UNIVERSIDADE DE SÃO PAULO

ESCOLA DE ENGENHARIA DE SÃO CARLOS

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

CÉLIO RICARDO CASTELANO

ESTUDO COMPARATIVO DA TRANSFORMADA WAVELET

NO RECONHECIMENTO DE PADRÕES DA ÍRIS HUMANA

**São Carlos - SP
2006**

CÉLIO RICARDO CASTELANO

**ESTUDO COMPARATIVO DA TRANSFORMADA WAVELET
NO RECONHECIMENTO DE PADRÕES DA ÍRIS HUMANA**

Dissertação apresentada à escola de Engenharia de São Carlos, da Universidade de São Paulo, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Área de Atuação: **Processamento de Sinais e Instrumentação**

Orientador: Prof. Dr. Adilson Gonzaga.

**São Carlos - SP
2006**

DEDICATÓRIA

À Deus, criador de tudo e de todos, que possibilitou a conclusão de mais uma etapa importante em minha existência. À Jesus, nosso Mestre e à todos os amigos do plano maior.

À pessoa que me concedeu a oportunidade dessa vida, meu grande exemplo, que com a sua honrosa profissão de manicure lutou para que, primeiramente, seu único filho conseguisse a graduação e agora, esse mestrado. Obrigado dona Maria de Fátima Castelano, tenho muito orgulho de ser seu filho.

À minha filha Isis que muito tem me ensinado com a sua energia e alegria. Espero prover-te não apenas os recursos materiais, mas principalmente, Amor e bons exemplos.

AGRADECIMENTOS

Aos meus avós Augusto e Dídima Castelano, pela criação com os melhores princípios de dignidade e honestidade. À minha esposa Renata e sogra Nair, pelo apoio e incentivo nas horas mais difíceis.

Ao professor Adilson Gonzaga por toda orientação, paciência e compreensão, ensinando-me o que é ser um verdadeiro docente. À professora Agma J. M. Traina, pela valiosa colaboração ao longo deste trabalho e também pelas aulas de Computação Gráfica.

Ao amigo e professor Celso Zácari Faria, por conceder-me a primeira oportunidade de lecionar, obrigado pela confiança e motivação. Ao grande amigo Henrique Rodrigo, uma pessoa que personaliza o sentido real de amizade; obrigado pela ajuda inestimável durante todo o curso, desde a inscrição como aluno especial, o apoio nas disciplinas, as longas estadias, enfim, grande parte desse trabalho devo à você. Obrigado também à Eleisy e família, pela acomodação e cordialidade.

Aos irmãos Ângelo e Ulisses Christianini, da empresa Eletro Ponto de Bauru, que me liberaram do trabalho para que eu cursasse todas as disciplinas. À todos os amigos dessa empresa: Luís Antonio, Heverton, Ricardo Christianini, Rose e Coleu, que sempre me apoiaram.

Ao Beto, Danielle e Jean pelas caronas de São Carlos à Bauru; Ronaldo Martins e Kátia pela companhia nas viagens; à Patrícia Bellin, pela força e grande ajuda desde a qualificação. Gabriel Calin, que tive o prazer de conhecer na disciplina do prof. Adilson, autor principal do método de extração da íris. Ao Leandro, pelos conselhos e motivação; Marcão, Menotti e Vanderlei do LCR.

Ao oráculo Adriano Múfalo, camarada que sabe de tudo um monte; obrigado pela ajuda na integração Delphi/Matlab e tantas outras desde os tempos da graduação. À Alternativa Sistemas, na figura dos não menos amigos Gustavo Lourenço e Miguel Damasceno.

RESUMO

Neste trabalho é apresentado um método para reconhecimento de seres humanos através da textura da íris. A imagem do olho é processada através da análise do gradiente, com uma técnica de dispersão aleatória de sementes. Um vetor de características é extraído para cada íris, baseado na análise dos componentes *Wavelet* em diversos níveis de decomposição. Para se mensurar as distâncias entre esses vetores foi utilizado o cálculo da Distância Euclidiana, gerando-se curvas Recall x Precision para se medir a eficiência do método desenvolvido. Os resultados obtidos com algumas famílias *Wavelets* demonstraram que o método proposto é capaz de realizar o reconhecimento humano através da íris com uma precisão eficiente.

Palavras chave: Reconhecimento pela íris; Biometria; Recuperação de imagem por conteúdo, *Wavelets*; Processamento de imagens; Análise de Textura.

ABSTRACT

This work presents a method for recognition of human beings by iris texture. The image of the eye is processed through gradient analysis, based on a random dispersion of seeds. So, it is extracted a feature vector for each iris based on Wavelet Transform in some levels of decomposition. To estimate the distances between these vectors it was used the Euclidean Distance, and Recall x Precision curves are generated to measure the efficiency of the developed method. The results gotten with some Wavelets families had demonstrated that the proposed methodology is capable to do human recognition through the iris with an efficient precision.

Keywords: *Iris Recognition; Biometrics; Content-Based Image Retrieval, Wavelets; Image Processing; Texture Analysis.*

LISTA DE ILUSTRAÇÕES

Figura 2.3 – Taxas de precisão de sistemas biométricos	18
Figura 3.1 – Esquema geral do funcionamento de um sistema CBIR	30
Figura 3.2 – Formas em imagens. a) Artificiais e b) Naturais (CASTAÑON, 2003).	32
Figura 3.3 – Categorias de representação de formas em imagens. (a) baseada em bordas. (b) baseada em regiões. (AL-TAYECHE; KHALIL, 2003)	32
Figura 3.4 – Imagens com textura	34
Figura 3.5 – Consulta por Abrangência (<i>Range Query</i>).....	36
Figura 3.6 – Consulta pelos k vizinhos mais próximos (<i>k Nearest-Neighbors Query</i>).....	37
Figura 3.7 – Propriedades da desigualdade triangular (VARGAS, 2004).	41
Figura 3.8 – Exemplo de uma R-Tree (FREDERICK, 1999).....	42
Figura 3.9 – Exemplo de uma V-Tree (FREDERICK, 1999).....	43
Figura 3.10 – Exemplo de uma Quad-Tree (FREDERICK, 1999).....	44
Figura 3.11 – Exemplo de uma Slim-Tree (BUENO; TRAINA; TRAINA JR, 2005). (a) Representação planar. (b) Estrutura.	46
Figura 4.1 – Exemplos de <i>wavelets</i>	49
Figura 4.2 – Processo de decomposição das <i>wavelets</i>	50
Figura 4.3 – Árvore de três níveis de decomposição <i>wavelet</i>	50
Figura 4.4 – (a) Janelas de análise no plano tempo-freqüência para a WFT(Transformada de Fourier por Janelas) e para as <i>wavelets</i> – (b) Cobertura do espectro de freqüências pela Transformada por Janelas de Fourier (acima) e pela Transformada Wavelet.	54
Figura 4.5 – (a) 3 exemplos de escala de sinal em <i>wavelet</i> . (b) posicionamento(deslocamento) de sinal em <i>wavelet</i> (MISITI et al, 2005).	55
Figura 4.6 – Análise de um sinal 1-D através da DWT	57
Figura 4.7 – (a) Esquema de organização dos coeficientes Wavelets. (b) Imagem original. (c) Decomposição em 3 níveis pela wavelet Daubechies-4. (d) os três níveis de decomposição(L1, L2, L3) vistos individualmente, com seus respectivos coeficientes de aproximação, detalhes horizontais, diagonais e verticais	58
Figura 4.8 – Árvore de decomposição de sinais na Transformada Wavelet Packet, onde <i>A</i> representa os coeficientes de aproximação (filtro passa-baixa) e <i>D</i> o coeficiente de detalhe (filtro passa-alta)	59
Figura 4.9 – Árvore da organização da <i>Wavelet Packets</i> . A escala <i>j</i> define os níveis de decomposição e <i>W(n)</i> a posição.	60
Figura 5.1 – Equalização de Histograma. (a) Imagem original. (b) Histograma da imagem original. (c) Imagem equalizada. (b) Histograma da imagem equalizada.....	66
Figura 5.2 – (a) Nove sementes na imagem. (b) Direção de varredura das sementes	67
Figura 5.3 – Localização da pupila através da dispersão de 9 sementes, onde cada semente possui uma cor diferente. (a) Imagem completa. (b) Destaque das sementes na região da pupila.	68

Figura 5.4 – Pupila não localizada.....	69
Figura 5.5 – Pupila localizada.....	70
Figura 5.6 – (a) Imagem original. (b) Imagem processada	71
Figura 5.7 – (a) Região inicial de análise das 6 sementes (retângulos)	72
Figura 5.8 – Íris normalizada em 64x512 pixels.....	73
Figura 5.9 – Separação da região de interesse.....	74
Figura 5.10 – imagem referente à uma íris não detectada.	75
Figura 5.11 – Região de interesse com influência da pálpebra.	76
Figura 5.12a – Região de interesse sem influência da pálpebra (Desvio padrão dos coeficientes de aproximação igual a 55).....	77
Figura 5.12b – Região de interesse com influência da pálpebra (Desvio padrão dos coeficientes de aproximação igual a 74).....	77
Figura 5.13a – Imagem de uma íris desfocada (indivíduo 96).....	80
Figura 5.13b – Identificação das 7 imagens com maior similaridade em relação à imagem desfocada (Figura 5.13a) – acerto de 100% (7/7).	80
Figura 5.13c – Imagem de uma íris desfocada (indivíduo 38).	80
Figura 5.13d – Identificação das 7 imagens com maior similaridade em relação à imagem desfocada (Figura 5.13c) – acerto de 85,71% (6/7).	80
Figura 5.14a – Região de interesse 31x224.	81
Figura 5.14b – Região de interesse com histograma equalizado.	81
Figura 5.15 – Decomposição da região de interesse.....	82
Figura 5.16 – Gráfico da Distância Euclidiana de uma íris	85
Figura 6.1 – Modelo das curvas Recall x Precision.	90
Figura 6.2 – Curva Recall x Precision do sistema de recuperação sem a verificação da qualidade de imagem.	91
Figura 6.3 – Curva Recall x Precision do sistema de recuperação com a verificação da qualidade de imagem.	92

LISTA DE TABELAS

Tabela 3.1 – Descrição formal dos algoritmos kAndRange e kOrRange	38
Tabela 3.2 – Símbolos	38
Tabela 5.1 – Dimensão dos vetores referentes aos componentes Wavelet.	83
Tabela 5.2 – Dimensão dos vetores referentes aos componentes Wavelet.	84
Tabela 6.1 – Avaliação do método de localização e extração da íris	88
Tabela 6.2 – Distâncias Euclidianas entre as imagens.....	88
Tabela 6.3 – Tamanho dos vetores referentes aos componentes Wavelet.	89
Tabela 6.4 – Taxa de Reconhecimento Correto.	95

LISTA DE ABREVIATURAS E SIGLAS

1-1	<i>Método de comparação um-para-um</i>
1-N	<i>Método de comparação um-para-muitos</i>
AFIS	<i>Automated Fingerprint Identification System</i>
CBIR	<i>Content-Based Image Retrieval</i>
COM	<i>Component Object Model</i>
CWT	<i>Continuous Wavelet Transform</i>
DWT	<i>Discrete Wavelet Transform</i>
EER	<i>Equal Error Rate</i>
FAR	<i>False Acceptance Rate</i>
FFT	<i>Fast Fourier Transform</i>
FRR	<i>False Rejection Rate</i>
k-NN	k-Nearest Neighbor
MAM	<i>Metric Access Method</i>
NLPR	<i>National Laboratory of Pattern Recognition</i>
QMF	<i>Quadrature Mirros Filters</i>
SAM	<i>Spatial Access Method</i>
SGBD	<i>Sistema de Gerenciamento de Banco de Dados</i>
WFT	<i>Windowed Fourier Transform</i>

SUMÁRIO

RESUMO.....	iii
ABSTRACT.....	iv
CAPÍTULO 1 – INTRODUÇÃO	11
1.1 CONSIDERAÇÕES INICIAIS.....	11
1.2 OBJETIVOS	12
1.3 ESTRUTURA DA DISSERTAÇÃO	13
CAPÍTULO 2 – BIOMETRIA	14
2.1 INTRODUÇÃO	14
2.2 MEDIDAS DE PRECISÃO	17
2.3 BIOMETRIA POR FACE	18
2.4 BIOMETRIA POR IMPRESSÃO DIGITAL	20
2.5 BIOMETRIA PELA RETINA	24
2.6 BIOMETRIA PELA ÍRIS.....	26
2.7 CONSIDERAÇÕES FINAIS	29
CAPÍTULO 3 – RECUPERAÇÃO DE IMAGENS BASEADA EM CONTEÚDO (CBIR).....	30
3.1 INTRODUÇÃO	30
3.2 REPRESENTAÇÃO DE IMAGENS	31
3.2.1 <i>Atributo Forma</i>	31
3.2.2 <i>Atributo Cor</i>	33
3.2.3 <i>Atributo Textura</i>	33
3.3 CONSULTA POR SIMILARIDADE.....	34
3.3.1 CONSULTA POR ABRANGÊNCIA (RANGE QUERY)	35
3.3.2 CONSULTA PELOS K VIZINHOS MAIS PRÓXIMOS (K-NN)	36
3.3.3 CONSULTA PELOS K VIZINHOS MAIS PRÓXIMOS E/OU ABRANGÊNCIA (KANDRANGE E KORRANGE)	37
3.3.4 ESTRUTURAS DE INDEXAÇÃO	38
3.3.4.1 R-TREE	41
3.3.4.2 V-TREE	43
3.3.4.3 QUAD-TREE.....	44
3.3.4.4 SLIM-TREE.....	45
3.4 CONSIDERAÇÕES FINAIS	46
CAPÍTULO 4 – WAVELETS.....	48
4.1 INTRODUÇÃO	48
4.2 HISTÓRICO	51
4.3 TRANSFORMADA DE FOURIER.....	51
4.4 COMPARAÇÃO ENTRE A TRANSFORMADA DE FOURIER E A TRANSFORMADA WAVELET	52
4.5 TRANSFORMADA WAVELET CONTÍNUA	54
4.6 TRANSFORMADA WAVELET DISCRETA	56
4.7 TRANSFORMADA “WAVELET PACKET”	58
4.8 CONSIDERAÇÕES FINAIS	61
CAPÍTULO 5 – METODOLOGIA.....	62

5.1	INTRODUÇÃO.....	62
5.2	MATERIAIS.....	63
5.3	EQUIPAMENTO UTILIZADO.....	63
5.4	SEGMENTAÇÃO DA ÍRIS.....	64
5.5	ALGORITMO GENERALIZADO DE SEGMENTAÇÃO DA ÍRIS	65
5.5.1	<i>Equalização da Imagem.....</i>	65
5.5.2	<i>Localização da pupila.....</i>	66
5.5.3	<i>Aumento do contraste da imagem.....</i>	70
5.5.4	<i>Localização da Borda Externa da Íris.....</i>	71
5.5.5	<i>Extração da Íris</i>	72
5.6	ANÁLISE DAS IRISES EXTRAÍDAS	73
5.6.1	AVALIAÇÃO DA QUALIDADE DAS IMAGENS	75
5.6.2	COMPOSIÇÃO DO VETOR DE CARACTERÍSTICAS	81
5.6.3	RECONHECIMENTO ATRAVÉS DAS CARACTERÍSTICAS EXTRAÍDAS	84
5.7	CONSIDERAÇÕES FINAIS	86
CAPÍTULO 6 – RESULTADOS E CONCLUSÕES		87
6.1	RESULTADOS.....	87
6.1.1	AVALIAÇÃO DA LOCALIZAÇÃO E EXTRAÇÃO DA ÍRIS	87
6.1.2	AVALIAÇÃO DO MÉTODO DE RECUPERAÇÃO DE IMAGENS	88
6.2	CONCLUSÃO.....	92
6.3	SUGESTÃO PARA TRABALHOS FUTUROS	95
REFERÊNCIAS BIBLIOGRÁFICAS.....		97
APÊNDICE A – SOFTWARE DESENVOLVIDO.....		102
APÊNDICE B – TABELA COM A QUALIDADE DAS IMAGENS DO BANCO		104
APÊNDICE C – CÓDIGO FONTE		122

Capítulo 1 – Introdução

1.1 Considerações iniciais

Medidas biométricas têm sido amplamente exploradas em aplicações de identificação e verificação de indivíduos, através da análise das características físicas que os distinguem, tais como voz, geometria da mão, impressão digital, forma de andar, retina e íris. Dentre todas essas características, o reconhecimento pela íris tem mostrado altos índices de acertos, como verificado nos trabalhos de (DAUGMAN, 2001, 2002; MA et al, 2003; WILDES, 1997). Impressão digital e reconhecimento da voz, por exemplo, podem sofrer alterações ao longo da vida do indivíduo, ocasionados por mudanças naturais, acidentes ou patologias, e dessa forma, grande influência negativa teria na eficiência de tais sistemas de reconhecimento. A íris, por outro lado, é uma medida biométrica que não sofre mudanças expressivas durante a vida, garantindo assim maior fidelidade na análise das suas características.

Um sistema completo de identificação pela íris pode ser dividido em três etapas distintas: a aquisição de imagens, detecção da íris viva, que garante que a imagem não é uma foto ou vídeo gravado previamente, e o reconhecimento, que pode ser uma autenticação ou identificação, que se diferem pela finalidade.

Na autenticação, os dados da pessoa que deseja ter acesso em um

determinado local são comparados aos do usuário cadastrado que ela afirma ser. Sistemas desse tipo são conhecidos como *1-1* (um-para-um), pois a medida biométrica que se apresenta é simplesmente checada com o que foi registrado no banco de dados, durante o cadastro dessa pessoa.

Já na identificação, o reconhecimento de uma pessoa ocorre quando se tem o dado biométrico dela e se faz uma busca num banco de dados, comparando as informações até que se encontre (ou não) um registro idêntico ao que é procurado, com certa margem de erro inclusa. Sistemas desse tipo são conhecidos por *1-N* (um-para-muitos), pois o dado de uma pessoa é comparado ao de várias outras. Essa alternativa é usada em sistemas de identificação civil ou criminal, quando não se conhece a identidade da pessoa cujos dados foram coletados.

Um fator chave em sistemas biométricos é a necessidade de serem não-invasivos, ou seja, não ter contato físico ou intervenção humana, e não necessitar de ajustes e manutenções constantes de operadores, garantindo assim um padrão único para todas as imagens capturadas. Nos sistemas atuais que utilizam a íris, as duas exigências fundamentais são de que o usuário deve permanecer parado em uma posição definida e com os olhos abertos, enquanto uma fonte de luz ilumina os olhos e uma câmera captura as imagens. Mesmo apresentando esses aspectos um tanto quanto invasivos, o reconhecimento pela íris mostra-se com maior eficiência em relação às demais características biométricas, conseguindo altas taxas de acertos e baixo tempo de processamento computacional (DAUGMAN, 2001, 2002; MA et al., 2003; WILDES, 1997).

1.2 Objetivos

A proposta deste trabalho é a análise comparativa de algoritmos baseados em *Wavelets* para o reconhecimento de imagens da íris, utilizando o banco CASIA, mantido pelo Laboratório Nacional de Reconhecimento de Padrões (NLPR), Instituto de Automação e Academia Chinesa de Ciências, contendo

756 imagens de íris humanas, de 108 indivíduos.

É proposto também um método de localização da pupila e íris, através da direção do gradiente. Localizadas as bordas internas e externas, é extraída a íris (região de interesse na imagem), para ser analisada através de diversas famílias de transformadas *Wavelets*, visando encontrar aquela que apresente os melhores resultados para reconhecimento.

Para avaliação dos resultados são utilizadas curvas Recall x Precision, técnica esta amplamente empregada em Sistemas de Recuperação de Imagens Baseada em Conteúdo (CBIR).

1.3 Estrutura da Dissertação

Este capítulo introduziu o assunto sobre o reconhecimento de íris.

O capítulo 2 enfoca os diversos tipos de medidas biométricas como meio de identificação ou autenticação de pessoas.

O capítulo 3 descreve sobre os Sistemas de Recuperação Baseados em Conteúdo (CBIR).

O capítulo 4 apresenta uma introdução sobre a Transformada *Wavelet*, bem como algumas de suas famílias e uma comparação com a Transformada de Fourier.

O capítulo 5 descreve a metodologia.

O capítulo 6 discute os resultados obtidos e as conclusões.

Capítulo 2 – Biometria

2.1 Introdução

Atualmente o termo biometria tem sido empregado para referenciar o emergente campo de tecnologia da informação voltado à identificação de indivíduos a partir das suas características biológicas, tais como: impressão digital, geometria das mãos, retina, face e íris, e também características comportamentais como a forma de andar ou porte, assinatura, voz e escrita. A Figura 2.1 representa algumas possibilidades de identificação biométrica.

Os recentes avanços nesta área do conhecimento, bem como a crescente necessidade por segurança em todo mundo, têm contribuído para o desenvolvimento de sistemas inteligentes de identificação pessoal baseados nas características biométricas. Diversas aplicações desses métodos tem sido empregadas na sociedade, dentre elas destacam-se:

- Controle de acesso a lugares restritos;
- Identificação pessoal para empresas;
- Criação de retrato falado;
- Monitoramento de aeroportos, estações e arenas esportivas;

Capítulo 2 – Biometria

- Envelhecimento computadorizado para auxiliar a busca por desaparecidos;
- Sistemas de controles criminais;

Figura 2.1 – Alguns indicadores biométricos. (a) face, (b) impressão digital, (c) geometria da mão, (d) retina, (e) voz, (f) assinatura, (g) íris e (h) forma de andar(porte)

O esquema básico utilizado pelos métodos de reconhecimento biométrico pode ser decomposto sucintamente em três etapas distintas: aquisição de imagens, extração de características e reconhecimento (Figura 2.2).

