

## Resumo

Os sistemas de recuperação de imagens baseada em conteúdo (CBIR – *Content-based Image Retrieval*) consideram o conteúdo visual de uma imagem de consulta como chave de busca para retornar outras imagens cujos conteúdos são similares à imagem consultada. Tal técnica tem como objetivo a extração de informações das imagens que possam caracterizar o seu conteúdo adequadamente. Este trabalho está direcionado na geração de vetores de características considerando um banco de imagens médicas de imuno-histoquímica a fim de propiciar consultas CBIR.

A textura é o conteúdo visual utilizado para representar as imagens neste trabalho. Para extrair a informação de textura, a transformada *wavelet* é aplicada às imagens, possibilitando sua representação e localização espacial. Descritores estatísticos são utilizados nos sub-espacos *wavelets* gerados para compor os vetores de características das imagens de imuno-histoquímica. Por fim, os resultados dos agrupamentos desses vetores são analisados através do algoritmo *k*-médias e comparados com a interpretação semântica dos usuários realizada através de consultas a um ambiente *web* desenvolvido.

## Abstract

Content-based image retrieval (CBIR) considers the visual content of a query image as a key to search for other images whose contents are similar to the consulted image. The objective of this technique is to extract information of the images to characterize its content adequately. This work address the generation of feature vectors for immunohistochemistry medical image databases in order to propitiate CBIR consultations.

Texture is the visual content used to represent the images in this work. To extract texture information, *wavelet* transform is applied to the images. This allow the representation and space localization of the texture. Statistics descriptors are used in the *wavelets* sub-spaces generated to compose the feature vectors of the immunohistochemistry images. Finally, the results of grouping these vectors are analyzed through the *k*-means clustering algorithm and compared with the semantics interpretation of the users through consultations in a web environment developed.

# Sumário

<b>Índice de Figuras</b>	<b>v</b>
<b>Índice de Tabelas</b>	<b>vi</b>
<b>Tabela de Símbolos e Siglas</b>	<b>vii</b>
<b>1 Introdução</b>	<b>9</b>
1.1 Organização do Trabalho	10
<b>2 Recuperação de Imagens Baseada em Conteúdo - CBIR</b>	<b>11</b>
2.1 Introdução	11
2.2 Atributos das imagens	12
2.2.1 Cor	12
2.2.2 Forma	14
2.2.3 Textura	15
2.3 Abordagens estatísticas	16
2.3.1 Momentos do histograma	16
2.3.2 Matriz de Co-Ocorrência de Níveis de Cinza - GLCM	17
<b>3 Wavelets e Extração de Características</b>	<b>20</b>
3.1 Introdução	20
3.2 Transformada <i>Wavelet</i>	21
3.3 Análise de Multiresolução	22
3.4 Algoritmo	23
3.5 Extração de Características	24
<b>4 Técnicas de Análise de Agrupamento</b>	<b>26</b>
4.1 Introdução	26
4.2 Algoritmos de Agrupamento	26
4.2.1 K - Médias	27
4.2.2 Mapas Auto-Organizados - SOM	28
4.3 Medidas de Distância	29
<b>5 Resultados e Discussões</b>	<b>31</b>
5.1 Introdução	31
5.2 Conjuntos de dados	31
5.3 Geração dos Vetores de Características	32
5.4 Resultados dos Agrupamentos	34
5.4.1 Agrupamento no Weka	34
5.4.2 Agrupamento Segundo Interpretação Semântica	34
5.4.3 Comparação dos Resultados	37
5.4.4 Dificuldades Encontradas	39
<b>6 Conclusão e Trabalhos Futuros</b>	<b>41</b>

6.1	Contribuições
6.2	Trabalhos Futuros

41

42

# Índice de Figuras

<b>Figura 1.</b>	(a) Imagem original e (b) imagem saturada .....	13
<b>Figura 2.</b>	(a) Histograma da imagem original e (b) histograma da imagem saturada .....	13
<b>Figura 3.</b>	Formas de imagens [20] .....	14
<b>Figura 4.</b>	Exemplos de texturas .....	15
<b>Figura 5.</b>	Orientações usadas para construir as matrizes de co-ocorrência [17].....	17
<b>Figura 6.</b>	Comparação entre a transformada de Fourier e <i>wavelet</i> em função do tempo [32]...	21
<b>Figura 7.</b>	Decomposição de imagem em um domínio 2D ao aplicar a transformada <i>wavelet</i> [40] .....	23
<b>Figura 8.</b>	Transformada <i>wavelet</i> sobre um domínio 2D. (a) Imagem original, (b) 1 nível de decomposição da Transformada e (c) 2 níveis de decomposição.....	24
<b>Figura 9.</b>	Processo de extração de características [31] .....	25
<b>Figura 10.</b>	(a) Dados a serem agrupados e (b) primeiros <i>k</i> centros são escolhidos aleatoriamente.....	27
<b>Figura 11.</b>	(a) Elementos são alocados aos <i>clusters</i> com o centro mais próximos e (b) um centróide é calculado para representar o <i>cluster</i> .....	28
<b>Figura 12.</b>	(a) Alocação dos elementos aos clusters e cálculo de novos centróides até um critério de convergência e (b) algoritmo converge resultando nos clusters finais.....	28
<b>Figura 13.</b>	(a) CD34 NP001a, (b) CD34 P003b, (c) CD68 NP993a e (d) CD68 P9824b .....	32
<b>Figura 14.</b>	Transformada <i>wavelet</i> Daubechies-4 nível 2 aplicada em imagem de imuno-histoquímica .....	33
<b>Figura 15.</b>	Parte do arquivo “vetores.arff” .....	35
<b>Figura 16.</b>	Tela para cadastrar imagens.....	35
<b>Figura 17.</b>	Visualização da imagem cadastrada no banco .....	36
<b>Figura 18.</b>	Tela para consultar a opinião dos usuários sobre a semelhança das imagens .....	37
<b>Figura 19.</b>	Visualização dos resultados da clusterização no <i>Weka</i> .....	38

# Índice de Tabelas

<b>Tabela 1.</b>	Resultados da clusterização através do algoritmo <i>k</i> -médias no <i>Weka</i> .....	38
<b>Tabela 2.</b>	Resultados da clusterização através da interpretação semântica dos usuários.....	38

# Tabela de Símbolos e Siglas

(Dispostos por ordem de aparição no texto)

CBIR - Content Based Image Retrieval (Recuperação de Imagens Baseada em Conteúdo)

GLCM - Grey Level Co-occurrence Matrix (Matriz de Co-Ocorrência de Tons de Cinza)

SOM - Self Organized Maps (Mapas Auto-Organizáveis)

SIVP - Scilab Image and Video Processing toolbox

SWT - Scilab Wavelet Toolbox

PHP - Hypertext Preprocessor (Preprocessador de Hipertexto)

SIP - Scilab Image Processing toolbox

# Agradecimentos

A Deus, Nele posso todas as coisas.

Aos meus pais, pelo apoio, preocupação e carinho sempre constantes e por serem meu refúgio em momentos difíceis.

Ao meu orientador, Prof. Wellington Pinheiro, não só por sua valiosa e imprescindível ajuda na orientação deste trabalho como também pelas conversas descontraídas e proveitosas que tivemos ao longo desta orientação.

A minhas irmãs, por serem minhas melhores amigas e me proporcionarem momentos maravilhosos junto a elas.

A minha tia Tânia, pelos conselhos, palavras de tranquilidade e por ter me levado a igreja quando estava precisando desopilar.

Ao meu amigo André Rodovalho, pela grande ajuda e paciência dispensadas para realização de uma importante etapa deste trabalho.

A minha prima e amiga Thalita, pela compreensão e companherismo.

A minha grande amiga Lúcia, pela amizade, presença constante e momentos de lazer vivenciados.

A Izaura, amiga inseparável, por tudo que passamos juntas até agora. Pelos momentos difíceis enfrentados durante o curso, horas de estudos e madrugadas dedicadas a projetos, mas também pelos momentos felizes que foram muitos e pelo apoio que recebi em sua casa, tudo isto só fez fortalecer nossa amizade.

A amiga Angélica, sempre paciente, prestativa e solícita quando precisei.

Aos amigos Alberto, Alexandre e Pedro por estarem sempre presentes, pelas dúvidas tiradas, pelas caronas, enfim, pela amizade que construímos.

A Máfia, queridos amigos, pelos momentos maravilhosos e por tornar o ambiente da POLI um local mais agradável e divertido mesmo em situações inesperadas.

A todos o meu muito obrigada!



# Capítulo 1

## Introdução

Nos últimos anos, o interesse no potencial das imagens digitais aumentou enormemente devido, em parte, ao rápido crescimento do uso de imagens na *web*. Usuários estão explorando as oportunidades oferecidas pela habilidade de acesso e liberdade de manipular imagens armazenadas remotamente de várias maneiras. Entretanto, o processo de localizar uma imagem desejada dentro de uma base de dados grande e variada pode ser uma fonte de frustração considerável. Os problemas de recuperação de imagens estão tornando-se extensamente reconhecidos, e a busca por soluções uma área cada vez mais ativa para pesquisa e desenvolvimento [1].

Os problemas com métodos tradicionais de indexação de imagens conduziram à ascensão de novas técnicas para recuperar imagens baseadas em características derivadas da própria imagem. *CBIR – Content-based Image Retrieval* (Recuperação de imagens baseada em conteúdo) é uma aplicação que usa conteúdo visual, através de diferentes técnicas, para procurar imagens a partir de bases de dados com larga escala de imagens de acordo com o interesse do usuário.

A medicina tem sido uma das áreas que mais tem se beneficiado da interação com a computação. Atualmente, existe uma demanda muito grande para armazenar e consultar dados complexos, por exemplo, imagens coletadas em exames médicos, tais como radiografia, tomografia, seqüências de DNA, etc. Dentro deste contexto, a recuperação de imagens em bancos de dados multimídia, tem sido de grande funcionalidade para a medicina e a técnica *CBIR* tem se mostrado uma opção eficiente para extração de informações de imagens médicas, podendo caracterizar adequadamente os seus conteúdos.

Imuno-histoquímica é a aplicação de métodos e técnicas imunológicas ao estudo de células e tecidos e consiste na detecção de moléculas (antígenos) teciduais. Essa detecção é feita por meio de um anticorpo que, ao entrar em contato com um determinado antígeno, libera uma substância cromógena, ou seja, uma substância que marca o antígeno com uma determinada cor [2].

A imuno-histoquímica pode ser utilizada para caracterização do local de origem de neoplasias malignas através do estudo imuno-histoquímico de material de metástases, identificação de agentes infecciosos, caracterização de determinadas proliferações celulares como benignas ou malignas, avaliações de tumores de difícil diferenciação, etc.

Assim, uma técnica *CBIR* para análise de imagens de imuno-histoquímica utilizando descritores estatísticos de texturas é a proposta deste trabalho, a fim de auxiliar os patologistas na recuperação das imagens.

Uma alternativa para representação do conteúdo de imagens é utilizar o domínio de transformadas, onde a vantagem esperada de uma transformação é a efetiva caracterização de propriedades da imagem.

O uso da transformada *wavelet* tem se mostrado uma solução significativa no processo de extração de características, em especial da textura. Isto porque a transformada *wavelet* representa o sinal em diferentes bandas de frequência e em diferentes escalas, sendo possível a representação e localização espacial da textura. As *wavelets* foram aplicadas com sucesso na análise, classificação, caracterização e recuperação de imagens.

Neste contexto, o objetivo deste trabalho foi desenvolver uma técnica para a extração de características de imagens médicas de imuno-histoquímica, para recuperação de imagens por conteúdo, através da análise multiresolução por *wavelets*.

Essa técnica, implicou na aplicação de medidas estatísticas aos sub-espacos *wavelets* para extração e geração dos vetores de características que identifiquem as imagens, a fim de analisar os resultados obtidos pelo agrupamento de vetores semelhantes.

## 1.1 Organização do Trabalho

Este trabalho está organizado em 6 capítulos. Este primeiro Capítulo apresenta as considerações iniciais, a motivação para o desenvolvimento objetivos do trabalho.

O Capítulo 2 introduz o conceito de CBIR e extração de características, expõe os principais atributos utilizados para descrever o conteúdo das imagens, com ênfase especial no atributo de textura, no qual está baseado este trabalho e mostra as principais abordagens estatísticas para descrever textura.

O Capítulo 3 mostra conceitos relacionados à teoria das *wavelets*, aplicação da transformada *wavelet* na caracterização de imagens e à extração de características.

No Capítulo 4, são apresentadas algumas técnicas de agrupamento e a definição de métricas de distâncias para determinar o grau de similaridade entre imagens.

O Capítulo 5 descreve os métodos utilizados para extração de características das imagens e analisa os resultados dos agrupamentos obtidos.

Finalmente as Conclusões, Contribuições e propostas de possíveis trabalhos futuros à esta monografia são apresentadas no Capítulo 6.

## Capítulo 2

# Recuperação de Imagens Baseada em Conteúdo - CBIR

### 2.1 Introdução

Recuperação de Imagens Baseada em Conteúdo (*Content-Based Image Retrieval* – CBIR) é uma aplicação que usa conteúdo visual para procurar imagens a partir de bases de dados com larga escala de imagens de acordo com o interesse do usuário e tem sido uma área de pesquisa explorada desde os anos 90 [3]. Tal aplicação, surgiu da necessidade de se estender as técnicas tradicionais de recuperação de imagens em bancos de dados multimídia, incluindo também informações visuais das imagens que podem ser obtidas através de um processo automático. O objetivo principal da abordagem CBIR consiste em extrair informações das imagens que possam caracterizar adequadamente o seu conteúdo [4].

O termo conteúdo significa referência a cores, formas, texturas ou qualquer outra informação que possa ser derivada a partir da própria imagem para representá-la. Os conteúdos visuais da imagem são extraídos e descritos por um vetor multidimensional conhecido como “vetor de características”, que se torna um perfil descritivo da imagem, a partir do qual esta pode ser identificada como uma imagem. Para recuperar as imagens, usuários fornecem ao sistema de recuperação exemplos de imagens. As similaridades ou distâncias entre os vetores de características do exemplo e aqueles das imagens na base de dados são calculadas e a recuperação é executada com a ajuda de um esquema de índices que fornece um eficiente modo de buscar imagens na base de dados [3].

