as n imagens que sejam semelhantes à passada de consulta, de acordo com um critério dado. Essa similaridade entre as imagens é medida através de características obtidas das mesmas, como histogramas, transformadas, cor, forma e textura. Várias técnicas de processamento de imagens podem ser utilizadas para extrair tais características de uma imagem, dentre as quais a morfologia matemática. A Morfologia Matemática representa uma técnica não-linear de processamento de imagens. Formalizada a partir dos anos 60, ela tem como fundamento a teoria dos conjuntos e a geometria integral, concentrando seu esforço na estrutura geométrica da imagem. A idéia básica dessa técnica consiste na comparação de uma imagem com um elemento estruturante, que é uma imagem menor e de forma conhecida, e que contém características geométricas e/ou topológicas relacionadas com a informação que pretendemos extrair da imagem de interesse. Sistemas CBIR, juntamente com a morfologia matemática, tem se mostrado uma poderosa ferramenta para a recuperação de imagens de maneira rápida e eficiente. Podemos encontrar aplicações nas mais diferentes áreas, entre elas na área médica. Na intenção de recuperar imagens semelhantes a uma dada imagem médica, esses sistemas se mostram mais eficientes dos que os sistemas em que as imagens estão associadas a uma palavra-chave ou um número.

Abstract

Content-Based Image Retrieval (CBIR) is a technique, or a set of techniques, able to search images using other images as keys. This technique, which uses visual characteristics to seek images in a large image database, according to the interests of the user, is becoming a very active area of research since 1990. CBIR systems are designed to look for similar images in a determined image database, given a certain image key. This search is performed according to a specific criterion. Degrees of similarity are measured according to certain feature vectors whose coordinates are determined by image characteristics like histogram, color, shape and texture. Various techniques of image processing can be used to extract features from an image. Mathematical morphology can be a way to extract such features. Mathematical morphology is a non-linear set of techniques of image processing. Formalized in the 60s, it is based in the theory of sets and integral geometry, concentrating their efforts on geometric structure of images. The basic idea of morphological techniques is making sorts of operations and comparisons with the image of interest and a structure element, a smaller well-known image whose characteristics are composed by geometric features and / or related to the topological information we want to extract from the image of interest. CBIR systems and mathematical morphology together have been powerful tools for quick and efficient recovery of images. We can find applications in many different areas, including medical-based ones. In order to retrieve images similar to a given medical image, these systems have shown that are more efficient than other systems in which images are associated with a keyword or a number, in an intuitive sense, connected to the skills of the specialist.

Sumário

Índice de Figuras	vi
Tabela de Símbolos e Siglas	vii
1 Introdução	9
1.1 Organização do Trabalho	10
2 Recuperação de Imagens Baseada em Conteúdo - CBIR	11
2.1 Introdução	11
2.2 Imagem e Atributos	13
2.2.1 Cor	13
2.2.2 Textura	15
2.2.3 Forma	17
2.3 Conclusão	19
3 Introdução à Morfologia Matemática	20
3.1 Introdução	20
3.2 Operações básicas da Morfologia Matemática	21
3.2.1 Dilatação e erosão	21
3.2.2 Abertura e fechamento	25
3.2.3 Gradiente Morfológico	29
3.3 Conclusão	30
4 Experimentos e Resultados	31
4.1 Introdução	31
4.2 Algoritmo k-médias	31
4.3 Conjunto de Dados	34
4.4 Geração dos Vetores de Características	34
4.5 Resultados das Classificações	38
4.6 Conclusão	42
5 Considerações Finais e Trabalhos Futuros	43
5.1 Trabalhos Futuros	44

Índice de Figuras

Figura 2.1 Diagrama de um sistema de recuperação de imagem por conteúdo [3]	12
Figura 2.2 Imagem original	15
Figura 2.3 Imagem saturada	15
Figura 2.4 Imagem original	15
Figura 2.5 Imagem original	15
Figura 2.6 Exemplos de textura	16
Figura 2.7 Exemplos de formas de imagens	17
Figura 2.8 Exemplos de similaridade de forma baseada em contorno e região [15]	18
Figura 2.9 Classificação das técnicas de representação e descrição de formas [18]	19
Figura 3.1 Exemplo de dilatação binária [22]	22
Figura 3.2 Exemplo de dilatação (adaptado de [21])	23
Figura 3.3 Exemplo ilustrativo de dilatação [24]	24
Figura 3.4 Exemplo de erosão binária [22]	24
Figura 3.5 Exemplo de erosão (adaptado de [21])	25
Figura 3.6 Exemplo ilustrativo de erosão [24]	25
Figura 3.7 Exemplo da operação de abertura [25]	27
Figura 3.8 Exemplo da operação de fechamento [25]	27
Figura 3.9 Detecção de bordas com gradiente morfológico binário [22]	29
Figura 3.10 Aplicação de um gradiente morfológico sobre uma imagem [24]	30
Figura 4.1 Etapas do algoritmo k-médias [30]	33
Figura 4.2 Tipos de imagens: (a) NP001a, (b) NP9759b, (c) P9824a e (d) P98289b	34
Figura 4.3 Tela principal do ambiente ANIMED	35
Figura 4.4 Seleção do espectro morfológico	35
Figura 4.5 Tela de seleção da banda de cor no ANIMED	36
Figura 4.6 Tela de seleção do número de iterações	36
Figura 4.7 Exemplos de espectros de padrões obtidos	37
Figura 4.8 Partes do arquivo ARFF	39
Figura 4.9 Resultados da classificação com 2 <i>clusters</i>	40
Figura 4.10 Resultado da classificação com 4 <i>clusters</i>	41

Tabela de Símbolos e Siglas

(Dispostos por ordem de aparição no texto)

CBIR – Content-Based Image Retrieval (Recuperação de Imagens Baseada em Conteúdo)
QBIC - Query By Image Content

Agradecimentos

À Deus.

Aos meus pais, pelo carinho, dedicação, e apoio em todos os momentos da minha vida, nas minhas escolhas certas ou erradas, enfim, por eu ser quem eu sou hoje.

Ao meu orientador Wellington Pinheiro dos Santos, pela grande ajuda e paciência ao longo desse período, e pelas conversas descontraídas sobre vários assuntos que, importantes ou não, serviram de alguma forma para o meu desenvolvimento como pessoa.

Aos amigos Samuel Sales e Arthur Campos, por todas as noites em claro, sejam elas dedicadas a projetos ou a farra. À amizade que construímos ao longo dessa etapa e que permaneça ao longo de nossas vidas.

Aos amigos Pedro Clericuzzi e Rubens Galvão, pela ajuda ao longo desse trabalho, seja esclarecendo dúvidas, revisando texto, ou apenas pelas longas conversas sobre qualquer coisa, menos o próprio trabalho.

A Máfia, grupo de excelentes amigos, que tornaram toda essa jornada mais suportável, através de vários momentos felizes que, para mim, serão inesquecíveis.

Muito obrigado a todos!

Capítulo 1

Introdução

Há tempos sabe-se que uma imagem pode representar idéias, ações, sentimentos até melhor do que as palavras. Na verdade, uma imagem ultrapassa qualquer barreira temporal ou idiomática e não exige conhecimento prévio como a escrita. Com o passar do tempo e com os avanços em termos de equipamento, as imagens foram se tornando cada vez mais importantes, com algumas áreas profissionais incorporando seu uso. Nas últimas décadas, com o rápido desenvolvimento tecnológico e a popularização da web, a quantidade de informação visual circulando, seja ela de forma digital ou não, é enorme. Por este motivo, é necessário que se tenha uma forma eficiente de armazenar, manter e recuperar as imagens.

Devido às dificuldades encontradas nos métodos de indexação tradicionais, que utilizavam anotações manuais como palavras chaves, novos modelos mais eficientes foram desenvolvidos para contornar essa situação, entre eles os sistemas de recuperação de imagens baseada em conteúdo (*Content-Based Image Retrieval* – CBIR). Sistemas CBIR utilizam o conteúdo visual da imagem, como cor, textura e forma para representar e indexar as imagens.

Essa técnica pode ser utilizada em qualquer área onde o armazenamento e a recuperação de imagens são relevantemente importantes, como na medicina. De fato, tem-se observado ultimamente que a quantidade de imagens adquiridas em hospitais e em centros médicos tem crescido bastante, permitindo que as técnicas de CBIR sejam utilizadas de maneira eficiente nessa área.

Neste trabalho, foi proposto utilizar técnicas CBIR à análise de imagens médicas de imunohistoquímica, a fim de ajudar a recuperação dessas imagens por parte dos médicos, gerando uma espécie de triagem virtual, para facilitar a análise e, principalmente, otimizar o tempo de análise por parte do especialista (patologista ou biomédico). Para isso, foram estudados os conceitos de

Morfologia Matemática, e em particular o espectro de padrões, que servem para identificar as características de uma imagem, em especial a forma. Nesse contexto, o objetivo deste trabalho é desenvolver uma aplicação CBIR que utiliza da Morfologia Matemática e espectro de padrões a fim de recuperar, de maneira eficiente, imagens médicas de imuno-histoquímica, diferenciando imagens de placenta das de pulmão, mesmo que aplicadas ao mesmo antígeno.

1.1 Organização do Trabalho

Este trabalho foi dividido em seis capítulos. Neste primeiro capítulo encontram-se as considerações iniciais e os objetivos do trabalho.

O Capítulo 2 mostra os conceitos de CBIR, bem como a importância da extração de características de uma imagem. Para isso, são mostradas as principais características de uma imagem, como cor, textura e forma, sendo esta última a utilizada na realização deste trabalho.

O Capítulo 3 apresenta o conceito da Morfologia Matemática, mostrando sua importância no processamento de imagens. São mostradas suas operações básicas e como as combinações das mesmas geram poderosos operadores morfológicos, utilizados para detectar as bordas dos objetos em uma imagem. Neste capítulo é apresentado o espectro de padrões, um vetor capaz de representar de forma única imagens binárias a partir da forma.

No Capítulo 4 é mostrado o experimento realizado, ressaltando os métodos utilizados e, por fim, os resultados obtidos.

Para finalizar, no Capítulo 5, são tecidos comentários e apresentadas as conclusões gerais e sugestões de trabalhos futuros.

Capítulo 2

Recuperação de Imagens Baseada em Conteúdo - CBIR

2.1 Introdução

O uso de imagens na comunicação humana é bastante comum; engenheiros as usam em seus projetos, cineastas para contar histórias, médicos para efetuar diagnósticos, entre outras situações. O uso de imagens está presente também no mundo digital e, nos últimos anos tem-se visto um rápido aumento na quantidade de imagens que são armazenadas. Entretanto, grandes quantidades de informação estão ficando sem utilidade, pois tais informações não são acessíveis se, previamente, elas não houverem sido organizadas para permitir a busca e recuperação eficiente desses dados [1]. Desde o começo dos anos 90, devido ao surgimento de bancos de imagens em grande escala, surgiram dois grandes problemas para o tratamento dessas imagens. O primeiro é o esforço em fazer as anotações manuais, já que tradicionalmente as imagens são armazenadas em bases de dados utilizando informação textual. Anotá-las manualmente em uma grande base de dados é uma tarefa praticamente impossível [2]. O segundo problema refere-se à subjetividade da percepção humana. As características visuais das imagens são difíceis de serem descritas com palavras, além de que diferentes informações podem ser obtidas por pessoas diferentes sobre a mesma imagem, gerando respostas não confiáveis para pesquisas sobre esses assuntos [1,2]. Para solucionar esses problemas, pesquisas foram realizadas e novas propostas de recuperação de imagens surgiram, entre elas a recuperação de imagens por conteúdo (*Content-Based Image Retrieval* – CBIR).

Os sistemas de recuperação de imagens por conteúdo possuem a habilidade de retornar imagens utilizando como chave de busca outras imagens. Considerando uma imagem de consulta, o foco de um sistema CBIR é pesquisar no banco de dados as n imagens mais similares a imagem de consulta de acordo com um critério dado.