Figura 2.2 – Esquema generalizado do funcionamento de sistemas biométricos

Na etapa de aquisição, uma seqüência de imagens é capturada para extração de características, onde operações de tratamento são executadas com a finalidade de garantir as imagens de melhor qualidade para o reconhecimento, que dependendo do contexto, pode operar nos modos de *autenticação* e/ou *identificação*.

No modo de autenticação, o indivíduo se apresenta como sendo uma determinada pessoa e o sistema confere a veracidade da informação, retornando verdadeiro ou falso, gerando uma comparação *um-para-um*. Já o modo de identificação lança os dados de determinado indivíduo em um

banco de dados, onde eles são comparados aos de várias outras pessoas até que seja possível determinar a identidade correta, gerando uma comparação *um-para-muitos*. Identificação é voltada para aplicações onde se deseja garantir que múltiplos indivíduos não utilizem a mesma identidade.

2.2 Medidas de Precisão

A precisão dos sistemas biométricos é medida através das *taxas de falsa aceitação* (FAR – *False Acceptance Rate*) e *falsa rejeição* (FRR – *False Rejection Rate*). A FAR representa a percentagem de usuários não-autorizados que são incorretamente identificados como usuários válidos. A FRR representa a percentagem de usuários autorizados que são incorretamente rejeitados. Essas taxas são interdependentes, ou seja, reduzindo-se a taxa de falsa aceitação, a taxa de falsa rejeição aumentará, e reduzir a taxa de falsa rejeição implica no aumento da taxa de falsa aceitação.

O nível de precisão configurado no algoritmo de comparação tem efeito direto nessas taxas. O modo como essas taxas são determinadas é fundamental para a operação de qualquer sistema biométrico e assim deve ser considerado um fator primário na sua avaliação. Os dispositivos biométricos físicos tendem a ter uma menor taxa de falsa aceitação por causa da estabilidade da característica medida e porque as características comportamentais são mais fáceis de serem adulteradas por outros usuários.

A configuração do valor limite para tolerância a estes erros é crítica no desempenho do sistema. A falsa rejeição causa frustração, gerada pela não identificação de um usuário autêntico, ao passo que a falsa aceitação causa fraude, permitindo acesso a um indivíduo impostor.

Muitos sistemas podem ser configurados para fornecer detecção sensível (baixa FAR e alta FRR), onde altos níveis de segurança são requeridos, ou detecção fraca (baixa FRR e alta FAR), que por sua vez são utilizados, por exemplo, em sistemas com um número reduzido de usuários. A medida

crítica é conhecida como taxa de cruzamento (*EER – Equal Error Rate*). Ela é o ponto onde a FAR e a FRR cruzam-se. A figura 2.3 mostra as taxas de falsa aceitação e falsa rejeição.

Figura 2.3 – Taxas de precisão de sistemas biométricos

2.3 Biometria por Face

Duas abordagens distintas são usualmente tratadas em sistemas biométricos por face: *detecção* e *reconhecimento*. Em se tratando de detecção, a função dos algoritmos é conseguir extrair faces humanas de uma imagem estática ou de uma seqüência de imagens. Detecção de faces é consideravelmente difícil porque envolve a localização da face sem nenhum conhecimento prévio de suas escalas, localizações, orientações (ereta, rotacionada) com ou sem oclusões, posicionadas de frente ou perfil (MARIN; BARRETO, 2003).

Três fatores são citados como problemas complexos na dissertação de mestrado de Marroni (MARRONI, 2002). Primeiro: variação de padrões devido a diferenças na aparência facial, discrepâncias entre imagens diferentes de uma mesma face, expressão e cor da pele. Segundo: ocultação de características faciais por objetos como bigode, óculos, cabelo na face e maquiagem. Terceiro: a escala e a orientação da face na imagem, dificultando o uso de *templates* fixos para localizar as características

procuradas.

Um sistema de reconhecimento de face tem como objetivo identificar, que consiste em investigar a face de uma pessoa em um conjunto de faces e encontrar a classe correta para a face investigada. Além disso, com base em uma face investigada, verificar a autenticidade e certificar ou não a informação declarada. Deve também determinar se uma face é de uma classe e em caso positivo, determinar sua identidade, através de características de cada indivíduo.

O reconhecimento de face tem início a partir da captura de imagens estáticas ou imagens de vídeo, seguida da detecção de um rosto que será comparada com modelos armazenados numa base de dados, complementada com a análise da cor da pele, detecção de linhas ou ainda de um modelo híbrido (POH; KORCZAK, 2001). É importante também adaptar o modelo a todas as condições que podem alterar a aparência de um indivíduo, como o uso de óculos, envelhecimento, barba, etc, conforme Figura 2.4.

Figura 2.4 – Sistema de verificação de face

Os métodos de reconhecimento de face podem ser agrupados em diversas categorias sendo elas: por atributos, holísticas, baseadas na transformada de Gabor, tridimensionais, seqüências de vídeo entre outras (CHELLAPPA

et. al., 1995).

As aplicações para os sistemas de reconhecimento de face envolvem: identificação pessoal para banco, passaporte, fichas criminais; sistemas de segurança e controle de acesso; monitoramento de multidões em estações, *shopping centers*, etc; criação de retrato falado; busca em fichas criminais; envelhecimento computadorizado para auxiliar a busca por desaparecidos; interfaces homem-máquina para o reconhecimento de expressões faciais; entre outras diversas aplicações (SCHWERDT; CROWLEY, 2000; GONG et al, 2000; PENTLAND, 2000).

2.4 Biometria por Impressão Digital

A impressão digital é o desenho formado pelas papilas que são compostas por linhas e vales, formando padrões que tornam cada pessoa única, pois, a impressão digital não muda durante a vida. A biometria digital é bastante utilizada, tendo um alto nível de aceitação em registros civis e investigações criminais, tendo como função identificar os padrões (minúcias).

Os AFIS (*Automated Fingerprint Identification System*) sistemas automáticos de identificação de impressões digitais são compostos por vários estágios de processamento, sendo eles: melhoria de imagem, cálculo da imagem direcional, redução de ruídos, segmentação e o afinamento (*thinning*) (JAIN et al. 1997). O objetivo desses sistemas é a comparação de imagens a serem identificadas com as armazenadas em um banco de dados de impressão digitais, conforme mostra a Figura 2.5.

Figura 2.5 – Comparação de minúcias

Durante décadas, vários métodos têm sido desenvolvidos para o propósito de classificação de impressões digitais automaticamente, como os métodos Estruturais, Estatísticos, Sintáticos, por Redes Neurais Artificiais, Híbridos, entre outros.

O desenho da impressão digital apresenta uma área útil onde é possível observar a área padrão, que é a parte principal da impressão do dedo formada por cristas e todos os seus aspectos, e tipos de linhas, que podem ser contínuas ou não, conforme mostra a Figura 2.6.

Figura 2.6 – Área padrão e Tipos de linhas

Os pontos singulares em impressão digital são conhecidos como: núcleo, que é um ponto localizado na área central da impressão digital, e delta, que é um ângulo ou triângulo formado pelas cristas papilares, que podem ser observados na Figura 2.7.

Figura 2.7 – Ponto delta e núcleo de uma impressão digital

A caracterização de aspectos de impressões digitais tem sido apresentada por diversos autores, sendo, Francis Galton (KARU; JAIN, 1996) o primeiro a fazer um estudo dos aspectos de impressões digitais como: cristas finais, cristas bifurcadas, cristas curtas, cruzamentos, esporas e ilhas, chamados de minúcias ou pontos característicos, que são acidentes que se encontram nas cristas papilares como, por exemplo, linhas que terminam abruptamente ou se bifurcam, e tem a finalidade de estabelecer a unicidade das impressões digitais, (Figura 2.8).

Figura 2.8 – Aspectos de impressões digitais (Detalhes de Galton)

Capítulo 2 – Biometria

A impressão digital pode ser dividida em várias classes, sendo a primeira tentativa proposta por Edward Henry (HENRY, 1905) que dividiu em cinco classes, uma delas pertinentes ao delta e outra às linhas do sistema nuclear (núcleo), sendo elas: Arco plano, Arco Angular, Presilha Interna (Direita), Presilha Externa (Esquerda) e Verticilo, conforme ilustrado na Figura 2.9.

Figura 2.9 – As cinco classes propostas por Henry (deltas e núcleos destacados em vermelho)

A verificação da impressão digital pode ser uma boa escolha para sistemas que operam em um ambiente controlado, além de o equipamento apresentar baixo custo, tamanho pequeno, e facilidade de integração de dispositivos de autenticação de impressão digital. Porém, dentre as biometrias físicas é a de menor confiabilidade devido ao fato dos equipamentos utilizados para a captura dos padrões não distinguirem, eficientemente, um dedo vivo de um morto, sendo, fácil produzir uma impressão digital sintética com ou sem a colaboração do proprietário (PUTTE; KEUNING, 2000). Existem atualmente leitores que tentam resolver o problema do “efeito dedo morto” recorrendo a sensores de tensão arterial, condutividade, temperatura e leitura de padrões existentes em camadas inferiores à epiderme, porém, estas tecnologias são

caras e ainda não atingiram o nível de maturidade desejado.

2.5 Biometria pela Retina

A retina é composta de células sensíveis à luz, os cones e os bastonetes. Essas células transformam a energia luminosa das imagens em sinais nervosos que são transmitidos ao cérebro pelo nervo ótico. Normalmente, as imagens dos objetos que olhamos diretamente formam-se na região da retina bem na linha que passa pela pupila e pelo centro do cristalino, isto é, pelo eixo do globo ocular (VESPER, 2005). A figura 2.10 ilustra o olho humano, com seus órgãos internos e veias da retina.

Figura 2.10 – Olho humano, onde é possível verificar a composição das veias na retina.

Fonte: (VESPER, 2005).

Capítulo 2 – Biometria

Os sistemas biométricos baseados na leitura de retina analisam a camada de vasos sanguíneos situada na parte de trás do olho, através da utilização de uma fonte de luz de baixa intensidade para óticamente reconhecer padrões únicos.

A captura das imagens é feita através do posicionamento de um indivíduo no aparelho de leitura (Figura 2.11), devendo este olhar para um determinado ponto fixo, onde uma seqüência de imagens é capturada para extração de características.

Figura 2.11 – Aparelho leitor de imagens da retina
Fonte: (RETICA, 2005).

As vantagens dos sistemas que utilizam a retina como medida biométrica são:

- Estabilidade ao longo da vida de um indivíduo, mantendo o mesmo padrão desde o nascimento até a sua morte, onde em segundos desaparece devido a sua estrutura aquosa;
- Proteção: por se tratar de um órgão interno e que se decompõe caso for extraído, praticamente exclui qualquer possibilidade de fraudes, onde indivíduos pudessem utilizar uma outra retina que não fosse a sua;

- Unicidade: probabilidade de duas retinas apresentarem o mesmo padrão está na ordem de 1 em 10.000.000 de indivíduos. (RETICA, 2005).

Atualmente os fatores que limitam o uso desta tecnologia são: o custo elevado de implantação, a necessidade de o usuário olhar através de um aparelho para um determinado ponto fixo, e o receio por possíveis danos causados pelo contato dos olhos no aparelho.

2.6 Biometria pela Íris

A íris, um órgão interno protegido do olho e de formato anelar, está situada atrás da córnea e do humor aquoso, entre a esclera (parte mais clara do olho) e a pupila (parte mais escura do olho). É a parte responsável pela coloração dos olhos, e ajuda a regular a quantidade de luz que entra no olho. A figura 2.12 mostra a composição do olho humano.

Figura 2.12 – O olho humano

A formação da íris inicia-se no terceiro mês de gestação e as estruturas que criam seu padrão completam-se, na sua maior parte, no oitavo mês, embora o crescimento da pigmentação possa continuar nos primeiros anos após o nascimento (DAUGMAN, 2001). Sua estrutura constitui-se de diversas

Capítulo 2 – Biometria

características de minúcias, tais como: sardas, sulcos de contração, listras, etc. Essas características, geralmente chamadas de textura da íris, são únicas e permanecem praticamente inalteradas por toda existência do indivíduo (BOLES, 1997; DAUGMAN, 2002; WILDES et al, 1996).

Todas essas características intrínsecas da íris produzem grande autenticidade e exclusividade dentre os indivíduos. Cálculos estatísticos revelam que a probabilidade de ocorrência do mesmo padrão de íris em diferentes indivíduos está na ordem de 1 em 10^{78} (GONZAGA, 2003). A anatomia da íris bem como suas características propícias para biometria são vistas na figura 2.13.

Figura 2.13 – Anatomia da íris humana. A estrutura da íris vista de forma frontal. A aparência visual da íris humana deriva desta estrutura anatômica

Um completo sistema de identificação pessoal baseado na análise da íris humana é constituído em 3 etapas distintas, como observado na figura 2.14.

Figura 2.14 – Diagrama esquemático do reconhecimento biométrico pela íris humana

De forma resumida, na primeira etapa de aquisição de imagens, uma seqüência é capturada para garantir um número suficiente de imagens, já que é comum ocorrer à movimentação da cabeça e o piscar dos olhos por parte do usuário. É necessário também um processamento dessas imagens, visando melhorar a sua qualidade e realçar as características de textura da íris. No processo de localização, os objetivos compreendem a detecção da íris viva, para que não ocorra que a imagem de entrada seja uma foto ou mesmo uma lente de contato, usada por um indivíduo impostor; a extração da íris, que tem a finalidade de obter somente a região da íris, descartando a pupila e esclera; e a Extração de características, onde o padrão da íris é codificado através de um vetor, contendo as informações processadas apenas desta região de interesse. A última etapa é o reconhecimento, onde este vetor de características é comparado com os demais padrões do banco de dados.

A variável de decisão, aqui denominada pela letra *D* representa as duas maneiras de reconhecimento: autenticação e identificação.

2.7 Considerações Finais

Neste capítulo foi abordado o assunto biometria, descrevendo algumas características físicas utilizadas para reconhecimento pessoal. Com o aumento da criminalidade global, a biometria está cada vez mais presente na sociedade, apresentando diversas vantagens em relação aos métodos tradicionais de autenticação, como cartões e senhas, pelo fato de analisar as características físicas inerentes a cada indivíduo.

A íris humana é uma das medidas de maior confiabilidade para biometria, tendo como vantagens a unicidade pessoal e dificuldade para fraudes, devido a sua composição fisiológica.

Capítulo 3 – Recuperação de Imagens Baseada em Conteúdo (CBIR)

3.1 Introdução

O termo CBIR (*Content-Based Image Retrieval*) refere-se aos sistemas de recuperação de imagens baseados em características visuais tais como: cor, forma e textura. O seu esquema geral de funcionamento está ilustrado na figura 3.1, onde, considerando-se uma imagem de entrada, o sistema CBIR irá pesquisar no banco de dados as n imagens de maior similaridade à imagem de consulta, de acordo com um critério fornecido previamente (CASTAÑON, 2003).

Figura 3.1 – Esquema geral do funcionamento de um sistema CBIR

3.2 Representação de Imagens

Alguns métodos como associação das imagens com uma determinada palavra chave, número, descrição por categorias dentre outros, não podem ser considerados CBIR, já que se mostram ineficientes na análise de grandes bancos de dados, consumindo demasiado tempo computacional. (AL-TAYECHE; KHALIL, 2003)

Para caracterização e representação de imagens, os sistemas de recuperação fazem uso de alguns atributos visuais, como: forma, cor e textura. Dependendo do tipo da aplicação e do tamanho do banco de dados, apenas um destes atributos pode ser suficiente para representá-las. Pode ocorrer também que em sistemas com grande quantidade e não uniformidade de imagens, apenas um atributo não seja suficiente para discriminar satisfatoriamente as suas características, bem como alcançar níveis adequados de recuperação, motivo pelo qual alguns sistemas procuram utilizar múltiplos atributos para a sua indexação e recuperação (CASTAÑON, 2003).

3.2.1 Atributo Forma

O atributo forma para sistemas de recuperação por conteúdo é uma abordagem tão importante quanto difícil, fundamentalmente pelo fato de ter que segmentar e conhecer o tamanho e orientação dos objetos contidos na imagem (CASTAÑON, 2003). Outro aspecto que torna a análise de formas uma tarefa computacionalmente difícil é a existência de ruídos, oclusões e distorções, introduzidos no processo de aquisição de imagens, significando a descaracterização e até mesmo a transformação de uma determinada forma em outra. Theodoridis (THEODORIDIS; KOUTROUMBAS, 2003) cita como exemplo o tamanho dos nódulos em aplicações médicas, como forma de classificação entre malignos e benignos. Os nódulos que apresentam bordas irregulares têm uma probabilidade maior de serem malignos, e aqueles com bordas mais regulares são geralmente benignos.

A figura 3.2 mostra alguns exemplos de formas em imagens.

Figura 3.2 – Formas em imagens. a) Artificiais e b) Naturais (CASTAÑON, 2003).

A representação de formas em imagens pode ser dividida em duas categorias: baseada na borda (*contour-based*) e baseada na região (*region-based*) (THEODORIDIS; KOUTROUMBAS, 2003). A caracterização baseada na borda utiliza apenas o limite externo da forma, considerando, por exemplo, somente os *pixels* que determinam o seu contorno (características morfológicas). Já a representação baseada na região utiliza a região completa da forma, como por exemplo, todos os *pixels* que estão contidos na forma determinada. A figura 3.3 ilustra estas categorias.

Figura 3.3 – Categorias de representação de formas em imagens. (a) baseada em bordas. (b) baseada em regiões. (AL-TAYECHE; KHALIL, 2003)

3.2.2 Atributo Cor

No reconhecimento automático de imagens, a cor é um poderoso descritor de propriedades de um objeto, exercendo significativa importância na segmentação e recuperação de imagens em um banco de dados. As representações mais comuns de cores são RGB (*red, green, blue*), que é baseado num sistema de coordenadas cartesianas, CMY (*cyan, magenta, yellow*), utilizado pelas impressoras e copiadoras coloridas, e HSI (*hue, saturation, intensity*), que é o modelo semelhante à forma com que o ser humano percebe as cores, uma vez que permite separar as componentes de matiz, saturação e intensidade da informação de cor em uma imagem (FILHO; NETO, 1999). Exemplos da utilidade do modelo HSI vão desde o projeto de sistemas de imageamento para determinação automática do amadurecimento de frutas e vegetais, a sistemas para correspondência de amostras de cores, ou para inspeção de qualidade de produtos coloridos (GONZALES; WOODS, 2000).

Em sistemas CBIR o atributo “cor” pode ser representado a partir de histogramas. Um histograma é uma função de distribuição de densidade que indica o percentual (ou o número) de *pixels* que apresenta uma determinada intensidade de cor. Em imagens coloridas, calcula-se o histograma correspondente a cada um de seus componentes. Para uma imagem do tipo RGB, por exemplo, são calculados três histogramas, um para cada componente (*R, G e B*).

Liapis e Tziritas (2004) utilizam um histograma de coordenadas 2-D no espaço de cores *CIE Lab*, para caracterizar o conteúdo de cromaticidade contido na imagem, relatando que para recuperação e classificação de imagens coloridas o fator de maior relevância é a distribuição de cromaticidade, excluindo os componentes de iluminação, que são utilizados para extrair características de textura.

3.2.3 Atributo Textura

Embora não exista nenhuma definição formal de textura, esse descritor

fornece intuitivamente medidas de propriedades como suavidade, rugosidade e regularidade (GONZALES; WOODS, 2000).

Pode-se também estabelecer textura em imagem como sendo a composição de determinados padrões que se repetem ao longo da superfície de um objeto. Tais padrões envolvem freqüentemente as mudanças de orientação, cor e escala, sendo que os sistemas de recuperação baseados em conteúdo devem ser capazes de classificar texturas de maneira independente de tais variações. Na figura 3.4 pode-se visualizar alguns exemplos de texturas.

Figura 3.4 – Imagens com textura

A análise de textura de imagens digitais 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 homogeneidade ou regularidade (CASTAÑON, 2003).

Um dos métodos mais utilizados na literatura para representar texturas em imagens é o processamento de sinais através de *Wavelets* (MANJUNATH; MA, 1996), onde esses padrões de textura podem ser decompostos e analisados separadamente em cada resolução.

3.3 Consulta por Similaridade

Em um passado recente, a necessidade de armazenamento e recuperação de informações em bancos de dados limitava-se a tipos simples de dados,

tais como: caracteres, números inteiros, lógicos, ponto flutuante, datas, etc. Até então, consultas de dados de maior complexidade, de características próprias em sua estrutura, como por exemplo: imagens, som, vídeos, DNA, dentre outros, não poderiam ser realizadas na maioria dos sistemas gerenciadores de bancos de dados existentes, já que estes não proviam tais recursos.

Atualmente, existe a necessidade de manipulação desses tipos de dados complexos, que devem ser armazenados e recuperados através de elementos que definam a sua característica. Com essa finalidade é que sistemas de recuperação baseados em conteúdo utilizam consultas por similaridades.