Diferentes implementações de CBIR fazem uso de diferentes tipos de consulta ao usuário, tais como a consulta por exemplo visual (*query by visual example* – QBVE) [4] e a recuperação semântica [6]. A consulta por exemplo visual envolve o fornecimento ao sistema CBIR de um exemplo de imagem na qual será baseada a busca das outras imagens a serem recuperadas. Os algoritmos de busca podem variar dependendo da aplicação, mas as imagens recuperadas devem compartilhar elementos em comum com o exemplo provido. Opções para o exemplo de imagem provido ao sistema incluem uma imagem pré-existente fornecida pelo usuário ou escolhida em um conjunto aleatório e também os usuários podem desenhar uma aproximação da imagem que estão procurando. Essa técnica de consulta por exemplo remove as dificuldades que aparecem ao tentar descrever imagens com palavras.

A técnica de recuperação semântica seria o sistema CBIR ideal do ponto de vista do usuário. Nessa técnica, o usuário faz um pedido como “encontre imagens de cães” ou “ache retratos do presidente Lula”. Esse tipo de tarefa é muito difícil para computadores executarem, pois imagens de cães da raças chihuahuas e dinamarqueses, por exemplo, são bastante diferentes. Sistemas CBIR atuais, geralmente, fazem uso de características de baixo nível como textura, cor ou forma, embora alguns sistemas tomem vantagem de características de alto nível muito comuns como faces usando-as para sistema de reconhecimento facial [7].

Outras técnicas incluem especificar as proporções das cores desejadas (por exemplo “30% verde, 70% vermelho”) e procurar por imagens que contêm um objeto dado em uma imagem de consulta [7]. Para gerar resultados mais significativos, sistemas CBIR podem também fazer uso de *feedback* de relevância do usuário [8] para modificar o processo de recuperação. O usuário refina progressivamente os resultados da busca marcando as imagens recuperadas como “relevantes”, “não relevantes”, ou “neutras”, então repete a busca com a nova informação.

O interesse em CBIR está crescendo por causa das limitações herdadas dos sistemas baseados em metadados como também a larga escala de possíveis usos para recuperação eficiente de imagens. A informação textual sobre imagens pode ser facilmente procurada usando tecnologia existente, mas requer que seres humanos descrevam pessoalmente cada imagem na base de dados. Essa descrição textual não é prática para base de dados muito grandes ou para imagens que são geradas automaticamente. Também é possível que imagens que usem diferentes sinônimos em suas descrições não sejam recuperadas.

O sistema de consulta pelo conteúdo da imagem da IBM conhecido como QBIC (*Query-By-Image-Content*) [9-10] é um dos mais poderosos sistemas de consulta a base de dados de imagens. Foi desenvolvido para consultar uma grande base de dados de imagens on-line usando o conteúdo da imagem como a base das consultas. Exemplos dos conteúdos usados pelo QBIC incluem cor, textura e forma dos objetos e regiões da imagem. Entre os potenciais usos de CBIR podemos citar coleções de arte, arquivos de fotografias, imagens médicas, entre outros.

## 2.2 Atributos das imagens

Nos sistemas de recuperação de imagens, alguns dos atributos mais utilizados para efetuar descrições sobre o conteúdo das imagens são cor, forma e textura. A recuperação da imagem está baseada na similaridade das características derivadas destes atributos.

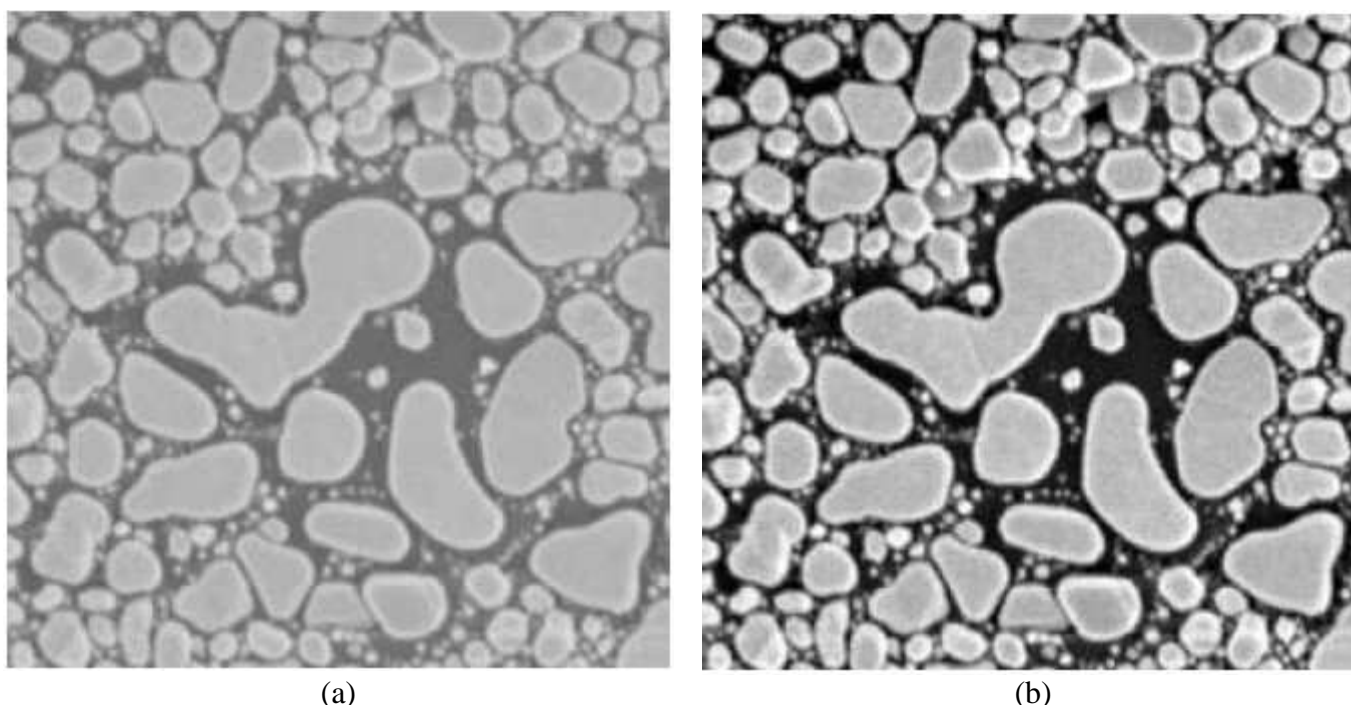
### 2.2.1 Cor

A cor é um dos atributos mais utilizados pelos seres humanos para reconhecimento e discriminação visual. A aparência de uma cor em objetos do mundo real geralmente é alterada pela textura da superfície, pela iluminação e sombra de outros objetos, e as condições de observação e captura. Examinar imagens baseadas nas cores que elas contêm é uma das técnicas mais extensamente utilizadas, pois as cores presentes em uma imagem possuem um papel significativo na indexação e recuperação da mesma. Existem diferentes representações de cores incluindo desde o tradicional modelo RGB (*red, green, blue*), o mais simples dos modelos 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 [11].

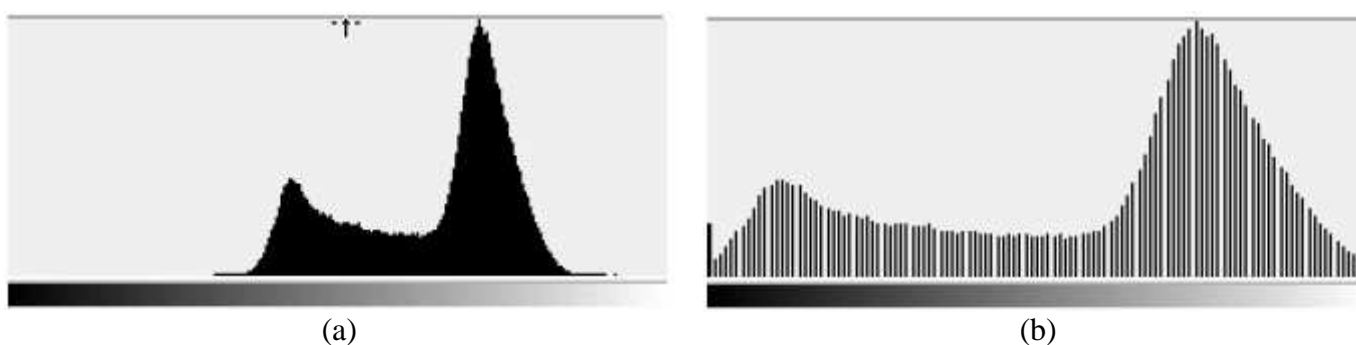
Muitos trabalhos recentes de extração de características baseadas em distribuição de cores estão concentrados nos histogramas de cor. Swain e Ballard [12] incluíram a indexação da cor usando intersecção de histogramas. Já Bueno [13] monta uma estrutura baseada em histogramas

de dimensão variável, nomeada histogramas métricos, sendo utilizados na recuperação de imagens similares. 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.

Mudando parâmetros e limiares, muitas operações podem ser feitas com os histogramas de cor. A Figura 2.(a), por exemplo, representa o histograma da imagem original (Figura 1.(a)). Ao se aplicar uma operação de alongamento (*stretch*) sobre a imagem, o histograma ficaria como na Figura 2.(b), tendo como resultado uma imagem melhor definida como mostra a Figura 1.(b).



(a) (b)  
**Figura 1.** (a) Imagem original e (b) imagem saturada



(a) (b)  
**Figura 2.** (a) Histograma da imagem original e (b) histograma da imagem saturada

Existem diversas dificuldades com a recuperação através de histogramas, relacionadas com a grande quantidade de dimensões dos histogramas de cores, e com o uso de histogramas para localizar espacialmente diversos tipos de características, onde os histogramas de cor não indicam a localização espacial dos pixels na imagem. Além disso, se a imagem contém um número alto de cores, então o vetor de características será maior e a indexação de vetores com essa dimensão torna-se problemática. Entre os sistemas comerciais populares que utilizam os histogramas de cores em sistemas de recuperação de imagens por conteúdo podemos citar o QBIC [10-11], Virage [14], VisualSEEK [15], entre outros.

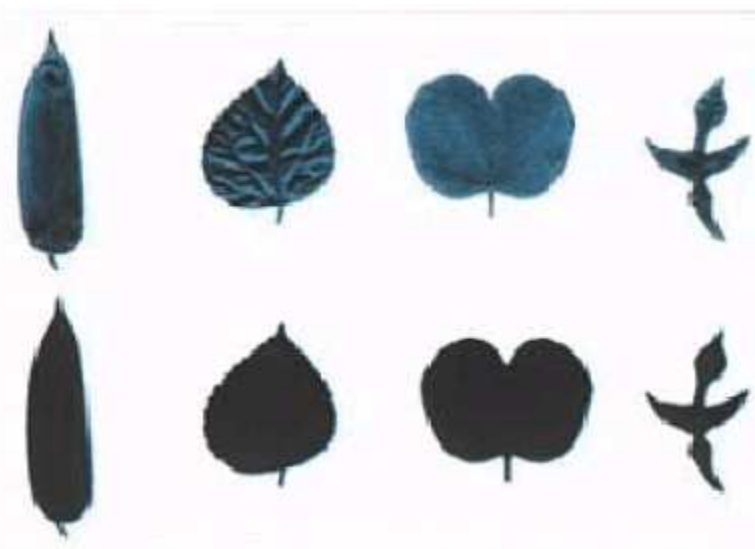
### 2.2.2 Forma

As formas de um objeto são características muito eficientes para a identificação do mesmo. Estas características podem ser descritas através da forma global do objeto, como a área, o seu alongamento em relação a um eixo, a orientação em relação a um eixo, ou através dos elementos locais, como os limites e contornos, pontos característicos, etc [16].

Em sistemas CBIR, o atributo forma é a abordagem que mostra maior dificuldade, especialmente pelo fato de ter que segmentar e conhecer o tamanho dos objetos contidos na imagem. Em aplicações médicas, por exemplo, a forma e tamanho dos tumores é de muita importância na hora de classificá-los como malignos ou benignos. Segundo Theodoridis [17], tumores com bordas irregulares têm uma alta probabilidade de serem malignos e aqueles que mostram bordas regulares geralmente são benignos.

Theodoridis [17] mostra também que existem duas abordagens pelas quais se pode obter a caracterização da imagem pela forma. Uma é desenvolver técnicas que transmitem uma descrição total da borda do objeto através de coeficientes de Fourier [18] e a outra é utilizar qualidades que descrevam as características morfológicas da região usando número de quinas na borda. Nessa abordagem estão as características baseadas em momentos. O trabalho recente de [29] envolve essa última abordagem.

Costa e Cesar [20] apresentam técnicas de análise e classificação de formas onde o processo de extração de características de formas é definido com base em três tarefas: (a) organizar e visualizar as características; (b) extrair características; e (c) mensurar as características extraídas dos objetos. Na Figura 3, apresentam-se alguns exemplos de forma como resultado do trabalho de [20].



**Figura 3.** Formas de imagens [20]

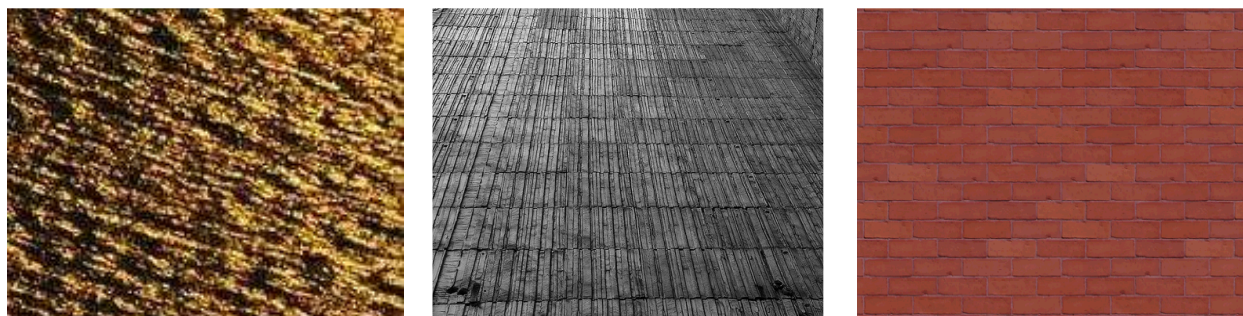


### 2.2.3 Textura

Apesar de não haver uma definição estrita do conceito de textura de imagem, os autores concordam em entendê-la como as mudanças na intensidade da imagem que formam determinados padrões repetitivos [21]. Esses padrões podem ser o resultado de propriedades físicas da superfície do objeto como rugosidade, ou ser o resultado de diferenças de reflexão tal como a cor na superfície.