Para recuperar imagens, o usuário fornece ao sistema imagens de exemplo. O sistema transforma então esses exemplos em sua representação interna, que são os vetores de características. As similaridades ou diferenças entre os vetores de características do exemplo passado como entrada e das imagens armazenadas na base de dados são calculadas e então a recuperação é executada com a ajuda de um esquema de indexação. Esse esquema fornece uma maneira eficiente de procurar as imagens pela base de dados. Os sistemas de recuperação recentes incorporaram o uso de *feedback* do usuário para modificar o processo de recuperação, permitindo recuperações mais significativas tanto perceptual quanto semanticamente [3]. A figura abaixo (figura 2.1) representa de maneira sucinta um sistema de recuperação de imagem por conteúdo.

Fundamentals of Content-Based Image Retrieval

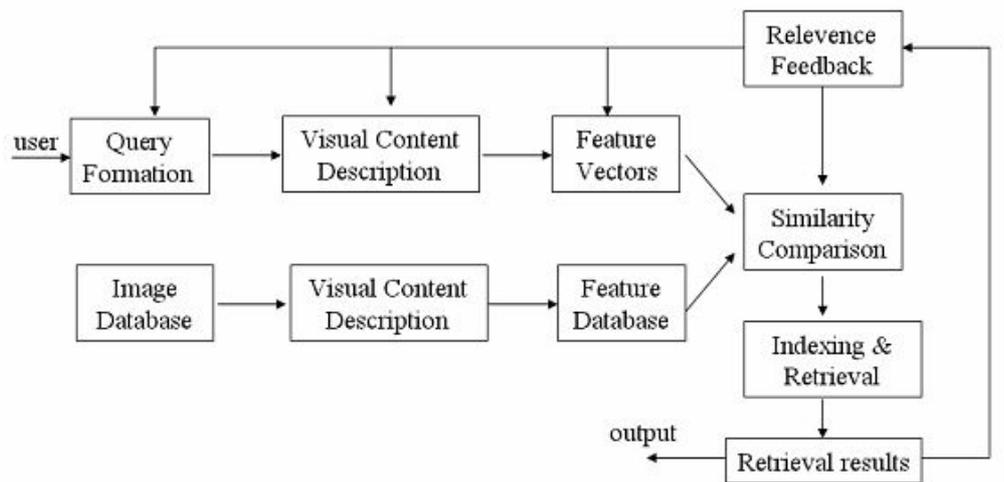


Figura 2.1 Diagrama de um sistema de recuperação de imagem por conteúdo [3]

Sistemas CBIR utilizam o conteúdo visual da imagem, como cor, textura e forma para representar e indexar as imagens, em vez de utilizar as anotações manuais via palavras chaves [3]. De maneira geral, são três as tarefas fundamentais da recuperação de imagens por conteúdo: extração de características visuais (*feature extraction*), indexação multidimensional e projetos de sistemas de recuperação.

A extração de características é a base dos sistemas CBIR, que podem ser classificadas como características gerais ou específicas de domínio. As primeiras incluem características de cor, textura e forma, enquanto as últimas dependem da aplicação [1]. Nos sistemas CBIR comuns, as características visuais presentes nos bancos de dados são descritas por um vetor de características multidimensional. Um vetor de características é uma representação numérica sucinta de uma imagem ou parte de uma imagem (um objeto da mesma) caracterizando medidas dos aspectos representativos do objeto [1,3]. O número de características dependerá do domínio de imagens e das propriedades que se deseja caracterizar, e o mesmo determinará a dimensão do vetor (vetor n -dimensional). Esta nova representação da imagem pode ser armazenada em um banco de dados que permita uma recuperação eficiente da imagem, porém essa nova representação deve atender as três seguintes considerações:

- Reduzir a dimensionalidade dos dados,
- Destacar características da imagem para facilitar a percepção humana,
- Ser invariante as transformações da imagem.

2.2 Imagem e Atributos

Como dito anteriormente, sistemas CBIR utilizam o conteúdo visual da imagem, como cor, textura e forma para representar e indexar as imagens. A recuperação da imagem está baseada na similaridade das características derivadas destes atributos.

2.2.1 Cor

Entre os atributos citados anteriormente, a cor é a característica visual mais usada para a indexação e recuperação de imagens. A utilização da cor no processamento de imagem é motivada por dois fatores principais. Primeiramente, a cor é um poderoso descritor que geralmente simplifica a identificação e extração de um objeto em uma cena. O outro fator é que humanos podem discernir milhares de tons de cores e intensidades, em comparação com apenas cerca de duas dezenas de tons de cinza. Esse segundo fator é particularmente importante na análise manual de imagem (ou seja, feita por humanos) [5].

Existem diferentes representações de cores que incluem desde o RGB (*red, green, blue*), o mais simples modelo que mapeia diretamente as características físicas do dispositivo de exibição, até o HSI (*hue, saturation, intensity*) que reflete mais precisamente o modelo de cores para a percepção humana.

Segundo [4], muitos trabalhos realizados recentemente de extração de características baseadas em distribuição de cores estão concentrados no histograma de cor. Alguns dos trabalhos anteriores incluíram a indexação da cor usando interseção de histogramas. Outros, até mais recentes, montam uma estrutura de histograma de dimensão variável, chamada histogramas métricos, os quais são utilizados na recuperação de imagens similares.

Um histograma de cor é uma aproximação da distribuição de probabilidade $p(c)$ das cores na imagem. Nesse histograma associamos a cada intensidade de cor c presente na imagem, a sua frequência de ocorrência, ou seja, o número de pixels da imagem que utilizam a cor c . Os histogramas de cor são invariantes à translação e rotação das imagens, sendo que, com a normalização dos histogramas, obtém-se também a invariância à escala. De qualquer forma, os histogramas de cor não indicam a localização espacial dos pixels na imagem.

Muitas operações podem ser realizadas utilizando os histogramas de cor, através da mudança de parâmetros e limiares. Na próxima página se tem um exemplo, onde foi aplicada uma operação de alongamento (*stretch*) sobre a imagem original (figura 2.2). O histograma da imagem original (figura 2.4) é alongado, ficando como na figura 2.5. Como resultado dessa operação, obtém-se uma imagem melhor definida (figura 2.3).

Em alguns trabalhos, onde a distribuição de cores na imagem é utilizada para montar o vetor de características, baseado nos coeficientes wavelets, são apresentados por [1], conseguindo diminuir o tempo de busca das imagens similares, mas com uma eficiência não tão boa, com no máximo 35% de eficiência.

Os histogramas de cores são utilizados atualmente por várias ferramentas comerciais em sistemas de recuperação de imagens por conteúdo, como o QBIC (*Query By Image Content*) da IBM [6], um dos mais poderosos sistemas de consulta a base de dados de imagens; o Virage [7]; o VisualSEEK [8], entre outros. Mesmo assim, algumas dificuldades são encontradas ao se utilizar os histogramas de cores. O vetor de características poderá ser muito grande se a imagem contém um número alto de cores, fazendo com que a indexação de vetores com essa dimensão torne-se problemática.

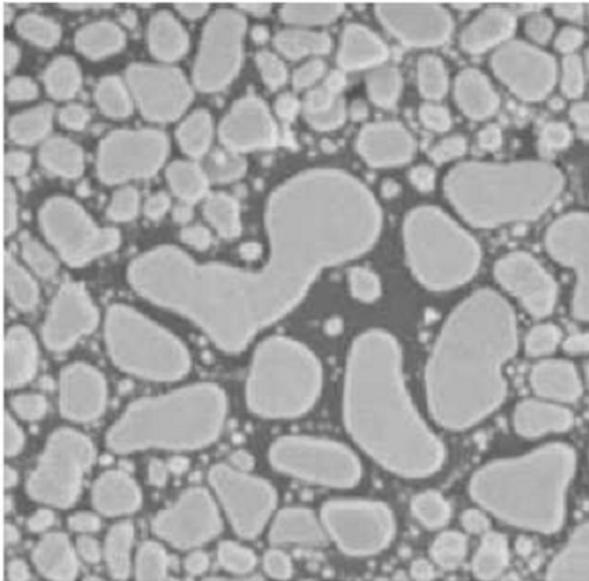


Figura 2.2 Imagem original

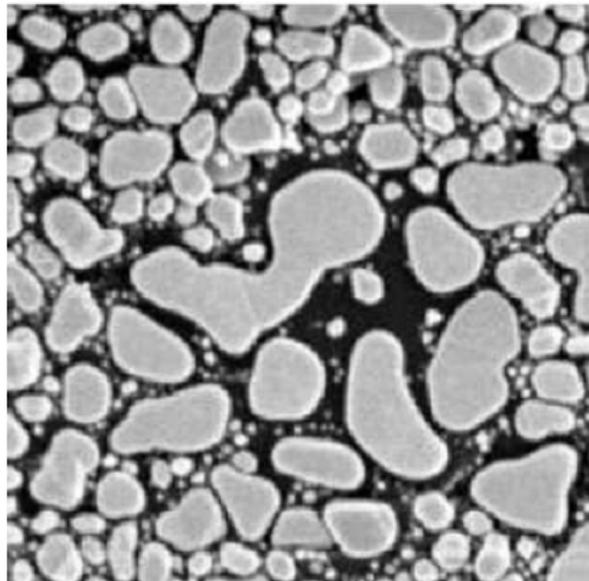


Figura 2.3 Imagem saturada

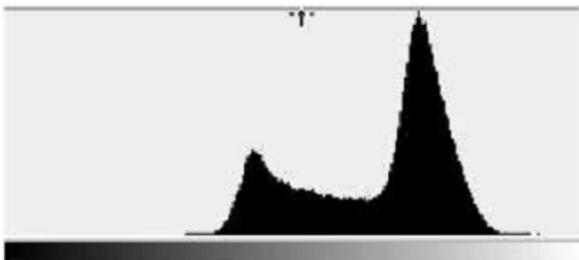


Figura 2.4 Imagem original



Figura 2.5 Imagem original

2.2.2 Textura

A análise de texturas é um dos aspectos fundamentais da visão humana, no qual é possível discriminar entre superfícies e objetos. De modo semelhante, utilizando-se da visão computacional, pode-se tirar proveito das pistas fornecidas pela textura de superfícies para distinguir e reconhecer objetos. Logo, a textura representa uma fonte de informação visual muito rica acerca da natureza e estrutura tridimensional dos objetos físicos.

Quando se fala em textura, entende-se como as mudanças na intensidade da imagem que formam determinados padrões repetitivos, já que para os autores não existe uma definição clara de textura [9]. Esses padrões podem ser o resultado de diferenças de reflexão tal como a cor na

superfície, ou ser o resultado de propriedades físicas da superfície do objeto, como rugosidade. De maneira genérica, pode-se considerar que texturas são padrões visuais complexos compostos por entidades ou sub-padrões que possuem propriedades características (como brilho, densidade, granulação, etc.) da textura como um todo.

Embora o reconhecimento de textura seja fácil para as pessoas, isso não ocorre com procedimentos automáticos, já que essa tarefa geralmente precisa de técnicas computacionais complexas. O tratamento de textura difere do realizado sobre as cores devido ao fato de que as texturas são definidas sobre janelas ou regiões da imagem e não sobre pixels como as cores. A segmentação de uma imagem utilizando textura determina quais regiões da imagem possuem textura uniforme [10]. A figura 2.6 apresenta alguns exemplos de textura.



Figura 2.6 Exemplos de textura

De acordo com [11], modelos computacionais de característica de textura podem ser agrupados em três categorias: estrutural, estatístico e espectral. Modelos estruturais caracterizam texturas de acordo com o relacionamento local entre pixels de imagens, ou seja, lidam com o arranjo espacial de primitivas estruturais. Modelos estatísticos categorizam texturas de acordo com medidas estatísticas de característica visual, tais como grossura, granularidade, regularidade, entre outros. Este modelo considera também a distribuição dos tons de cinza e o inter-relacionamento entre eles. Por fim, modelos espectrais caracterizam textura como propriedades da transformada de Fourier ou nos resultados de filtragem das texturas por filtros apropriados.