O conceito de consulta por similaridade considera o quanto dois dados são semelhantes entre si. Para medir esta semelhança utiliza-se de uma função de distância ($d()$) adequada ao domínio dos dados. Essa função recebe dois objetos (O_i e O_j) pertencentes ao mesmo domínio (D) e retorna o grau de dissimilaridade destes objetos (LOPES, 2005). Caso O_i e O_j forem idênticos, a função $d()$ retorna zero; se eles forem diferentes, o valor será um inteiro positivo relacionado à distância entre O_i e O_j , ou seja, quanto maior for a distância entre os objetos, maior será o valor retornado pela função de distância $d()$.

Existem basicamente dois operadores mais comuns para consultas sobre domínio de dados métricos, os quais podem ser chamados de operadores básicos, são eles: consulta por abrangência (*range query*) e consulta pelos k vizinhos mais próximos (*k-nearest-neighbors query*).

3.3.1 Consulta por Abrangência (Range Query)

Dados um conjunto de objetos $O = \{O_1, O_2, \dots, O_n\}$ com $O_1, O_2, \dots, O_n \in D$, uma função de distância métrica $d()$, um objeto de consulta $Q \in D$ e uma abrangência máxima de busca $r(Q)$, a consulta por abrangência é definida por pela equação 3.1 (LOPES, 2005):

$$range(Q, r(Q)) = \{O_i | O_i \in O \text{ e } d(O_i, Q) \leq r(Q)\} \quad (3.1)$$

Essa definição pode ser compreendida como a seleção de todos os objetos O_i do conjunto O , tal que a distância de O_i para Q seja menor ou igual à abrangência da consulta ($r(Q)$). Exemplo: “Encontre as cidades que estão em um raio de até 100 km de distância da Capital”. Nessa consulta, o objeto foco é a Capital, o domínio D é o conjunto de cidades, e a abrangência ou raio de busca (distância máxima) é 100 km. A distância utilizada é a medida em km que separa as cidades. A figura 3.5 ilustra este exemplo.

Figura 3.5 – Consulta por Abrangência (*Range Query*)

3.3.2 Consulta pelos k Vizinhos mais Próximos (k -NN)

Dados um conjunto de objetos $O = \{O_1, O_2, \dots, O_n\}$ com $O_1, O_2, \dots, O_n \in D$, uma função de distância métrica $d()$, um objeto de consulta $Q \in D$ e um inteiro $k \geq 1$, a consulta pelos k vizinhos mais próximos é definida pela equação 3.2 (LOPES, 2005):

$$k-NN(Q) = \{A_i | A_i \in A, A \subseteq O, |A| = k \text{ e } \forall A_i \in A, O_i \in O - A, d(Q, A_i) \leq d(Q, O_i)\} \quad (3.2)$$

De acordo com essa definição, o resultado será a seleção dos k objetos do

conjunto O que estão mais próximos de Q . Utilizando o exemplo, a consulta “Selecione as 5 cidades mais próximas da capital”, é uma busca K -NN para o objeto “Capital” sobre o mesmo domínio de objetos da consulta-exemplo anterior. A figura 3.6 ilustra este exemplo.

Figura 3.6 – Consulta pelos k vizinhos mais próximos (k Nearest-Neighbors Query)

3.3.3 Consulta pelos k Vizinhos mais Próximos e/ou Abrangência ($kAndRange$ e $kOrRange$)

Com o propósito de integrar em uma mesma consulta, os algoritmos baseados nos k -vizinhos mais próximos e abrangência, surgiram dois novos métodos de consulta por similaridade: $kAndRange$ e $kOrRange$, propostos por Arantes (ARANTES et al., 2003). Nesses novos algoritmos, existe a possibilidade de gerar consultas que envolvam conjunções/disjunções de critérios de seleção por similaridade, muito freqüentes em aplicações reais, e que, nos algoritmos $RangeQuery$ e k -NN somente poderiam ser realizadas de maneira isolada, havendo a necessidade de execução de operadores de união (“ou”) e intersecção (“e”) desses resultados intermediários. A tabela 3.1 mostra a descrição formal dos algoritmos $kAndRange$ e $kOrRange$. Os

símbolos utilizados estão representados na tabela 3.2.

Tabela 3.1 – Descrição formal dos algoritmos *kAndRange* e *kOrRange*

	Operador Composto	Composição do conjunto-resposta dos algoritmos básicos
Conjunção	$kAndRange: \sigma_{(RQ(o_q, r_q) \wedge kNNQ(o_q, k))}(\mathcal{D})$	$\sigma_{RQ(o_q, r_q)}(\mathcal{D}) \cap \sigma_{kNNQ(o_q, k)}(\mathcal{D})$
Disjunção	$kOrRange: \sigma_{(RQ(o_q, r_q) \vee kNNQ(o_q, k))}(\mathcal{D})$	$\sigma_{RQ(o_q, r_q)}(\mathcal{D}) \cup \sigma_{kNNQ(o_q, k)}(\mathcal{D})$

Tabela 3.2 – Símbolos

σ	Operador de seleção da álgebra relacional
RQ	Range Query
O_q	Objeto de consulta (centro da consulta)
r_q	Raio da busca por abrangência
$kNNQ$	k Vizinhos mais próximos
k	Número de vizinhos em uma consulta $kNNQ$
$\mathcal{D}$	Domínio dos objetos

Os algoritmos completos dos métodos *kOrRange* e *kAndRange*, bem como os resultados obtidos, podem ser observados integralmente no trabalho de Arantes e outros (2003).

3.3.4 Estruturas de Indexação

O processo de organização de informação efetuado pelos sistemas de gerenciamento de bancos de dados (SGBD) utiliza como base as estruturas de indexação ou métodos de acesso para acelerar a busca aos dados (BUENO, 2001). Logo, as estruturas de indexação são ferramentas fundamentais que habilitam os sistemas de gerenciamento de bancos de dados a eficientemente armazenar e recuperar informações de um grande volume de dados. Entretanto, os SGBDs comerciais não possibilitam a recuperação de informações baseados em similaridades, como por exemplo,

em dados complexos como imagens, utilizados nos sistemas CBIR. Para tal finalidade, várias estruturas de indexação para dados complexos já foram propostas, entre elas, destacam-se basicamente dois tipos (LOPES, 2005):

- Estruturas Espaciais;
- Estruturas Métricas.

As Estruturas Espaciais foram as primeiras estruturas de indexação propostas para suportar dados espaciais em espaços de poucas dimensões. Essas estruturas são conhecidas como Métodos de Acessos Espaciais – SAM (*Spatial Access Methods*) (BUENO, 2001), que são técnicas para indexar e recuperar informações em um conjunto de dados vetoriais. Os SAM tiveram início através do surgimento da estrutura R-Tree, mas não se mostraram adequadas com dados de alta dimensionalidade e adimensionais, sendo sucedidas pelas estruturas métricas (LOPES, 2005).

As estruturas métricas, ou os Métodos de Acesso Métricos – MAM (*Metric Access Methods*) suportam naturalmente consultas por proximidade ou similaridade, além de se mostrarem eficientes para dados de alta dimensão (BUENO, 2001). Dessa forma, utilizar MAM para indexar imagens, ou mais propriamente, as características que foram extraídas das imagens suportando consultas por similaridade têm-se mostrado bastante apropriado (BUENO, 2001).

Um espaço métrico é definido pelo par $M = (D, d())$, onde D é o domínio dos conjuntos de características dos objetos (utilizando como chave na construção da estrutura de indexação) e d é uma função de distância métrica adequada ao domínio D (LOPES, 2005).

Para que $d()$ seja uma função métrica, ela deve possuir as seguintes características:

- Simetria:

$$\forall O_i, O_j \in D \Rightarrow d(O_i, O_j) = d(O_j, O_i);$$

- Não negatividade:

$$\forall O_i, O_j \in D \mid O_i \neq O_j \Rightarrow 0 < d(O_i, O_j) < \infty \text{ e } d(O_i, O_j) = 0;$$

- Desigualdade triangular:

$$\forall O_i, O_j, O_k \in D \Rightarrow d(O_i, O_j) \leq d(O_i, O_k) + d(O_k, O_j).$$

De acordo com Arantes e outros (2003), dessas características, a mais importante em termos de otimização da busca por similaridade, é a desigualdade triangular. É esta propriedade que torna possível o descarte (poda) de ramos da árvore. Ela possibilita reduzir a quantidade de cálculos de distância necessários para responder a uma consulta, proporcionando um melhor desempenho nas operações de seleção, uma vez que evita a necessidade de se calcular todas as distâncias entre cada objeto armazenado e o objeto de busca O_q .

Para descartar elementos utilizando a desigualdade triangular, pode-se considerar o seguinte cenário: dados um espaço métrico $M = (S, d)$, o conjunto $S \subseteq \mathbb{S}$, o objeto de busca $O_q \in \mathbb{S}$, o raio de busca r_q e um objeto representativo O_{rep} . Um objeto $s \in S$ poderá ser descartado se, e somente se, as equações 3.3 e 3.4 forem satisfeitas (VARGAS, 2004).

Na Figura 3.7 observa-se que os objetos compreendidos na região escura são aqueles que, provavelmente, fazem parte do conjunto de resposta.

$$d(O_{rep}, s) < d(O_{rep}, O_q) + r_q \quad (3.3)$$

$$d(O_{rep}, s) > d(O_{rep}, O_q) - r_q \quad (3.4)$$

Figura 3.7 – Propriedades da desigualdade triangular (VARGAS, 2004).

3.3.4.1 R-Tree

A estrutura R-Tree (Guttman, 1984) foi a primeira proposta para dados não pontuais existente na literatura. A estrutura R-Tree é amplamente difundida nesta área e isso pode ser comprovado porque, quase sempre, é a base de comparação para novas estruturas deste tipo (VARGAS, 2004).

A R-tree é uma extensão da B-tree para o espaço multidimensional. Ela representa o espaço de dimensão n como uma hierarquia de intervalos n -dimensionais (retângulos no caso bidimensional). Assim como a B-tree, ela é uma estrutura balanceada, em que todos os objetos estão armazenados nas folhas da árvore. Toda R-tree possui uma ordem (m, M) , indicando que seus nós possuem entre $m \leq \lfloor m/2 \rfloor$ e M registros (FREDERICK, 1999).

Todo registro da estrutura é formado pelo par (r, p) , onde r é um intervalo (retângulo) e p um ponteiro. Nos nós externos (folhas), r é o retângulo envolvente do objeto apontado por p , e enquanto que, nos nós internos, r é o retângulo que envolve todos os retângulos armazenados na sub-árvore apontada por p (FREDERICK, 1999).

A Figura 3.8 mostra o exemplo de uma R-tree para um determinado conjunto de dados.

Figura 3.8 – Exemplo de uma R-Tree (FREDERICK, 1999).

(a) Representação planar. (b) Estrutura.

Um estudo mais completo sobre as árvores R-Tree e suas diversas variações, tais como R^+ -Tree, R^* -Tree, X-Tree e SR-Tree, podem ser encontrado no trabalho de Vargas (2004).

3.3.4.2 V-Tree

As V-trees são estruturas de dados hierárquicas projetadas para armazenar linhas poligonais longas, que são representadas por uma sequência de pontos. Sua principal funcionalidade consiste em recuperar eficientemente aproximações de linhas poligonais armazenadas em memória secundária (FREDERICK, 1999). Além disso, elas otimizam a recuperação de fragmentos da linha poligonal que interceptam uma dada região retangular (*range query* ou *clipping*) e a operação de ponto em polígono. Esta testa se um dado ponto está dentro de uma linha poligonal fechada que não se cruza.

A figura 3.9 ilustra a representação e estrutura de uma árvore V-Tree.

Figura 3.9 – Exemplo de uma V-Tree (FREDERICK, 1999).

(a) Representação planar. (b) Estrutura.

3.3.4.3 Quad-Tree

A Quad-Tree é um tipo de estrutura de dados organizada em árvore, em que cada nó ou tronco gera sempre, e exatamente, quatro folhas. A idéia fundamental da Quad-Tree é que qualquer imagem pode ser dividida em quatro quadrantes, onde cada quadrante pode ser novamente dividido em quatro subquadrantes e assim sucessivamente.

A figura 3.10 ilustra um exemplo de uma Quad-Tree.

Figura 3.10 – Exemplo de uma Quad-Tree (FREDERICK, 1999).

(a) Representação planar. (b) Estrutura.

Em uma Quad-Tree cada nó é associado a um quadrante, onde os nós internos são formados por um valor e quatro ponteiros para os seus nós

filhos. Cada um desses nós filhos corresponde a um quadrante da região associada ao nó, sendo que a raiz da árvore representa todo o objeto. Cada quadrante possui um valor armazenado correspondente a uma versão simplificada da parte do objeto, e as folhas da árvore representam os quadrantes que não são mais divididos, armazenando o valor real do bloco (FREDERICK, 1999).

Com base nessas características, as estruturas Quad-Tree se mostram úteis para o armazenamento e visualização de imagens grandes, pois elas permitem a recuperação de versões simplificadas da imagem bem como a recuperação somente do bloco visível da imagem.

3.3.4.4 *Slim-tree*

A *Slim-tree* é uma estrutura balanceada e dinâmica, que tem crescimento das folhas para a raiz, e permite inserções de dados de um domínio métrico. Os objetos são armazenados nas folhas, organizados numa estrutura hierárquica que utiliza um objeto representativo como centro de uma região de cobertura dos objetos em uma subárvore, delimitada por um raio. A *Slim-tree*, assim como a maioria dos MAMs, utiliza a propriedade de desigualdade triangular para “podar” cálculos de distâncias em buscas por similaridade em domínios métricos. (BUENO; TRAINA; TRAINA JR, 2005).

A Figura 3.11 ilustra a disposição de 17 objetos em uma *Slim-tree* e sua estrutura lógica.

Figura 3.11 – Exemplo de uma Slim-Tree (BUENO; TRAINA; TRAINA JR, 2005). (a) Representação planar. (b) Estrutura.

Para fazer a medição de uma Slim-Tree foram propostos o *fat-factor* e o *bloat-factor*, que representam o grau absoluto e relativo (à árvore ótima) de sobreposição de uma árvore, respectivamente. (YAMAMOTO; BIAJIZ, 2002).

Para fazer a diminuição da sobreposição, foi proposto o Slim-down cuja idéia é fazer a reinserção de objetos de forma conveniente a diminuir o raio de cobertura dos nós. A Slim-tree também propôs o algoritmo de inserção por mínima ocupação e o algoritmo MST (*Minimal Spanning Tree*) de divisão de nó e escolha dos representativos. (YAMAMOTO; BIAJIZ, 2002).

3.4 Considerações Finais

Neste capítulo foi abordado o assunto referente à Recuperação de Imagens Baseado em Conteúdo (CBIR).

O funcionamento básico de tais sistemas pode ser resumido como sendo uma pesquisa em uma determinada base de dados, que retorne n imagens

de maior similaridade com uma imagem de consulta, baseadas em características visuais tais como: cor, forma e textura. Essas consultas são possíveis através de algumas estruturas de indexação, dentre as quais se destacam as Estruturas Espaciais (SAM) e Estruturas Métricas (MAM). Diversas árvores de indexação dão suporte às estruturas métricas, dentre as quais, as descritas neste capítulo: R-Tree e suas variações, V-Tree, Quad-Tree e Slim-Tree.

Capítulo 4 – Wavelets

4.1 Introdução

Wavelets são funções que satisfazem requisitos matemáticos utilizados na representação de dados ou outras funções. A idéia de utilizar funções para representar outras funções não é nova, e existe desde que Fourier descobriu que senos e cossenos podem ser utilizados para representar outras funções. Porém, na análise de *wavelet*, a escala possui um papel importante no processamento dos dados. A *wavelet* pode ser processada utilizando diferentes escalas ou resoluções. Inicialmente, a transformada *wavelet* é considerada como um sinal contínuo $f(t)$, e dada pela equação 4.1, onde o parâmetro a representa a escala (resolução) e o parâmetro b representa o deslocamento (translação). A função $\psi_{a,b}(t)$ chamada “wavelet”, é derivada de uma função $\psi(t)$ através da transformação apresentada na equação 4.2 (GALVÃO et al, 2001).

$$F(a, b) = \int f(t) \psi_{a,b}(t) dt \quad (4.1)$$

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad (4.2)$$

A análise de *wavelet* permite a utilização de intervalos longos de tempo para encontrar as informações de baixa frequência, e intervalos curtos de tempo onde estão localizadas as informações de alta frequência (MISITI et al, 2005).

Existem diversas possibilidades de escolha para a função $\psi(t)$, denominada “wavelet-mãe”. Algumas alternativas são apresentadas na figura 4.1.

Figura 4.1 – Exemplos de *wavelets*.

No processo de decomposição dos sinais, dois coeficientes são gerados: aproximação (A), que são os componentes de baixa-frequência (*Low-Pass*), e detalhe (D), que corresponde aos componentes de alta-frequência (*High-Pass*), conforme ilustrado na figura 4.2.

Figura 4.2 – Processo de decomposição das *wavelets*.

Para decomposição em dois ou mais níveis, o processo adota o último coeficiente de aproximação (A) para gerar outros dois coeficientes de aproximação e detalhe, e assim, sucessivamente até que o último nível seja decomposto, como observado na figura 4.3.

Figura 4.3 – Árvore de três níveis de decomposição *wavelet*

4.2 Histórico

Embora desde o início do século XIX a literatura registre trabalhos matemáticos semelhantes à Transformada *Wavelet* com Alfred Haar em 1910 (agora conhecida como base de Haar), foi no final da década de 70 que a mesma passou a sua própria identidade (GALVÃO et al, 2001). Foi nesse período que o termo “*wavelets*” foi introduzido por Jean Morlet, propondo uma modificação da Transformada de Fourier para melhor tratar sinais geofísicos. Os dados sísmicos estudados por Morlet exibiam conteúdos de frequência que mudavam rapidamente ao longo do tempo, além disso, mostraram que qualquer tipo de sinal pode ser analisado em termos de escalas e translações de uma simples função *wavelet mãe* (CASTAÑON, 2003). O matemático Yves Meyer colaborou no amadurecimento desta nova idéia, e Stephane Mallat, em seus estudos de processamento de imagens desenvolveu um algoritmo para calcular de forma eficiente a Transformada *Wavelet*, teoria esta denominada *análise de multiresolução*.

4.3 Transformada de Fourier

Análise de Fourier consiste em funções de sobreposição de senos e cosenos, transformando um sinal (função) do domínio do espaço para o domínio da frequência (GONZALES; WOODS, 2000). A função responsável pela transformação é definida pela equação 4.3.

$$F(u) = \int f(x) e^{-j2\pi ux} dx \quad (4.3)$$

A equação 4.3 representa a transformada de Fourier de um sinal contínuo $f(x)$, a qual apresenta um sério problema para análise de sinais que mudam durante o tempo, já que na transformação para o domínio da frequência, a informação do tempo é perdida. Para sinais estacionários, ou seja, aqueles que não mudam durante o tempo esse problema é indiferente, entretanto, a maioria dos sinais interessantes contém muitas características não

estacionárias ou transitórias (fluxos, tendências, mudanças abruptas, início e final de um evento). Como estas características são freqüentemente as partes mais importantes do sinal, a análise de Fourier torna-se inadequada para esta detecção (MISITI et al, 2005).

Visando suprir a deficiência da análise de Fourier em observar os sinais que variam no tempo, Dennis Gabor (1946) adaptou esta transformada para analisar apenas uma pequena porção do sinal em um tempo, introduzindo a chamada transformada por janelas de Fourier (*Windowed Fourier Transform* – *WFT*) (CASTAÑON, 2003).

A transformada por janelas de Fourier pode ser usada para obter informação sobre sinais no domínio do tempo e no domínio da frequência simultaneamente (GRAPS, 1995), onde um sinal de entrada $f(t)$ é observado através do deslocamento da janela no domínio do tempo, e a Transformada de Fourier calculada para cada posição da janela.

4.4 Comparação entre a Transformada de Fourier e a Transformada Wavelet

No trabalho de Amara Graps (GRAPS, 1995) encontra-se uma comparação entre a Transformada de Fourier e a Transformada *Wavelet*, sendo as principais similaridades e diferenças descritas a seguir:

i. Similaridades

- Ambas são operações lineares que geram uma estrutura de dados que contém $\log_2 n$ segmentos de vários tamanhos; usualmente os preenchem e os transformam em um vetor de dados diferente de tamanho 2^n ;
- As propriedades matemáticas das matrizes envolvidas nas transformadas também são similares. A inversa da matriz da transformada tanto para FFT (*Fast Fourier Transform*) quanto para a

DWT (*Discrete Wavelet Transform*) é a transposta da original. Como resultado, ambas podem ser vistas como uma rotação no espaço da função para um domínio diferente. Para a FFT, este novo domínio contém funções base que são senos e cosenos. Para a transformada *wavelet*, este novo domínio contém funções base mais complicadas, chamadas *wavelets*, ou *wavelets-mãe*.

ii. Diferenças

- A mais importante diferença entre estes dois tipos de transformadas é que funções individuais *wavelets* estão localizadas no espaço. As funções seno e coseno usadas na FFT não estão.
- O tamanho da janela varia na transformada *wavelet*, sendo uma vantagem em relação à transformada de Fourier, que utiliza apenas janelas de observação fixas.
- A transformada *wavelet* não possui um único conjunto de funções base, como acontece com a transformada de Fourier, que utiliza apenas as funções seno e coseno. A transformada *wavelet* possui um conjunto infinito de funções base, assim, a análise *wavelet* fornece acesso imediato à informação que pode estar escondida em outros métodos tempo-freqüência, como a análise de Fourier (SILVA; ENYNG, 2000).