A textura refere-se a um padrão visual que tem algumas propriedades de homogeneidade que não resultam simplesmente de uma cor ou intensidade. Então, podemos considerar, de forma genérica, que texturas são padrões visuais complexos compostos por entidades ou sub-padrões que apresentam propriedades características (brilho, uniformidade, densidade, dureza, linearidade, frequência, fase, direcionalidade, suavidade, granulação, aleatoriedade, grossura, etc) da textura como um todo [22]. Gonzalez [11], descreve a textura por medidas que quantificam suas propriedades rugosidade e regularidade.

Embora seja fácil, para as pessoas, o reconhecimento de textura, isto não acontece com procedimentos automáticos onde esta tarefa, às vezes, precisa de técnicas computacionais complexas. A extração de características a partir da textura é considerada como um descritor importante para imagens naturais, sendo muito útil para sistemas CBIR, pois a grande maioria de superfícies naturais exibem texturas. Além disso, é facilmente perceptível pela visão humana e representa uma fonte de informação visual muito rica. A Figura 4, mostra alguns exemplos de imagens com textura.



**Figura 4.** Exemplos de texturas

De acordo com Traina [23], 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). Geralmente, a análise de textura é utilizada para:

- Segmentação: divisão de uma imagem em regiões com mesmo perfil textural;
- Descrição: extração de características baseada na quantificação de seu conteúdo de textura para discriminação entre classes de objetos;
- Classificação: rotulação de uma região com determinada textura com base em exemplos de texturas conhecidas;
- Forma: empregar a informação de textura para derivar a geometria de uma superfície tridimensional;
- Réplica: descrever uma textura visando sua síntese.

Aplicações práticas e científicas do uso de textura podem ser encontradas em áreas como computação gráfica, processamento de imagens e reconhecimento de padrões. Além disso,

existem pesquisas, em processamento de imagens, visando a procura de algoritmos que gerem textura automaticamente. Mello [24] cita que essa síntese de texturas procura definir um padrão da textura e reproduzi-lo.

Segundo Gonzalez [11], as principais abordagens utilizadas para descrever texturas podem ser divididas em Estatísticas, Estruturais e Espectrais. Para Sonka [25], a abordagem estatística é a que tem sido mais amplamente utilizada, gerando resultados satisfatórios com um custo computacional reduzido.

As abordagens estatísticas (momentos do histograma, matriz de co-ocorrência de níveis de cinza [26], etc.) descrevem a textura através de um conjunto de medidas locais extraídas da imagem.

As abordagens estruturais (conceitos estruturais) descrevem a textura como arranjos de primitivas da imagem dispostas de forma aproximadamente regular e repetitiva. Esta primitiva é um elemento fixo que se repete numa área da imagem, de acordo com regras bem definidas.

As abordagens espectrais lidam com as regiões no domínio das frequências espaciais (espectro de Fourier). Utilizando-se das propriedades do espectro de Fourier, é possível identificar características como periodicidade global, através da identificação de picos de alta energia no espectro da imagem.

A caracterização de uma textura não é uma tarefa fácil. Várias dificuldades estão presentes, da mesma forma que no caso de cores, como a invariância e a dimensionalidade do vetor de características. Este trabalho dará ênfase as abordagens estatísticas como métodos para descrição de texturas.

## 2.3 Abordagens estatísticas

As abordagens estatísticas (momentos do histograma, matriz de co-ocorrência de níveis de cinza [26] e medidas estatísticas) descrevem a textura através de grandezas que as caracterizam como suave, áspera ou granular.

### 2.3.1 Momentos do histograma

Os momentos do histograma podem ser utilizados para descrever as propriedades de uma textura, já que a forma em que se apresenta o histograma da imagem fornece informação sobre a natureza dessa imagem. Assim, a partir do histograma de intensidade de uma região, podem ser calculados os momentos.

Seja  $I$  a variável aleatória que representa os níveis de cinza na região de interesse, o histograma  $P(I)$  é definido como:

$$P(I) = \frac{\text{número de pixels com nível de cinza } I}{\text{número total de pixels na região}} \quad (1)$$

O histograma  $P(I)$  de uma imagem digital fornece a frequência com que cada nível de cinza  $I$  ocorre na imagem. Isto significa que  $P(I)$  é a fração de pixels com nível de cinza  $I$ . O gráfico de um histograma é uma função  $I \times P(I)$  (onde  $P(I)$  representa o eixo da porcentagem e  $I$  para  $I = 0, 1, \dots, N_g - 1$  representa o eixo da intensidade, sendo  $N_g$  o máximo tom da resolução de cinza da imagem) que provê uma percepção da distribuição de cores. Medidas estatísticas calculadas sobre o histograma nos dão informações gerais sobre esta população de níveis de cinza [27].

O momento do histograma pode ser representado pela equação (2):



$$m_i = E[I^i] = \sum_{I=0}^{N_g-1} I^i P(I), \quad i = 1, 2, 3, \dots \quad (2)$$

Se  $i = 1$  então  $m_1 = E[I]$  que representa a média do valor  $I$ . Assim o primeiro momento, média dos níveis de cinza, indica a média de intensidade da região.

O momento central do histograma pode ser calculado pela equação (3):

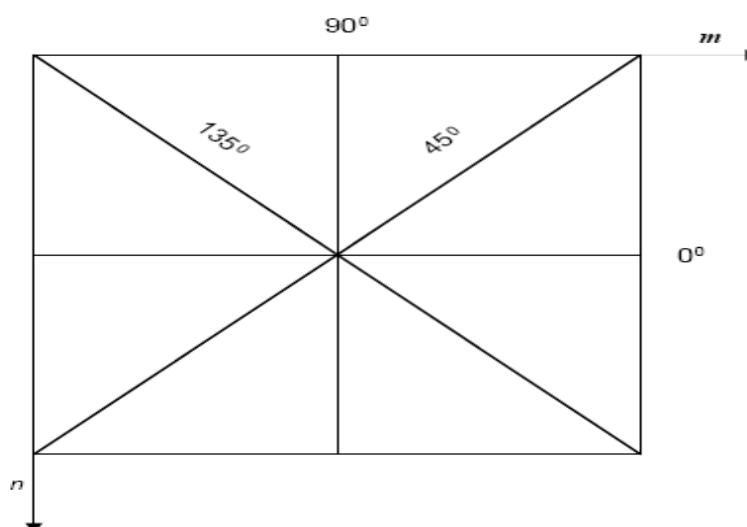
$$\mu_i = E[(I - E[I])^i] = \sum_{I=0}^{N_g-1} P(I)(I - m_1)^i \quad (3)$$

O segundo momento central  $\mu_2$  é a variância,  $\sigma^2$ , e é útil na descrição da uniformidade de uma determinada região da imagem, ou seja, indica como estão distribuídas as intensidades iguais dentro da região. O terceiro momento central, às vezes denotado como  $\sigma^3$ , é uma medida de não centralismo (assimetria) que calcula o grau da assimetria do histograma ao redor da média.

### 2.3.2 Matriz de Co-Ocorrência de Níveis de Cinza - GLCM

O cálculo dos momentos de intensidade possui a vantagem de ser fácil de realizar, mas não oferece nenhuma informação sobre a natureza repetitiva da textura, ou seja, não dá informação sobre a posição relativa dos níveis de cinza distintos dentro da imagem. Um método estatístico que soluciona este problema é a Matriz de Co-Ocorrência de Níveis de Cinza (GLCM, *Gray Level Co-Occurrence Matrix*), proposto por Haralick *et al* [26], que consiste em experimentos estatísticos realizados sobre como um certo nível de cinza ocorrer em relação a outros níveis de cinza.

A GLCM contém informações sobre as posições dos pixels que têm valores de níveis de cinza similares, usando de uma medida de probabilidade de ocorrência de cores separadas por uma dada distância em uma dada direção. Para isso, são empregadas múltiplas matrizes, uma para cada direção de interesse (horizontal, vertical e as duas diagonais). Para construção dessas matrizes, define-se um operador  $(d, \theta)$ , onde  $d$  é a distância relativa medida em número de pixels ( $d = 1$  para pixels vizinhos, etc) e  $\theta$  pode ser uma das 4 direções, conforme a figura 5:



**Figura 5.** Orientações usadas para construir as matrizes de co-ocorrência [17]

A dimensão da GLCM depende da resolução de cor da imagem, ou seja, se a imagem possuir  $k$  níveis de cinza, a matriz terá dimensão  $k \times k$ . Cada elemento  $(i,j)$  da matriz indica o número de vezes que um ponto com o tom  $i$  aparece na posição especificada por  $(d, \theta)$  em relação a um ponto com tom  $j$ . Por exemplo, para a imagem binária dada pela matriz abaixo que possui 2 níveis de cinza (0 e 1):

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

A GLCM correspondente para essa imagem e operador  $(1, 135^\circ)$  é a matriz 2x2:

$$A = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}$$

Por exemplo, o elemento  $a_{0,0}$  (canto superior esquerdo), representa o número de vezes que um elemento com o tom 0 aparece à distância 1 e à direita e abaixo de outro ponto com o tom 0, ou seja, 10 vezes. Já o elemento  $a_{0,1}$  (canto superior direito), nos mostra que nenhum elemento com o tom 0 aparece à distância 1 nessa mesma posição  $(135^\circ)$  de outro ponto com o tom 1.

A matriz de co-ocorrência não caracteriza univocamente as texturas, mas a comparação de algumas características extraídas dessas matrizes ajuda a averiguar a similaridade de duas texturas. Algumas das características têm uma interpretação física direta em relação à textura, por exemplo, onde é possível quantificar a rugosidade, suavidade, etc. Por outro lado, outras características não têm tais propriedades, mas ainda assim codificam informação relacionada à textura com uma grande capacidade discriminatória [17] [28]. Haralick [26] propõe 28 descritores de textura que podem ser extraídas das matrizes de co-ocorrência. Normalmente, a escolha dos descritores a serem adotados baseia-se em testes empíricos sobre um domínio específico, verificando-se quais deles produzem resultados mais satisfatórios. Exemplos de descritores de textura que podem ser calculados após a obtenção da matriz de co-ocorrência são: a variância (4), a entropia (5), a energia (6), a homogeneidade (7), a assimetria (8) e a variância inversa (9).

$$VAR = \sum_{i=0}^{Ng-1} \sum_{j=0}^{Ng-1} (i-j)^2 P(i, j) \quad (4)$$

$$H = - \sum_{i=0}^{Ng-1} \sum_{j=0}^{Ng-1} P(i, j) \log_2 P(i, j) \quad (5)$$

$$E = \sum_{i=0}^{Ng-1} \sum_{j=0}^{Ng-1} [P(i, j)]^2 \quad (6)$$

$$HOM = \sum_{i=0}^{Ng-1} \sum_{j=0}^{Ng-1} \frac{[P(i, j)]}{1 + (i-j)^2} \quad (7)$$

$$M_3 = \sum_{i=0}^{Ng-1} \sum_{j=0}^{Ng-1} (i-j)^3 P(i, j) \quad (8)$$

$$IVAR = \sum_{i=0}^{Ng-1} \sum_{j=0}^{Ng-1} \frac{P(i, j)}{(i-j)^2} \quad (9)$$

A abordagem GLCM possui a vantagem de ser invariante às transformações na imagem, entretanto não permite revelar a forma da textura. Além disso, para uma imagem armazenada com um grande número de níveis de cinza, será necessário um grande espaço de armazenamento para a GLCM, como também o tempo de processamento necessário para seu cálculo é alto. Por esses motivos, a GLCM não é utilizada nesse trabalho. Outras técnicas têm sido desenvolvidas tais como o Histograma da Soma e da Diferença [29] e o LPB - *Local Binary Pattern* [30].

Caracterizar textura é uma tarefa difícil. Os métodos para representação de textura ainda não produziram uma solução efetiva para o problema da análise, classificação e segmentação de texturas. Tais métodos incluem diferentes abordagens como o uso de momentos do histograma, matrizes de co-ocorrência e análise no domínio espaço-frequência, chamada de análise de multiresolução.

Métodos baseados em análises de multiresolução estão sendo desenvolvidos sob a alegação de serem capazes de suprir as deficiências encontradas em métodos anteriores. Tais deficiências são devidas a estes métodos não possuírem recursos para caracterização eficiente da textura em diferentes escalas.

Trabalhos recentes relatam o uso de transformadas baseadas em *wavelets* como solução para uma análise multiresolução efetiva e consistente. A transformada *wavelet* produz decomposições no domínio da frequência que atingem uma solução comum máxima no domínio espacial e no domínio da frequência, onde é possível atingir uma resolução em ambos os domínios. Isto é altamente significativo no processo de extração de informação de textura o qual possui o conflitante objetivo de acurácia na representação da textura e localização espacial desta [16]. No Capítulo seguinte, iremos apresentar a base teórica da transformada de *wavelet* e o conceito de análise de multiresolução para caracterizar texturas na geração de um vetor de características que possa ser capaz de representar suficientemente uma imagem, dado um determinado problema de classificação.

## Capítulo 3

# Wavelets e Extração de Características

### 3.1 Introdução

Uma imagem é, antes de tudo, um sinal, e, para seu processamento e análise, conta-se com uma série de ferramentas, onde uma das mais conhecidas é a transformada de Fourier que faz a representação de um sinal por meio de uma série de senos e cossenos. Através da transformada de Fourier, um sinal representado no domínio do tempo é transformado para o domínio da frequência. A expressão da transformada de Fourier para um sinal contínuo  $f(t)$  é dada por:

$$F(w) = \int_{-\infty}^{+\infty} f(t)e^{-j2\pi wt} dt \quad (10)$$

Onde  $w$  é a frequência. Essa transformada detecta a variação de tons presentes na imagem, mas não gera informação sobre a localização espacial destes, então ela não pode representar adequadamente as mudanças que acontecem no sinal no espaço. Esse não seria um problema se o sinal não mudasse durante o tempo (sinais estacionários), mas acontece que muitos sinais reais contêm muitas características não estacionárias ou transitórias (fluxos, tendências, mudanças repentinas). Essas características, geralmente, são a parte mais importante do sinal [31].