A análise de textura tem por objetivo estabelecer o relacionamento de vizinhança dos elementos de textura e seu posicionamento em relação aos demais (conectividade), o número de elementos por unidade espacial (densidade) e a sua regularidade (homogeneidade) [4]. O uso dessa análise pode ser encontrado em diversas áreas computacionais, como reconhecimento de padrões e processamento de imagens. Porém, da mesma forma que no caso das cores, algumas dificuldades são encontradas, como o problema da invariância e o da dimensionalidade do vetor de características, ocasionando a “maldição da alta dimensionalidade”, onde o processo de

indexação e recuperação de imagens se torna mais lento devido à alta dimensionalidade dos vetores de características.

2.2.3 Forma

Da mesma maneira que a cor e textura de um objeto são importantes para o reconhecimento do mesmo, a forma também é uma característica muito importante. De fato, é possível identificar diferentes objetos em uma imagem analisando apenas as formas dos mesmos. Em muitos problemas reais, tais como a recuperação em bancos de dados biológicos, o reconhecimento de caracteres, o reconhecimento de digitais, etc., as formas são o objeto de busca [12]. A figura 2.7 mostra alguns exemplos de formas de imagens.



Figura 2.7 Exemplos de formas de imagens

Apesar da importância, a recuperação baseada em forma não é simples. Na verdade, em sistemas CBIR, o atributo forma é o que mostra a maior dificuldade quando usado na recuperação de imagens, especialmente pelo fato de ter que segmentar os objetos de interesse e conhecer o tamanho dos mesmos contidos na imagem, levando-se a recuperação por formas ser tipicamente limitada aos poucos objetos mais bem discriminados presentes na imagem [13]. Por exemplo, em aplicações médicas, a forma e o tamanho de tumores são importantes para classificá-los como benigno ou maligno. Tumores com bordas irregulares têm uma maior probabilidade de serem malignos, enquanto aqueles que mostram bordas regulares geralmente são benignos [14]. Outro problema se deve a própria natureza da imagem digital. Afinal, muitos dos objetos do mundo real são 3D enquanto imagem e vídeo são normalmente projeções 2D de objetos 3D [15]. Além disso, durante a projeção no plano 2D, informações sobre o objeto podem ser perdidas, somando-se ainda as oclusões, os ruídos, os defeitos, as distorções, etc.

Existe uma variedade de técnicas que tem sido propostas na literatura para a representação de formas. Segundo [16], existem basicamente dois tipos de descritores de forma: os baseados em contorno (*contour-based shape descriptors*) e os baseados em região (*region-based shape descriptors*). Os métodos baseados em contorno usam apenas o contorno (ou bordas) da forma de

um objeto e ignora completamente seu interior. Por outro lado, as técnicas baseadas em região levam em conta não só a borda da forma do objeto, como também os detalhes internos da forma do mesmo, ou seja, todos os *pixels* presentes no interior da forma do objeto analisado são utilizados para obtenção da representação da mesma [16-17]. Sendo assim, a recuperação e/ou classificação das imagens se dará de acordo com a similaridade da forma, baseada nas técnicas de contorno ou região. A figura 2.8 servirá de exemplo para melhor ilustrar as técnicas citadas.

Os objetos na primeira linha possuem uma similar distribuição espacial de pixels, sendo semelhantes de acordo com critérios baseados em região. Entretanto, é fácil observar que eles possuem contornos diferentes. Quando a similaridade baseada em contorno é empregada, os objetos mostrados em cada coluna são similares. Nesse caso, se fosse realizada uma consulta a partir de uma imagem exemplo, como a localizada na primeira linha e segunda coluna, as imagens consideradas semelhantes seriam as da primeira linha (se a similaridade fosse baseada em região) ou da segunda coluna (se fosse baseada em contorno) [15].

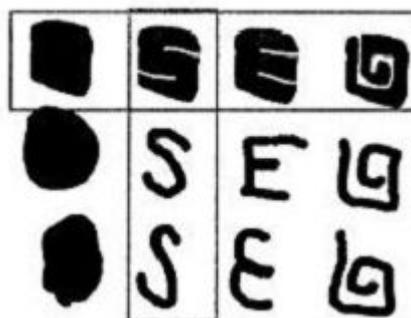


Figura 2.8 Exemplos de similaridade de forma baseada em contorno e região [15]

Dentro dessas duas abordagens citadas anteriormente, uma nova divisão pode ser feita com base no modo como a forma é representada. Se primitivas tais como segmentos e figuras geométricas são utilizadas, considera-se uma abordagem estrutural. Já no caso da forma ser tratada como um todo, a abordagem é considerada global. Dentro de cada uma dessas abordagens vários descritores podem ser enumerados. Uma divisão bastante coerente quanto a esses aspectos na representação e descrição de formas pode ser encontrada em [18]. A figura 2.9 ilustra essa divisão.

Para se projetar um bom descritor, várias características devem ser analisadas. Dentre elas, pode-se destacar que o mesmo deve representar bem a forma, ser fácil de calcular e comparar, ser conciso, além de possuir invariância a mudanças de orientação (rotação e translação) bem como a variações de escala, pois, por exemplo, objetos semelhantes mas com tamanhos diferentes devem

ser considerados similares, independente de seu tamanho. Assim sendo, são considerados métodos eficazes aqueles que possuem tais características aliadas também à robustez em relação ao ruído [2]. Na prática é difícil encontrar descritores que tenham pelo menos uma boa parte dessas características.

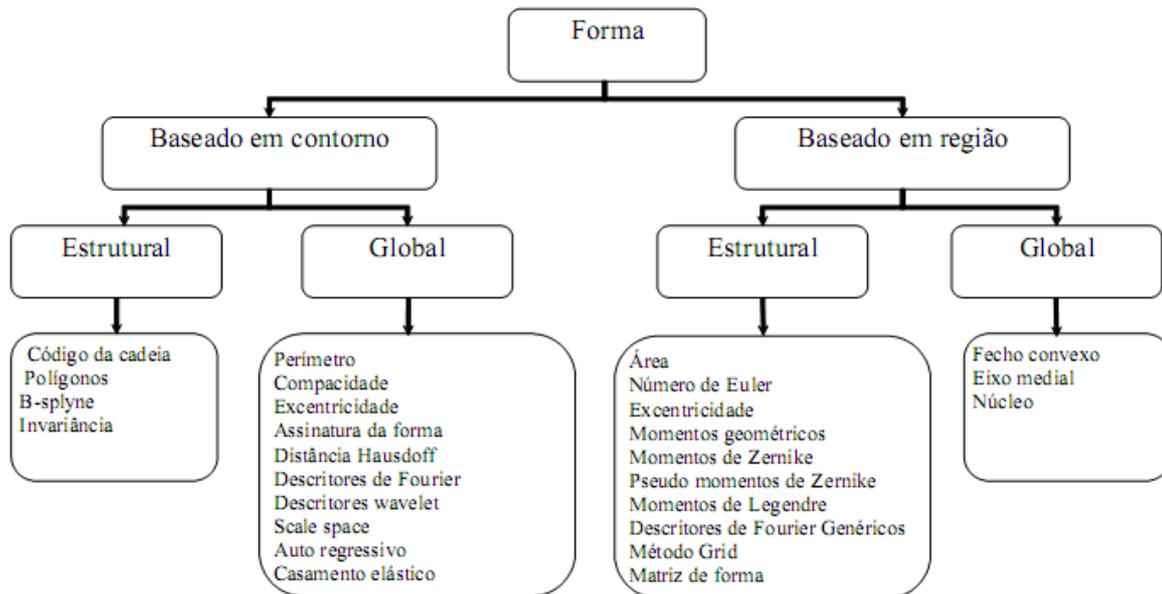


Figura 2.9 Classificação das técnicas de representação e descrição de formas [18]

2.3 Conclusão

Nesse capítulo, foram introduzidos os conceitos de sistemas CBIR, ferramentas muito importantes na recuperação de imagens. Tal recuperação se dá através da extração de características importantes da imagem, como cor, textura e forma, que foram apresentadas ao longo da seção 2.2. Nesse trabalho, o atributo forma foi o utilizado para recuperar as imagens.

No capítulo seguinte, serão introduzidos os conceitos da Morfologia Matemática, uma poderosa técnica utilizada no processamento de imagens, que se concentra na estrutura geométrica (forma) das imagens.

Capítulo 3

Introdução à Morfologia Matemática

3.1 Introdução

A Morfologia Matemática ou simplesmente morfologia diz respeito ao ramo de processamento não-linear de imagens que se concentra na estrutura geométrica da imagem. Esta estrutura pode ser de natureza macroscópica, onde o intuito é a análise de formas como caracteres impressos, por exemplo, ou pode ser de natureza microscópica onde pode haver interesse na distribuição de partículas ou texturas geradas por pequenas primitivas. De maneira geral, o objetivo básico da Morfologia Matemática é descrever quantitativamente as estruturas geométricas e funcionar como uma técnica na concepção de algoritmos na área de processamento digital de imagens (PDI), dispondo de ferramentas básicas como detectores de borda e filtros morfológicos [19]. Logo, a morfologia não é apenas uma teoria matemática, mas uma poderosa técnica utilizada em PDI.

As origens da Morfologia Matemática são os estudos da geometria de mídia porosa em meados da década de sessenta, pelo grupo liderado por Georges Matheron e Jean Serra, da École Supérieure des Mines de Paris, em Fontainebleau. Eles introduziram um formalismo relacionado à teoria de conjuntos para a análise de imagens binárias sendo possível processá-las com operações simples, como uniões, intersecções, complementações e translações. Até o final dos anos setenta, o grande potencial da morfologia para a análise de imagens tinha sido reconhecido e bastante utilizado na Europa, principalmente nas áreas envolvendo microscopia. A partir dos anos oitenta, a Morfologia Matemática começou também a ser difundida nos Estados Unidos e hoje é um fértil campo de pesquisas, tanto teóricas quanto práticas, sendo vigorosamente explorado em todo o mundo.

O princípio básico da Morfologia Matemática consiste em extrair de uma imagem desconhecida a sua geometria e a topologia através da utilização da transformação de uma outra imagem completamente definida, chamado elemento estruturante.

Segundo [20], o elemento estruturante é um conjunto completamente definido e conhecido (forma, tamanho), o qual é comparado, a partir de uma transformação, ao conjunto desconhecido da imagem. A morfologia age sobre imagens digitais a partir de elementos estruturantes geralmente definidos em uma malha retangular. De uma forma comparativa, pode-se dizer que o elemento estruturante tem a mesma importância para a Morfologia Matemática que uma máscara tem para o processamento linear. Ele é uma imagem de dimensões menores do que as da imagem a ser aplicada a transformação morfológica. As diversas transformações morfológicas são baseadas em comparações entre o elemento estruturante e a imagem em questão [21]. É importante salientar que uma das chaves para a obtenção de bons resultados na Morfologia Matemática é a escolha adequada do elemento estruturante.

De maneira geral, a Morfologia Matemática apresenta duas operações fundamentais que são a erosão, em que *pixels* que não atendem a um dado padrão são apagados da imagem, e a dilatação, em que uma pequena área relacionada a um *pixel* é alterada para um dado padrão. Entretanto, dependendo do tipo de imagem sendo processada (preto e branco, tons de cinza ou colorida) a definição dessas operações muda, devendo cada tipo ser considerado separadamente.

3.2 Operações básicas da Morfologia Matemática

Como dito anteriormente, as operações fundamentais da Morfologia Matemática são a erosão e a dilatação. A partir da combinação destas, podem-se originar outras operações aplicáveis à morfologia. Por esse motivo, pode-se dizer que a Morfologia Matemática é uma teoria construtiva, já que operações mais complexas podem ser obtidas a partir das mais simples.