A figura 4.4 apresenta de uma forma generalizada janelas de análise no plano tempo-freqüência para as Transformadas de Fourier e *Wavelet*.

Figura 4.4 – (a) Janelas de análise no plano tempo-freqüência para a WFT(Transformada de Fourier por Janelas) e para as *wavelets* – (b) Cobertura do espectro de freqüências pela Transformada por Janelas de Fourier (acima) e pela Transformada Wavelet.

Fonte: (FARIA, 1997)

4.5 Transformada Wavelet Contínua

A Transformada *Wavelet* Contínua (CWT – *Continuous Wavelet Transform*) é definida a partir de um espaço de funções ortonormais, denominado baixas *Wavelets*, as quais formam uma base de funções da Transformada *Wavelet*. Assim, a CWT é definida como o produto interno na forma dada pela equação 4.4.

$$CWT_{f(a,b)} = \langle f(t), \psi_{a,b}(t) \rangle = \int f(t) \cdot \psi_{a,b}(t) \cdot dt \quad (4.4)$$

onde $\psi_{a,b}(t)$ é conhecida como “*Wavelet-Mãe*” ou “*Wavelet-Básica*”, e os parâmetros a e b estão apresentados pela transformação da equação 4.2.

A CWT constitui-se da soma de todos os sinais $f(t)$ no tempo, multiplicados por versões de escala e deslocamento da função *wavelet* ψ (MISITI et al, 2005), conforme a equação 4.5, tendo como resultados vários *coeficientes wavelet* C , que são uma função de escala e posição.

$$C(escala, posição) = \int f(t) \psi(escala, posição, t) dt \quad (4.5)$$

O fator escala em *wavelet* representa simplesmente a forma de expandir ou comprimir um sinal, e o fator posição pode ser entendido como o deslocamento do sinal, através do seu retardo ou avanço do seu ponto inicial. O escalonamento em *Wavelets* é ilustrado na figura 4.5.

(a)

(b)

Figura 4.5 – (a) 3 exemplos de escala de sinal em *wavelet*. (b) posicionamento(deslocamento) de sinal em *wavelet* (MISITI et al, 2005).

4.6 Transformada Wavelet Discreta

Translações e dilatações da “Função-Mãe” ou “Análise Wavelet” $\Phi(x)$, definem uma base ortogonal, denominada bases *wavelet* (GRAPS, 1995), conforme mostra a equação 4.6, onde as variáveis s e l são inteiras, e representam respectivamente a escala e localização da função mãe $\Phi(x)$ para gerar as *wavelets*, como por exemplo, a família de *wavelets* Daubechies. O índice de escala s indica o tamanho da *wavelet*, e o índice de localização l a sua posição (GRAPS, 1995).

$$\Phi_{(s,l)}(x) = 2^{-\frac{s}{2}} \Phi(s^{-s}x - l) \quad (4.6)$$

Para estender o domínio de dados em diferentes resoluções, a análise *wavelet* é usada em uma equação escalar, dada pela equação 4.7, onde $W(x)$ é a função escala da função-mãe Φ , e c_k são os *coeficientes wavelets*, que devem satisfazer as condições linear e quadrática na forma (GRAPS, 1995), dada pela equação 4.8, onde δ é a função delta e l é o índice de localização.

$$W(x) = \sum_{k=-1}^{N-2} (-1)^k c_{k+1} \Phi(2x + k) \quad (4.7)$$

$$\sum_{k=0}^{N-1} c_k = 2, \quad \sum_{k=0}^{N-1} c_k c_{k+2l} = 2\delta_{l,0} \quad (4.8)$$

A DWT é implementada por uma seqüência de filtragens digitais sobre o sinal original, de acordo com um algoritmo conhecido como codificação sub-banda, onde um par de filtros digitais, definidos como filtros em quadratura ou QMF (*Quadrature Mirros Filters*), é representado pelas funções $h(n)$ e $g(n)$, que são, respectivamente, as funções resposta ao impulso de um filtro Passa-Baixa (*Low-Pass*) e Passa-Alta (*High-Pass*), cada um com metade da banda do sinal original.

A figura 4.6 apresenta a execução da Transformada *Wavelet* Discreta em

um sinal unidimensional, representado por uma senóide com ruído de alta frequência adicionado a ele (MISITI et al, 2005).

Figura 4.6 – Análise de um sinal 1-D através da DWT
Fonte: (MISITI et al, 2005)

Na análise de imagens, onde os sinais são tratados em 2-D, a DWT produz uma matriz de coeficientes, conhecidos como coeficientes *wavelets*. Aplicando a DWT em uma imagem, o resultado são quatro tipos de coeficientes: aproximação, detalhes horizontais, detalhes verticais e detalhes diagonais. A figura 4.7 mostra o esquema e imagens geradas pela decomposição em 3 níveis da *wavelet* Daubechies-4. Neste exemplo, cada nível de decomposição fornece uma subimagem com os coeficientes de aproximação (passa-baixa), que é uma versão de baixa resolução da imagem original, e também três subimagens que correspondem ao coeficiente de detalhe (passa-alta), contendo respectivamente os detalhes horizontais, diagonais e verticais. Na decomposição dos subníveis da transformada, apenas os coeficientes de aproximação são novamente divididos em coeficientes de aproximação e detalhes, chamado de segundo nível de decomposição. E assim pode ser feito n vezes, onde $2n$ é o tamanho do sinal amostrado.

Figura 4.7 – (a) Esquema de organização dos coeficientes Wavelets. (b) Imagem original. (c) Decomposição em 3 níveis pela wavelet Daubechies-4. (d) os três níveis de decomposição (L1, L2, L3) vistos individualmente, com seus respectivos coeficientes de aproximação, detalhes horizontais, diagonais e verticais

4.7 Transformada “Wavelet Packet”

A Transformada *Wavelet Packet* é uma generalização do conceito da Transformada Wavelet Discreta (DWT), na qual a resolução tempo-freqüência pode ser escolhida de acordo com o sinal. A diferença entre elas está no número de parâmetros de decomposição. Na transformada discreta os sinais são decompostos pelos parâmetros escala e posição, gerando dois coeficientes para cada nível: aproximação e detalhe, derivados unicamente

do coeficiente de aproximação (A) analisado, conforme observado na figura 4.8. Já a “Wavelet Packet” adiciona um novo parâmetro: a frequência. Dessa forma, os coeficientes de detalhe (D) resultantes dos filtros de passa-alta, descartados pela transformada discreta, são utilizados para uma nova análise, gerando também, os coeficientes aproximação e detalhe para todos os níveis da decomposição, oferecendo uma análise mais rica. Na figura 4.8 este processo está genericamente ilustrado.

Figura 4.8 – Árvore de decomposição de sinais na Transformada Wavelet Packet, onde A representa os coeficientes de aproximação (filtro passa-baixa) e D o coeficiente de detalhe (filtro passa-alta)

O cálculo numérico dos coeficientes da *Wavelet Packet* é a seqüência do produto interno de um sinal $x = x(t)$ em $L^2(\mathfrak{R})$ com as funções *Wavelet Packet* $W_{sf}(t)$ (PARRAGA, 2002); conforme mostra a equação 4.9, onde $\lambda_{sf}(p)$ são os coeficientes da expansão de x nas funções *Wavelet Packet*, $W_{sf}(t)$ são as funções *Wavelet Packet* básicas, s é o índice de escala (ou dilatação), p é o índice de posição (ou translação), e f é o índice de frequência; $s, p \in \mathbb{Z}$, e $f \geq 0$ (PARRAGA, 2002).

$$\lambda_{sf}(p) \stackrel{\text{def}}{=} \langle x, w_{sf} \rangle = \int_{\mathfrak{R}} x(t) \cdot 2^{-s/2} w_f(2^{-s}t - p) dt \quad (4.9)$$

Na utilização de *wavelet* ortogonal, o esquema computacional para gerar *Wavelet Packets* é de fácil implementação. Inicia-se com dois filtros de tamanho $2N$ contendo $h(n)$ e $g(n)$ que correspondem à *wavelet* (MISITI et al,

2005). A equação 4.10 define a seqüência de funções, onde $W_0(x) = \phi(x)$ é a função escala e $W_1(x) = \psi(x)$ é a função “wavelet” ou “wavelet-mãe” (MISITI et al, 2005), e $h(k)$ e $g(k)$ são os Filtros de Quadratura H e G. Esta seqüência de funções está organizada na figura 4.9, sendo o parâmetro de escala j (número de níveis de decomposições), $W_{j,n}$ a *Wavelet-Packet* na posição n .

$$(W_n(x), n = 0, 1, 2, \dots)$$

por

$$W_{2n}(x) = \sqrt{2} \sum_{k=0}^{2N-1} h(k) W_n(2x - k) \quad (4.10)$$

$$W_{2n+1}(x) = \sqrt{2} \sum_{k=0}^{2N-1} g(k) W_n(2x - k)$$

Figura 4.9 – Árvore da organização da *Wavelet Packets*. A escala j define os níveis de decomposição e $W(n)$ a posição.

Fonte: (MISITI et al, 2005).

4.8 Considerações Finais

Neste capítulo foi abordado o assunto referente às Transformadas *Wavelets*.

Essas transformadas são constituídas de funções que satisfazem requisitos matemáticos, e podem ser processadas utilizando diferentes escalas ou resoluções. O processo de decomposição *Wavelet* de um sinal gera informações de baixa e alta frequência, tratadas respectivamente como coeficientes de aproximação e detalhes (horizontais, verticais e diagonais). Existem diversas possibilidades de escolha para uma função *Wavelet*, dentre as quais se destacam: Haar, Daubechies, Meyer, Morlet, dentre outras.

Capítulo 5 – Metodologia

5.1 Introdução

Este trabalho tem por objetivo o estudo comparativo do uso de Transformadas *Wavelets* na análise de textura de íris humana para aplicação em sistemas de reconhecimento biométrico. De modo geral, pode ser dividido em duas etapas principais: localização e extração da íris e o reconhecimento utilizando a análise *Wavelet*.

Na primeira etapa, foi desenvolvido um algoritmo programado no ambiente Delphi, contendo os métodos de localização, tratamento e extração da íris, onde as técnicas empregadas e os resultados obtidos são descritos nos próximos tópicos. Com as imagens das íris extraídas e normalizadas no tamanho 64x512(altura/largura), uma nova base de dados foi criada, com a finalidade de analisar somente a área de interesse e facilitar os passos seguintes de processamento.

A segunda etapa compreende um estudo comparativo sobre qual família *Wavelet* apresenta o melhor resultado para reconhecimento das imagens da íris, através do desenvolvimento de um algoritmo utilizando a interface com o usuário no ambiente Delphi, o processamento das *Wavelets* e o cálculo da distância Euclidiana na ferramenta Matlab.

Um método foi desenvolvido para avaliar a influência nos resultados das

imagens borradas, desfocadas e que possuem uma grande quantidade de cílios ou pálpebras, reduzindo ou mesmo tornando imprópria tal imagem para o reconhecimento.

5.2 Materiais

Todas as imagens utilizadas no desenvolvimento e avaliação desse trabalho foram obtidas do banco de imagens CASIA, fornecido pelo *Íris Recognition Research Group - Nation Laboratory of Pattern Recognition* (NLPR), mantido pelo Instituto de Automação da Academia Chinesa de Ciências.

No total são 756 imagens de olhos referentes a 108 indivíduos, onde cada indivíduo possui 7 imagens, capturadas em duas etapas distintas. Na primeira, foram obtidas três imagens e na segunda, um mês após, outras quatro imagens. Através dessa construção, pode-se comprovar que não existe variação nas íris de um mesmo indivíduo com o decorrer do tempo, e também simular o momento do cadastro de uma pessoa e uma tentativa de identificação ou autenticação, um mês depois.

As imagens possuem resolução de 320x280 *pixels* e 256 níveis de cinza, capturadas através de um sensor ótico digital desenvolvido pelo NLPR.

Com a finalidade de testar os sistemas de análise da qualidade de imagens, também estão inseridas no banco CASIA imagens borradas, desfocadas e com forte oclusão da íris, geralmente pela influência dos cílios.

5.3 Equipamento Utilizado

Para avaliar o desempenho dos métodos de localização, tratamento e extração da íris, foi utilizado um Pentium IV, com processador de 3.06 GHz e 512 MB de memória RAM.

Para o desenvolvimento do software de análise Wavelet foi utilizado um Pentium IV com processador de 2.80 GHz com 512 MB de memória RAM.

5.4 Segmentação da Íris

A segmentação da íris utilizando algoritmos propostos por Wildes (WILDES, 1997), Daugman (DAUGMAN, 2001, 2002) e Li Ma (MA et al., 2003) apresenta resultados expressivos na identificação de indivíduos, mas do ponto de vista computacional, exige tempos de processamento por quadro além do necessário para análise em tempo real (mínimo de 15 quadros por segundo).

A íris humana e estruturas adjacentes possuem morfologia e simetria muito regulares o que permite a identificação de regiões baseada em um conjunto mínimo de características obtidas de uma imagem. Dessa forma, uma análise amostral de características das imagens e a inferência indireta das regiões de interesse foi desenvolvida neste trabalho, apesar do mesmo processar na seqüência imagens de um banco de dados. A idéia é poder estudar, no futuro, os resultados obtidos aqui, para a análise e o reconhecimento dos padrões da íris em tempo real.

A análise amostral de características da imagem provocaria baixa performance do método – uma vez que grandes erros podem ocorrer ao se inferir características globais baseado em observações locais; entretanto uma filtragem estatística é utilizada para minimizar o efeito destes erros sobre os resultados finais obtidos das imagens.

Para garantir alto desempenho computacional dos processos de detecção e extração da íris, utilizou-se de algoritmos simples e limitou-se as regiões de processamento.

5.5 Algoritmo Generalizado de Segmentação da Íris

O método desenvolvido pode ser sintetizado nos seguintes passos:

1. Equalização do histograma da imagem;
2. Dispersão de sementes;
3. Localização da pupila;
4. Aumento do contraste da imagem (método Sat-Quad);
5. Dispersão orientada de sementes;
6. Localização da borda externa da íris;
7. Extração da íris;

5.5.1 Equalização da Imagem

Um pré-processamento da imagem, realizando-se a equalização de seu histograma, faz-se necessária para minimização das diferenças de contraste e brilho em diferentes imagens. A figura 5.1 mostra um exemplo de uma imagem do banco e sua correspondente equalização.

Figura 5.1 – Equalização de Histograma. (a) Imagem original. (b) Histograma da imagem original. (c) Imagem equalizada. (d) Histograma da imagem equalizada.

5.5.2 Localização da pupila

Nove sementes são aleatoriamente dispersadas sobre uma região central e limitada da imagem. A aleatoriedade das coordenadas das sementes é obtida utilizando-se a função *Random* – do ambiente de programação Delphi – que gera seqüências pseudo-aleatórias baseadas em uma semente de 32-bits. Apesar da característica pseudo-aleatória da função *random*, uma inicialização baseada no relógio do computador garantiu independência amostral das sementes dispersas.

Uma análise local é iniciada sobre cada semente para localização da pupila, calculando-se o gradiente em 4 direções pré-definidas (para cima, para baixo, para a direita e para a esquerda) a partir de cada uma das sementes,

determinam-se os pontos cuja ocorrência de gradiente é um valor superior a um limiar pré-definido (figura 5.2).

Figura 5.2 – (a) Nove sementes na imagem. (b) Direção de varredura das sementes

O cálculo do gradiente é realizado observando-se regiões adjacentes em cada uma das quatro direções indicadas (Figura 5.2b). Experimentos iniciais mostraram que a análise de regiões com apenas 1 *pixel* resultavam em excessivos erros devido aos ruídos da imagem. Para minimizar o efeito de ruídos da imagem sobre o cálculo do gradiente sem a necessidade de se aplicar um filtro sobre toda a imagem (o que resultaria em um gasto expressivo de tempo computacional), escolheu-se analisar regiões sucessivas de 5 *pixels* para cada direção de varredura, onde a média de intensidade é aplicada ao cálculo do gradiente. A figura 5.3 ilustra esse procedimento.

(a) (b)
Figura 5.3 – Localização da pupila através da dispersão de 9 sementes, onde cada semente possui uma cor diferente. (a) Imagem completa. (b) Destaque das sementes na região da pupila.

Cada semente retorna de 0 a 4 pontos (referente a cada direção de varredura) dependendo da identificação com sucesso do gradiente. O retorno de 3 ou mais pontos permite então a realização de uma regressão linear iteracional circular (CHERNOV; OSOSKOV, 1984). Esse algoritmo de regressão proposto por Chernov é altamente otimizado (pois utiliza mecanismos iteracionais simples) e se mostrou bastante adequado às necessidades deste trabalho. O processo se repete para cada uma das sementes, armazenando-se todos os círculos encontrados. Caso um número menor que 3 círculos não seja encontrado, uma nova dispersão das sementes é realizada, e assim sucessivamente, até que sejam encontrados 3 círculos ou que a dispersão tenha atingido metade da largura da imagem, onde nesse caso, o processo é finalizado sem a detecção da pupila (Figura 5.4).

Figura 5.4 – Pupila não localizada.

Caso a coleção de 3 ou mais círculos tenha sido detectada, essa é então analisada e uma filtragem estatística descarta as circunferências com menor probabilidade de estarem representando o centro e raio da pupila da imagem. Essa filtragem baseia-se no fato que grande parte das circunferências encontradas se agrupará em uma determinada classe amostral, enquanto o restante estará disperso.

Este subconjunto amostral mais denso, ou seja, que contém o maior número de círculos coerentes (com raios e centros próximos), é utilizado para cálculo do raio e centro exatos da pupila. Estes valores são obtidos pela média aritmética de cada uma das características das circunferências encontradas. O resultado é mostrado na figura 5.5.

Figura 5.5 – Pupila localizada

5.5.3 Aumento do contraste da imagem

A borda externa da íris não possui uma envoltória bem definida, apresentando na imagem uma transição gradual para a esclera. Este fato dificulta a detecção de sua fronteira.

Para minimizar esse efeito, foi aplicado um método para realce do contraste da imagem, permitindo a detecção da borda da íris com maior eficiência.

Este método modifica a intensidade de cada *pixel*, utilizando como referência a média de intensidade da imagem, onde os valores de intensidades dos *pixels* superiores a esta referência são saturados. Isto basicamente remove a pele e regiões da esclera. Os *pixels* com intensidades inferiores à média calculada têm seus valores de intensidades reprocessados utilizando uma função exponencial, definida na equação 5.1.

$$f(x) = a \cdot x^2 \quad (5.1)$$

Onde a é o valor do *pixel* e x a razão entre o valor do *pixel* e a média de intensidade da imagem.

O resultado desse processo é ilustrado na figura 5.6.

Figura 5.6 – (a) Imagem original. (b) Imagem processada

5.5.4 Localização da Borda Externa da Íris

A íris possui formato circular e centro próximo ao da pupila; por isso foi adotada a detecção de Gradiente, semelhante à detecção da pupila, para definição da borda externa da íris.

Foram utilizadas empiricamente, 6 sementes para análise de mudança do gradiente. Apenas as direções para a direita e para a esquerda foram utilizadas, uma vez que a análise para cima e para baixo detectaria bordas da pálpebra e os cílios (Figura 5.7b).

O posicionamento dessas 6 sementes é fixo e baseado na localização da pupila. Foi escolhida como região de interesse a área logo abaixo ao centro da pupila, área que normalmente está visível (Figura 5.7a).

Figura 5.7 – (a) Região inicial de análise das 6 sementes (retângulos)
(b) Direção de varredura das sementes

De forma análoga à detecção da pupila, até 6 círculos são identificados a partir das características locais observadas através de cada semente. Desse conjunto de círculos é separado um subconjunto de elementos contendo os círculos com maior coerência de características (centros e raios); o cálculo da região circular final é então efetuado - identificando, portanto, o limite externo da íris - pela média dos círculos coerentes. A região delimitada da íris é mostrada na figura 5.7a.

5.5.5 Extração da Íris

Obtidos o centro e os raios da pupila e da íris é possível delimitar a região para extração da íris.

A extração tem como objetivo remover as regiões nas quais não há interesse na imagem, como o esclera e a pupila, e normalizá-la para uma figura padrão no sistema cartesiano.

A rotina executa uma varredura circular, normalizando a diferença dos raios para uma altura de 64 *pixels* e a rotação de 360 graus para 512 *pixels* de largura da figura normalizada. (Figura 5.8)

Figura 5.8 – Íris normalizada em 64x512 pixels

Esta imagem pode então ser realçada e processada para a extração de características e reconhecimento.