A solução para este problema, uma variação da transformada de Fourier, foi elaborada por Gabor (1946) e chamada de transformada por janelas de Fourier (*Windowed Fourier Transform - WFT*). A WFT se utiliza de uma janela de observação para analisar só uma pequena porção do sinal. Essa janela é deslocada no domínio do tempo e para cada posição da janela é calculada a transformada de Fourier do sinal dentro dela.

No entanto, a transformada por janelas de Fourier traz outro problema: a definição do tamanho da janela. Pois uma vez definido o tamanho da janela, ela permanece constante para todas as frequências e pode haver sinais que precisem de uma janela maior para detectar o conteúdo local da frequência. Por outro lado, uma janela grande perde a consideração de estacionaridade do sinal.

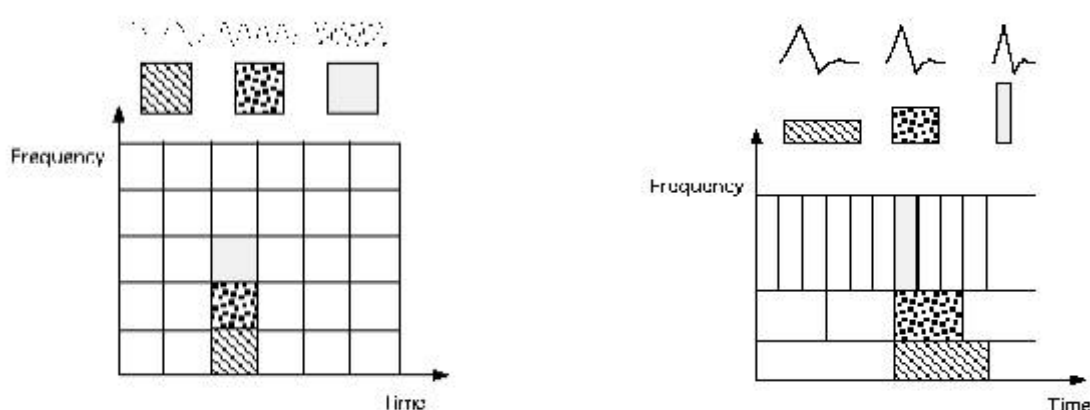
As transformadas *wavelets* suprem estas falhas presentes na abordagem da transformada por janelas de Fourier para a análise de sinais não periódicos. As *wavelets* foram desenvolvidas com base na matemática, física quântica e estatística, e são funções que decompõem os sinais em diferentes componentes de frequência e analisam cada componente em diferentes resoluções [32].

Segundo Castañón [31], as *wavelets* estão sendo utilizadas em inúmeras áreas de aplicação, por exemplo na mecânica de fluidos, computação numérica, análise de imagens, processamento de sinais, sistemas de controle, fenômenos biológicos, medicina, medidas e psicologia. No futuro próximo, espera-se que a análise de *wavelets* seja uma ferramenta de análise e visualização tão popular quanto a transformada de Fourier. Outras aplicações comuns das *wavelets* são: remoção de ruído de sinal, compressão de imagens, suavização de imagens e análise fractal [32]. A seguir, é mostrada uma introdução à sua teoria.

### 3.2 Transformada Wavelet

A transformada *wavelet* apresenta-se como uma ferramenta alternativa para o processamento de sinais, mudando o paradigma de representação dos mesmos, ao utilizar funções base de *wavelets* (em vez das senoidais de Fourier) para transformar um sinal no espaço de tempo-escala [31]. O que a análise de *wavelets* faz é decompor um sinal. O processo inverso é a reconstrução ou síntese do sinal.

Pode-se interpretar a transformada *wavelet* como uma técnica por janelas, mas com janelas de tamanho variável, e que tem como base uma função de duração limitada definida por dois parâmetros: posição e escala. Esses parâmetros indicam a translação (localização da janela) e compressão ou expansão das janelas, respectivamente. Isso permite que características de um sinal examinado sejam extraídas em função de determinada posição e tamanhos desejados, dando à *wavelet* uma flexibilidade na representação do domínio do tempo. A Figura 6 compara as ondas geradas por Fourier e *wavelet*. Observa-se que a janela de tempo varia para *wavelets*, o que não ocorre para Fourier.



**Figura 6.** Comparação entre a transformada de Fourier e *wavelet* em função do tempo [32]

Já o fato da função ser de duração limitada significa ter suporte compacto, ou seja, possuir a propriedade na qual seu domínio é diferente de zero em uma extensão finita e igual a zero em todo o resto. Castañón [31] afirma que isto torna interessante a utilização das *wavelets* no caso específico da análise de imagens, pois as mudanças de regiões ou bordas podem ser detectadas

mais facilmente. A definição da transformada *wavelet*, considerando um sinal contínuo  $f(t)$  é dada por:

$$F(a,b) = \int f(t)\psi_{a,b}(t)dt \quad (11)$$

Pela equação 11, percebe-se que a transformada *wavelet* depende de dois parâmetros  $a$  e  $b$ , que variam continuamente em  $R$ , correspondendo às informações de escala e tempo respectivamente [32]. As *wavelets*, funções que analisam o sinal, são representadas por  $\psi_{a,b}$  e assim definidas:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi_{a,b}\left(\frac{t-b}{a}\right) \quad (12)$$

Para sinais discretos, a transformada *wavelet* é definida como:

$$F_{m,n}(a,b) = a_0^{-m/2} \int f(t)\psi(a_0^{-m}t - nb_0)dt \quad (13)$$

Pela equação 13, observa-se que o comportamento dessa função está baseado em dilatações e translações da *wavelet* mãe ( $\psi$ ). A transformada contínua de *wavelets*, obtém os coeficientes de *wavelets* em cada escala possível, isso requer uma grande quantidade de cálculo e torna o trabalho bastante custoso. Por esse motivo, a transformada discreta de *wavelets* escolhe um subconjunto de escalas e locações sobre as quais os cálculos vão ser realizados.

### 3.3 Análise de Multiresolução

As primeiras construções de bases de *wavelets* eram muito trabalhosas. Mallat, em 1986, formulou a análise de multiresolução que surgiu como solução para este problema. A análise de multiresolução é uma estratégia de processamento de sinais onde é utilizado um conjunto de filtros especializados em extrair as informações do sinal, como as frequências presentes nesse e sua localização no tempo de duração do sinal, em diferentes resoluções [34]. Segundo diversos estudos sobre o sistema visual humano, o uso da multiresolução é útil em sistemas de classificação de imagem [35] [36]. Estes estudos indicaram que o córtex visual humano detecta as imagens em diferentes resoluções, onde as informações sobre as frequências e a orientação dos tons presentes na imagem são interpretadas separadamente. A análise da imagem é feita através de um mapeamento desta para um conjunto de imagens em diferentes resoluções, a partir de uma transformada matemática [37]. A transformada utilizada detecta a variação dos tons de cores da vizinhança de um *pixel* e a sua localização espacial e transpõe isto em um espaço multiresolução.

A idéia de se estudar imagens em várias escalas simultaneamente era popular quando Mallat estava trabalhando com análise de imagens e usou *wavelets* pela primeira vez. Isso o estimulou a ver bases ortonormais de *wavelets* como uma ferramenta para descrever matematicamente o “incremento na informação” necessário para se ir de uma aproximação grosseira para uma aproximação com maior resolução [32].

Segundo Mallat [38], uma decomposição multiresolução permite obter uma interpretação invariante de escala da imagem. A escala de uma imagem muda em razão da distância entre a cena e o centro óptico da câmera. Quando a escala da imagem é modificada, a interpretação da

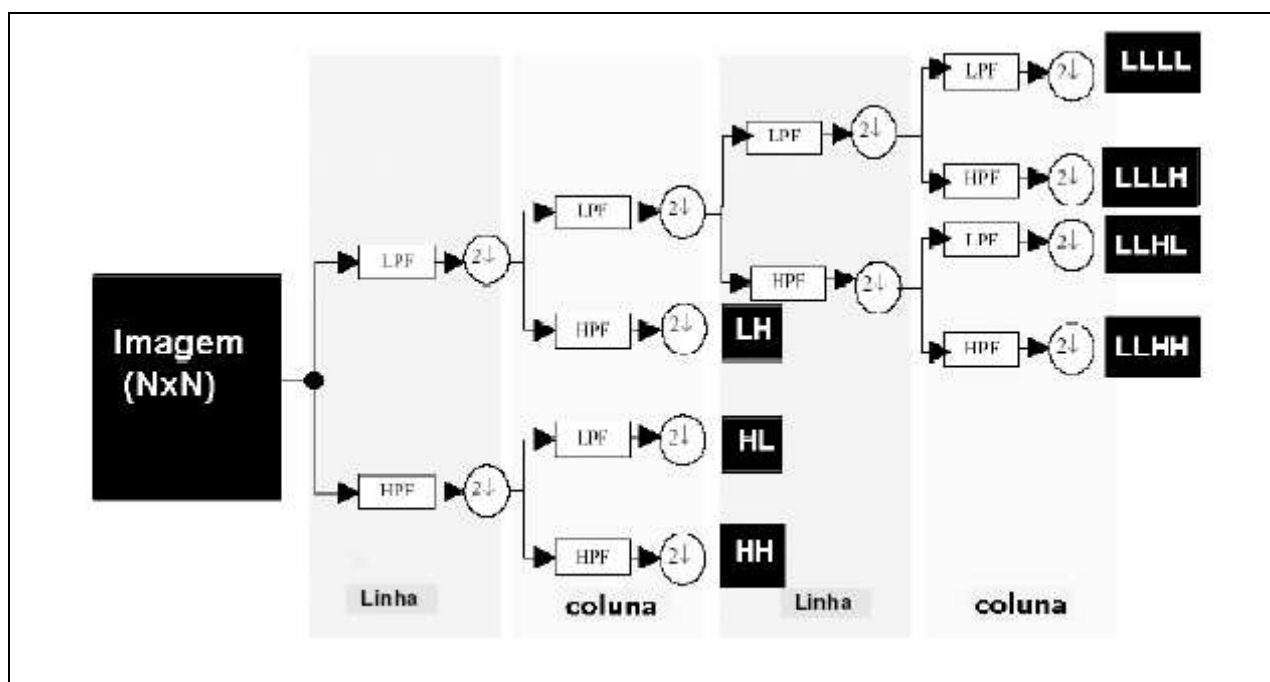
imagem não deve mudar. Uma representação multiresolução pode ser parcialmente invariante de escala, se a sequência de parâmetros da resolução varia exponencialmente.

Assim, na análise de multiresolução, qualquer tipo de sinal pode ser analisado em termos de escalas e translações de uma simples função *wavelet* mãe. Essa estratégia se mostra bastante adequada para a extração de características das imagens, pois poucos coeficientes representam bem a imagem, armazenam informações de arestas e bordas e, praticamente, não dependem da resolução original da imagem.

### 3.4 Algoritmo

O algoritmo da transformada de *wavelets* é feito da seguinte forma: o sinal é passado por dois filtros, um passa baixa e um passa-alta, os quais irão dividir o sinal em dois sinais, um de frequências maiores que um dado limiar e outro de frequências menores. O operador ( $\downarrow 2$ ) é o operador de sub-amostragem (do inglês *downsampling*) [39] e quando é aplicado ao sinal reduz o sua taxa de amostragem pela metade. Este processo é repetido sobre o sinal de menores frequências até um nível pré-determinado e os sinais de maiores frequências são armazenados a cada iteração com os filtros. Estes sinais de maiores frequências são conhecidos como coeficientes de detalhes e os de menores frequências são os de aproximação [37].

A aplicação destes filtros em uma imagem é feita pela iteração entre linhas e colunas da matriz de imagem, o que gera a cada nível de iteração 3 sub-imagens, que são os coeficientes de *wavelets* armazenados. Assim, tanto o processo de decomposição quanto o de reconstrução da imagem podem ser implementados em duas dimensões, ou seja, aplicando o algoritmo nas linhas e colunas da imagem, como mostra a Figura 7.

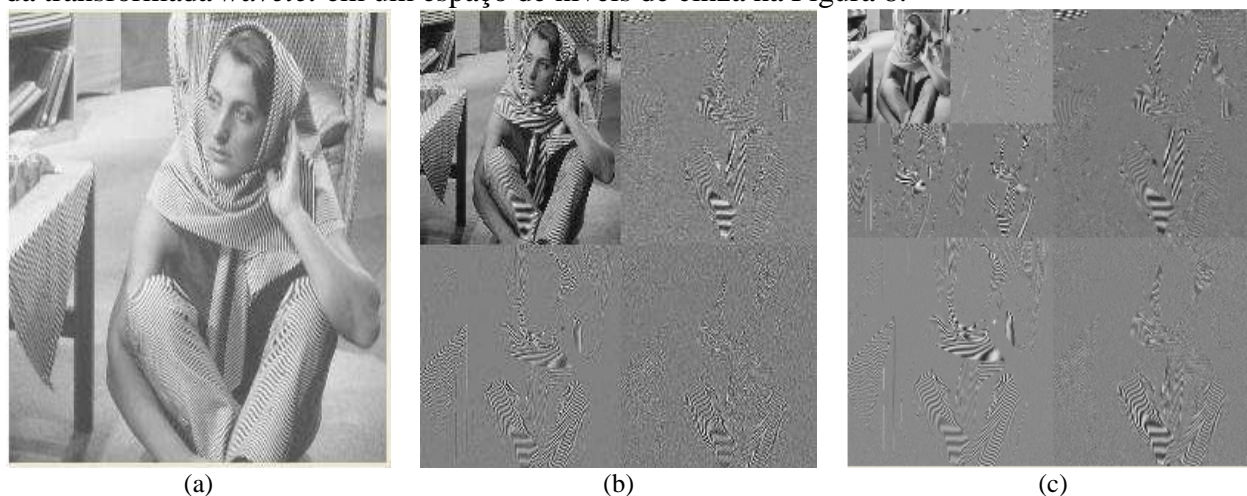


**Figura 7.** Decomposição de imagem em um domínio 2D ao aplicar a transformada *wavelet* [40]

Assim, as sub-imagens que compõe o resultado da transformada estão organizadas em diferentes níveis de resolução, cada um responsável por espectro de frequências, e em cada nível, têm-se a angulação destas texturas, obtendo-se assim, uma descrição das variações dos tons de cinza e a localização destes. Isso mostra a eficácia das *wavelets* na caracterização de imagens.