3.2.1 Dilatação e erosão

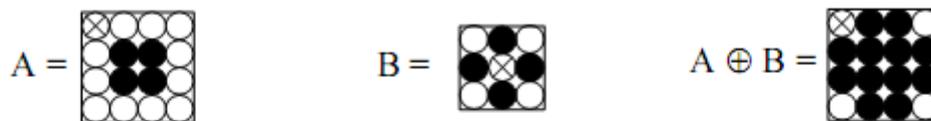
A dilatação, chamada as vezes de dilatação, é uma transformação morfológica a qual combina dois conjuntos usando adição vetorial [22]. Pode-se definir a dilatação de um conjunto A pelo conjunto B como:

$$A \oplus B = \{c \mid c = a + b, a \in A, b \in B\} \quad (1)$$

onde A representa a imagem sendo processada e B o elemento estruturante, onde sua composição define a natureza específica da dilatação. O símbolo \oplus se refere à dilatação binária (ou soma de Minkowsky) [23]. Outra forma de definir a dilatação se baseia no uso de funções, podendo ser aplicada a qualquer tipo de imagem:

$$\delta_g(f)(u) = (f \oplus g)(u) := \bigvee_{v \in S} f(v) \wedge g(u - v), \forall u \in S \quad (2)$$

onde, nesse caso, g é o elemento estruturante. Na prática, a imagem original é transformada pela dilatação de forma a fazer com que o elemento estruturante se encaixe na imagem, fazendo com que a imagem original “engorde”, ou seja, que a imagem seja expandida, como o nome dilatação sugere. A figura 3.1 mostra um exemplo de dilatação binária. No exemplo, o conjunto A representa a imagem, enquanto o conjunto B representa o elemento estruturante. O *pixel* marcado com um “x” representa a origem (0,0) de cada imagem.



$$A = \{(1,1), (1,2), (2,1), (2,2)\}$$

$$B = \{(-1,0), (0,-1), (0,1), (1,0)\}$$

$$A \oplus B = [A + \{(-1,0)\}] \cup [A + \{(0,-1)\}] \cup [A + \{(0,1)\}] \cup [A + \{(1,0)\}]$$

$$A + \{(-1,0)\} = \{(0,1), (0,2), (1,1), (1,2)\}$$

$$A + \{(0,-1)\} = \{(1,0), (1,1), (2,0), (2,1)\}$$

$$A + \{(0,1)\} = \{(1,2), (1,3), (2,2), (2,3)\}$$

$$A + \{(1,0)\} = \{(2,1), (2,2), (3,1), (3,2)\}$$

$$A \oplus B = \{(0,1), (0,2), (1,0), (1,1), (1,2), (1,3), (2,0), (2,1), (2,2), (2,3), (3,1), (3,2)\}$$

Figura 3.1 Exemplo de dilatação binária [22]

De maneira geral, as operações morfológicas podem ser representadas na forma de matrizes. A figura 3.2 ilustra essa situação, levando-se em conta que, usualmente, os *pixels* mais claros são representados por valores mais próximos de 1, enquanto os mais escuros por valores mais próximos de 0 [21]. No exemplo, f representa a imagem, g o elemento estruturante e h_1 o resultado da dilatação de f por g , onde $h_1 = f \oplus g$.

Pelo exemplo dado, pode-se observar que, através da dilatação, ocorre o crescimento das áreas claras e a eliminação das áreas escuras menores do que o elemento estruturante. De fato, esse resultado pode ser observado mais claramente no exemplo da figura 3.3.

$$f = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}_{(4,4)}$$

$$g = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}_{(2,2)}$$

$$h_1 = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}_{(4,4)}$$

Figura 3.2 Exemplo de dilatação (adaptado de [21])



Figura 3.3 Exemplo ilustrativo de dilatação [24]

Ao contrário da dilatação, a erosão basicamente “encolhe” uma imagem e pode ser vista como uma transformação morfológica que combina dois conjuntos usando vetores de subtração. Baseando-se na erosão binária, tal operação é definida como:

$$A \ominus B = \{x \mid x + b \in A, \forall b \in B\} \quad (3)$$

onde B é o elemento estruturante. Como na dilatação, a erosão pode ser definida para o caso de imagens em níveis de cinza. Nesse caso:

$$\varepsilon_g(f)(u) = (f \ominus g)(u) := \bigwedge_{v \in S} f(v) \vee \bar{g}(v - u), \forall u \in S \quad (4)$$

Na prática, a erosão transforma a imagem original f de modo a fazer as áreas semelhantes ao elemento estrutural g “diminuírem”. A figura 3.4 ilustra um exemplo de erosão binária. Como na dilatação, a erosão pode ser representada na forma de matrizes. Aproveitando a imagem f e o elemento estruturante g utilizados no exemplo da figura 3.2, a erosão de f por g será uma imagem $h_2 = f \ominus g$, dada pela figura 3.5.

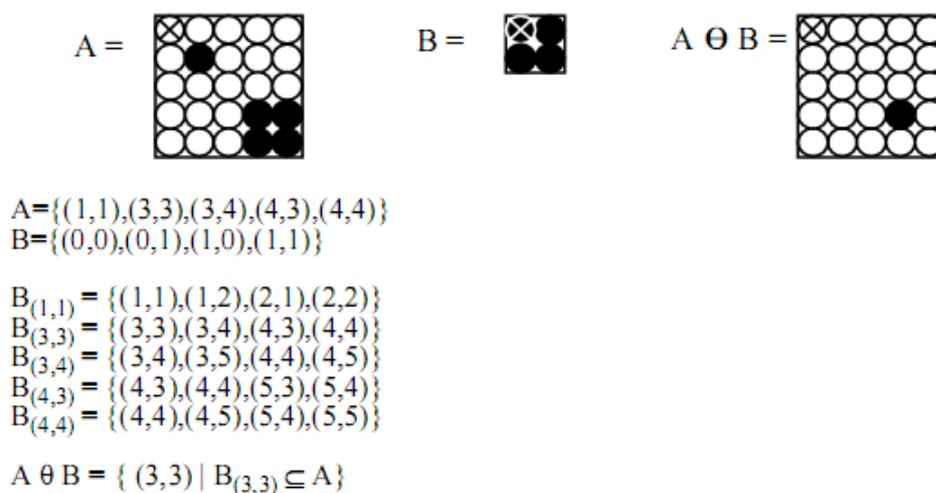


Figura 3.4 Exemplo de erosão binária [22]

$$h_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} (4,4)$$

Figura 3.5 Exemplo de erosão (adaptado de [21])

De acordo com os dois exemplos mostrados, pode-se observar que na operação de erosão as áreas escuras da imagem crescem, enquanto as áreas claras menores do que o elemento estruturante são eliminadas, considerando o fato de que, como na figura 3.2, os pixels mais claros são representados por valores mais próximos de 1, enquanto os mais escuros pelos valores mais próximos de 0. De forma mais prática, pode-se observar essa conclusão no exemplo mostrado na figura 3.6.



Figura 3.6 Exemplo ilustrativo de erosão [24]

3.2.2 Abertura e fechamento

Como visto na seção anterior, a dilatação tem a capacidade de expandir uma imagem, enquanto a erosão a encolhe. Nessa seção, será introduzido os conceitos de outras duas operações bastante importantes para a Morfologia Matemática: a abertura (*opening*) e o fechamento (*closing*).

A abertura e o fechamento são filtros morfológicos, ou seja, são filtros não-lineares baseados em operações morfológicas. Eles são baseados exatamente na combinação das operações de dilatação e erosão.

A operação de abertura é obtida quando se aplica a uma imagem uma erosão seguida imediatamente de uma dilatação, usando o mesmo elemento estrutural. A abertura em geral suaviza o contorno de uma imagem, quebra estreitos e elimina proeminências delgadas. Esse filtro é utilizado para eliminar regiões claras na imagem que não possam estar contidas no elemento estruturante. Assim, a abertura realiza na imagem um “peneiramento” dos pontos claros, podendo ser usado nesse caso para remover ruídos de uma imagem [21-22].

Baseando-se na teoria dos conjuntos, a abertura de um conjunto A pelo elemento estruturante B de forma binária é definida como:

$$A \circ B = (A \ominus B) \oplus B \quad (5)$$

Segundo [23], a abertura de uma imagem f pelo elemento estruturante g em operações em níveis de cinza é definida por:

$$\gamma_g(f) := \delta_g \varepsilon_g(f) \quad (6)$$

A figura 3.7 ilustra um exemplo da operação de abertura numa imagem, utilizando um elemento estruturante circular. No alto da figura, tem-se o conjunto original A (a imagem). Na linha intermediária é mostrada a etapa da erosão e na linha inferior o resultado da operação de dilatação aplicada ao conjunto resultante da erosão.

Basicamente a operação de fechamento é igual à abertura. Eles se diferenciam na ordem em que as são realizadas, ou seja, no fechamento primeiramente é feita a dilatação e depois é feita a erosão. A operação de fechamento une pequenas quebras, alarga golfos estreitos e elimina pequenos buracos na imagem. Desse modo, essa operação é utilizada para “eliminar” regiões escuras na imagem que não possam estar contidas no elemento estruturante. Assim como na abertura, o fechamento pode ser usado para remover ruídos de uma imagem, só que, nesse caso, ocorre um “peneiramento” dos pontos escuros [21-22].

De maneira geral, o fechamento de um conjunto A pelo elemento estruturante B é definido como:

$$A \bullet B = (A \oplus B) \ominus B \quad (7)$$

Como na abertura, existe uma outra forma de definir a operação de fechamento, relacionando uma imagem f com o elemento estruturante g (caso em níveis de cinza):

$$\phi_g(f) := \varepsilon_g \delta_g(f) \quad (8)$$

A figura 3.8 mostra um exemplo da operação de fechamento em uma imagem, utilizando um elemento estruturante circular. Como na figura 3.7, no alto da figura tem-se o conjunto original A, que representa a imagem. A diferença está nas linhas abaixo: agora na linha intermediária ocorre a dilatação da imagem e subseqüentemente a erosão do resultado, mostrada na linha inferior.

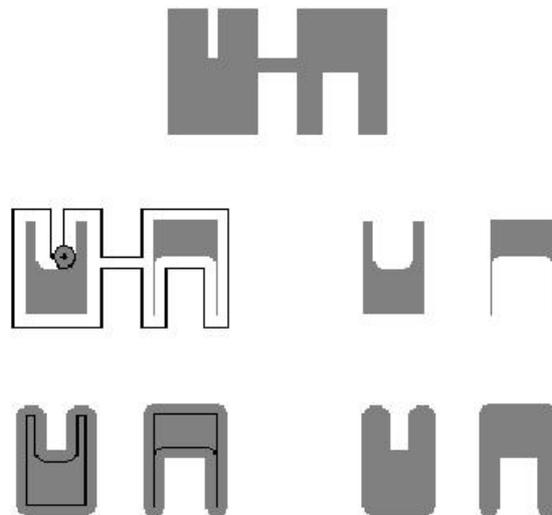


Figura 3.7 Exemplo da operação de abertura [25]

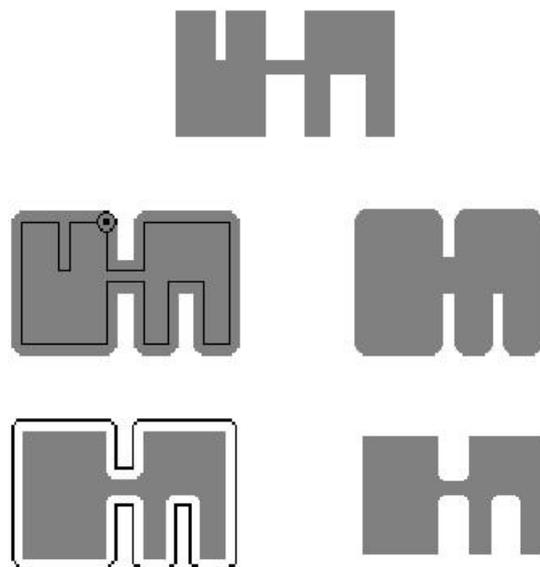


Figura 3.8 Exemplo da operação de fechamento [25]

Propriedades da Abertura e fechamento

Nessa seção foram mostrados os filtros morfológicos de abertura e fechamento. É importante observar que esses filtros possuem propriedades relevantes. As operações de abertura e fechamento são consideradas duais, idempotentes e crescentes.