5.6 Análise das irises extraídas

Com base nas imagens das irises normalizadas em 64x512 *pixels*, é necessário identificar partes nas imagens que possibilitem a extração de características, reduzindo a influência das informações não relevantes para o reconhecimento, como cílios, pálpebras e esclera (Figura 5.9a). A metodologia empregada foi segmentar as irises extraídas em oito partes iguais de tamanho 64x64 *pixels* (Figura 5.9b), no intuito de identificar a possível existência de regiões que contivessem menor interferência dos elementos não relevantes ao processamento, e ao mesmo tempo, com um número aceitável de *pixels* para a construção do vetor de características.

Foi identificado uma região nas irises segmentadas que possui pouca ou nenhuma interferência de cílios e pálpebras, localizada entre a primeira e a quarta parte da imagem segmentada (Figura 5.9c). Através dessa região, maximizou-se a utilização das imagens contidas no banco CASIA, havendo apenas o descarte de uma pequena quantidade dessas regiões, que ainda contém a influência da pálpebra, já que, de acordo com o tamanho e localização desta região, conseguiu-se eliminar a influência dos cílios (Figura 5.9d). Essa região de tamanho 31x224 forma a região de interesse, compondo o vetor de características de cada indivíduo, que será decomposta pela análise multiresolução *Wavelet*. (Figura 5.9e).

Figura 5.9 – Separação da região de interesse.

- (a) Íris normalizada em 64x512 pixels. (b) Localização das oito partes de 64x64 pixels. (c) Localização da região de interesse na imagem da íris segmentada. (d) Região de interesse com a influência da pálpebra. (e) Região de interesse para extração de características, 31x224.

Identificada a região de interesse, o próximo passo é a análise *Wavelet* para geração do vetor de características. Para tal, buscou-se o desenvolvimento de um método que proporcionasse a análise das diversas famílias *Wavelets*, bem como a influência dos níveis e dos componentes gerados pela decomposição multiresolução. A implementação ocorreu em duas plataformas de softwares, Delphi e Matlab. A plataforma Delphi foi escolhida devido à um conjunto maior de componentes gráficos para a interface com o usuário, ficando para a plataforma Matlab, a decomposição *Wavelet* para montagem do vetor de características, a análise da qualidade das imagens e o cálculo da distância Euclidiana.

5.6.1 Avaliação da qualidade das imagens

Para garantir que uma imagem da íris de um indivíduo contenha apenas os componentes necessários para compor o vetor de características, ou seja, a textura das imagens, um algoritmo para avaliar a qualidade das irises extraídas foi desenvolvido. As características que não possuem qualidade aceitável para a sequência do processamento são: as irises não detectadas pelo algoritmo de localização e extração da íris, e as regiões de interesse que contenham qualquer porção de pálpebras, que serão denominadas *Q1* e *Q2*, respectivamente.

O primeiro fator de descarte refere-se às irises não detectadas. Para isso, avalia-se a imagem gerada e se a mesma apresentar uma grande quantidade de *pixels* tendendo a zero, resultando em uma imagem escura, ela será descartada (Figura 5.10).

Figura 5.10 – imagem referente à uma íris não detectada.

O método utilizado foi calcular a média aritmética de intensidade dos *pixels*, considerando-se que as imagens de irises não detectadas possuem uma média de intensidade dos *pixels* inferior ao valor 80. Esse índice de qualidade *Q1* é dado pela equação 5.2.

$$Q_1 = \frac{\sum_{i=1}^n \sum_{j=1}^m p_{i,j}}{n \cdot m} \quad (5.21)$$

Onde $p_{i,j}$ é o valor de intensidade do pixel p na posição i,j da imagem, e n e m representam a altura e largura da imagem.

O segundo fator de descarte são as pálpebras contidas na região de interesse (Figura 5.11).

Figura 5.11 – Região de interesse com influência da pálpebra.

Para verificar se uma região de interesse possui ou não uma parte da pálpebra, realizou-se uma análise da *Wavelet* de Haar no nível de decomposição 1 (um) (Figura 5.12a). As pálpebras possuem uma grande quantidade de *pixels* tendendo ao branco (255), e na análise *Wavelet pixels* nesse intervalo geram os componentes de baixa frequência. Com base nessa observação, foi desenvolvido um cálculo estatístico do desvio padrão sobre os componentes de aproximação *Wavelet*, onde se averiguou que tais imagens possuem um desvio padrão maior ou igual ao valor 74, sendo esse considerado o fator de descarte (Figura 5.12b). Esse índice de qualidade Q_2 é dado pela equação 5.3.

$$Q_2 = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2} \quad (5.3)$$

Figura 5.12a – Região de interesse sem influência da pálpebra (Desvio padrão dos coeficientes de aproximação igual a 55).

Figura 5.12b – Região de interesse com influência da pálpebra (Desvio padrão dos coeficientes de aproximação igual a 74).

Existe ainda outro fator para avaliação da qualidade das imagens: as imagens capturadas desfocadas por movimento. Zhenan (ZHENAN, 2003) propõe um método para avaliar a qualidade de imagens da íris no domínio freqüência, cílios e pálpebras apresentam altas freqüências, as imagens desfocadas estão localizadas no domínio de freqüência mais baixa e as borradas por movimento estão dispersas por todo espectro. Nas imagens nítidas e desobstruídas, o espectro aglomera-se na freqüência vertical baixa e horizontalmente na freqüência baixa ou média.

(MORENO, 2005) analisa duas porções da imagem onde é mais provável que ocorram pálpebras e cílios, de dimensão 22×22 *pixels*. Para cada uma delas, é calculado o espectro de Fourier 2D, resultando em duas matrizes de 22×22 *pixels* que são separadas em três sub-regiões representando as altas, médias e baixas freqüências. O descritor de qualidade então é calculado, contendo dois valores: o primeiro, que são os componentes de alta freqüência, e o segundo, a média aritmética dos componentes de média freqüência pela soma dos componentes de alta e baixa freqüências.

Para avaliar a influência apenas das imagens desfocadas, foi implementado o algoritmo proposto por Moreno (MORENO, 2005), que é baseado no algoritmo de Li Ma (MA et al., 2003). Os resultados não foram os mesmos obtidos pelo autor principal, já que o método de extração da íris e as regiões de interesse são diferentes, e talvez por isso, não se tenha conseguido descartar efetivamente as imagens desfocadas.

Para avaliar a influência nos resultados das imagens borradas e desfocadas, um teste foi realizado considerando-se duas imagens desfocadas. Conforme pode ser observado nas figuras 5.13, a porcentagem de acerto de cada imagem desfocada não sofre variações relevantes em relação a uma imagem nítida e focalizada.

A Figura 5.13b mostra a tentativa de identificar o indivíduo referente à imagem 96, retornando corretamente as 7 imagens que mais se assemelham à este, resultando em um acerto de 100%. Já a figura 5.13d, o acerto foi de 6 entre as 7 imagens consultadas, resultando em uma precisão

de 85,71%.

Após a comprovação de que os vetores de características gerados pela análise *Wavelet* não sofrem influência das imagens desfocadas, o algoritmo adaptado de Moreno (MORENO, 2005) não foi adotado para a análise de qualidade das imagens.

096_1_3.bmp

Figura 5.13a – Imagem de uma íris desfocada (indivíduo 96).

Figura 5.13b – Identificação das 7 imagens com maior similaridade em relação à imagem desfocada (Figura 5.13a) – acerto de 100% (7/7).

038_1_3.bmp

Figura 5.13c – Imagem de uma íris desfocada (indivíduo 38).

Figura 5.13d – Identificação das 7 imagens com maior similaridade em relação à imagem desfocada (Figura 5.13c) – acerto de 85,71% (6/7).

5.6.2 Composição do vetor de características

O primeiro passo para se obter um vetor de características de cada indivíduo consiste em aplicar um realce região de interesse. Algumas técnicas foram testadas com esse propósito, tais como a equalização convencional de histograma e filtragens passa-alta. A técnica que apresentou os melhores resultados para a etapa de reconhecimento é a equalização adaptiva de histograma (Figuras 5.14a e 5.14b). A diferença entre a equalização convencional de histograma e a equalização adaptiva está na região analisada. O método de equalização convencional de histograma utiliza a imagem em uma única iteração, já no modelo adaptivo, a equalização se dá por regiões automaticamente selecionadas na imagem, chamadas de *telhas*, conforme pode ser conferido nas linhas horizontais e verticais da figura 5.14b (MATHWORKS, 2006). O contraste de cada região é realçado, de modo que o histograma da região de saída seja compatível com o histograma especificado por um parâmetro de distribuição. As vizinhanças dessas regiões são combinadas, usando interpolação bi-linear para eliminar os limites artificialmente induzidos. O contraste, especificamente em áreas homogêneas pode ser limitado para evitar a amplificação de todo o ruído que possa estar presente na imagem (MATHWORKS, 2006).

Figura 5.14a – Região de interesse 31x224.

Figura 5.14b – Região de interesse com histograma equalizado.

A análise *Wavelet* é realizada na região de interesse, agora já realçada. Esse processo de decomposição retorna quatro vetores, que correspondem respectivamente, aos componentes de baixa frequência (vetor de

aproximação) e alta frequência (vetores dos detalhes horizontal, vertical e diagonal) para o nível de decomposição selecionado (Figura 5.15).

Figura 5.15 – Decomposição da região de interesse

A dimensão dos vetores referente aos componentes decompostos também sofre variação de acordo com a família *Wavelet* e nível de decomposição selecionados, conforme pode ser observado na tabela 5.1.

Tabela 5.1 – Dimensão dos vetores referentes aos componentes Wavelet.		
Família Wavelet	Nível de decomposição	Dimensão dos vetores de Componentes Wavelet
Haar	1	1792
Haar	4	28
Daubechies 2	1	1921
Daubechies 2	4	64
Daubechies 8	1	2737
Daubechies 8	4	448
Symlets 4	1	2185
Symlets 8	3	697
Biortogonal 3.3	2	793

O vetor de características é gerado a partir da concatenação dos vetores com os componentes *Wavelet*, ordenadamente formados pelos coeficientes de aproximação, detalhes horizontais, detalhes verticais e detalhes diagonais, dado pela equação 5.4. Diversas combinações entre os vetores com os coeficientes foram testadas para a composição do vetor de características, na procura pela ordem que diferenciase um indivíduo do outro. Dentre esses experimentos, utilizou-se a inversão na seqüência dos vetores de coeficientes, e o acúmulo de cada vetor-componente quando o nível de decomposição escolhido tenha sido maior ou igual a dois. Por fim, apenas os componentes do nível de composição escolhido foi utilizado, sem acumular os valores de cada componente em cada nível inferior ao selecionado.

$$\text{Vetor de Características} = [cA_n \cup cD_n^h \cup cD_n^v \cup cD_n^d] \quad (5.4)$$

A dimensão final do vetor de características para cada indivíduo do banco CASIA também será variável, dependendo da escolha da família *Wavelet* e nível de decomposição, conforme pode ser observado nos exemplos da

tabela 5.2.

Tabela 5.2 – Dimensão dos vetores referentes aos componentes Wavelet.		
Família Wavelet	Nível de decomposição	Dimensão do vetor de características
Haar	1	7168
Haar	4	112
Daubechies 2	1	7684
Daubechies 2	4	256
Daubechies 8	1	10948
Daubechies 8	4	1792
Symlets 4	1	8740
Symlets 8	3	2788
Biorotogonal 3.3	2	3172

5.6.3 Reconhecimento através das características extraídas

Para o reconhecimento de um indivíduo através de autenticação ou identificação, calcula-se o vetor de características com os componentes *Wavelet* decompostos para cada uma das 756 imagens do banco. Pode-se então operar no modo de autenticação, que consiste em comparar um determinado vetor de características de entrada com todos os demais, formando uma comparação um-para-um, ou no modo de identificação, onde é calculada a distância de cada vetor de característica das 756 imagens em relação às demais, retornando a semelhança entre eles de forma decrescente, numa comparação um-para-muitos.

Para determinação de quão semelhantes são duas irises, é calculada a distância Euclidiana entre todos os vetores de características das imagens, dada pela equação 5.5.

$$D(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (5.5)$$

Onde x_i representa o vetor de características do indivíduo de entrada, y_i o vetor de características do indivíduo no banco de imagens, e n o número de características do vetor.

O cálculo dessa equação retorna a distância de um vetor de características para os demais. Quanto menor o valor da distância Euclidiana entre os vetores característicos, mais próximas serão as íris desses indivíduos, podendo ser ou não a mesma pessoa. Para exemplificar esse conceito, a figura 5.16 mostra o cálculo das distâncias entre a primeira imagem da íris do indivíduo número 1 e as próximas 99 imagens do banco. O número de 100 comparações foi utilizado para que fosse possível uma visualização aceitável do gráfico, já que o cálculo da distância Euclidiana retorna 756 comparações, representando cada imagem do banco.

Figura 5.16 – Gráfico da Distância Euclidiana de uma íris

Dessa forma, analisando o gráfico da figura 5.16, verifica-se na região em destaque que as distâncias entre os vetores de características de íris de uma mesma pessoa apresentam valores muito inferiores comparados à distância entre os vetores de íris de indivíduos diferentes.

No mesmo gráfico pode-se notar ainda que não existe distância entre o vetor de características de uma mesma imagem de íris, ou seja, sua distância tem valor igual a zero.

5.7 Considerações Finais

Neste capítulo foi apresentada a proposta para reconhecimento biométrico pela íris humana, onde foram escolhidas algumas famílias *Wavelets* e níveis de decomposição para compor o vetor de características de cada indivíduo.

Técnicas de processamento de imagem, tais como dispersão aleatória de sementes para localizar regiões de interesse, equalização adaptiva de histograma e aumento de contraste foram utilizadas na etapa de pré-processamento das imagens.

Capítulo 6 – Resultados e Conclusões

6.1 Resultados

Os resultados obtidos com o método proposto no capítulo 5 estão separados em duas partes. A primeira avalia a localização e extração da íris, e a segunda a recuperação de imagens.

6.1.1 Avaliação da localização e extração da íris

Na tabela 6.1 encontram-se os resultados do método de localização e extração. Como este processamento utiliza a técnica aleatória de dispersão de sementes, no propósito de localizar a íris na imagem, a taxa de acerto pode sofrer uma pequena variação, sempre menor que 1%. Dentre as 756 imagens do banco CASIA, aproximadamente 4 pupilas não foram localizadas e 13 irises não foram detectadas.

Capítulo 6 – Resultados e Conclusões

Tabela 6.1 – Avaliação do método de localização e extração da íris	
Imagens Analisadas	756
Pupilas localizadas	99,60%
Íris localizadas	97,09%
Acerto Global	96,43%
Tempo Médio/Frame	21,45ms
Frame/Segundo	46,6

6.1.2 Avaliação do método de recuperação de imagens

O primeiro passo para avaliar o método de recuperação é calcular a Distância Euclidiana entre todas as imagens, resultando em uma matriz de dimensão 756x756. O primeiro valor de cada uma das 756 linhas representa uma imagem do banco CASIA, e os valores contidos nas 755 colunas subsequentes (DE1...DE756), as respectivas distâncias desta. A ilustração desse conceito pode ser observada na Tabela 6.2.

Tabela 6.2 – Distâncias Euclidianas entre as imagens							
Entrada	DE1	DE2	DE3	DE4	DE5	...	D756
001	006	003	002	005	004	...	108
002	001	005	007	003	006	...	343
003	007	001	004	005	002	...	030
004	002	007	001	003	006	...	099
005	003	006	002	001	007	...	224
006	005	001	002	007	004	...	026
007	001	005	003	002	004	...	118
008	009	013	011	014	010	...	295
009	012	014	011	008	013	...	066
010	008	011	014	009	012	...	328
...	...	...	...	...	...	...	651
756	753	751	754	752	755	...	556

Capítulo 6 – Resultados e Conclusões

Na tabela 6.3 encontram-se os resultados da análise de algumas famílias *Wavelets* em determinados níveis de decomposição, utilizando todos os componentes (aproximação e os detalhes horizontais, verticais e diagonais), equalização adaptiva de histograma e avaliação da qualidade das imagens, onde podem ser observadas as respectivas taxas de acertos e tempo computacional gasto.

Tabela 6.3 – Tamanho dos vetores referentes aos componentes Wavelet.					
Wavelet	Nível	Total de comparações	Total de acertos	Taxa de acertos	Tempo de processamento
Haar	1	4872	3066	62,93%	3m27s
Haar	4	4872	3945	80,97%	1m39s
Daubechies 2	2	4872	3222	66,13%	2m14s
Daubechies 2	5	4872	2142	43,66%	1m42s
Daubechies 8	1	4872	3066	62,93%	5m08s
Daubechies 8	3	4872	3367	69,11%	2m40s
Symlets 4	2	4872	3183	65,33%	2m28s
Symlets 17	4	4872	3711	76,17%	8m53s
Biorotogonal 3.3	4	4872	2806	57,59%	1m44s
Discrete Meyer	1	4872	3106	63,75%	10m46s
Discrete Meyer	4	4872	3971	81,51%	15m06s

O total de comparações é calculado através da multiplicação do número de imagens com índice de qualidade e o número 7, que representa a quantidade de imagens de um mesmo indivíduo. O total de acertos é incrementado para cada uma das 7 imagens seguintes com índice de qualidade, que possua um valor correspondente à mesma classe da imagem de entrada.

O segundo e último passo para avaliação é através das curvas *Recall x*

Precision. A figura 6.1 ilustra o modelo geral dessas curvas, que tem o propósito de avaliar as informações relevantes e irrelevantes retornadas por um sistema de busca.

Figura 6.1 – Modelo das curvas Recall x Precision.

A medida *recall* corresponde à proporção de respostas corretas e é uma estimativa da probabilidade de se obter pelo menos uma imagem relevante (Equação 6.1).

$$Recall = \frac{\text{Quantidade de imagens relevantes recuperadas}}{\text{Quantidade total de imagens relevantes}} \quad (6.1)$$

Já a medida *precision* corresponde à qualidade de cada recuperação – estimativa da probabilidade de cada imagem retornada estar correta (Equação 6.2).

$$Precision = \frac{\text{Quantidade de imagens relevantes recuperadas}}{\text{Quantidade total de imagens recuperadas}} \quad (6.2)$$

As curvas das figuras 6.2 e 6.3 representam o sistema de recuperação de imagem com base na *Wavelet* de Haar, no nível 4 de decomposição; comparando cada imagem do banco e retornando as n imagens de maior semelhança. A escolha dessa família *Wavelet* e nível de decomposição é devido ao tempo computacional inferior e uma pequena diferença na taxa de precisão, comparando-se com a *Wavelet* Discrete Meyer, que obteve a maior taxa mas com tempo computacional elevado, conforme pode ser observado na tabela 6.3.

Capítulo 6 – Resultados e Conclusões

Como no banco CASIA existem 7 imagens para cada um dos 108 indivíduos, utilizou-se a variável n com valor igual a 10, procurando dessa maneira, mostrar a eficiência do método tanto para as imagens que devem ser consideradas do mesmo indivíduo, ou seja, as 7 primeiras imagens, quanto avaliar os resultados para as demais imagens retornadas (3 imagens), que não se referem a mesma pessoa.

A Figura 6.2 mostra a recuperação de indivíduos do banco de imagens sem a utilização do algoritmo de verificação da qualidade da imagem.

Figura 6.2 – Curva Recall x Precision do sistema de recuperação sem a verificação da qualidade de imagem.

Verifica-se na figura 6.2, que representa a recuperação dos 10 indivíduos que mais se assemelham com a imagem original de entrada, sem utilizar o processo de avaliação da qualidade de imagens, uma precisão de aproximadamente 79,19% na recuperação das 7 imagens do mesmo indivíduo, representando a identificação correta de 519 imagens, e a não identificação de 237 imagens.

Capítulo 6 – Resultados e Conclusões

A Figura 6.3 mostra a recuperação de indivíduos do banco de imagens considerando o algoritmo de qualidade.

Figura 6.3 – Curva Recall x Precision do sistema de recuperação com a verificação da qualidade de imagem.

Verifica-se na figura 6.3, que representa a mesma tentativa de recuperação dos 10 indivíduos que mais se assemelham com a imagem original de entrada, utilizando o procedimento de verificação da qualidade de imagens, a precisão de aproximadamente 80,97%, representando a identificação correta de 612 imagens, e a não identificação de 144 imagens.

6.2 Conclusão

Para a localização e extração da íris, o algoritmo proposto foi rápido o suficiente para trabalhos de reconhecimento em tempo real, podendo ser utilizado para vídeos capturados, ao invés de imagens estáticas,

Capítulo 6 – Resultados e Conclusões

processando um frame em aproximadamente de 46 milissegundos. Para avaliar o desempenho deste método, pode-se compará-lo à segmentação da íris proposto por Moreno (MORENO, 2005), que consome cerca de 5,7 segundos para processar cada frame, valor este insuficiente para trabalhos em tempo real.

Já para o reconhecimento dos padrões das íris contidas no banco CASIA, o propósito foi pesquisar as diversas famílias *Wavelets*, seus níveis e componentes de decomposição para geração do vetor de características. Como diversas adaptações se fizeram necessárias para que fosse possível efetuar uma variedade grande de análises, o tempo de processamento não foi considerado um fator importante.

Nos experimentos realizados esse tempo variou entre 1 e 15 minutos, dependendo da família *Wavelet* e nível de decomposição selecionados, o que o torna impróprio para trabalhos de processamento em tempo real. Visando minimizar esse fator, uma opção para salvar e carregar os resultados processados foi implementada.