Após 2 níveis de decomposição da imagem original, podemos observar o resultado da aplicação da transformada *wavelet* em um espaço de níveis de cinza na Figura 8.



**Figura 8.** Transformada *wavelet* sobre um domínio 2D. (a) Imagem original, (b) 1 nível de decomposição da Transformada e (c) 2 níveis de decomposição.

Um dos pontos fortes da transformada *wavelet*, para processamento de imagens, vem do fato que com uma quantidade mínima de valores (em uma escala menor da imagem) é possível representar a imagem toda, ou seja, através dos coeficientes de *wavelet*. Esses coeficientes podem ser aproveitados na montagem do vetor de características para representação de uma imagem.

### 3.5 Extração de Características

A parte crítica do processo de recuperação de imagens é definir o espaço de características a ser utilizado para realizar o casamento entre atributos similares. Espaços de características comuns incluem bordas, contornos, superfícies e outras características salientes tais como esquinas, interseção de linhas e pontos de alta curvatura da imagem [31].

Os dados que serão usados no processo de recuperação de imagens por conteúdo representam o espaço de características e sua escolha determina o que será casado no momento da busca das imagens mediante a métrica de semelhança que determina a similaridade entre duas imagens.

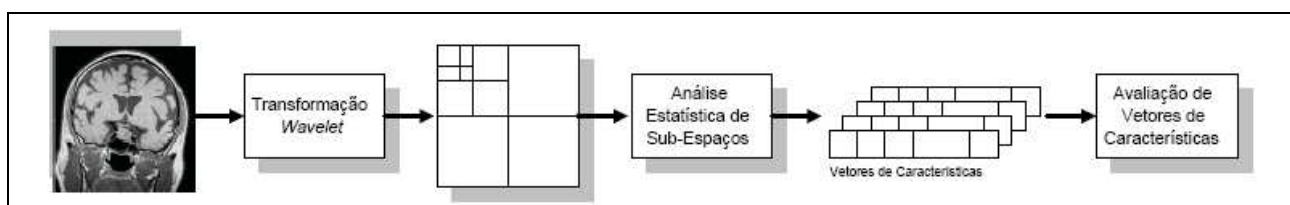
*Wavelets* modelam a textura no domínio da frequência espacial e sua aplicação para a geração do vetor de características é efetuada através de uma análise estatística em cada uma das sub-bandas, onde cada elemento do vetor pode corresponder a uma descrição de cada sub-imagem, de maneira que o conteúdo presente na imagem seja a correta descrição feita pelo vetor.

Existe uma variedade de funções *wavelets*, chamadas de “famílias de *wavelets*”, que têm comportamento distinto de acordo com o domínio de imagens. As *wavelets* Daubechies [32] apresentam-se como as que melhor caracterizam os detalhes da imagem através das *wavelets* mãe que ficam nas sub-bandas de baixa resolução (os coeficientes de *wavelets*) enquanto que as *wavelets* que ficam nas sub-bandas de alta resolução caracterizam a tendência do sinal [41]. Estas propriedades podem ser aproveitadas para representar uma imagem e gerar o vetor de características aplicando descritores estatísticos sobre esses vetores. Assim, pode-se avaliar a uniformidade da imagem (através da energia), a suavidade (através da entropia), entre outros descritores vistos na Seção 2.3.2.

O processo geral de extração de características está representado na Figura 9, onde, dependendo da abordagem, realiza-se a obtenção dos coeficientes das sub-imagens e as análises



estatísticas desses a fim de gerar o vetor de característica da imagem. Assim, o número de valores gerados para cada vetor de características vai depender de cada abordagem.



**Figura 9.** Processo de extração de características [31]

Para Traina *et al* [42], a representação de uma imagem mediante um conjunto de valores característicos é equivalente a associá-la com um ponto no espaço de dimensionalidade igual ao número de características no vetor, onde aqueles que têm valores próximos entre si possuirão maior grau de similaridade.

Depois de realizada a extração de características, a partir de alguma métrica, como a distância euclidiana, por exemplo, é possível recuperar imagens semelhantes, dado um certo grau de aproximação de seus vetores como critério de semelhança.

No Capítulo seguinte, mostraremos algumas técnicas de agrupamento, também conhecidas como clusterização, que agrupam as imagens similares em *clusters*. Também é apresentada a definição de medidas de distância ou dissimilaridade, que são utilizadas para o agrupamento das imagens segundo seu grau de semelhança.

## Capítulo 4

# Técnicas de Análise de Agrupamento

### 4.1 Introdução

Na maioria dos sistemas CBIR, o usuário geralmente pensa em termos de conceitos semânticos e não está interessado nas características de baixo nível das imagens (cor, textura e forma). Por exemplo, alguma imagem que ele se lembra de já ter analisado e que deseja recuperar do banco de imagens, ou mesmo buscar as imagens mais parecidas com uma determinada imagem sob análise [31].

Assim, depois de ocorrida a extração de características, a definição de uma métrica capaz de responder às consultas por similaridade é de grande importância, além da capacidade de agrupar (*clustering*) e diferenciar cada uma das classes semânticas.

### 4.2 Algoritmos de Agrupamento

Os algoritmos de agrupamento ou *Clustering* particionam os objetos em conjuntos (*clusters*) baseados na similaridade entre eles. Um agrupamento pode ser visto como uma partição sobre um espaço de atributos, definidos sobre algum critério de semelhança. Esse critério, faz parte da definição do problema e, depende, do algoritmo utilizado.

Assim, *Clustering* é uma técnica de *data mining* para fazer agrupamentos automáticos de dados, explorando as semelhanças entre esses dados e agrupando-os em categorias. O principal objetivo da clusterização é particionar conjuntos de objetos em grupos homogêneos de maneira que se maximize a similaridade de objetos dentro de um mesmo *cluster* e se minimize a similaridade de objetos de *clusters* distintos.

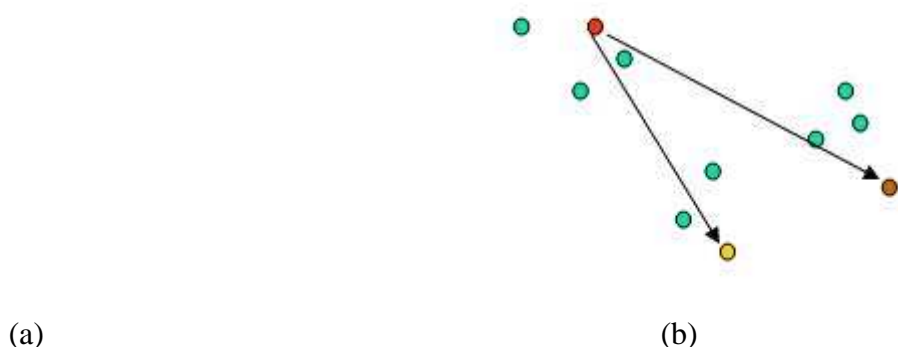
Agrupar ou classificar objetos em conjuntos ou categorias tem sido uma atividade bastante comum que vem sendo intensificada devido ao número elevado de informações disponíveis atualmente. Segundo [43], a clusterização é um método que utiliza o aprendizado não supervisionado [44], ou seja, não há um “professor” que lhe indique o que cada padrão representa. A aprendizagem não supervisionada busca extrair informação relevante de dados não rotulados e os conjuntos aos quais os dados pertencem são definidos por um algoritmo de agrupamento. Já na aprendizagem supervisionada, o usuário é quem define os conjuntos ou classes aos quais os dados pertencerão.

### 4.2.1 K - Médias

O  $k$ -médias [45] é um tipo de algoritmo de agrupamento cuja aprendizagem é não supervisionada. O objetivo desse algoritmo é encontrar a melhor divisão dos dados em  $k$  *clusters* de maneira que a distância total entre os dados de um *cluster* e o seu respectivo centro, somada por todos os *clusters* seja minimizada. Dessa maneira, o centro do *cluster* inicial para cada caso é formado em torno dos dados mais próximos e então comparado com os pontos mais distantes e os outros *clusters* formados. A partir daí, dentro de um processo de atualização contínua e de um processo iterativo encontram-se os centros dos *clusters* finais [46].

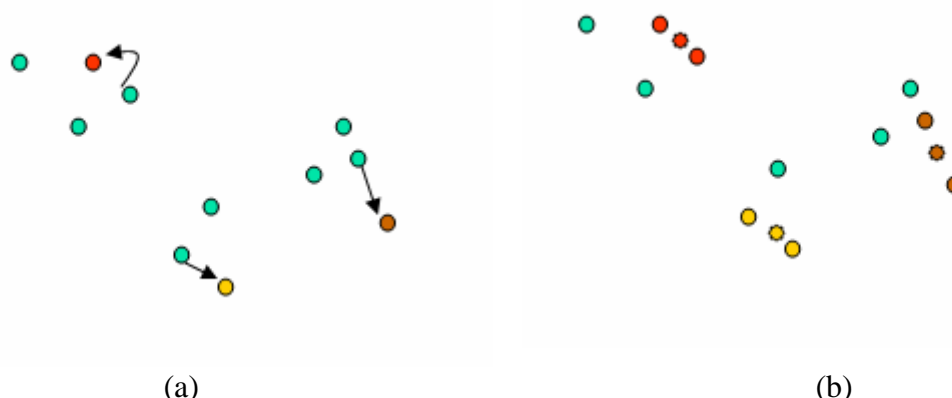
Assim, o algoritmo atribui aleatoriamente os dados a  $k$  grupos e calcula as médias dos vetores de cada grupo. Em seguida, cada ponto é deslocado para o grupo correspondente ao vetor médio do qual ele está mais próximo. Com este novo rearranjo dos pontos em  $k$  grupos, novos vetores médios são calculados. O processo de re-alocação de pontos a novos grupos cujos vetores médios são os mais próximos deles continua até que se chegue a uma situação em que todos os pontos já estejam nos grupos dos seus vetores médios mais próximos [47].

O primeiro passo do algoritmo  $k$ -médias é a determinação das posições iniciais dos  $k$  centros dos *clusters* onde o número de *clusters* tem que ser definido inicialmente pelo usuário. A Figura 10 mostra esse primeiro passo, com um  $k = 3$ .



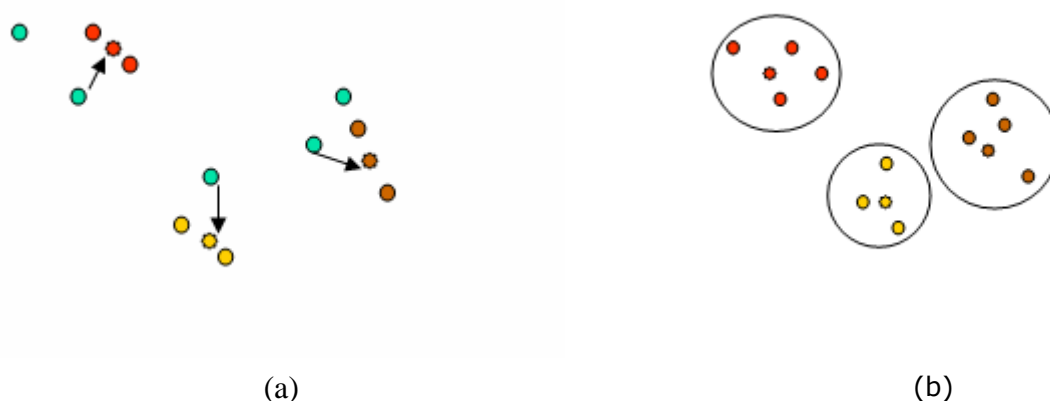
**Figura 10.** (a) Dados a serem agrupados e (b) primeiros  $k$  centros são escolhidos aleatoriamente.

O segundo passo é a alocação de cada elemento ao *cluster* associado com o centro mais próximo. Em seguida, no terceiro passo, é computado um novo centro para cada *cluster*, ou seja, a média dos valores de todos os elementos chamado de centróide. Ambos os passos são apresentados na Figura 11.



**Figura 11.** (a) Elementos são alocados aos *clusters* com o centro mais próximos e (b) um centróide é calculado para representar o *cluster*.

Feito isso, repete-se o segundo e terceiro passos com os novos centróides, ou seja, a alocação de elementos aos *clusters* e o cálculo de um novo centróide até algum critério de convergência, como observa-se na Figura 12.



**Figura 12.** (a) Alocação dos elementos aos clusters e cálculo de novos centróides até um critério de convergência e (b) algoritmo converge resultando nos clusters finais.

Como critério de convergência pode ser empregado um número máximo de iterações ou executar o algoritmo até que os centros não se movam mais, ou apresentem uma mudança muito pequena de posição. No último caso, deve ser definido um erro mínimo no início da execução.

O *k*-médiãs, no entanto, apresenta como dificuldade a definição inicial de *k* clusters distintos e se são realmente suficientes para representar os dados.

#### 4.2.2 Mapas Auto-Organizados - SOM

As redes SOM (*Self-Organized Maps*) [48] são redes neurais artificiais que possuem a capacidade de auto-organização. Em contraste com outros modelos de redes neurais, tem uma forte inspiração fisiológica, pois são baseadas no mapa topológico que existe no córtex do cérebro. O córtex é organizado de modo que neurônios topologicamente próximos tendem a produzir respostas ao mesmo tipo de estímulo, esta é uma das razões pelas quais as redes SOM são amplamente empregadas em reconhecimento de padrão visual.

As SOMs utilizam-se do paradigma de aprendizagem não supervisionada e são baseadas no aprendizado competitivo, onde os neurônios competem entre si para serem ativados. O neurônio vencedor tem o direito de atualizar seus pesos [48]. O algoritmo SOM, similar ao  $k$ -médias, é um algoritmo de partição onde o número de *clusters* é definido inicialmente; diferente dos *clusters* gerados pelo  $k$ -médias, no entanto, nas SOMs há uma ordenação topológica dos *clusters*.

Segundo [49], quando um padrão de entrada é apresentado, a rede SOM procura a unidade mais semelhante a esse padrão. Durante o treinamento, a rede aumenta a semelhança do nodo escolhido e de seus vizinhos a esse padrão de entrada. Desta forma, um mapa topológico é construído pela rede onde nodos que estão topologicamente próximos respondem de forma semelhante a padrões de entrada semelhantes.