Diz-se que tais operações são duais já que uma abertura pode ser implementada usando um fechamento e vice-versa. Definindo:

$$\gamma_B(f) = -\phi_B(-f) \quad (9)$$

Segundo [23], a transformação ξ é crescente se, dadas duas imagens f e g , a seguinte condição é satisfeita:

$$f(u) \leq g(u) \Rightarrow \xi(f)(u) \leq \xi(g)(u), \forall u \in S \quad (10)$$

De mesma forma, a transformação é idempotente se [24]:

$$\xi(\xi(f))(u) = \xi(f)(u), \forall u \in S \quad (11)$$

ou seja, a imagem f não é alterada por aplicações sucessivas da mesma transformação ξ , se ξ é idempotente. São as propriedades de crescimento e idempotência que fazem com que a abertura e o fechamento sejam classificados como filtros morfológicos, já que uma transformação ξ é um filtro morfológico se possuir ao mesmo tempo as duas características [23].

Existem também algumas propriedades que são específicas para cada operação: a anti-extensividade e a extensividade. Em álgebra, uma operação é dita ser extensiva se o conjunto resultante da operação for superconjunto do conjunto inicial, e é chamada de anti-extensiva no caso em que o conjunto resultante é subconjunto do conjunto inicial [26].

Nesse caso, a abertura é uma transformação anti-extensiva, pois:

$$A_B \subset A \quad (12)$$

Já o fechamento é uma transformação extensiva, ou seja:

$$A^B \supset A \quad (13)$$

Segundo [27], a extensão das características de abertura/fechamento no caso binário para a abertura/fechamento em níveis de cinza, é imediata. Dessa forma, as características da anti-extensividade da abertura (equação 14) e a extensividade do fechamento (equação 15) são definidas como:

$$\gamma_g(f)(u) \leq f(u), \forall u \in S \tag{14}$$

$$f(u) \leq \phi_g(f)(u), \forall u \in S \tag{15}$$

3.2.3 Gradiente Morfológico

A partir dos elementos básicos da Morfologia Matemática (erosão e dilatação), vários operadores puderam ser definidos, entre eles os responsáveis para realizar as operações de abertura e fechamento citados anteriormente. Um outro operador bastante importante em muitas aplicações de processamento de imagens é o gradiente morfológico.

O gradiente morfológico tem como característica a capacidade de extrair as bordas de uma imagem, gerando uma outra imagem composta pelos contornos dos objetos presentes na imagem [21]. Esta operação é composta de três outras operações básicas da morfologia: a dilatação, erosão e subtração, e é definida, para o caso da morfologia binária, da seguinte forma:

$$X = (A \oplus B) - (A \ominus B) \tag{16}$$

onde, A é a imagem original e B o elemento estruturante X o gradiente morfológico binário. Quando aplicado à níveis de cinza, o gradiente morfológico de uma imagem f dado um elemento estruturante g é uma transformação dada por:

$$\psi_g(f)(u) := (\delta_g(f) \sim \varepsilon_g(f))(u), \forall u \in S \tag{17}$$

A figura 3.9 ilustra como é feita a detecção de bordas com gradiente morfológico binário, enquanto a figura 3.10 mostra na prática o resultado da aplicação de um gradiente morfológico.

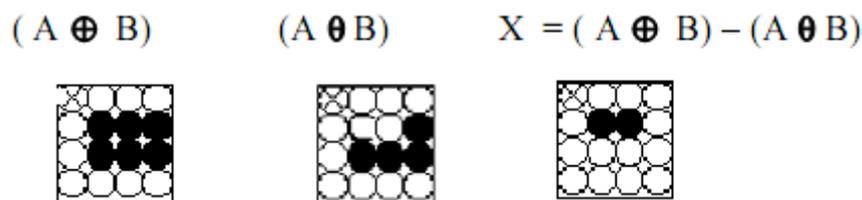


Figura 3.9 Detecção de bordas com gradiente morfológico binário [22]

É importante ressaltar que o gradiente depende da forma e do tamanho do elemento estruturante. Como a dilatação e a erosão representam respectivamente a filtragem máxima e mínima, cada ponto do gradiente morfológico é a diferença entre os valores máximos e mínimos

do ponto determinado pelo elemento estruturante. Em geral escolhe-se o elemento estruturante simétrico, para evitar deformações na extração das bordas.

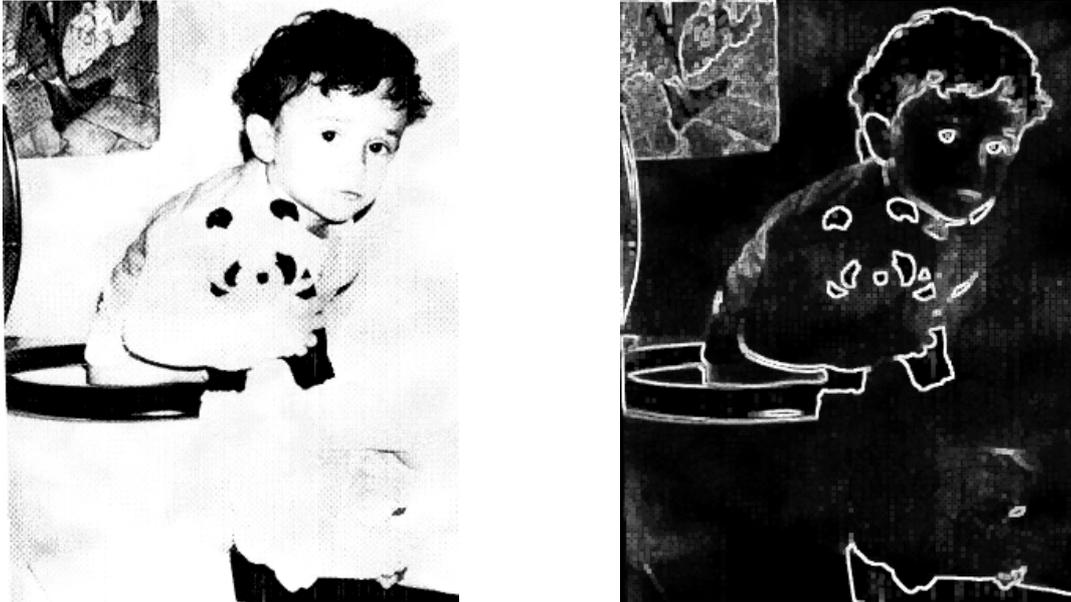


Figura 3.10 Aplicação de um gradiente morfológico sobre uma imagem [24]

3.3 Conclusão

Ao longo desse capítulo foram vistos vários conceitos sobre Morfologia Matemática, sendo esta uma técnica bastante interessante na área de processamento de imagens. Suas operações básicas, dilatação e erosão, são de extrema importância no processo de aprendizado dessa abordagem, já que a partir da combinação das mesmas é possível construir vários operadores específicos para as mais variadas situações, seja na eliminação de ruídos de uma imagem através da abertura ou fechamento, ou a detecção de bordas de um objeto através do operador gradiente morfológico.

No próximo capítulo será mostrado o experimento realizado com as técnicas descritas ao longo deste trabalho e, ao final, os resultados obtidos.

Capítulo 4

Experimentos e Resultados

4.1 Introdução

Nesse capítulo serão apresentados, de uma forma detalhada, os experimentos desenvolvidos, que envolvem a extração de características de imagens de imuno-histoquímica baseada na Morfologia Matemática, e a classificação dessas imagens através de técnicas de agrupamento, introduzidos em capítulos anteriores.

Em primeiro lugar, será mostrado o conceito do algoritmo de classificação utilizado (k -médias) e o seu funcionamento. Logo após, será mostrado o conjunto de dados utilizado no experimento. Em seguida, será analisado o processo de extração de características e como o vetor de características é gerado. Por fim, serão apresentados os resultados dos agrupamentos das imagens realizados no software *Weka* [28] pelo k -médias, onde será analisado o melhor desempenho desse algoritmo para diferentes configurações.

4.2 Algoritmo k -médias

O algoritmo k -médias é um dos mais simples algoritmos de aprendizado não supervisionado que resolve o problema de segmentação por agrupamento [29]. O objetivo deste algoritmo é encontrar a melhor divisão de P dados em k grupos $C_i, i = 1, \dots, k$, de maneira que a distância

total entre os dados de um conjunto e o seu respectivo centro, somada por todos os conjuntos, seja minimizada.

De maneira geral, esse método consiste em usar os valores dos primeiros n casos em um arquivo de dados, como estimativas temporárias das médias dos k conjuntos, onde k é o número de conjuntos especificado pelo usuário. Assim, o centro do conjunto inicial, chamado de centróide, é criado para cada caso em torno dos dados mais próximos e depois comparados com os pontos mais distantes e os outros conjuntos formados. Após essa etapa, ocorre um processo de atualização contínua e interativa, encontrando os centros dos conjuntos finais ao término do processo [30].

O método segue os passos definidos para cumprir o objetivo de agrupar os objetos em k conjuntos. Primeiramente, definem-se k centros, um para cada conjunto, sendo suas posições iniciais de muita influência no resultado do algoritmo. Em seguida, a região de propriedade de cada centróide é determinada, a qual agrupa um grupo de objetos similares. Depois de determinar as regiões, inicia-se a etapa interativa do algoritmo, no qual o centro do conjunto é recalculado de modo a minimizar uma determinada função objetivo. Esta função é definida através do método dos mínimos quadrados, por:

$$J = \sum_{j=1}^k \sum_{i=1}^n \|x_i^{(j)} - c_j\|^2 \quad (18)$$

onde $\|x_i^{(j)} - c_j\|^2$ é a medida de distância entre um ponto qualquer $x_i^{(j)}$ até o centro do conjunto c_j em questão. Então J (função objetivo) é o indicador de distância dos n objetos contidos em seus respectivos conjuntos [31]. Depois de realizar os cálculos, um novo centróide para cada conjunto é escolhido, reiniciando o processo. O processo termina quando a variação de cada centro do conjunto é mínima, chegando a uma situação em que todos os pontos já estejam nos conjuntos dos seus vetores médios mais próximos.

De uma forma prática e ilustrada (figura 4.1), será mostrado passo a passo como o k -médias funciona:

1. Aleatoriamente são gerados 3 centróides ($k=3$);
2. Atribuir a cada um dos objetos o conjunto que tem o centróide mais próximo
3. As posições dos centróides são recalculadas (nessa etapa nota-se que alguns pontos são deslocados de acordo com o resultado);
4. Se a posição dos centróides não mudar, passa-se para a próxima etapa. Se não, voltará para a segunda etapa.
5. Resultado do processo, com todos os pontos agrupados.