Com base nos gráficos gerados nas seções anteriores, verificou-se que os melhores resultados foram obtidos utilizando a seguinte combinação de técnicas:

- Equalização adaptiva de histograma;
- Análise da qualidade das imagens;
- Todos os componentes da decomposição Wavelet; aproximação, detalhe horizontal, detalhe vertical e detalhe diagonal;
- *Wavelet* de Haar no nível de decomposição 4.

Avaliando todas as imagens do banco, a decomposição pela *Wavelet Discrete Meyer* no nível 4 obteve a melhor taxa, com 81,51% de precisão. Porém, o processamento dessa *Wavelet* consumiu cerca de 15 minutos. Já a *Wavelet de Haar* também no nível 4 resultou em uma precisão de 80,97%, com tempo de processamento aproximadamente de 1m39s, ou seja, 0,11s

Capítulo 6 – Resultados e Conclusões

em média por imagem.

Como o software de reconhecimento desenvolvido permite a escolha de diversos parâmetros para cada processamento, notou-se uma considerável variação no resultado, dos quais se destacam:

- Utilização de apenas um dos componentes de decomposição *Wavelet*;
- Não utilização da equalização adaptiva de histograma;
- Nível de decomposição *Wavelet* diferente de 4. Entre os níveis 1 e 3, nota-se uma queda moderada na taxa de precisão, comparando-se com o nível 4. Já para níveis acima de 4, a queda na taxa de precisão é bastante acentuada, concluindo-se que, quanto maior for o nível de decomposição escolhido, menor será a precisão. Provavelmente a razão dessa queda acentuada deve-se à menor dimensão do vetor de características, já que, quanto maior o nível de composição, menor será a dimensão dos componentes gerados, aumentando a probabilidade de serem semelhantes vetores característicos de indivíduos diferentes.

Para avaliar o desempenho geral deste trabalho, considerando-se o descritor de qualidade, pode-se utilizar o trabalho de Li Ma (MA et al., 2003), que apresenta uma comparação entre os principais métodos já desenvolvidos de identificação (um-para-muitos) pela íris, conforme a tabela 6.4. O método proposto utilizado nesta comparação foi a análise *Wavelet* na família *Discrete Meyer*, no nível de decomposição 4, que obteve a maior taxa de acertos na identificação (81,51). Já no modo de autenticação (um-para-um), a taxa de acertos não variou entre as diversas famílias *Wavelets* e níveis de decomposição, sempre obtendo 100% em todos os experimentos realizados.

Tabela 6.4 – Taxa de Reconhecimento Correto.
Fonte: (MA et al., 2003), exceto Método Proposto.

Método	Taxa de Reconhecimento Correto (%)
Daugman	100
Li Ma	99,43
Boles	92,64
Método Proposto	81,51

6.3 Sugestão para trabalhos futuros

No propósito de evoluir o método de identificação pessoal baseada na íris, podem ser consideradas algumas melhorias no método desenvolvido. O algoritmo de localização e extração da íris mostrou-se apropriado para trabalhos em tempo real, mas o sistema de reconhecimento, compreendido pela extração de características, análise *Wavelet* e o cálculo da Distância Euclidiana, mesmo apresentando um tempo de processamento satisfatório na verificação de todo o banco de imagens, mostrou-se impróprio para tempo real através da busca seqüencial utilizada nesse trabalho.

Para tentativas de se trabalhar completamente em tempo real, pode-se implementar a família *Wavelet* que apresentou os melhores resultados nesse trabalho, que é a de *Haar*, por exemplo, no ambiente de programação Delphi, mantendo o mesmo padrão da implementação de todos os algoritmos de localização e extração das irises.

Algumas modificações no algoritmo de montagem do vetor de características podem ser feitas, buscando outras fórmulas para a composição do mesmo, considerando, por exemplo, a média ou o desvio padrão entre os componentes de aproximação, detalhes horizontais, detalhes verticais e detalhes diagonais.

Capítulo 6 – Resultados e Conclusões

Podem ser pesquisadas outras regiões de interesse, considerando uma ou mais partes menores distribuídas ao longo da imagem. Também outras técnicas de realce como métodos de equalização de histograma e convolução de filtros passa-alta, tanto no domínio do espaço, quando no domínio da frequência.

Melhorias no método de avaliação da qualidade de imagem também influenciarão diretamente nos resultados, já que com esse passo procura-se descartar as imagens que não possuem qualidade mínima de processamento.

Nesse trabalho foi utilizada uma decomposição da *Wavelet* de Haar para detectar a influência da pálpebra nas imagens da íris, resultando em um custo computacional relativamente alto, e também poderiam ser substituídos por outras técnicas de processamento de imagens, como filtragem e morfologia matemática para a detecção e eliminação de tais ruídos.

Outro fator primordial é a utilização da métrica de distância. Somente a distância Euclidiana foi utilizada para medir a distância entre uma região de interesse e as demais. Uma avaliação interessante seria a implementação de outras métricas, como as distâncias de Hamming e Mahalanobis.

Referências Bibliográficas

AL-TAYECHE, Rami; KHALIL, Ahmed (2003). *CBIR: Content Based Image Retrieval* - A report submitted in partial fulfillment of the requirements Engineering Project – Department of Systems and Computer Engineering Faculty of Engineering – Carleton University – April 4, 2003. Disponível em: <<http://www.sce.carleton.ca/faculty/cuhadar/CBIR/files/finalreport.doc>>. Acesso em: 16/mai/2005.

ARANTES, Adriano S. et. al. (2003). *Operadores de Seleção por Similaridade para Sistemas de Gerenciamento de Bases de Dados Relacionais*. 18º Simpósio Brasileiro de Banco de Dados, Manaus, Amazonas, 2003. Disponível em: <<http://www.google.com.br/url?sa=U&start=6&q=http://www.lbd.dcc.ufmg.br/sbbd2003/artigos/paper026.pdf&e=747>>. Acesso em: 17/mai/2005.

BOLES, W. W. (1997). *A Security System Based on Human Iris Using Wavelet Transform*, International Conference on Knowledge-Basic Intelligent Eletronic System, Vol. 2, pages 533-541.

BUENO, Josiane M. (2001). *Suporte à Recuperação de Imagens Médicas Baseada em Conteúdo através de Histogramas Métricos*. Tese (Doutorado em Ciências da Computação), Universidade de São Paulo, São Carlos, 2001.

BUENO, Renato; TRAINA, Agma J. M. ; TRAINA JR, Caetano. (2005). *Algoritmos Genéticos para Consultas por Similaridade Aproximadas*. Disponível em: <<http://www.sbbd-sbes2005.ufu.br/arquivos/artigo-13-BuenoTraina.pdf>>. Acesso em 25/jul/2006.

CASTAÑÓN, César A. B. (2003). *Recuperação de imagens por conteúdo através da análise multiresolução por Wavelets*. Dissertação (Mestrado em Ciências da Computação), Universidade de São Paulo, São Carlos, 2003. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/55/55134/tde-29072004-194807/>>. Acesso em: 30/mar/2005.

CHELLAPPA, R.; WILSON, C. L.; SIROHEY, S. (1995). *Human and Machine Recognition of Face: A survey*. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 83, nº 5.

CHERNOV, N.I.; OSOSKOV, G.A. (1984). *Effective Algorithms for Circle Fitting*. Journal of Computer Physics Communications 33., pages 329-333.

DAUGMAN, John. (2001). *High Confidence Recognition of Persons by Iris Patterns*. IEEE 35th International Carnahan Conference on Security Technology, 16-19 Oct. 2001, pages 254-263.

DAUGMAN, John. (2002). *How Iris Recognition Works*. International Conference on Image Processing, vol. 1, pages 33-36.

FARIA, Regis Rossi Alves. (1997). *Aplicação de Wavelets na análise de gestos musicais em timbres de instrumentos acústicos tradicionais*. Dissertação (Mestrado em Engenharia de Sistemas Eletrônicos), Universidade de São Paulo, São Paulo, 1997.

FILHO, Ogê Marques; NETO, Hugo Vieira. (1999). *Processamento Digital de Imagens*. Rio de Janeiro: Brasport.

FREDERICK, Paula. (1999). *Visualização Eficiente de Objetos Gráficos*. Dissertação (Mestrado em Ciências em Informática), Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 1999.

GALVÃO, Roberto K. H. et al. (2001). *Estudo comparativo sobre filtragem de sinais instrumentais usando transformadas de Fourier e Wavelet*. Química Nova, vol. 24, n.6, p.874-884.

GONG, S. MCKENNA; PSARROU, A. (2000). *Dynamic Vision: From Images to Face Recognition*. Imperial College Press.

GONZAGA, Adilson. (2003). *Biometria – A Senha que Nunca se Esquece*. Disponível em: <<http://www.fafica.br/computacao/biometria.pdf>>. Acesso em: 18/mai/2005.

GONZALEZ, R.; WOODS, R. (2000). *Processamento de Imagens Digital*. São Paulo: Edgar Blucher.

GRAPS, Amara. (1995). *An Introduction to Wavelets*. IEEE Computacional Science and Engineering, Vol. 2, nº 2.

HENRY, E. R. (1905). *Classification and Uses of Fingerprints*. Wyman and Sons Ltda.

JAIN, Anil et al. (1997). *An Identity-Authentication System using Fingerprints*. Proceeding of the IEEE, vol. 85, n. 9, pp. 1365-1388.

KARU, Kalle; JAIN, Anil. (1996). *Fingerprint Classification*. Pattern Recognition, vol. 29, n. 3, pp. 389-404.

LIAPIS, Spyros; TZIRITAS, Georgios. (2004). *Color and Texture Image Retrieval Using Chromaticity Histograms and Wavelet Frames*. IEEE Transactions on Multimedia, vol. 6, n. 5, pp. 676-686.

LOPES, Odnei Cuesta. (2005). *iRIS Servidor: Sistema Servidor para Recuperação de Imagens por Conteúdo*. Dissertação (Mestrado em Ciências da Computação), Universidade de São Paulo, São Carlos, 2005.

MA, Li et al. (2003). *Personal Identification Based on Iris Texture Analysis*, IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. 25, no. 12.

MANJUNATH, B.S.; MA, W.Y. (1996). *Texture Features for Browsing and Retrieval of Image Data*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, nº 8.

MARIN, Luciene de Oliveira; BARRETO, Jorge Muniz. (2003). *Reconhecimento de Faces*. Universidade Federal de Santa Catarina. Disponível em: <http://www.inf.ufsc.br/~l3c/artigos/MarinR03_2003.PDF>. Acesso em 16/mai/2005.

MARRONI, Lílían Saldanha. (2002). *Aplicação da Transformada de Hough para Localização dos Olhos em Faces Humanas*. Dissertação (Mestrado em Engenharia Elétrica), Universidade de São Paulo, São Carlos, 2002.

MATHWORKS, Inc. (2006). Image Processing Toolbox. Disponível em: <http://www.mathworks.com/access/helpdesk/help/toolbox/images/adaphist_eq.html>. Acesso em 21/jul/2006.

MISITI, M.; MISITI, Y.; OPPENHEIM, G.; POGGI, J-M. (2005). *Wavelet Toolbox – for use Matlab – user guide version 3*.

MORENO, Raphael Pereira. (2005). *Extração de Características de Textura de Haralick em Imagens de Iris Aplicada à Identificação Pessoal*. Dissertação (Mestrado em Engenharia Elétrica), Universidade de São Paulo, São Carlos, 2005.

PARRAGA, Adriane. (2002). *Aplicação da Transformada Wavelet Packet na análise e classificação de sinais de vozes patológicas*. Dissertação (Mestrado em Engenharia Elétrica), Universidade Federal do Rio Grande do Sul, Porto Alegre, 2002.

PENTLAND, A. (2000). *Looking at people: Sensing for ubiquitous and wearable computing*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1):107-119.

POH, N., KORCZAK, J. (2001). *Hybrid Biometric Person Authentication Using Face and Voice Features*. Proceedings of the Third International Conference, Audio- and Video- based Biometric Person Authentication AVBPA 2001, Halmstad, Sweden, 348-353.

PUTTE, T.; KEUNING, J. (2000). *Biometrical Fingerprint Recognition: don't get your fingers burned*, Proceedings of IFIP TC8/WG8.8 Forth Working Conference on Smart Card Research and Advanced Applications, Kluwer Academic Publishers, 289-303.

RETICA, Systems Inc. (2005). *Retinal Biometrics Technology*. Disponível em: <<http://www.retinaltech.com/site/technology/index.html>>. Acesso em: 13/mai/2005.

SCHWERDT, Karl, CROWLEY, James L. (2000). *Robust Face Tracking using Color*. 4th IEEE International Conference on Face and Gesture Recognition, Grenoble, France. Processing. Disponível em: <<http://www-prima.inrialpes.fr/Prima/Hompages/jlc/papers/fg2000-Schwerdt.pdf>>. Acesso em: 17/mai/2005.

THEODORIDIS, Sergios; KOUTROUMBAS, Konstantinos. (2003). *Pattern Recognition – Second Edition*, Elsevier Academic Press.

VARGAS, Ernesto Cuadros. (2004). *Recuperação de Informação por Similaridade Utilizando Técnicas Inteligentes*. Tese (Doutorado em Ciências – Ciências da Computação e Matemática Computacional), Universidade de São Paulo, São Carlos, 2004.

VESPER, Escola (2005). *Olho Humano*. Disponível em: <www.escolavesper.com.br/olho_humano.htm>. Acesso em: 13/mai/2005.

WILDES, R. et. al. (1996). *A Machine-vision System for Iris Recognition*, *Machine Vision and Applications*, vol.9, pp.1-8.

WILDES, Richard P. (1997). *Iris Recognition: An Emerging Biometric Technology*, Proceedings of the IEEE, vol. 85, no. 9.

YAMAMOTO, Cláudio Haru; BIAJIZ, Mauro. (2002). *Uso de Lógica Nebulosa na Construção e na Utilização da Árvore Métrica Slim-tree*. Disponível em: <www.icmc.usp.br/~haruo/pubs/wtdbd2002pt_final.pdf>. Acesso em: 25/jul/2006.

ZHENAN, Sun. (2003). *Iris Image Quality Assessment Based on Texture Analysis in the Frequency Domain*. Center for Biometrics Authentication and Testing, National Laboratory of Pattern Recognition □ Institute of Automation, Chinese Academy of Sciences.

Disponível em:

<<http://www.sinobiometrics.com/download/Iris%20Image%20Quality%20Assessment%20Based%20on%20Texture%20Analysis%20in%20Frequency%20Domain.doc>>. Acesso em: 16/mai/2005.

Apêndice A – Software Desenvolvido

Nesse apêndice são apresentados os softwares de localização/extração das imagens das íris e também o de reconhecimento, desenvolvidos na plataforma Delphi.

O programa para a localização e extração da íris efetua a separação da região de interesse (íris) do restante da imagem. Para facilitar as etapas seguintes de processamento, cada imagem da íris extraída é gravada em um arquivo em disco com a mesma identificação dos indivíduos do banco CASIA.

Programa para extração da íris.

O segundo programa tem o objetivo de fornecer opções de interação com o usuário, integrando a plataforma de desenvolvimento Delphi com a plataforma Matlab. A linguagem Delphi trabalha como cliente, passando os parâmetros à serem processados em um servidor Matlab. Essa interação é possível através dos objetos da interface COM. Neste é permitida a escolha da família *Wavelet* e nível de decomposição a serem analisados e também quais os componentes que formarão o vetor de características. Todo esse processamento pode ser gravado em arquivo e depois recarregado, evitando o re-processamento e reduzindo drasticamente o tempo computacional, já que o processo de recarga do arquivo processado consome menos de 2 segundos.

Programa para o sistema de reconhecimento.

Apêndice B – Tabela com a qualidade das imagens do banco

Na tabela à seguir é mostrado o valor do indicador de qualidade Q1 e Q2 das regiões de interesse de todas as imagens do banco CASIA. Na coluna imagem, os três primeiros números representam o indivíduo cadastrado, o quarto número representa a primeira etapa da captura da imagem, e o quinto número é a ordem sequencial da captura. Por exemplo, o número 066_1_2 representa o indivíduo 66, na primeira etapa de captura e a segunda imagem capturada. O indicador Q1 é a média aritmética da intensidade dos pixels de toda a região de interesse, e Q2 o desvio padrão arredondado da decomposição dos componentes de aproximação Wavelet de Haar, no nível 1. Na coluna Qualidade, os números 0 são as imagens identificadas como de qualidade aceitável para processamento; o número -1 (em destaque) são as imagens de íris não detectadas, e o número -2 (em destaque) são as imagens das regiões de interesses que contém pálpebras.

Imagem	Indicador Q1	Indicador Q2	Qualidade	
001_1_1	144,7	54	0	
001_1_2	142,98	55	0	
001_1_3	147,57	54	0	
001_2_1	144,98	54	0	
001_2_2	148,2	57	0	
001_2_3	149,75	58	0	
001_2_4	148,45	57	0	
002_1_1	158,84	56	0	

002_1_2	162,14	55	0	
002_1_3	157,81	54	0	
002_2_1	159,79	56	0	
002_2_2	158,75	54	0	
002_2_3	157,42	54	0	
002_2_4	159,28	57	0	
003_1_1	154,95	60	0	
003_1_2	155,39	60	0	
003_1_3	153,57	60	0	
003_2_1	155,82	58	0	
003_2_2	155,58	56	0	
003_2_3	154,85	58	0	
003_2_4	154,07	56	0	
004_1_1	138,84	73	0	
004_1_2	139,56	70	0	
004_1_3	137,73	69	0	
004_2_1	137,87	67	0	
004_2_2	139,05	69	0	
004_2_3	139,13	70	0	
004_2_4	139,74	69	0	
005_1_1	146,02	46	0	
005_1_2	146,44	51	0	
005_1_3	149,15	48	0	
005_2_1	150,77	53	0	
005_2_2	153,31	53	0	
005_2_3	149,49	50	0	
005_2_4	150,98	53	0	
006_1_1	141,83	74	-2	Região com pupila
006_1_2	147,17	70	0	
006_1_3	145,35	70	0	
006_2_1	147,84	71	0	
006_2_2	148,84	66	0	
006_2_3	147,51	70	0	
006_2_4	146,91	64	0	
007_1_1	145,44	70	0	
007_1_2	150,09	66	0	
007_1_3	143,72	64	0	
007_2_1	141,89	63	0	
007_2_2	145	65	0	
007_2_3	144,32	74	-2	Região com pupila
007_2_4	143,83	72	0	
008_1_1	160,96	65	0	
008_1_2	161	63	0	
008_1_3	161,28	61	0	
008_2_1	160,86	62	0	
008_2_2	160,23	61	0	

008_2_3	159,37	64	0	
008_2_4	161,31	58	0	
009_1_1	158,68	47	0	
009_1_2	155,31	46	0	
009_1_3	155,46	47	0	
009_2_1	151,99	48	0	
009_2_2	155,24	45	0	
009_2_3	149,76	50	0	
009_2_4	151,24	43	0	
010_1_1	146,84	65	0	
010_1_2	144,91	66	0	
010_1_3	52,057	1	-1	Íris não detectada
010_2_1	154,08	58	0	
010_2_2	148,72	60	0	
010_2_3	152,11	56	0	
010_2_4	152,14	59	0	
011_1_1	137,91	60	0	
011_1_2	138,03	61	0	
011_1_3	136,07	58	0	
011_2_1	138,15	60	0	
011_2_2	138,18	59	0	
011_2_3	132,82	55	0	
011_2_4	132,31	55	0	
012_1_1	147,87	53	0	
012_1_2	145,36	59	0	
012_1_3	144,11	56	0	
012_2_1	145,19	60	0	
012_2_2	146,49	54	0	
012_2_3	143,64	54	0	
012_2_4	146,62	55	0	
013_1_1	148,5	67	0	
013_1_2	151,81	60	0	
013_1_3	152,39	63	0	
013_2_1	151,26	64	0	
013_2_2	155,61	62	0	
013_2_3	156,06	59	0	
013_2_4	151,56	55	0	
014_1_1	161,57	68	0	
014_1_2	163,61	67	0	
014_1_3	164,09	68	0	
014_2_1	162,06	70	0	
014_2_2	162,31	69	0	
014_2_3	163,68	62	0	
014_2_4	163,63	60	0	
015_1_1	141,65	72	0	
015_1_2	149,65	58	0	

015_1_3	145,08	59	0	
015_2_1	146,63	63	0	
015_2_2	144,82	63	0	
015_2_3	151,36	61	0	
015_2_4	149,38	63	0	
016_1_1	155,62	52	0	
016_1_2	157,11	50	0	
016_1_3	159,39	51	0	
016_2_1	160,14	55	0	
016_2_2	158,81	54	0	
016_2_3	160,57	54	0	
016_2_4	159,26	54	0	
017_1_1	155,32	52	0	
017_1_2	155,7	53	0	
017_1_3	153,65	57	0	
017_2_1	153,72	57	0	
017_2_2	153,2	54	0	
017_2_3	152,94	55	0	
017_2_4	153,29	54	0	
018_1_1	156,97	49	0	
018_1_2	148,33	49	0	
018_1_3	149,71	47	0	
018_2_1	160,95	50	0	
018_2_2	154,72	60	0	
018_2_3	154,77	47	0	
018_2_4	152,02	49	0	
019_1_1	148,67	89	-2	Região com pupila
019_1_2	151,45	76	-2	Região com pupila
019_1_3	152,44	78	-2	Região com pupila
019_2_1	151,87	76	-2	Região com pupila
019_2_2	154,04	74	-2	Região com pupila
019_2_3	151,73	77	-2	Região com pupila
019_2_4	150,5	78	-2	Região com pupila
020_1_1	147,75	60	0	
020_1_2	150,2	59	0	
020_1_3	148,26	64	0	
020_2_1	150,29	62	0	
020_2_2	151,23	61	0	
020_2_3	135,76	76	-2	Região com pupila
020_2_4	148,65	61	0	
021_1_1	156,64	67	0	
021_1_2	157,26	65	0	
021_1_3	158,27	64	0	
021_2_1	157,86	64	0	
021_2_2	157,63	65	0	
021_2_3	158,37	64	0	