Após o treinamento, a rede SOM organiza os padrões de entrada em *clusters*. Algumas aplicações requerem a rotulação dos nós de saída para indicar os *clusters* que representam. Isto é útil para a classificação de padrões desconhecidos [49]. Desde que seus parâmetros sejam ajustados adequadamente, o algoritmo SOM geralmente leva a uma representação organizada de padrões de ativação retirados do espaço de entrada [50].

As redes SOM são utilizadas geralmente para explorar grandes conjuntos de dados, extraindo as características e relações mais relevantes entre estes dados [51]. Imagens são consideradas dados que, às vezes, escondem informações úteis. O uso de redes SOM permite a criação de conjuntos de características que mantêm os dados mais relevantes contidos em uma imagem.

A dificuldade apresentada pelo algoritmo SOM é a dependência de muitos parâmetros para a obtenção do resultado, como a inicialização dos pesos, a taxa de aprendizagem, taxa de vizinhança e número de iterações.

## 4.3 Medidas de Distância

Na procura de imagens visualmente similares, dada uma imagem de consulta, em um sistema de recuperação de imagens é importante levar em consideração dois aspectos: uma representação apropriada das características visuais das imagens e uma medida que determine o grau de similaridade ou dissimilaridade das imagens de resposta. Considerando que não existe nenhuma anotação textual da imagem, as características que podem ser utilizadas são descrições do conteúdo da imagem que podem ser representados pela cor, textura e forma. Estas características mudam sobre uma mesma imagem, por variações inerentes da aparência da superfície e como resultado de mudanças na iluminação, sombreamento, entre outros [31].

Para classificação, segmentação e recuperação de imagens, foram desenvolvidas medidas de similaridade baseadas em estimativas empíricas da distribuição das características dessas imagens. Para dados numéricos, a distância entre dois objetos é um dos métodos mais simples, onde valores pequenos indicam maior similaridade entre os objetos, assim como valores maiores indicam menor similaridade ou maior dissimilaridade entre os objetos. Então, deve-se definir uma medida de distância entre dois objetos. Sejam dois objetos,  $S$  e  $Q$  pertencentes a um mesmo domínio, a distância entre os dois objetos é denotada por :  $d(S, Q)$ .

Por exemplo, se os objetos têm sido caracterizados por vetores de características de igual tamanho, a distância  $d()$  pode ser a distância Euclidiana entre eles:

$$d(S, Q) \equiv \left( \sum_{i=1} (S[i] - Q[i])^2 \right)^{1/2} \quad (14)$$

Onde  $S$  e  $Q$  são dois objetos,  $S[i]$  e  $Q[i]$  correspondem ao valor da  $i$ -ésima dimensão de  $S$  e  $Q$  respectivamente.

Medidas de distância são utilizadas em várias áreas, incluindo aprendizagem baseada em instâncias, redes neurais, estatísticas, reconhecimento de padrões e psicologia cognitiva. Muitos sistemas inteligentes dependem da eficiência de uma função para cálculo de distância entre dois vetores. Uma variedade de funções de distância está disponível para tais usos, incluindo as métricas de distância Minkowsky, Mahalanobis, Canberra, Chebychev, Quadrática, Correlação e Qui-quadrado [31].

No próximo Capítulo, mostramos a aplicação da transformada *wavelet* às imagens médicas de imuno-histoquímica a fim de avaliar estatisticamente os sub-espacos *wavelets* formados para a geração dos vetores de características das imagens e são expostos os resultados do agrupamento desses vetores.

## Capítulo 5

# Resultados e Discussões

### 5.1 Introdução

Os experimentos desenvolvidos neste trabalho envolvem a extração de características de imagens de imuno-histoquímica baseada em *wavelets* detalhadas em Capítulos anteriores. Especificamente, descrevem-se os conjuntos de imagens usados nos experimentos, as abordagens adotadas na extração e geração do vetor de características, e a comparação dos resultados obtidos pelo agrupamento das imagens com a interpretação semântica dos usuários e com o algoritmo de clusterização *k*-médias.

Para todos os testes, o banco de imagens utilizados foi gerado em 2001 no hospital das clínicas da UFAL [2].

Para a extração de características das imagens, foi aplicada a transformada wavelet, que modela a textura no domínio da frequência espacial, com o filtro Daubechies-4, onde o número 4 representa o tamanho do filtro. Tanto a aplicação da transformada wavelet quanto a geração dos vetores de características das imagens foram todas realizadas no software Scilab [52] e com o uso das toolboxes SIVP (*Scilab Image and Vídeo Processing toolbox*) [53] e SWT (*Scilab Wavelet Toolbox*) [54].

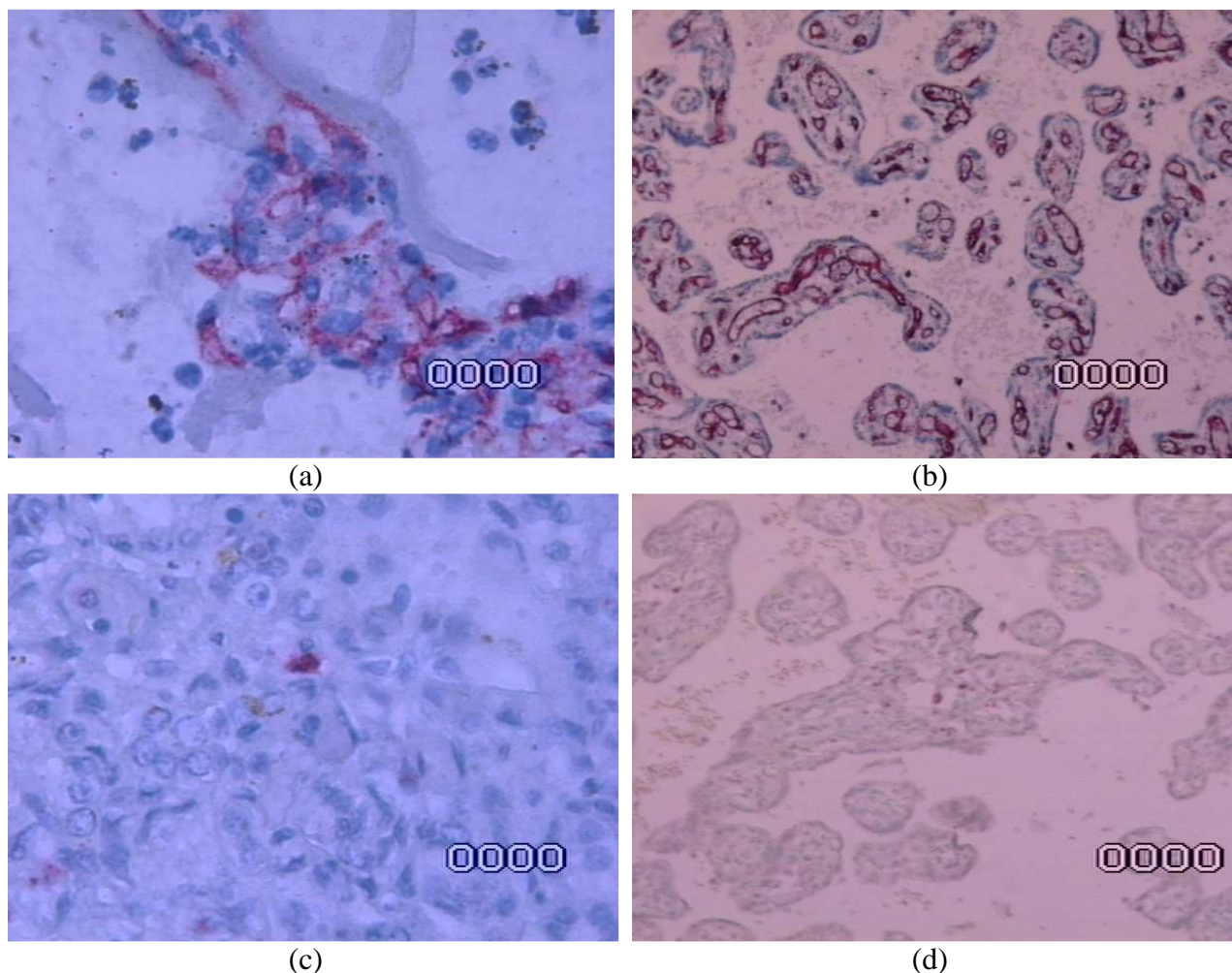
Por fim, foi criado um arquivo ARFF, com os vetores de características gerados das imagens, para ser usado no software *Weka* [55], com a finalidade de obter os resultados da clusterização das imagens através do algoritmo *k*-médias. Esses resultados foram, então, comparados com os resultados obtidos pela interpretação semântica dos usuários através de consultas ao ambiente *web* desenvolvido.

### 5.2 Conjuntos de dados

As imagens utilizadas no trabalho, são imagens de imuno-histoquímica da placenta e do pulmão e podem ser do tipo CD68 e CD34, que significam o tipo de antígeno. Na nomenclatura



das imagens, NP significa placenta, enquanto P quer dizer pulmão. Logo, uma imagem pode ser CD68 ou CD34, que por sua vez podem ser P ou NP. Assim, tem-se 4 tipos de imagens como pode ser observado na figura 13.



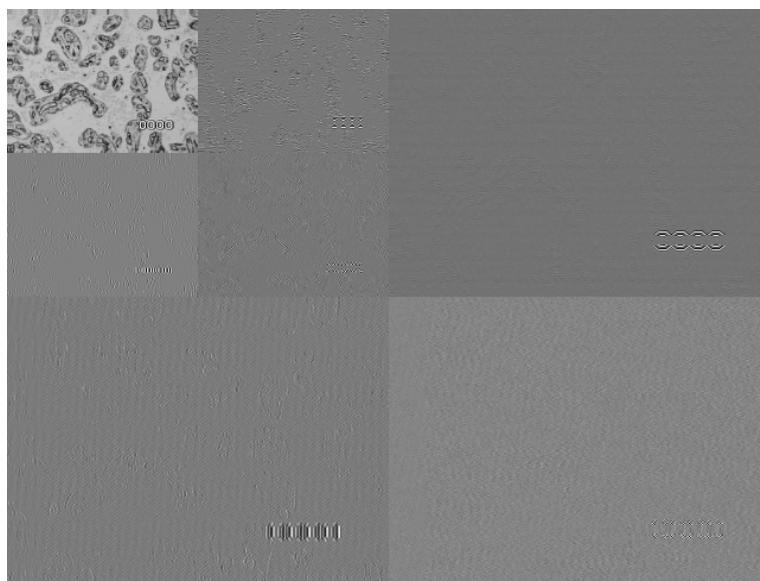
**Figura 13.** (a) CD34 NP001a, (b) CD34 P003b, (c) CD68 NP993a e (d) CD68 P9824b

Essas imagens estão armazenadas em formato bitmap, mas também encontra-se imagens do tipo CD68 no formato JPEG. Todas as imagens têm dimensões 640 x 480 *pixels*.

### 5.3 Geração dos Vetores de Características

Para geração dos vetores de características, foi separado um grupo de cem imagens de imuno-histoquímica, vinte e cinco imagens de cada classe, sobre o qual foi aplicada a *wavelet* Daubechies-4 em dois níveis de decomposição, gerando, assim, seis sub-espacos *wavelets* como mostra a Figura 14.





**Figura 14.** Transformada *wavelet* Daubechies-4 nível 2 aplicada em imagem de imuno-histoquímica

Para o cálculo das *wavelets*, as imagens foram convertidas para tons de cinza e, em seguida, para o tipo *double* pois a SWT [54] só aceita imagens desse tipo como entrada para suas funções [56]. A função de decomposição *wavedec2* [57] é aplicada à imagem, recebendo como parâmetros de entrada a quantidade de nível de decomposição e filtro desejados, além da matriz que representa a imagem. As funções *appcoef2* [57] e *detcoef2* [57] são aplicadas às saídas da *wavedec2* (vetores de decomposição), calculando os coeficientes de aproximação e detalhes, respectivamente. A função *detcoef2* é aplicada duas vezes para calcular os coeficientes de detalhes horizontal, vertical e diagonal nos dois níveis de decomposição. Feito isso, os coeficientes são “cortados” para seus tamanhos apropriados através da função *wkeep* [56], para formarem as sub-imagens que representam os sub-espacos *wavelets* na imagem resultante. Por fim, esses mesmos coeficientes são normalizados e utilizados para o cálculo das medidas estatísticas que formarão os vetores de características.

Nos testes realizados, após aplicar a transformada *wavelet* em dois níveis de decomposição, é gerada uma sub-imagem de baixa frequência (coeficientes de aproximações), e seis sub-espacos *wavelets* (cada três nas direções horizontal, vertical e diagonal) que correspondem as sub-imagens direcionais de alta frequência (coeficientes de detalhes), os quais são aproveitados para realizar uma análise estatística para a extração de características e assim, gerar o vetor de características. Cada elemento do vetor corresponde a uma descrição estatística de cada sub-imagem, de maneira que o conteúdo presente na imagem seja a correta descrição feita pelo vetor.

As medidas estatísticas utilizadas para extrair as características e compor os vetores de características, neste trabalho, foram a média, calculada a partir da função *mean2* [53] e avalia a luminosidade da imagem; a variância, calculada a partir da função *variance* [52], indicando o contraste da imagem; e o momento de terceira ordem, calculado pela função *moment* [52], cuja ordem é um parâmetro de entrada da função, neste caso o número três, e indica a distorção da imagem.

Assim, obtém-se, para cada imagem, um vetor de características composto de dezoito posições, já que tem-se seis sub-espacos *wavelets* e para cada um é calculado a média, a variância e o momento de terceira ordem. Foram então gerados cem vetores, vinte e cinco para cada tipo de imagem, com dezoito posições cada, representando as cem imagens separadas para os testes.

## 5.4 Resultados dos Agrupamentos

Os testes de clusterização das imagens foram realizados em duas etapas. Na primeira, os vetores de características gerados no Scilab, foram agrupados no *Weka* através do algoritmo *k*-médias, a fim de avaliar os resultados da clusterização dos vetores em suas respectivas classes. Na segunda, o agrupamento é feito levando em conta a interpretação semântica do conteúdo das imagens pelos usuários em um ambiente *web* desenvolvido para esse fim. Por fim, os resultados das duas etapas foram avaliados e comparados.

### 5.4.1 Agrupamento no Weka

Depois de gerados os vetores de características, foi criado um um arquivo *ARFF* para realizar a clusterização dos vetores no *Weka*.