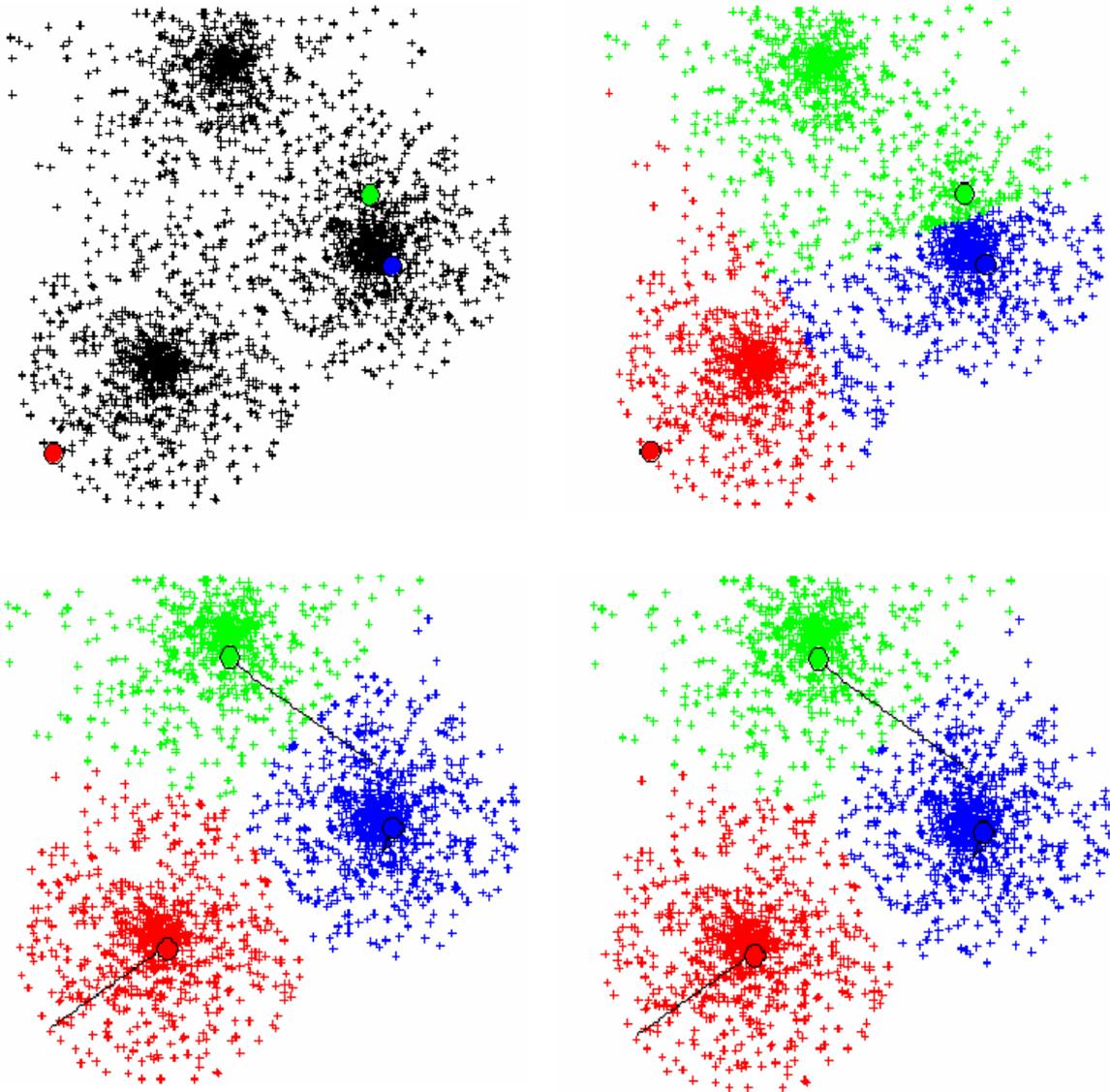


Figura 4.1 Etapas do algoritmo k-médias [30]

De maneira geral, o k -médias apresenta bons resultados para conjuntos de dados com *clusters* densos, compactos e bem separados uns dos outros. Entretanto, a necessidade de especificar a quantidade de conjuntos k (o número de *clusters*) é visto como uma desvantagem em relação a outros algoritmos de agrupamento, visto que não se sabe se os *clusters* escolhidos são de fato suficientes para representar os dados, influenciando no resultado do algoritmo.

4.3 Conjunto de Dados

Nesse trabalho, foram usadas imagens de imuno-histoquímica da placenta e do pulmão, do tipo CD68, que representa o tipo de antígeno. Para diferenciar os tipos de imagens, foi utilizado na nomenclatura as letras P e NP, onde P significa imagens de pulmão, enquanto NP quer dizer placenta. Outro detalhe importante é que as imagens podem ter dois tipos de zoom cada. A figura 4.2 ilustra os tipos de imagens utilizadas.

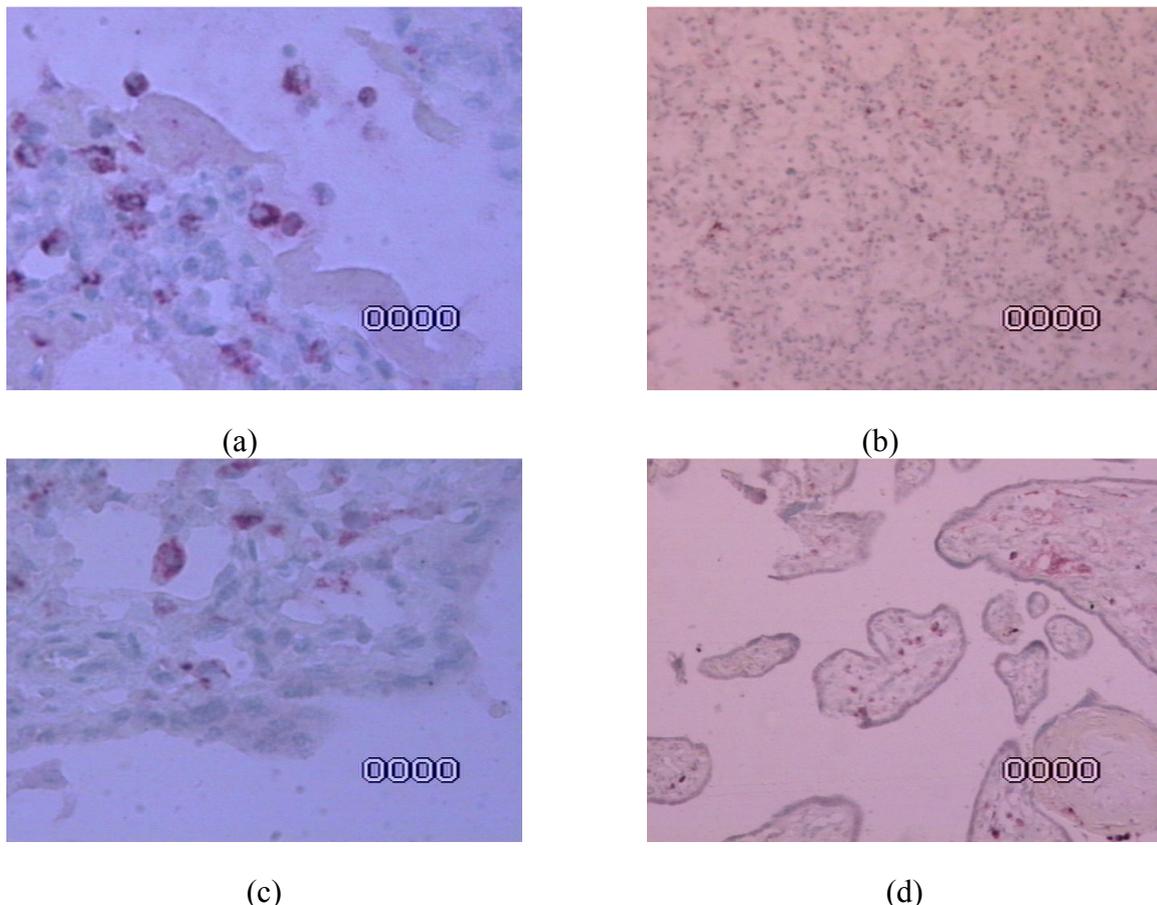


Figura 4.2 Tipos de imagens: (a) NP001a, (b) NP9759b, (c) P9824a e (d) P98289b

4.4 Geração dos Vetores de Características

Nos sistemas CBIR comuns, as características visuais presentes nos bancos de dados são descritas por um vetor de características multidimensional, como dito no Capítulo 2. Os vetores de características utilizados nesse trabalho foram obtidos através da ferramenta chamada ANIMED, que na verdade é um sistema para classificação de imagens médicas por fotomicrometria [21]. Na figura 4.3 é mostrada a visão geral do programa.

Para se obter os vetores de características, foi separado um grupo de quarenta imagens coloridas de imuno-histoquímica, sendo vinte da classe NP (placenta) e vinte da classe P (pulmão). Cada uma dessas imagens foi aberta no ANIMED, onde era aplicado a mesma um espectro morfológico do tipo quadrado 3x3 (figura 4.4). A cada aplicação do espectro, uma banda de cor era selecionada, representada pelos valores 0, 1 e 2. Com isso, foi possível tanto a visualização da imagem de entrada em bandas de cores específicas como também retirar as características dessa imagem de acordo com as mesmas, associando cada umas das 3 bandas escolhidas ao vermelho, azul e verde. A figura 4.5 ilustra essa etapa.

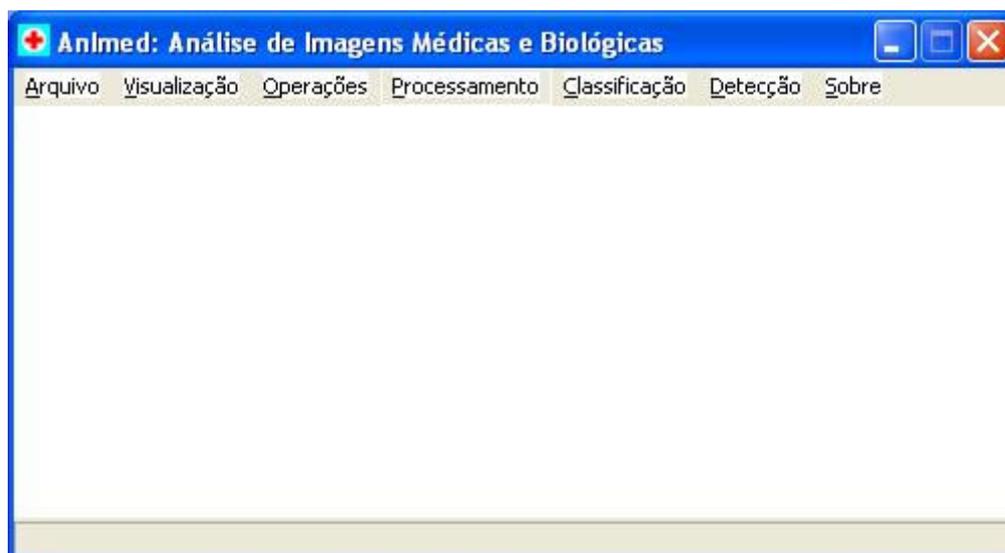


Figura 4.3 Tela principal do ambiente ANIMED



Figura 4.4 Seleção do espectro morfológico

Depois de selecionar que banda de cor utilizar na extração, foi necessário escolher quantas interações seriam realizadas para se obter as características. O número de iterações, nesse caso, define quantos atributos a imagem vai ter ao final da extração. Nesse trabalho, para todas as imagens foram utilizadas 25 iterações (figura 4.6).

Após realizar todas essas etapas, o sistema nos repassa como resultado um gráfico que mostra o espectro de padrões obtido. Para cada banda de cor selecionada, um novo espectro será obtido, como mostrado na figura 4.7. Cada um desses espectros pode ser salvo em um arquivo de texto no formato TXT, onde são armazenados todos os valores para essa configuração.



Figura 4.5 Tela de seleção da banda de cor no ANIMED



Figura 4.6 Tela de seleção do número de iterações



Figura 4.7 Exemplos de espectros de padrões obtidos

Dessa forma, foi obtido para cada imagem um vetor de características composto por vinte e cinco posições, de acordo com o número de iterações realizadas no ANIMED. Entretanto, as características obtidas dizem respeito à aplicação do espectro a uma banda de cor da imagem. Como as imagens analisadas são coloridas, foi necessário juntar os arquivos de cada banda em um único arquivo, gerando assim um vetor final com setenta e cinco características. Esses vetores foram usados para criar um arquivo *ARFF* para realizar a classificação em uma outra ferramenta denominada *Weka*, que será mostrada na próxima seção.

4.5 Resultados das Classificações

Para realizar os testes de classificação das imagens, foi utilizada uma ferramenta chamada *Weka*, que contém uma coleção de algoritmos e ferramentas de visualização para a análise de dados.

A primeira etapa na utilização do *Weka* foi a importação da base de dados que seria classificada. Para isso, o arquivo criado foi salvo em formato *ARFF*, um dos aceitos pelo software. Este tipo de arquivo é composto de três partes: a primeira, chamada relação, indica o que será estudado. Localizada na primeira linha do arquivo, é representada pelo código '@relation' seguido de uma palavra chave que identifica essa relação. A segunda parte é composta pelos atributos, onde cada linha deve ser iniciada com o código '@atributte', seguido do nome do atributo e do seu tipo. Por fim, na terceira parte se encontra os dados, que são dispostos depois de uma linha que contém o código '@data'. Cada linha deve corresponder a uma instância e deve ter valores separados por vírgula correspondentes e na mesma ordem dos atributos da parte atributos [32].