021_2_4	160,17	63	0
022_1_1	145,61	58	0
022_1_2	151,15	58	0
022_1_3	151,85	59	0
022_2_1	152,14	57	0
022_2_2	150,23	66	0
022_2_3	151,01	63	0
022_2_4	155	63	0
023_1_1	153,52	59	0
023_1_2	148,12	57	0
023_1_3	152,59	55	0
023_2_1	150,21	59	0
023_2_2	151,36	57	0
023_2_3	151,52	58	0
023_2_4	156,19	56	0
024_1_1	160,99	63	0
024_1_2	160,01	66	0
024_1_3	159,09	64	0
024_2_1	161,51	61	0
024_2_2	161,72	59	0
024_2_3	160,02	60	0
024_2_4	155,61	56	0
025_1_1	153,72	61	0
025_1_2	152,78	60	0
025_1_3	152,72	62	0
025_2_1	153,18	61	0
025_2_2	149,81	61	0
025_2_3	152,03	61	0
025_2_4	151,69	62	0
026_1_1	152,67	62	0
026_1_2	151,93	61	0
026_1_3	151,97	62	0
026_2_1	152,35	63	0
026_2_2	149,31	60	0
026_2_3	151,3	60	0
026_2_4	150,93	60	0
027_1_1	155,23	65	0
027_1_2	155,19	69	0
027_1_3	161,23	67	0
027_2_1	154,14	64	0
027_2_2	157,19	65	0
027_2_3	154,62	65	0
027_2_4	156,86	64	0
028_1_1	154,84	55	0
028_1_2	152,28	56	0
028_1_3	157,1	53	0

028_2_1	154,86	56	0	
028_2_2	155,05	53	0	
028_2_3	156,63	53	0	
028_2_4	158,08	51	0	
029_1_1	149,85	62	0	
029_1_2	150,59	57	0	
029_1_3	148,33	58	0	
029_2_1	149,37	61	0	
029_2_2	151,81	62	0	
029_2_3	152,35	56	0	
029_2_4	151,9	58	0	
030_1_1	141,17	68	0	
030_1_2	55	0	-1	Íris não detectada
030_1_3	138,92	68	0	
030_2_1	143,83	60	0	
030_2_2	140,93	65	0	
030_2_3	144,29	57	0	
030_2_4	142,85	63	0	
031_1_1	146,86	64	0	
031_1_2	148,14	59	0	
031_1_3	147,99	57	0	
031_2_1	149,12	60	0	
031_2_2	148,7	59	0	
031_2_3	148,06	57	0	
031_2_4	148,65	58	0	
032_1_1	146,71	59	0	
032_1_2	141,99	56	0	
032_1_3	138,02	60	0	
032_2_1	139,14	58	0	
032_2_2	142,39	54	0	
032_2_3	139,27	53	0	
032_2_4	140,99	56	0	
033_1_1	144,42	59	0	
033_1_2	143,57	61	0	
033_1_3	143,6	53	0	
033_2_1	143,86	48	0	
033_2_2	144,71	60	0	
033_2_3	144,85	61	0	
033_2_4	147,65	58	0	
034_1_1	143,98	62	0	
034_1_2	144,31	66	0	
034_1_3	147,02	57	0	
034_2_1	145,6	58	0	
034_2_2	145,89	58	0	
034_2_3	146,35	60	0	
034_2_4	148,12	61	0	

035_1_1	152,69	61	0	
035_1_2	148,78	63	0	
035_1_3	151,34	59	0	
035_2_1	152,92	62	0	
035_2_2	152,72	59	0	
035_2_3	154,61	62	0	
035_2_4	151,92	63	0	
036_1_1	138,08	58	0	
036_1_2	138,11	55	0	
036_1_3	137,39	55	0	
036_2_1	136,05	58	0	
036_2_2	137,78	57	0	
036_2_3	139,67	59	0	
036_2_4	137,37	57	0	
037_1_1	155,88	60	0	
037_1_2	155,09	58	0	
037_1_3	154,73	60	0	
037_2_1	155,27	60	0	
037_2_2	156,82	61	0	
037_2_3	155,48	60	0	
037_2_4	155,03	60	0	
038_1_1	149,14	61	0	
038_1_2	147,34	57	0	
038_1_3	145,78	62	0	
038_2_1	148,1	61	0	
038_2_2	147,33	58	0	
038_2_3	148,64	60	0	
038_2_4	147,64	64	0	
039_1_1	143,35	67	0	
039_1_2	142,08	65	0	
039_1_3	142,95	64	0	
039_2_1	141,25	74	-2	Região com pupila
039_2_2	139,68	75	-2	Região com pupila
039_2_3	144,25	67	0	
039_2_4	142,06	71	0	
040_1_1	155,34	70	0	
040_1_2	59,254	3	-1	Íris não detectada
040_1_3	155,25	58	0	
040_2_1	156,48	56	0	
040_2_2	158,38	56	0	
040_2_3	155,94	58	0	
040_2_4	157,32	60	0	
041_1_1	146,68	69	0	
041_1_2	148,45	61	0	
041_1_3	147,27	61	0	
041_2_1	143,55	64	0	

041_2_2	146,59	66	0	
041_2_3	144,09	61	0	
041_2_4	146,83	54	0	
042_1_1	160,69	56	0	
042_1_2	158,57	53	0	
042_1_3	153,1	54	0	
042_2_1	158,8	52	0	
042_2_2	156,9	55	0	
042_2_3	156,89	51	0	
042_2_4	156,57	53	0	
043_1_1	145,23	83	-2	Região com pupila
043_1_2	144,91	76	-2	Região com pupila
043_1_3	148,36	72	0	
043_2_1	146,27	72	0	
043_2_2	149,22	66	0	
043_2_3	149,04	68	0	
043_2_4	151,2	60	0	
044_1_1	155,16	61	0	
044_1_2	158,97	57	0	
044_1_3	158,62	59	0	
044_2_1	158,05	59	0	
044_2_2	160,54	60	0	
044_2_3	158,37	59	0	
044_2_4	159,29	59	0	
045_1_1	147,54	53	0	
045_1_2	149,29	54	0	
045_1_3	146,13	56	0	
045_2_1	150,71	51	0	
045_2_2	155,31	51	0	
045_2_3	152,06	54	0	
045_2_4	154,79	48	0	
046_1_1	147,52	63	0	
046_1_2	145,92	63	0	
046_1_3	140,51	62	0	
046_2_1	140,2	62	0	
046_2_2	138,51	56	0	
046_2_3	139,18	60	0	
046_2_4	141,21	59	0	
047_1_1	143,12	68	0	
047_1_2	142,89	68	0	
047_1_3	142,07	65	0	
047_2_1	142,17	64	0	
047_2_2	142,62	66	0	
047_2_3	143,23	65	0	
047_2_4	142,28	66	0	
048_1_1	156,02	75	-2	Região com pupila

048_1_2	157,27	73	0	
048_1_3	155,63	74	-2	Região com pupila
048_2_1	157,42	70	0	
048_2_2	157,74	72	0	
048_2_3	154,65	66	0	
048_2_4	154,72	65	0	
049_1_1	149,72	54	0	
049_1_2	148,74	54	0	
049_1_3	147,48	56	0	
049_2_1	146,59	55	0	
049_2_2	148,03	52	0	
049_2_3	145,25	57	0	
049_2_4	148,61	55	0	
050_1_1	160,74	51	0	
050_1_2	156,04	48	0	
050_1_3	160,6	51	0	
050_2_1	159,35	51	0	
050_2_2	156,88	56	0	
050_2_3	156,71	53	0	
050_2_4	158	53	0	
051_1_1	162,23	55	0	
051_1_2	159,49	55	0	
051_1_3	160,82	58	0	
051_2_1	157,67	57	0	
051_2_2	159,73	59	0	
051_2_3	155,09	54	0	
051_2_4	160,96	58	0	
052_1_1	154,19	60	0	
052_1_2	156,25	62	0	
052_1_3	152,81	63	0	
052_2_1	154,4	62	0	
052_2_2	157,4	64	0	
052_2_3	158,89	61	0	
052_2_4	153,07	60	0	
053_1_1	143,8	57	0	
053_1_2	144,19	58	0	
053_1_3	142,29	56	0	
053_2_1	142,39	56	0	
053_2_2	144,05	58	0	
053_2_3	145,26	59	0	
053_2_4	144,57	54	0	
054_1_1	150,48	86	-2	Região com pupila
054_1_2	148,74	86	-2	Região com pupila
054_1_3	149,3	94	-2	Região com pupila
054_2_1	55,766	6	-1	Íris não detectada
054_2_2	149,96	79	-2	Região com pupila

054_2_3	150,02	82	-2	Região com pupila
054_2_4	151,92	85	-2	Região com pupila
055_1_1	159,06	61	0	
055_1_2	162,09	61	0	
055_1_3	162,91	45	0	
055_2_1	159,91	60	0	
055_2_2	156,21	57	0	
055_2_3	158,71	58	0	
055_2_4	157,56	60	0	
056_1_1	146,32	52	0	
056_1_2	148,02	59	0	
056_1_3	149,04	56	0	
056_2_1	149,65	57	0	
056_2_2	145,9	57	0	
056_2_3	147,04	54	0	
056_2_4	149,48	57	0	
057_1_1	147,03	52	0	
057_1_2	146,12	53	0	
057_1_3	148,26	54	0	
057_2_1	146,03	53	0	
057_2_2	145,89	50	0	
057_2_3	146,75	50	0	
057_2_4	148,17	53	0	
058_1_1	138,83	76	-2	Região com pupila
058_1_2	136,43	75	-2	Região com pupila
058_1_3	136,56	55	0	
058_2_1	134,92	51	0	
058_2_2	133,1	52	0	
058_2_3	131,88	52	0	
058_2_4	132,48	51	0	
059_1_1	146,93	64	0	
059_1_2	146,74	62	0	
059_1_3	147,59	62	0	
059_2_1	146,9	60	0	
059_2_2	145,71	62	0	
059_2_3	145,61	62	0	
059_2_4	146,03	62	0	
060_1_1	143,74	53	0	
060_1_2	143,62	54	0	
060_1_3	145,92	52	0	
060_2_1	144,53	50	0	
060_2_2	142,06	52	0	
060_2_3	142,31	51	0	
060_2_4	143,34	52	0	
061_1_1	154,72	49	0	
061_1_2	154,83	51	0	

061_1_3	153,76	49	0	
061_2_1	154,14	47	0	
061_2_2	153,1	50	0	
061_2_3	154,3	50	0	
061_2_4	151,23	49	0	
062_1_1	153,35	65	0	
062_1_2	149,97	70	0	
062_1_3	149,68	69	0	
062_2_1	153,63	60	0	
062_2_2	154,28	60	0	
062_2_3	154,34	61	0	
062_2_4	154,03	60	0	
063_1_1	154,11	50	0	
063_1_2	154,64	50	0	
063_1_3	152,34	59	0	
063_2_1	156,04	52	0	
063_2_2	156,15	52	0	
063_2_3	154,24	56	0	
063_2_4	157,97	55	0	
064_1_1	149,17	64	0	
064_1_2	151,15	64	0	
064_1_3	148,13	63	0	
064_2_1	149,83	59	0	
064_2_2	150,91	59	0	
064_2_3	150,6	60	0	
064_2_4	150,87	61	0	
065_1_1	141,36	68	0	
065_1_2	141,95	69	0	
065_1_3	139,18	65	0	
065_2_1	143,47	70	0	
065_2_2	142,85	69	0	
065_2_3	140,55	69	0	
065_2_4	140,27	69	0	
066_1_1	153,01	60	0	
066_1_2	153,59	59	0	
066_1_3	155,11	59	0	
066_2_1	154,69	61	0	
066_2_2	61,775	13	-1	Íris não detectada
066_2_3	157,85	58	0	
066_2_4	156,99	58	0	
067_1_1	148,47	65	0	
067_1_2	146,98	64	0	
067_1_3	148,12	65	0	
067_2_1	148,37	66	0	
067_2_2	146,18	72	0	
067_2_3	150,69	67	0	

067_2_4	148,09	66	0	
068_1_1	140,34	61	0	
068_1_2	144,75	66	0	
068_1_3	139,69	61	0	
068_2_1	143,76	64	0	
068_2_2	143,44	65	0	
068_2_3	136,15	71	0	
068_2_4	140,36	61	0	
069_1_1	156,31	48	0	
069_1_2	153,57	47	0	
069_1_3	150,73	50	0	
069_2_1	149,5	42	0	
069_2_2	152,88	45	0	
069_2_3	62,939	16	-1	Íris não detectada
069_2_4	152,41	46	0	
070_1_1	151,09	80	-2	Região com pupila
070_1_2	151,12	76	-2	Região com pupila
070_1_3	151,06	74	-2	Região com pupila
070_2_1	150,29	75	-2	Região com pupila
070_2_2	147,91	71	0	
070_2_3	157,81	64	0	
070_2_4	156,66	60	0	
071_1_1	134,12	52	0	
071_1_2	140,69	57	0	
071_1_3	137,93	59	0	
071_2_1	133,35	55	0	
071_2_2	132,61	54	0	
071_2_3	134,5	52	0	
071_2_4	139,69	59	0	
072_1_1	154,26	69	0	
072_1_2	156,9	63	0	
072_1_3	150,81	81	-2	Região com pupila
072_2_1	157,27	63	0	
072_2_2	153,14	66	0	
072_2_3	155,73	63	0	
072_2_4	154,18	65	0	
073_1_1	149,56	55	0	
073_1_2	150,78	49	0	
073_1_3	146,21	72	0	
073_2_1	150,9	59	0	
073_2_2	151,98	57	0	
073_2_3	151,46	57	0	
073_2_4	148,53	67	0	
074_1_1	158,25	73	0	
074_1_2	76,931	4	-1	Íris não detectada
074_1_3	155,31	76	-2	Região com pupila

074_2_1	154,4	78	-2	Região com pupila
074_2_2	153,44	75	-2	Região com pupila
074_2_3	159,01	68	0	
074_2_4	159,58	66	0	
075_1_1	152,52	56	0	
075_1_2	155,3	55	0	
075_1_3	156,13	56	0	
075_2_1	155,31	58	0	
075_2_2	158,35	58	0	
075_2_3	155,14	57	0	
075_2_4	154,19	59	0	
076_1_1	159,7	49	0	
076_1_2	153,5	60	0	
076_1_3	152,04	63	0	
076_2_1	157,62	53	0	
076_2_2	152,69	65	0	
076_2_3	160,92	55	0	
076_2_4	161,13	51	0	
077_1_1	148,76	44	0	
077_1_2	154,45	48	0	
077_1_3	142,61	44	0	
077_2_1	150,4	49	0	
077_2_2	150,15	51	0	
077_2_3	154,3	44	0	
077_2_4	150,59	46	0	
078_1_1	154,43	75	-2	Região com pupila
078_1_2	153,28	77	-2	Região com pupila
078_1_3	153,59	74	-2	Região com pupila
078_2_1	151,56	77	-2	Região com pupila
078_2_2	152,84	73	0	
078_2_3	151,26	76	-2	Região com pupila
078_2_4	150,3	75	-2	Região com pupila
079_1_1	150,43	60	0	
079_1_2	140,89	67	0	
079_1_3	148,21	58	0	
079_2_1	149,78	60	0	
079_2_2	135,35	67	0	
079_2_3	139,63	59	0	
079_2_4	140,77	60	0	
080_1_1	154,65	58	0	
080_1_2	156,98	63	0	
080_1_3	155,12	60	0	
080_2_1	156,85	60	0	
080_2_2	154,29	64	0	
080_2_3	153,06	54	0	
080_2_4	153,17	54	0	

081_1_1	149,36	60	0	
081_1_2	151,01	57	0	
081_1_3	150,18	57	0	
081_2_1	150,57	58	0	
081_2_2	150,78	59	0	
081_2_3	149,83	59	0	
081_2_4	148,93	57	0	
082_1_1	154,45	55	0	
082_1_2	156,28	57	0	
082_1_3	56,339	14	-1	Íris não detectada
082_2_1	155,15	54	0	
082_2_2	154,01	58	0	
082_2_3	153,68	55	0	
082_2_4	154,47	54	0	
083_1_1	159,8	50	0	
083_1_2	157,16	51	0	
083_1_3	66,046	60	-1	Íris não detectada
083_2_1	160,3	50	0	
083_2_2	156,28	49	0	
083_2_3	157,79	49	0	
083_2_4	159,26	52	0	
084_1_1	153,88	69	0	
084_1_2	156,03	58	0	
084_1_3	155,13	58	0	
084_2_1	157,65	59	0	
084_2_2	157,07	59	0	
084_2_3	153,36	65	0	
084_2_4	154,87	62	0	
085_1_1	151,22	59	0	
085_1_2	154,45	62	0	
085_1_3	153,67	61	0	
085_2_1	153,52	62	0	
085_2_2	153,65	63	0	
085_2_3	150,35	62	0	
085_2_4	149,57	62	0	
086_1_1	157,81	65	0	
086_1_2	156,41	66	0	
086_1_3	155,68	67	0	
086_2_1	154,54	68	0	
086_2_2	150,25	72	0	
086_2_3	154,43	64	0	
086_2_4	157,36	68	0	
087_1_1	153,4	53	0	
087_1_2	157,24	53	0	
087_1_3	153,25	51	0	
087_2_1	152,32	51	0	

087_2_2	152	51	0	
087_2_3	153,29	50	0	
087_2_4	155,24	50	0	
088_1_1	152,17	57	0	
088_1_2	147,29	55	0	
088_1_3	153,81	57	0	
088_2_1	151,09	58	0	
088_2_2	152,71	59	0	
088_2_3	151,74	59	0	
088_2_4	150,2	59	0	
089_1_1	151,25	56	0	
089_1_2	147,73	57	0	
089_1_3	146,29	54	0	
089_2_1	146,74	57	0	
089_2_2	147,85	57	0	
089_2_3	146,61	55	0	
089_2_4	149,75	53	0	
090_1_1	141,85	61	0	
090_1_2	140,49	65	0	
090_1_3	140,7	62	0	
090_2_1	140,72	61	0	
090_2_2	141,43	62	0	
090_2_3	140,45	64	0	
090_2_4	139,33	62	0	
091_1_1	138,16	74	-2	Região com pupila
091_1_2	138,03	74	-2	Região com pupila
091_1_3	136,79	81	-2	Região com pupila
091_2_1	139,75	73	0	
091_2_2	144,84	65	0	
091_2_3	139,62	71	0	
091_2_4	149,08	60	0	
092_1_1	143,67	64	0	
092_1_2	145,38	58	0	
092_1_3	144,16	62	0	
092_2_1	142,42	63	0	
092_2_2	144,06	62	0	
092_2_3	146,63	63	0	
092_2_4	143,5	63	0	
093_1_1	147,6	66	0	
093_1_2	148,68	65	0	
093_1_3	147,92	65	0	
093_2_1	148,25	61	0	
093_2_2	146,96	62	0	
093_2_3	146,51	61	0	
093_2_4	147,15	64	0	
094_1_1	61,839	10	-1	Íris não detectada

094_1_2	65,939	28	-1	Íris não detectada
094_1_3	149,17	58	0	
094_2_1	145,21	52	0	
094_2_2	145,82	56	0	
094_2_3	151,24	50	0	
094_2_4	152,32	45	0	
095_1_1	134,7	48	0	
095_1_2	142,86	61	0	
095_1_3	141,76	56	0	
095_2_1	143,22	61	0	
095_2_2	143,34	61	0	
095_2_3	142,71	59	0	
095_2_4	143,86	62	0	
096_1_1	135,46	48	0	
096_1_2	137,44	53	0	
096_1_3	136,58	50	0	
096_2_1	133,9	49	0	
096_2_2	137,18	48	0	
096_2_3	137,72	55	0	
096_2_4	135,4	53	0	
097_1_1	157,63	60	0	
097_1_2	154,8	62	0	
097_1_3	158,18	59	0	
097_2_1	57,3	2	-1	Íris não detectada
097_2_2	157,11	59	0	
097_2_3	57,507	5	-1	Íris não detectada
097_2_4	155,4	61	0	
098_1_1	147,52	56	0	
098_1_2	148,06	56	0	
098_1_3	147,21	58	0	
098_2_1	150,02	56	0	
098_2_2	147,15	57	0	
098_2_3	149,45	58	0	
098_2_4	146,7	60	0	
099_1_1	145,03	65	0	
099_1_2	146,83	63	0	
099_1_3	146,85	65	0	
099_2_1	142,29	76	-2	Região com pupila
099_2_2	139,81	72	0	
099_2_3	144,49	66	0	
099_2_4	142,5	67	0	
100_1_1	151,22	60	0	
100_1_2	150,29	61	0	
100_1_3	149,48	62	0	
100_2_1	150,62	59	0	
100_2_2	149,98	57	0	