Um arquivo no formato *ARFF* é um arquivo de texto puro, composto de três partes. A primeira é a relação, primeira linha do arquivo, que deve ser igual a *@relation* seguida de uma palavra chave que identifique a relação a ser estudada. A segunda parte são os atributos, composta de um conjunto de linhas onde cada uma inicia com *@attribute* seguida do nome do atributo e seguida do seu tipo, que pode ser nominal ou numérico. E a terceira parte é a dos dados, começando após uma linha contendo *@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.

O arquivo “vetores.arff” gerado neste trabalho é composto de dezoito atributos numéricos, que são os componentes dos vetores de características, e um atributo nominal representando uma das quatro classes que as imagens podem fazer parte : CD34NP, CD34P, CD68NP ou CD68P. Já a parte de dados é composta de cem instâncias, correspondendo aos cem vetores. A Figura 15 mostra uma parte do arquivo contendo as primeiras cinco instâncias de dados.

No software *Weka*, o arquivo é aberto e na aba de *cluster* escolhe-se o algoritmo *SimplekMeans*, ou seja, o algoritmo de clusterização *K*-médias com número de clusters igual a quatro. Modificou-se apenas o número de iterações do algoritmo a fim de analisar a menor porcentagem de clusterizações incorretas das instâncias.

### 5.4.2 Agrupamento Segundo Interpretação Semântica

O mesmo conjunto de cem imagens utilizado para o cálculo dos vetores de características e geração do arquivo *ARFF* é cadastrado no ambiente *web* desenvolvido na linguagem PHP. As Figuras 16 e 17 mostram as telas do ambiente para o cadastro de imagens visualização da imagem cadastrada no banco de dados, respectivamente.

```
@RELATION vetores

@ATTRIBUTE medh1      REAL
@ATTRIBUTE varh1      REAL
@ATTRIBUTE mh1        REAL
@ATTRIBUTE medv1      REAL
@ATTRIBUTE varv1      REAL
@ATTRIBUTE mv1 REAL
@ATTRIBUTE medd1      REAL
@ATTRIBUTE vard1      REAL
@ATTRIBUTE md1 REAL
@ATTRIBUTE medh2      REAL
@ATTRIBUTE varh2      REAL
@ATTRIBUTE mh2 REAL
@ATTRIBUTE medv2      REAL
@ATTRIBUTE varv2      REAL
@ATTRIBUTE mv2 REAL
@ATTRIBUTE medd2      REAL
@ATTRIBUTE vard2      REAL
@ATTRIBUTE md2 REAL
@ATTRIBUTE class      {CD34NP,CD34P,CD68NP,CD68P}

@DATA
%CD34NP
0.454716, 0.000624, 0.094942, 0.518975, 0.001127, 0.141519, 0.541715, 0.000552,
0.159861, 0.460559, 0.003091, 0.102032, 0.501467, 0.001536, 0.128411, 0.504815,
0.001531, 0.130954, CD34NP
0.447640, 0.000541, 0.090486, 0.488341, 0.000920, 0.117802, 0.523369, 0.000729,
0.144498, 0.475491, 0.000816, 0.108701, 0.516943, 0.000583, 0.139039, 0.557132,
0.000698, 0.174087, CD34NP
0.451886, 0.000509, 0.093024, 0.436279, 0.000802, 0.084105, 0.495034, 0.000530,
0.122100, 0.463818, 0.000924, 0.101105, 0.527504, 0.000575, 0.147686, 0.546376,
0.000699, 0.164243, CD34NP
0.453754, 0.000626, 0.094351, 0.473425, 0.001053, 0.107623, 0.515491, 0.000704,
0.138068, 0.455726, 0.002127, 0.097598, 0.517849, 0.001130, 0.140625, 0.506115,
0.001326, 0.131653, CD34NP
0.454531, 0.000688, 0.094915, 0.426479, 0.001439, 0.079437, 0.476481, 0.000796,
0.109315, 0.445191, 0.003387, 0.092814, 0.518553, 0.001682, 0.142056, 0.531487,
0.002587, 0.154250, CD34NP
```

**Figura 15.** Parte do arquivo “vetores.arff”



**Figura 16.** Tela para cadastrar imagens



**Figura 17.** Visualização da imagem cadastrada no banco

A tela principal do ambiente, onde são realizadas as consultas, é mostrada na Figura 18. Nessa tela, é apresentada uma imagem principal e outras nove imagens que são selecionadas de acordo com a opinião do usuário em relação à semelhança delas com a imagem principal. Ou seja, só são selecionadas as imagens que a pessoa consultada achar que os conteúdos são similares ao da imagem principal. Todas as imagens são escolhidas aleatoriamente no banco de dados do sistema e mudam a cada nova consulta.

As imagens selecionadas são cadastradas em uma tabela no banco e indexadas à imagem principal da consulta. Assim, é formada uma relação identificando, por exemplo, que a imagem NP001b do tipo CD34 foi associada, segundo a interpretação semântica do usuário, as imagens P9422b e NP992a dos tipos CD34 e CD68 respectivamente. Foram consultados dez usuários leigos em análise de imagens de imuno-histoquímica, com a finalidade, apenas, de agrupar as imagens segundo a interpretação semântica desses usuários. O agrupamento das imagens nas quatro classes mencionadas anteriormente foi realizado através da análise das relações resultantes das consultas.



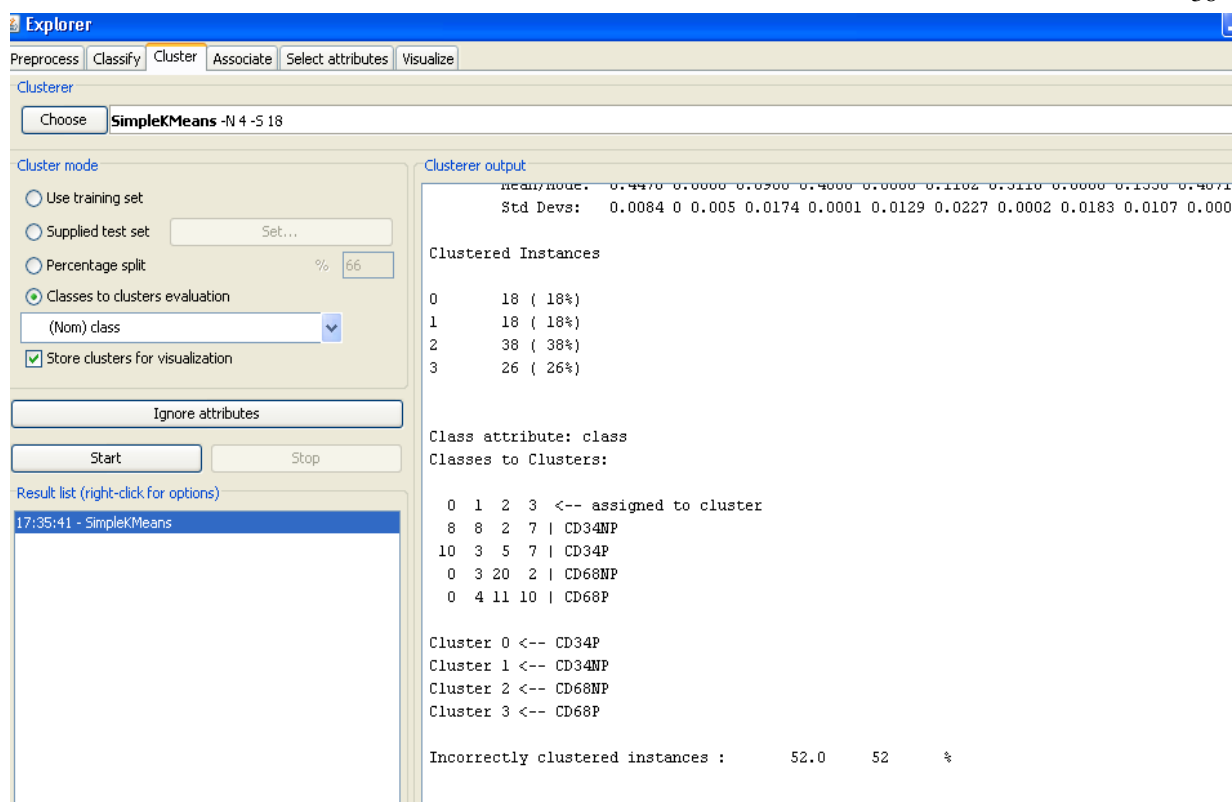
**Figura 18.** Tela para consultar a opinião dos usuários sobre a semelhança das imagens

### 5.4.3 Comparação dos Resultados

Após vários testes no Weka, o melhor resultado obtido foi com seis iterações do algoritmo k-médias, com o número de clusters igual a quatro previamente conhecido. A Figura 19 mostra a tela de visualização dos resultados da clusterização dos vetores obtidos no Weka. Como pode ser observado na figura, o modo de cluster escolhido avalia o cluster para a sua respectiva classe. A porcentagem de clusterização incorreta das instâncias foi de 52% e representa o melhor resultado encontrado. Para sete iterações do algoritmo, por exemplo, a porcentagem de erro foi de 60% e o pior resultado foi para cinco iterações que obteve erro de 62%.

Ainda analisando a Figura 19, observa-se as porcentagens de instâncias atribuídas a cada classe e a distribuição das instâncias nos quatro *clusters*. Para melhor visualização, a Tabela 1 mostra esses valores. As linhas representam a distribuição das vinte e cinco instâncias de cada classe nos quatro *clusters* representados pelas colunas. Os números em vermelho representam a classificação correta da classe ao seu *cluster* correspondente.





**Figura 19.** Visualização dos resultados da clusterização no *Weka*

**Tabela 1.** Resultados da clusterização através do algoritmo *k*-médias no *Weka*

Clusters				
Classes	CD34-NP	CD34-P	CD68-NP	CD68-P
CD34-NP	8	8	2	7
CD34-P	3	10	5	7
CD68-NP	3	0	20	2
CD68-P	4	0	11	10
% atribuída a cada classe	18%	18%	38%	26%

Na Tabela 2, estão os resultados obtidos a partir das consultas aos usuários, onde a clusterização foi realizada observando-se as relações existentes no banco entre imagens e as selecionadas como semelhantes a elas. Foram consultadas dez pessoas e a cada uma delas foram feitas 10 consultas. Assim, por exemplo, nove imagens das vinte e cinco da classe CD34-NP foram selecionadas como sendo desse mesmo tipo, como pode ser observado na tabela.

**Tabela 2.** Resultados da clusterização através da interpretação semântica dos usuários

Clusters				
Classes	CD34-NP	CD34-P	CD68-NP	CD68-P
CD34-NP	9	4	7	5
CD34-P	5	12	2	6
CD68-NP	4	3	9	9
CD68-P	2	3	9	11
% atribuída a cada classe	20%	22%	27%	31%

Observando-se os resultados das duas tabelas, conclui-se que mesmo com uma porcentagem alta de erro, a técnica proposta se mostrou mais eficiente do que a interpretação do conteúdo das imagens pelos usuários consultados. A clusterização semântica feita pelos usuários classificou 41% das instâncias corretamente a seus respectivos *clusters* obtendo um erro de 59% contra 52% de erro obtido pela clusterização dos vetores de características gerados pela técnica proposta neste trabalho. Além disso, os resultados da Tabela 1 mostram que vinte das vinte e cinco instâncias da classe CD68-NP foram atribuídas corretamente ao seu *cluster* correspondente contra nove da Tabela 2. Constata-se ainda que nenhuma instância das classes CD68-NP e CD68-P foram atribuídas ao *cluster* correspondente a classe CD34-P na Tabela 1 contra seis instâncias dessas classes atribuídas ao mesmo *cluster* na Tabela 2. Na Tabela 1, o *cluster* que mais agrupou instâncias foi o CD68-NP com 38% das instâncias, sendo que vinte dessas corretamente classificadas e na Tabela 2 o *cluster* que mais agrupou instâncias foi o CD68-P com 31% das instâncias onde onze dessas foram classificadas corretamente.

A partir dos resultados obtidos, conclui-se que seria necessário a inclusão de informação de textura das diferentes bandas da imagem colorida. Uma vez que, no processo de formação de imagens na técnica de imuno-histoquímica a informação de cor é essencial e integra a informação semântica da imagem num sistema CBIR.

Neste trabalho, não houve preocupação em acrescentar informação de cor porque o interesse inicial é investigar a possibilidade de construir um sistema CBIR apenas com descritores de textura atingindo um desempenho razoável para a tarefa de diferenciação de tipos de imagens de imuno-histoquímica a serem analisadas pelo patologista, dado que para a tarefa de triagem esperava-se que uma abordagem mais simples fosse suficiente.

#### 5.4.4 Dificuldades Encontradas

Nesta Seção, julga-se importante também relatar alguns problemas encontrados durante a realização deste trabalho. Inicialmente, a proposta era utilizar a *toolbox* SIP (*Scilab Image Processing toolbox*) [58] para o processamento das imagens no Scilab. No entanto, ao tentar realizar os primeiros testes com *wavelets* no Scilab, os resultados não eram gerados devido a *toolbox* SWT não ser compatível com a SIP, por esse motivo nenhuma função da SWT aceitava as matrizes das imagens processadas com o uso da SIP. Daí passou-se a usar a *toolbox* SIVP e o problema foi solucionado.

Outro problema encontrado, foi com a *toolbox* SIVP que não era carregada ao iniciar o Scilab pois não conseguia encontrar o arquivo “loader.sce”, mesmo esse estando presente na pasta contendo os arquivos de instalação da *toolbox*. Depois de reinstalar várias vezes tanto o Scilab quanto a SIVP e não solucionar o problema, a solução encontrada foi executar o arquivo “loader.sce” a partir do seu local de instalação através de um comando `exec`. Para o trabalho em questão o comando executado era:

```
exec ("c:\arquivos de programas\scilab-4.1.2\contrib\sivp/loader.sce")
```

Para não ficar executando esse comando toda vez que precisar-se utilizar a *toolbox* SIVP ao iniciar o Scilab, o comando foi incorporado ao *script* responsável pela aplicação da transformada *wavelet* e geração dos vetores de características, sendo necessário apenas a execução do *script* uma única vez.