O arquivo criado contém quarenta instâncias na parte de dados, que são todos os vetores de características das imagens discretizadas no ANIMED, onde cada vetor deste é composto por setenta e cinco atributos numéricos. A figura 4.8 ilustra uma parte do arquivo *ARFF* criado.

Depois de abrir o arquivo no *Weka*, foi selecionado o algoritmo de classificação *SimplekMeans*, na aba de *cluster*. Na realização dos testes com este algoritmo, os parâmetros *numClusters* e *seed* foram alterados várias vezes com o objetivo de achar o melhor resultado de classificação. O parâmetro *numClusters*, como o nome já diz, representa o número de *clusters* que serão usados na classificação, enquanto o parâmetro *seed* é usado para gerar um valor aleatório que é usado para a designação inicial de instâncias ao *cluster* [33].

```

@RELATION vetores

@ATTRIBUTE caract1 REAL
@ATTRIBUTE caract2 REAL
@ATTRIBUTE caract3 REAL
@ATTRIBUTE caract4 REAL
@ATTRIBUTE caract5 REAL
@ATTRIBUTE caract6 REAL
@ATTRIBUTE caract7 REAL
@ATTRIBUTE caract8 REAL
@ATTRIBUTE caract9 REAL
@ATTRIBUTE caract10 REAL
@ATTRIBUTE caract11 REAL
@ATTRIBUTE caract12 REAL
@ATTRIBUTE caract13 REAL
@ATTRIBUTE caract14 REAL
@ATTRIBUTE caract15 REAL
@ATTRIBUTE caract16 REAL
@ATTRIBUTE caract17 REAL
@ATTRIBUTE caract18 REAL
@ATTRIBUTE caract19 REAL
@ATTRIBUTE caract20 REAL
@ATTRIBUTE caract21 REAL
@ATTRIBUTE caract22 REAL
@ATTRIBUTE caract23 REAL
@ATTRIBUTE caract24 REAL
@ATTRIBUTE caract25 REAL
@ATTRIBUTE caract26 REAL
@ATTRIBUTE caract27 REAL
@ATTRIBUTE caract28 REAL
@ATTRIBUTE caract29 REAL
@ATTRIBUTE caract30 REAL

@ATTRIBUTE caract71 REAL
@ATTRIBUTE caract72 REAL
@ATTRIBUTE caract73 REAL
@ATTRIBUTE caract74 REAL
@ATTRIBUTE caract75 REAL

@DATA
%NP
0.000000000000000E+0000, 2.56610107943516E-0002, 1.02889463510553E-0002, 7.85178491184323E-0003,
7.73565231772084E-0003, 5.67776908218903E-0003, 5.51531585609689E-0003, 5.32832486784099E-0003,
4.99739609282800E-0003, 4.40718709985989E-0003, 4.48073631908710E-0003, 4.06374385258583E-0003,
4.13084363389010E-0003, 4.51447841153163E-0003, 3.80484803821551E-0003, 3.06228709113394E-0003,
3.51680161866369E-0003, 3.51718602224851E-0003, 4.50138733389334E-0003, 4.18564250048040E-0003,
4.46017072729969E-0003, 3.79165018180369E-0003, 2.93577560022162E-0003, 2.98382604832304E-0003,
2.91234833730286E-0003, 0.000000000000000E+0000, -3.00907209404041E-0002, 1.08776804396552E-0002,
8.30106448588653E-0003, 8.17828677820006E-0003, 6.00265134824734E-0003, 5.83090253942591E-0003,
5.63321191631031E-0003, 5.28334737818794E-0003, 4.65936659346364E-0003, 4.73712430314958E-0003,
4.29627150427392E-0003, 4.36721073883641E-0003, 4.77279712002061E-0003, 4.02256165685153E-0003,
3.23751133063491E-0003, 3.71803320498043E-0003, 3.71843960415997E-0003, 4.75895697018394E-0003,
4.42514519965354E-0003, 4.71538194704402E-0003, 4.00860861835386E-0003, 3.10376084510078E-0003,
3.15456074254368E-0003, 3.07899307288083E-0003, 0.000000000000000E+0000, 3.21489343163091E-0001,
7.16502143931025E-0003, 5.46782977680105E-0003, 5.38695730470340E-0003, 3.95388758122589E-0003,
3.84075832502012E-0003, 3.71054145012373E-0003, 3.48008910962638E-0003, 3.06907908547005E-0003,
3.12029732634983E-0003, 2.82991191072307E-0003, 2.87663888890088E-0003, 3.14379466101594E-0003,
2.64962191821860E-0003, 2.13251696652561E-0003, 2.44903201317026E-0003, 2.44929970432473E-0003,
3.13467829003472E-0003, 2.91479975013709E-0003, 3.10597584958811E-0003, 2.64043118858337E-0003,
2.04441683325857E-0003, 2.07787822756156E-0003, 2.02810254456948E-0003

%2
0.000000000000000E+0000, 2.59266188914309E-0002, 9.17711767424405E-0003, 6.58235657519324E-0003,
6.68406931754578E-0003, 4.89421326333730E-0003, 4.79176923884477E-0003, 5.27232914630574E-0003,
4.47841484186143E-0003, 4.23947915677060E-0003, 4.21792784193931E-0003, 3.66454083665517E-0003,
4.22719791847850E-0003, 3.72564591194034E-0003, 3.84002704195320E-0003, 4.19033269530401E-0003,
4.50730048639081E-0003, 3.13758752990163E-0003, 2.99989140460222E-0003, 4.01020069758724E-0003,
3.78249234417501E-0003, 4.24666292838105E-0003, 3.60384441903645E-0003, 3.42070125956680E-0003,

```

Figura 4.8 Partes do arquivo ARFF

Nos primeiros testes realizados, foi definido como dois o número de *clusters* iniciais, já que a intenção era classificar as imagens em dois *clusters* distintos, um para as imagens da classe NP e

outro para a classe P. Como o algoritmo k -médias é não-supervisionado, não foi definido previamente qual *cluster* representaria cada classe. O sistema ficou responsável por fazer uma “triagem” das imagens, de forma que o *cluster* que tivesse o maior número de um tipo de imagem, seria daquele tipo, ou seja, se a maioria de imagens naquele *cluster* for de P, então ele é considerado P. O melhor resultado obtido nessa primeira etapa foi com dois *clusters* e com o valor de *seed* igual a cinco. Dessa forma, o sistema classificou 31 imagens no *cluster* 0 e 9 no *cluster* 1. A figura 4.9 mostra o resultado encontrado.

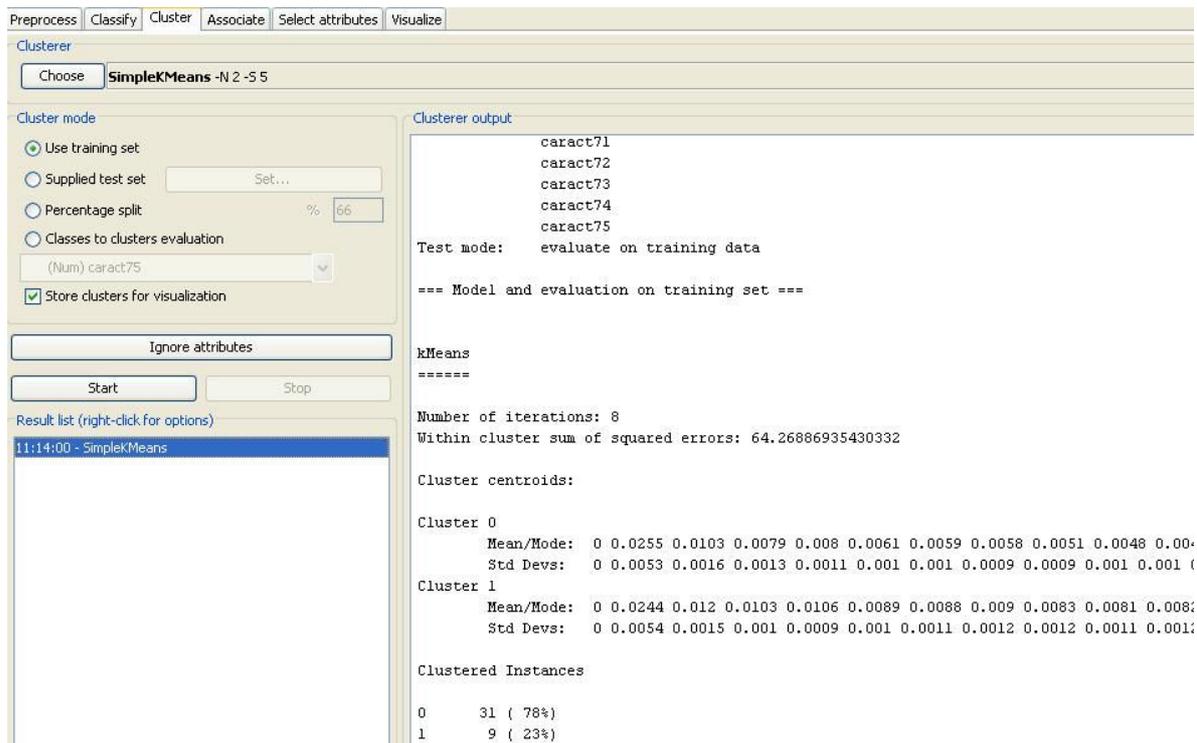


Figura 4.9 Resultados da classificação com 2 *clusters*

Para identificar quais imagens pertenciam a qual *cluster*, foi calculada a distância euclidiana entre o vetor e o centro de cada *cluster*, de forma que, a menor distância indicaria em qual *cluster* o vetor foi separado. Esse cálculo foi realizado numa planilha do *Excel* e foi constatado que, para o *cluster* 0 foram atribuídas 11 imagens da classe NP e 20 da classe P. Nesse caso, o *cluster* 0 foi considerado P, tendo classificado corretamente todas as imagens da classe. Entretanto, foram classificadas também 11 imagens do tipo NP, ocasionando um erro de 55% na classificação. Essa alta taxa de erro indica uma menor similaridade na classe NP, tornando mais difícil a classificação correta das imagens dessa classe. As imagens restantes (9 imagens do tipo NP) foram classificadas corretamente no *cluster* 1.

Devido a alta taxa de erro encontrada, foi decidido tentar uma nova classificação, só que utilizando agora quatro *clusters*. O melhor resultado obtido foi o realizado com quatro *clusters* e o *seed* igual a vinte e dois. Para essa configuração, o sistema classificou 7 imagens no *cluster* 0, 5 no *cluster* 1, 9 no *cluster* 2 e as 19 restantes no *cluster* 3, que pode ser verificado na figura 4.10. Depois do cálculo da distância euclidiana, foi verificado que dentre as 7 imagens atribuídas ao *cluster* 0, 4 eram do tipo NP e 3 do tipo P, sendo esse *cluster* definido com NP. Outros dois *clusters* foram classificados como NP: o *cluster* 1, que conteve 3 imagens da classe NP e 2 da classe P e o *cluster* 3, com 12 NP e 7 P. Apenas o *cluster* 2 foi definido como P, já que conteve apenas 1 imagem do tipo NP e 8 do tipo P.

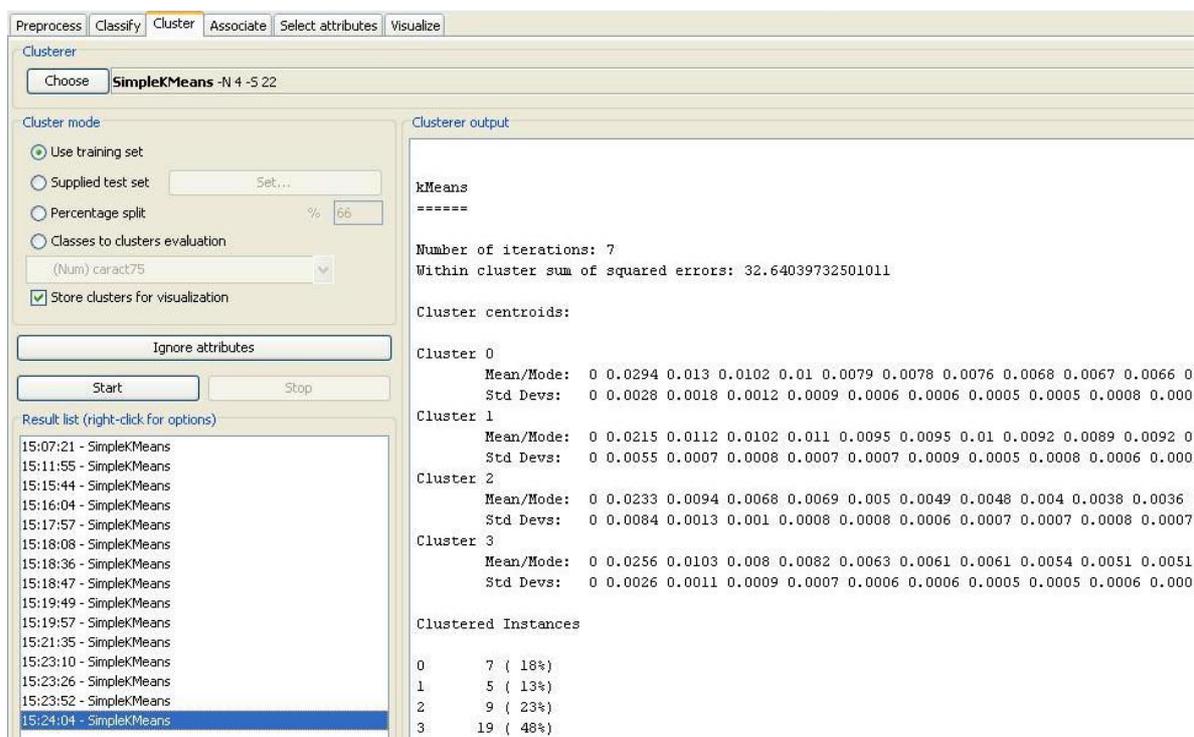


Figura 4.10 Resultado da classificação com 4 *clusters*

Mesmo sabendo que os dados foram divididos em duas classes, foram usados quatro *clusters* nessa etapa apenas para gerar resultados melhores. Na verdade, os 3 *clusters* definidos como NP serviram para modelar o conjunto NP real, que nesse caso teria 19 imagens NP e 12 P, enquanto o P teria as características do *cluster* 2. Dessa forma, o cluster NP classificou corretamente 95% das imagens dessa classe, mas classificou também 60% das imagens classe P.

Depois de analisados os resultados obtidos, pode-se concluir que os resultados foram satisfatórios, visto que é a primeira vez que é usado informação de cor e forma de uma imagem ao mesmo tempo para a classificação de imagens. O sistema conseguiu realizar a “triagem” das imagens, classificando-as mesmo que nem sempre de maneira correta, em grupos distintos, facilitando o trabalho do especialista em analisar tais imagens.

4.6 Conclusão

Nesse capítulo foram mostrados alguns resultados obtidos através da combinação das técnicas de CBIR, em conjunto com a Morfologia Matemática, para a recuperação e classificação de imagens. Foi visto como o espectro de padrões consegue extrair características que possam ser utilizadas para identificar imagens mediante um vetor de características.

Também foi mostrado como o algoritmo k -médias foi utilizado para realizar a classificação das imagens na ferramenta *Weka*.

No próximo capítulo, serão dadas as considerações finais deste trabalho, bem como a sugestão de trabalhos futuros.

Capítulo 5

Considerações Finais e Trabalhos Futuros

O estudo em recuperação de imagens por conteúdo ainda possui muitos desafios a serem atingidos, mas mesmo assim é uma área de aplicação onde o interesse vem crescendo bastante, devido aos problemas encontrados em outros métodos de recuperação de imagens.

Este trabalho apresentou uma nova proposta para a recuperação de imagens baseada em conteúdo, ao utilizar as informações de cor e forma da imagem simultaneamente para representá-la. Esta extração de características das imagens se deu através do espectro de padrões, uma técnica da Morfologia Matemática, que foi aplicado utilizando o ambiente ANIMED.

As imagens utilizadas neste trabalho foram imagens médicas de imuno-histoquímica de placenta e pulmão. A classificação dessas imagens foi realizada no *software Weka*, utilizando como algoritmo de agrupamento o k -médias.

Para a realização deste trabalho, foi necessário um estudo prévio das técnicas de CBIR e da Morfologia Matemática, apresentados na forma de revisão bibliográfica nos capítulos 2 e 3 respectivamente.

Com isso, a proposta de utilizar técnicas CBIR à análise de imagens médicas de imuno-histoquímica, a fim de ajudar a recuperação dessas imagens por parte dos médicos, gerando uma espécie de triagem virtual, para facilitar a análise e, principalmente, otimizar o tempo de análise por parte do especialista (patologista ou biomédico) foi alcançada, gerando resultados interessantes, visto que, como já foi dito, foi apresentada uma proposta inédita para realizar este experimento.

Na próxima seção, serão discutidos os trabalhos futuros que podem ajudar a incrementar o trabalho realizado.

5.1 Trabalhos Futuros

Após a realização deste trabalho, novas idéias surgiram para a continuação futura do mesmo. Pode-se, por exemplo, melhorar a parte da classificação das imagens, experimentando outros classificadores, como Redes Neurais e Lógica Fuzzy, através do algoritmo *fuzzy c-médias (fuzzy c-means)*.

Outra alternativa seria gerar vetores de características de dimensões maiores para uma caracterização mais precisa da imagem, fazendo com que o índice de acerto na classificação se torne maior.

Bibliografia

- [1] CASTAÑÓN, C. A. B. **Recuperação de Imagens por Conteúdo através de Análise Multiresolução por Wavelets**. 2003. Dissertação de Mestrado em Ciência de Computação e Matemática Computacional, Universidade de São Paulo - USP, São Carlos.
- [2] ERPEN, L. R. C. **Recuperação de Informação Visual**. Universidade Federal do Rio Grande do Sul – UFRGS, Porto Alegre.
- [3] FENG, D; LONG, F. e ZHANG, H. **Fundamentals of Content-Based Image Retrieval**. Disponível em: <http://old.hki.unikoeln.de/teach/ws0708/VIP/tag9/ch01_Long_v40-proof.pdf> Acesso em: 26 de agosto de 2008.
- [4] CASTAÑÓN, C. A. B; TRAINA, A. J. M. **Caracterização de Imagens através de Análise Multiresolução por Wavelets para Sistemas CBIR**. Universidade de São Paulo – USP, São Carlos.
- [5] GONZALES, R. C. e WOODS, E. R. **Processamento de Imagens Digitais**. São Paulo: Edgard Blucher, 2000.
- [6] QBIC em: <http://www.qbic.almaden.ibm.com>. Acesso em 31 de agosto de 2008.
- [7] Virage em : <http://www.virage.com>. Acesso em 31 de agosto de 2008.
- [8] VisualSEEK em : <http://www.ctr.columbia.edu/VisualSEEK>. Acesso em 1 de setembro de 2008.
- [9] TUCERYAN, M. e JAIN, A. K. **Texture Analysis in Handbook of Pattern Recognition and Computer Vision**. World Scientific Publishing Company; 1993. p. 235–276.

- [10] BUENO, J. M. **Suporte à Recuperação de Imagens Médicas Baseada em Conteúdo Através de Histogramas Métricos**. 2002. Tese de doutorado, Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo – USP, São Carlos.
- [11] LONG, H. Z. e LEOW, W. K. **Perceptual Texture Space for Content-Based Image Retrieval**. World Scientific, 2000.
- [12] SANTOS, C. F. **Uso de Saliências do Contorno via Esqueletização para Recuperação de Imagens**. 2007. Dissertação de Mestrado em Ciência da Computação, Universidade Federal de Uberlândia.
- [13] FALOUTSOS, C. **Searching Multimedia Database by Content**. Kluwer Academic Publishers, 1996.
- [14] THEODORIDIS, S. **Pattern Recognition**. Academic Press. Greece, 1999.
- [15] BOBER, M. **MPEG-7 Visual Shape Descriptors** In: IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, No. 6, Junho 2001, p.716-719.
- [16] ZHANG, D. e LU, G. **A Comparative Study of Three Region Shape Descriptors**. Gippsland School of Computing and Information Technology, Monash University, Australia.
- [17] SAFAR, M; SHAHABI, C e SUN, X. **Image Retrieval by Shape: A Comparative Study**. Departament of Computer Science, University of Southern California, Los Angeles.
- [18] ZHANG, D. S. e LU, G. J. **Shape Retrieval Using Fourier Descriptors**. In Proc. Int. Conference on Multimedia and Distance Education. USA, Junho 2001, pp.1-9.
- [19] STATELLA, T. e DA SILVA, E. A. **Morfologia Matemática: Extração de Feições a partir de Imagens Orbitais**. Departamento de Cartologia, Universidade Estadual Paulista – UNESP, São Paulo.
- [20] FACON, J. **Morfologia Matemática: Teorias e Exemplos**. Editora Universitária Champagnat da Pontifícia Universidade Católica do Paraná. Curitiba. 1996.

- [21] SANTOS, W. P. **Análise de Imagens Digitais em Patologia utilizando Morfologia Matemática e Lógica Nebulosa**. 2003. Dissertação de Mestrado em Engenharia Elétrica, Universidade Federal de Pernambuco - UFPE, Recife.
- [22] WANGENHEIM, A; DE SOUZA, A. I. e SANTOS, C. A. **Morfologia Matemática**. Universidade Federal de Santa Catarina - UFSC, Santa Catarina, p. 49-80.
- [23] CANDEIAS, A. L. B. **Aplicação da Morfologia Matemática à Análise de Imagens de Sensoriamento Remoto**. PhD thesis, Departamento de Computação Aplicada do Instituto Nacional de Pesquisas Espaciais, São José dos Campos, Brasil, 1997.
- [24] CAMAPUM, J. F. **Introdução ao Processamento de Imagens Digitais**. Departamento de Engenharia Elétrica, Universidade de Brasília.
- [25] **Processamento de imagens e Morfologia Matemática**. Disponível em: <http://www.joinville.udesc.br/processamentodeimagens/morfologia_matematica.html> Acesso em: 10 de outubro de 2008.
- [26] SERRA, J. **Image Analysis and Mathematical Morphology – Volume 2: Theoretical Advances**, Academic Press Inc., Ecole des Mines de Paris, 1988.
- [27] DOUGHERTY, E. R; LOTUFO, R. A. **Hands-on Morphological Image Processingods in Imaging**, SPIE Press, 2003.
- [28] WEKA em: <http://www.cs.waikato.ac.nz/ml/weka/>. Acesso em 25 de outubro de 2008.
- [29] JAIN, A. e DUBES, R. **Algorithms for Clustering Data**. Prentice-Hall, p. 155-166. 1988.
- [30] **Clusterização**. Disponível em: <<http://www.fredbf.com/disciplinas/unibrathec/dsi/seminario/cluster/Trabalho%20e%20agrupamento.pdf>> Acesso em: 26 de outubro de 2008.
- [31] MOORE, A. **K-means and Hierarchical Clustering - Tutorial Slides**. Disponível em: <<http://www-2.cs.cmu.edu/~awm/tutorials/kmeans.html> > Acesso em: 26 de outubro de 2008.

- [32] SANTOS, R. **Weka na Munheca: um guia para o uso do Weka em scripts e integração com aplicações Java**. Disponível em: <<http://www.lac.inpe.br/~rafael.santos/Docs/CAP359/2005/weka.pdf>> Acesso em: 26 de outubro de 2008.
- [33] **K-Means Clustering in Weka**. Disponível em: <<http://maya.cs.depaul.edu/~Classes/Ect584/Weka/k-means.html>>