Por último, os testes com a *toolbox* SWT foram inicialmente realizados com as imagens da ordem  $2^9$  pixels, providas pela própria *toolbox*. Os sub-espacos e coeficientes *wavelets* foram gerados corretamente com essas imagens, porém quando os mesmos testes foram realizados com as imagens de imuno-histoquímica (640 x 480 pixels), ocorria um erro e os resultados não eram gerados. Isso porque, a função *wkeep*, responsável pelo “corte” das sub-imagens, não aceitava como entrada imagens com 320 x 240 pixels, só aceitava imagens da ordem  $2^8$  pixels, como as

imagens dos testes providas pela *toolbox*. A solução foi aplicar a transformada *wavelet* para um nível de decomposição e descobrir o tamanho das sub-imagens geradas (323 x 243 *pixels*), já que para uma única decomposição *wkeep* não é utilizada. Feito isso, o tamanho encontrado foi testado na função *wkeep* para dois níveis de decomposição, o que finalmente funcionou.



## Capítulo 6

# Conclusão e Trabalhos Futuros

A Recuperação de Imagens Baseada em Conteúdo (*Content-Based Image Retrieval* – CBIR) é uma aplicação cujo interesse vem crescendo devido as limitações herdadas das técnicas tradicionais de recuperação de imagens. Em vários trabalhos, o uso da transformada *wavelet* para representar textura tem sido citado como uma boa maneira de caracterizar o conteúdo da imagem, no entanto, por tratar-se de uma abordagem relativamente nova, muito estudo na área ainda precisa ser realizado. No caso de recuperação de imagens médicas, a análise e escolha da família *wavelet* a ser utilizada é de grande importância para melhor precisão na caracterização das mesmas, pois existem várias funções *wavelets* que têm comportamento distinto de acordo com o domínio das imagens.

### 6.1 Contribuições

Este trabalho procurou ser uma contribuição aos estudos de recuperação de imagens, que ganharam bastante repercussão na última década devido ao rápido crescimento do uso de imagens digitais na *web* e à ineficiência das técnicas de indexação textual em localizar imagens em grandes bases de dados.

A aplicação da transformada *wavelet* foi sugerida como uma solução para o processo de caracterização de imagens. Propõe-se o tratamento de textura na análise de imagens médicas mediante o uso de filtros *wavelets* e testes para extração de texturas e geração de vetores de características a partir da análise estatística dos sub-espacos *wavelets* em imagens de imuno-histoquímica foram realizados.

Além disso, uma ferramenta foi desenvolvida com a finalidade de agrupar as imagens cujos conteúdos fossem semelhantes segundo a interpretação semântica dos usuários consultados. A partir dessa ferramenta, lança-se a idéia de formação de um ambiente *web* para triagem de imagens similares, podendo auxiliar patologistas na difusão de informação destas imagens.

## 6.2 Trabalhos Futuros

Para uma melhor caracterização das imagens, novos testes podem ser realizados, aplicando-se a transformada *wavelet* em mais de dois níveis de decomposição, gerando mais sub-espacos *wavelets*, pois para cada nível de decomposição, três sub-espacos são gerados. Já que esses sub-espacos correspondem as sub-imagens de alta frequência, as imagens seriam caracterizadas com maior detalhamento e consequentemente a recuperação mais eficiente.

Um outro trabalho futuro sugerido, é a utilização de outros descritores estatísticos para a análise dos sub-espacos *wavelets* como os vistos na Seção 2.3.2. A entropia e a energia, que indicam, respectivamente, a suavidade e a uniformidade da imagem, são descritores que devem ser utilizados na geração de novos vetores de características a fim de testar se caracterizam melhor ou não as imagens, observando para isso, os resultados do agrupamento desses novos vetores.

A geração de vetores de características com maiores dimensões para uma caracterização mais precisa da imagem, deve ser realizada. Para isto, é necessária a adição de informação de textura em cada banda da imagem colorida, ou seja, separar a imagem nos três níveis de cinza RGB, aplicar a transformada *wavelet* em cada banda, aplicar os descritores estatísticos e gerar um vetor com os valores obtidos para as três bandas.

Análise e testes com novos filtros *wavelets* podem ser feitos, uma vez que, o uso de filtros que melhor caracterizem a textura não só acrescenta a eficiência computacional, mas também, extrai maior informação significativa da imagens.

## Bibliografia

- [1] BAKINS, John; GRAHAM, Margaret. *Content-Based Image Retrieval*. University of Northumbria, Newcastle.
- [2] 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.
- [3] FENG, D; LONG, F. e ZHANG, H. *Fundamentals of Content-Based Image Retrieval*. Disponível em: <[http://old.hki.uni-koeln.de/teach/ws0708/VIP/tag9/ch01\\_Long\\_v40-proof.pdf](http://old.hki.uni-koeln.de/teach/ws0708/VIP/tag9/ch01_Long_v40-proof.pdf)> Acesso em: 03 de março de 2008.
- [4] MÜLLER, H. *et al. Evaluation axes for medical image retrieval systems: the image CLEF experience*. In: The 13th annual ACM international conference on Multimedia, 2005, Hilton, Cingapura: ACM Press; 2005. p. 1014-1022.
- [5] RASIWASIA, N. *et al. Query by semantic example*. Statistical Visual Computing Lab, University of California, San Diego.
- [6] MAFRA, P. M. e MOLL, V. *Recuperação Semântica de informação na Web*. Disponível em : < [http://www.das.ufsc.br/~gb/pg-ia/InfoWeb/sem\\_ia.pdf](http://www.das.ufsc.br/~gb/pg-ia/InfoWeb/sem_ia.pdf)> Acesso em : 04 de março de 2008.
- [7] DATTA, Ritendra *et al.* Wang. *Image Retrieval: Ideas, Influences, and Trends of the New Age*. *ACM Computing Surveys*, 2008.
- [8] DAS, G. e RAY, S. *A Comparison of Relevance Feedback Strategies in CBIR*. 6th IEEE/ACIS International Conference, p. 100-105. 2007.
- [9] FLICKNER, M, *et al. Query by image and video content system*. *IEEE Computer* 28 (5);1995. p. 23-32.
- [10] LEE, D. *et al. Query by image content using multiple objects and multiple features: user interfaces issues*. *IEEE ICIP'94*; p. 76-80. 1994.

- [11] GONZALES, R.C. e WOODS, E.R. *Processamento de Imagens Digitais*. São Paulo: Edgard Blucher, 2000.
- [12] SWAIN, M. e BALLARD, D. Color Indexing. *International Journal of Computer Vision*, v.7 n° 1, 1991. p.11–32.
- [13] 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.
- [14] Virage em : <http://www.virage.com>. Acesso em 10 de março de 2008.
- [15] VisualSEEK em : <http://www.ctr.columbia.edu/visualSEEK>. Acesso em 10 de março de 2008.
- [16] OLIVEIRA , R. A. R.; CLAUDINO L. M. B.; ARAÚJO A. DE A. *Recuperação de Informação Visual com Base no Conteúdo Através do Uso de Wavelets*. Anais do II Workshop em tratamento de imagens, NPDI/DCC-UFMG, Belo Horizonte, 2001.
- [17] THEODORIDIS, S. *Pattern Recognition*. Academic Press, Greece, 1999.
- [18] GARCIA, A. F. *Análise de Fourier*. Disponível em : < <http://www.fisica.ufs.br/CorpoDocente/egsantana/ondas/fourier/Fourier.html> >. Acesso em : 15 de março de 2008.
- [19] CHÁVEZ, G. C. *Sistema Celular Evolutivo para Reconhecimento de Padrão Invariante*. Tese de mestrado, Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo, 2002.
- [20] DA FONTOURA COSTA, L. e CESAR, R. M.. *Shape Analysis and Classification: Theory and Practice*. CRC Press, 2000.
- [21] TUCERYAN, M. e JAIN, A.. *Texture Analysis in Handbook of Pattern Recognition and Computer Vision*. World Scientific Publishing Company; 1993. p. 235–276.
- [22] BELINO, N.J.R.; NUNES, M.F.; GERALDES, M.J. *Caracterização de Mantos Agulhados por Recuperação de Conteúdo de Imagem*. Workshops Estruturas Têxteis Inteligentes, Technical University of Liberec, Guimarães, Portugal, 2005.
- [23] TRAINA, A. J. *Suporte à Visualização de Consultas por Similaridade em Imagens Médicas através de Estruturas de Indexação Métrica*. Tese de livre-docente em computação, Instituto de Ciências Matemática e de Computação - Universidade de São Paulo, São Paulo, 2001.
- [24] MELLO, C. *Synthesis of Images of Historical Documents for Web Visualization*. IEEE Multimedia Modelling, Brisbane, Austrália, 2004.

- [25] SONKA, M. ; HLAVAC, V. e BOYLE, R. *Image Processing, Analysis, and Machine Vision*. Pacific Grove: Brooks/Cole, 1999.
- [26] HARALICK, R. *et al.* Textural Features for Image Classification. *IEEE Transactions on Systems, Man and Cybernetics*, p. 610–621, 1973.
- [27] ROSENFELD, A. e KAK, A. C. *Digital picture processing*. San Diego: Academic Press, 1982.
- [28] LOEW, M. H. *Feature Extraction*. SPIE, Bellingham, WA, M. Sonka and J. Michael Fitzpatrick Edition, 2000.
- [29] UNSER, M. Sum and Differenc Histograms for texture classification. *IEEE Transactions on Pattern Analysis and Machine Inteligence*, volume 8, p. 118-125. 1986.
- [30] OJALA, T.; PIETIKÄINEN, M., *Unsupervised Texture Segmentation Using Feature Distributions*. Pattern Recognition. v. 32, n. 3, p. 477-486, 1999.
- [31] CASTAÑÓN, César A. Beltrán. *Recuperação de imagens por conteúdo através de análise multiresolução por Wavelets*. 2003. Dissertação de mestrado em Ciências da Computação e Matemática Computacional, ICMC – USP, São Carlos.
- [32] DAUBECHIES, I. *Ten Lectures on Wavelets*, volume 61. CBMS-NSF Regional Conference Series in Applied Mathematics, Rutgers University and AT&T Bell Laboratories, 1992.
- [33] GRAPS, A. L. An Introduction to Wavelets. *IEEE Computational Sciences and Engineering*, volume 2, p. 50-61. 1995.
- [34] CASTLEMAN, K. R. *Digital Imaging Processing*. 1ª edição, Prentice Hall, 1996.
- [35] FRISBYL, J. P. *Seeing*. Oxford University, p. 39-69. 1980.
- [36] BRADLEY, A. P. A Wavelet Visible Difference Predictor. *IEEE Transactions On Image Processing* 8, 1999. p. 717-730.
- [37] OLIVEIRA, R. A. R. *et al.* *Extração de Características Utilizando Wavelets para Indexação de Imagens*. Anais do I Workshop em tratamento de imagens, NPDI/DCC-UFGM, Belo Horizonte, 2000.
- [38] MALLAT, S. G. A Theory for Multiresolution Signal Decomposition: TheWavelet Representation. *IEEE Transaction on Pattern analysis and Machine Intelligence*, 1989, 11(7):674–693.
- [39] SCHNITER, P. *Downsampling*. Disponível em : < <http://cnx.org/content/m10441/latest> > Acesso em : 18 de abril de 2008.

- [40] ALBUZ, E.; KOCALAR, E.; KHOKHAR, A. A. Scalable Color Image Indexing and Retrieval using Vector Wavelets. *IEEE Transaction on Knowledge and Data Engineering*, 2001, 13(5):851–861.
- [41] WANG, J. Z. *Semantics-Sensitive Integrated Matching for Picture Libraries and Biomedical Image Databases*. Phd. thesis, Departament of Biomedical Informatics of Stanford University, 2000.
- [42] TRAINA, C. *et al.* Slim-Trees: High Performance Metric Trees Minimizing Overlap Between Nodes. *Proc. International conference on Extending Database Technology*, Konstanz, Germany, 2000. p. 51–65.
- [43] JAIN, A., e DUBES, R. *Algorithms for Clustering Data*. Prentice-Hall, p. 155-166. 1988.
- [44] GHAHRAMANI, Zoubin. *Unsupervised Learning*. Disponível em: < <http://www.inf.ed.ac.uk/teaching/courses/pmr/docs/ul.pdf> > Acesso em: 21 de abril de 2008.
- [45] ARTHUR, D. e VASSILVITSKII, S. *k-means++: The Advantages of Careful Seeding*. Disponível em: < <http://www.stanford.edu/~darthur/kMeansPlusPlus.pdf> > Acesso em: 23 de abril de 2008.
- [46] MONTEIRO E SILVA, A. B. *et al.* *Redes Neurais Artificiais e Análise de Sensibilidade: Uma Aplicação à Demanda de Importações Brasileira*. Revecap vol. 5 n. 4, 2001.
- [47] PIMENTEL, E. P. *et al.* *A Identificação de Grupos de Aprendizes no Ensino Presencial Utilizando Técnicas de Clusterização*. XIV Simpósio Brasileiro de Informática na Educação, NCE – IM/UFRJ, 2003.
- [48] KOHONEN, T. *The Self-Organizing Map*. *Proceedings of the IEEE*, vol. 78, n. 9, p. 1464-1480. 1990.
- [49] BRAGA, A.; CARVALHO, André; LUDERMIR, Teresa. *Redes Neurais Artificiais – Teoria e Aplicações*. 1. ed. Rio de Janeiro: LTC, 2000.
- [50] HAYKIN, S. *Redes Neurais: Princípios e Prática*. Bookman, 2 ed. Porto Alegre, 2001.
- [51] KASKI, S. *Data exploration using self organizing maps*. PhD thesis, University of Technology, Espoo, Finland, 1997.
- [52] Scilab em: <http://www.scilab.org/>. Acesso em 02 de março de 2008.
- [53] SIVP em : <http://sivp.sourceforge.net/>. Acesso em 27 de março de 2008.
- [54] SWT em : <http://scwt.sourceforge.net/>. Acesso em 25 de março de 2008.
- [55] Weka em : <http://www.cs.waikato.ac.nz/ml/weka/>. Acesso em 06 de maio de 2008.

- [56] ZHI, I. e LIU, R. *Tutorial of SCILAB Wavelet Toolbox (SWT) and SCILAB Image and Video Processing Toolbox (SIVP)*, 2007.
- [57] Wavedec2 em : <http://www.mathworks.com/>. Acesso em 08 de abril de 2008.
- [58] SIP em: <http://siptoolbox.sourceforge.net/>. Acesso em 02 de março de 2008.