100_2_3	150,83	59	0	
100_2_4	149,54	58	0	
101_1_1	139,99	63	0	
101_1_2	137,93	69	0	
101_1_3	140,97	69	0	
101_2_1	139,81	75	-2	Região com pupila
101_2_2	139,28	71	0	
101_2_3	139,94	78	-2	Região com pupila
101_2_4	139,71	64	0	
102_1_1	155,3	60	0	
102_1_2	159,94	59	0	
102_1_3	158,64	60	0	
102_2_1	158,66	59	0	
102_2_2	156,9	56	0	
102_2_3	156,37	58	0	
102_2_4	156,51	56	0	
103_1_1	161,12	52	0	
103_1_2	160,56	53	0	
103_1_3	160,05	49	0	
103_2_1	160,96	51	0	
103_2_2	158,56	49	0	
103_2_3	159,37	51	0	
103_2_4	160,74	51	0	
104_1_1	159,93	54	0	
104_1_2	152,15	52	0	
104_1_3	157,08	52	0	
104_2_1	145,06	54	0	
104_2_2	143,8	54	0	
104_2_3	157,49	61	0	
104_2_4	161,38	60	0	
105_1_1	143,54	49	0	
105_1_2	150,69	55	0	
105_1_3	147,4	53	0	
105_2_1	151,76	58	0	
105_2_2	148,94	51	0	
105_2_3	147,48	59	0	
105_2_4	150,02	56	0	
106_1_1	148,86	60	0	
106_1_2	145,88	57	0	
106_1_3	144,9	59	0	
106_2_1	143,45	58	0	
106_2_2	144,69	57	0	
106_2_3	142,99	57	0	
106_2_4	143,91	55	0	
107_1_1	138,98	65	0	
107_1_2	139,37	62	0	

107_1_3	139,94	62	0	
107_2_1	141,03	65	0	
107_2_2	144,53	84	-2	Região com pupila
107_2_3	144,12	86	-2	Região com pupila
107_2_4	142,63	65	0	
108_1_1	157,82	67	0	
108_1_2	157,99	68	0	
108_1_3	157,68	69	0	
108_2_1	159,04	67	0	
108_2_2	157,14	71	0	
108_2_3	159	68	0	
108_2_4	156,84	72	0	

Apêndice C – Código fonte

Na sequência são listados os códigos-fontes do algoritmo desenvolvido, seguindo a metodologia proposta no capítulo 5 da Dissertação. Primeiramente são listados os algoritmos programados na ferramenta Matlab, onde são analisadas as famílias *Wavelet* e calculada a Distância Euclidiana entre os vetores de características. Logo em seguida, o código fonte do algoritmo criado em Delphi para a interação com o usuário, e o cálculo dos resultados.

```
*** IRIS.M ***

function [D, qualidade, Q_Resultados] = Biometria_Iris(wname, nivel, w_ap, w_hz,
w_vt, w_dg, hist_equaliza, quali);
warning off all;
dir_name = 'D:\IRIS\';
files = dir([dir_name, 'iris*.bmp']);
im = {};
for i = 1:length(files)
    im{i}=files(i).name;
end;

tot_img = length(files);

bioiris = struct('caracteristica', [] );

%*****
% DECOMPOSICAO WAVELET PARA OS COMPONENTES E NÍVEL SELECIONADOS
%*****

warning off all;

qualidade = zeros(756,1);
q_resultado = zeros(756,2);
format short;
for i=1:tot_img
    % seleciona a img_original escolhida
    img = im{i};
    [img_read, map] = imread([dir_name, img]);

    % img = img_read(5:28, 49:232); % 44x64      75,00%
    % img = img_read(2:31, 40:237); % 32x198     77,09%
```

```

% img = img_read(2:32, 40:237); % 32x198 77,44%
% img = img_read(2:33, 40:237); % 32x198 77,69%
% img = img_read(2:33, 38:237); % 32x198 78,11%
% img = img_read(2:33, 18:237); % 32x198 80,18%
% img = img_read(3:33, 20:243); % 32x198 80,2647

% seleciona a parte 4 da iris segmentada - posição 1:64 a 192:255
img = img_read(3:33, 20:243); % 31x224

aux_carac = 0;

% equaliza o histograma da porção retirada
if hist_equaliza ~= 0
    img = adapthisteq(img); %80,18%
end;

% teste da qualidade da região de interesse
if quali ~= 0
    [qualidade(i), q_resultado(i,:)] = iris_qualidade(img);
else
    q_resultado(i,:) = [100, 50];
end;
v_ap = [];
v_hz = [];
v_vt = [];
v_dg = [];

% decompõem wavelet no nível selecionado
[C,S]=wavedec2(img, nivel, wname);
aux_carac = 0;

ap = [];
hz = [];
vt = [];
dg = [];

% componentes de aproximação Wavelet para o nível selecionado
if w_ap ~= 0
    ap=appcoef2(C,S,wname,nivel);
    ap=ap(:);
end;

% componentes de detalhes horizontais Wavelet para o nível selecionado
if w_hz ~= 0
    hz=detcoef2('h',C,S,nivel);
    hz=h(:);
end;

% componentes de detalhes verticais Wavelet para o nível selecionado
if w_vt ~= 0
    vt=detcoef2('v',C,S,nivel);
    vt=vt(:);
end;

% componentes de detalhes diagonais Wavelet para o nível selecionado
if w_dg ~= 0
    dg=detcoef2('d',C,S,nivel);
    dg=dg(:);
end;

% monta o vetor de características com os componentes Wavelet
bioiris(i).caracteristica = [ ap ; hz ; vt ; dg ];
end;
clear v_ap;
clear v_hz;
clear v_vt;
clear v_dg;

D = zeros(756,756);

%*****
% calculo da distancia euclidiana
%*****
for i = 1 : tot_img,
    distancia = zeros(1,756);
    for base = 1: tot_img,
        aux_dist = bioiris(i).caracteristica - bioiris(base).caracteristica;
    end;
end;

```

```

        distancia(base) = sqrt(aux_dist' * aux_dist);
    end;
    [valor, pos] = sort(distancia);
    for x = 1 : tot_img,
        D(i, x) = pos(x);
    end;
end;

D = D;
qualidade = qualidade;
Q_Resultados = q_resultado;
warning on all;

```

*** IRIS_QUALIDADE.M ***

```

function [v_qualidade, v_resultado] = Qualidade(img);
niv = 1;

[B,R]=wavedec2(img, niv, 'haar');

qualidade = 0;

% componentes de aproximação Wavelet de Haar no nível de decomposição 1
qap=appcoef2(B,R,'haar',niv);
qap=qap(:);
q1 = mean(mean(img));
q2 = round(std(qap));

% teste da qualidade da parte selecionada
% se a média geral da intensidade dos pixels da região de interesse é
% inferior ao limiar 80, significa que a iris não foi detectada corretamente,
% e essa porção refere-se a um nível de cinza que tende ao preto (zero),
% portanto, será descartada.

if q1 < 80
    qualidade = -1;
end;

% se o desvio padrão dos componentes de baixa frequência(aproximacao) for
% maior que o limiar 74, a região de interesse contém pálpebra e será
% descartada.
if q2 >= 74 % região contém pálpebra
    qualidade = -2;
end;

v_qualidade = qualidade;
v_resultado = [q1 ; q2];

```

*** IRIS_RECALL_PRECISION.M ***

```

v_qualidade = qualidade;
v_resultado = [q1 ; q2];
function recall_precision(search, D, Q, titulo);

warning off all;
dir_name = 'D:\IRIS\';
files = dir([dir_name, 'iris*.bmp']);
dist = {};

classes = 108;
img_classes = 7;
imagens = {};
for i = 1:756,
    for j = 1: 756,
        dist{i, j} = files(D(i,j)).name;
    end;
end;

aux = 1;
for i = 1:classes,

```

```

        for j = 1: img_classes,
            imagens{i,j}=files(aux).name;
            aux = aux + 1;
        end;
    end;

    aux = 1;
    for i = 1 : classes,
        for j = 1 : img_classes,
            for n = 1 : img_classes,
                c_imagens{aux,n}=imagens{i,n};
            end;
            aux = aux + 1;
        end;
    end;

    for aux = 1 : search,
        lin=1;
        for i = 1 : classes * img_classes,
            % acertos - apenas das imagens que tem qualidade, ou seja, Q(i) = 0
            if Q(i) == 0
                recall(lin,aux)= sum(strcmp(dist{lin,aux},c_imagens(lin,:)));
                precision(lin,aux)= sum(strcmp(dist{lin,aux},c_imagens(lin,:)));
                lin = lin+1;
            end;
        end;
    end;

    for aux = 2 : search,
        lin=1;
        for i = 1 : classes * img_classes,
            if Q(i) == 0
                recall(lin,aux)= recall(lin,aux) + recall(lin,aux-1);
                precision(lin,aux)= precision(lin,aux) + precision(lin,aux-1);
                lin = lin+1;
            end;
        end;
    end;

    recall = recall./img_classes;

    lin = 1;
    for i = 1 : classes * img_classes,
        if Q(i) == 0
            precision(lin,:)= precision(lin,:)/[ 1 : search ] ;
            lin = lin+1;
        end;
    end;

    m_recall = mean(recall);
    m_precision = mean(precision);

    figure,
    plot(m_recall,m_precision,'-bo','LineWidth',2 )
    title('Avaliação do Sistema de Reconhecimento da Textura da Iris Humana')
    xlabel('recall')
    ylabel('precision')
    legend(titulo)

    grid on;

```

*** UNT_IRIS.PAS ***

```
unit untIris;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, StdCtrls, Buttons, ExtCtrls, ComCtrls, ComObj, MLib_TLB, ActiveX,
Menus, imageenvirow, ievect, Mask, CheckLst, iemview, ImageEnIO, ievew,
```

```

ShellCtrls, TEdit, Math, StrUtils, clipbrd;

type
TfrmIris = class(TForm)
  GroupBox1: TGroupBox;
  CheckBox2: TCheckBox;
  CheckBox3: TCheckBox;
  CheckBox7: TCheckBox;
  CheckBox8: TCheckBox;
  ImageEnMView1: TImageEnMView;
  DirectoryListBox1: TShellTreeView;
  chk_recurso: TCheckBox;
  ImageEnMView2: TImageEnMView;
  btnRecallPrecision: TBitBtn;
  btnWavelet: TBitBtn;
  btnReconhece: TBitBtn;
  btnAcertoGeral: TBitBtn;
  iris_segmentada: TImage;
  Label1: TLabel;
  PageControl1: TPageControl;
  TabSheet1: TTabSheet;
  TabSheet2: TTabSheet;
  wavelet_mae: TRadioGroup;
  wavelet: TRadioGroup;
  niveis: TRadioGroup;
  GroupBox2: TGroupBox;
  edtAPROXIMACAO: TCheckBox;
  edtHORIZONTAL: TCheckBox;
  edtVERTICAL: TCheckBox;
  edtDIAGONAL: TCheckBox;
  edtEQUALIZA: TCheckBox;
  RadioGroup1: TRadioGroup;
  edtSEARCH: TEdit;
  Label2: TLabel;
  UpDown1: TUpDown;
  chk_salvar: TCheckBox;
  btnAbrir: TBitBtn;
  GroupBox3: TGroupBox;
  edtQuali1: TRealEdit;
  Label8: TLabel;
  edtQuali2: TRealEdit;
  Label9: TLabel;
  GroupBox4: TGroupBox;
  Label3: TLabel;
  Label4: TLabel;
  Label5: TLabel;
  Label6: TLabel;
  edtINICIO: TMaskEdit;
  edtTERMINO: TMaskEdit;
  edtTEMPOGASTO: TMaskEdit;
  edtacerto: TEdit;
  porc_acerto: TRealEdit;
  Label7: TLabel;
  txtResultado: TLabel;
  chkQUALIDADE: TCheckBox;
  procedure FormClose(Sender: TObject; var Action: TCloseAction);
  procedure FormCreate(Sender: TObject);
  procedure wavelet_maeClick(Sender: TObject);
  procedure ImageEnMView1BeforeImageDraw(Sender: TObject; idx, Left,
    Top: Integer; Canvas: TCanvas);
  procedure FormActivate(Sender: TObject);
  procedure ImageEnMView1ImageSelect(Sender: TObject; idx: Integer);
  procedure chk_recursoClick(Sender: TObject);
  procedure CheckBox2Click(Sender: TObject);
  procedure CheckBox3Click(Sender: TObject);
  procedure DirectoryListBox1Change(Sender: TObject; Node: TTreeNode);
  procedure CheckBox8Click(Sender: TObject);
  procedure CheckBox7Click(Sender: TObject);
  procedure btnWaveletClick(Sender: TObject);
  procedure btnRecallPrecisionClick(Sender: TObject);
  procedure btnReconheceClick(Sender: TObject);
  procedure btnAcertoGeralClick(Sender: TObject);
  procedure ImageEnMView1DbClick(Sender: TObject);
  procedure ImageEnMView2BeforeImageDraw(Sender: TObject; idx, Left,
    Top: Integer; Canvas: TCanvas);
  procedure btnAbrirClick(Sender: TObject);
  procedure iris_segmentadaDbClick(Sender: TObject);

```



```

private
  { Private declarations }
  procedure Reconhecimento(indice: Integer);
  procedure AcertoGeral(posicao, maximo: Integer);
  procedure LocalizaRegiaoDeInteresse;
  procedure SalvarProcessamento;
public
  Matlab: Variant;
  zr, zi: OleVariant;
  { Public declarations }
end;

const
  TOT_IMG = 756;
var
  frmIris: TfrmIris;
  Lista: TStringList;
  Distancia: array[0..755, 0..755] of Integer;
  Qualidade: array[0..755] of Integer;
  Resultado: array[0..755, 0..1] of Real; // matriz para os dois indices de qualidade

implementation

uses untZoom;

{$R *.dfm}

procedure TfrmIris.FormClose(Sender: TObject; var Action: TCloseAction);
begin
  MatLab := Unassigned;
end;

procedure TfrmIris.FormCreate(Sender: TObject);
begin
  Matlab := CreateOleObject('Matlab.Application');
  Matlab.Visible:=True;
end;

procedure TfrmIris.wavelet_maeClick(Sender: TObject);
var i: Integer;
begin
  wavelet.Items.Clear;
  wavelet.Columns:=1;
  case wavelet_mae.ItemIndex of
    0: wavelet.Items.Add('Haar');
    1: begin
        for i:=2 to 45 do
          wavelet.Items.Add('db'+IntToStr(i));
        wavelet.Columns:=3;
        end;
    2: begin
        for i:=2 to 45 do
          wavelet.Items.Add('sym'+IntToStr(i));
        wavelet.Columns:=3;
        end;
    3: begin
        for i:=1 to 5 do
          wavelet.Items.Add('coif'+IntToStr(i));
        end;
    4: begin
        wavelet.Items.Add('bior1.1');
        wavelet.Items.Add('bior1.3');
        wavelet.Items.Add('bior1.5');
        wavelet.Items.Add('bior2.2');
        wavelet.Items.Add('bior2.4');
        wavelet.Items.Add('bior2.6');
        wavelet.Items.Add('bior2.8');
        wavelet.Items.Add('bior3.1');
        wavelet.Items.Add('bior3.3');
        wavelet.Items.Add('bior3.5');
        wavelet.Items.Add('bior3.7');
        wavelet.Items.Add('bior3.9');
        wavelet.Items.Add('bior4.4');
        wavelet.Items.Add('bior5.5');
        wavelet.Items.Add('bior6.8');
        wavelet.Columns:=2;
        end;
  end;
end;

```

```

5: begin
    wavelet.Items.Add('rbio1.1');
    wavelet.Items.Add('rbio1.3');
    wavelet.Items.Add('rbio1.5');
    wavelet.Items.Add('rbio2.2');
    wavelet.Items.Add('rbio2.4');
    wavelet.Items.Add('rbio2.6');
    wavelet.Items.Add('rbio2.8');
    wavelet.Items.Add('rbio3.1');
    wavelet.Items.Add('rbio3.3');
    wavelet.Items.Add('rbio3.5');
    wavelet.Items.Add('rbio3.7');
    wavelet.Items.Add('rbio3.9');
    wavelet.Items.Add('rbio4.4');
    wavelet.Items.Add('rbio5.5');
    wavelet.Items.Add('rbio6.8');
    wavelet.Columns:=2;
end;
6: wavelet.Items.Add('dmey');
end;
wavelet.ItemIndex:=0;
end;

procedure TfrmIris.ImageEnMView1BeforeImageDraw(Sender: TObject; idx, Left,
Top: Integer; Canvas: TCanvas);
begin
    ImageEnMView1.ImageInfoText[idx].Background:=clYellow;
end;

procedure TfrmIris.FormActivate(Sender: TObject);
begin
    ImageEnMView1.EnableAlphaChannel := true;
    ImageEnMView1.SoftShadow.Enabled := true;
    ImageEnMView1.FillThumbnail := false;
    ImageEnMView1.ShowText := True;
    DirectoryListBox1Change(self, nil);
    Lista:=TStringList.Create;
    PageControl1.ActivePage:=TabSheet1;
end;

procedure TfrmIris.ImageEnMView1ImageSelect(Sender: TObject; idx: Integer);
var img: String;
    bmp: TBitmap;
begin
    img:='D:\iris\iris_'+extractfilename(ImageEnMView1.ImageFileName[idx]);
    bmp:=TBitmap.Create;
    bmp.LoadFromFile(img);
    iris_segmentada.Canvas.Draw(0,0,bmp);
    FreeAndNil(bmp);
    LocalizaRegiaoDeInteresse;
    ImageEnMView2.Clear;
    edtQuali1.Value:=Resultado[idx,0];
    edtQuali2.Value:=Resultado[idx,1];
    txtResultado.Caption:='Com Qualidade';
    if edtQuali1.Value < 80 then
        txtResultado.Caption:='Íris Não Detectada';
    if edtQuali2.Value >= 74 then
        txtResultado.Caption:='Região com Pupila';
end;

function xcompare(i1, i2: Integer): integer;
var
    s1, s2: string;
begin
    with Form1.ImageEnMView1 do
        begin
            s1 := ImageFileName[i1];
            s2 := ImageFileName[i2];
        end;
        result := AnsiCompareStr(s1, s2);
    end;
end;

procedure TfrmIris.chk_recursoClick(Sender: TObject);
var
    sr:TSearchRec;
    Found,idx:integer;
    fpath:string;

```

```

procedure ListaDiretorio(Dire:String);
var
  sr: TSearchRec;
  Found: integer;
  idx: integer;
  fpath: string;
begin
  // testar as irises segmentadas
  // Dire:='D:\IRIS';
  Found:=FindFirst(Dire+'*.*',faArchive+faDirectory ,sr);

  if Found=0 then begin
    while Found=0 do begin
      fpath:=Dire+'\' +sr.Name;
      if (chk_recurso.Checked) and (sr.Name <> '.') and (sr.Name <> '..') and
        ((Sr.Attr and faDirectory) = Sr.Attr) then
        ListaDiretorio(fpath);
      if IsKnownFormat(fpath) then
        begin
          idx:=ImageEnMView1.AppendImage;
          ImageEnMView1.ImageFileName[idx]:=fpath;
          ImageEnMView1.ImageBottomText[idx].Caption:='['+IntToStr(idx+1)+']' -
            '+extractfilename(fpath);
          ImageEnMView1.ImageBottomText[idx].Background:=$00EFEFD0;
        end;
      Found:=FindNext(sr);
    end;
  end;
  FindClose(sr);
end;

begin
  ImageEnMView1.Clear;
  ImageEnMView1.LockPaint;
  ListaDiretorio(DirectoryListBox1.Root);
  ImageEnMView1.UnlockPaint;
  ImageEnMView1.SelectedImage:=0;
  ImageEnMView1.Sort(xcompare);
end;

procedure TfrmIris.CheckBox2Click(Sender: TObject);
begin
  ImageEnMView1.SoftShadow.Enabled := CheckBox2.Checked;
  ImageEnMView1.Update;
end;

procedure TfrmIris.CheckBox3Click(Sender: TObject);
begin
  if CheckBox3.Checked then
    begin
      ImageEnMView1.Style := iemsFlat;
      ImageEnMView1.FillThumbnail := false;
    end
  else
    begin
      ImageEnMView1.Style := iemsACD;
      ImageEnMView1.FillThumbnail := true;
    end;
end;

procedure TfrmIris.DirectoryListBox1Change(Sender: TObject; Node: TTreeNode);
begin
  chk_recursoClick(Self);
end;

procedure TfrmIris.CheckBox8Click(Sender: TObject);
begin
  ImageEnMView1.ThumbnailsInternalBorder:=CheckBox8.Checked;
end;

procedure TfrmIris.CheckBox7Click(Sender: TObject);
begin
  if CheckBox7.Checked then
    ImageEnMView1.ThumbsRounded := 4
  else
    ImageEnMView1.ThumbsRounded := 0;
  ImageEnMView1.Update;
end;

```

```

end;

procedure TfrmIris.btnWaveletClick(Sender: TObject);
var wname, variavel: string;
    Tamanho, i, j: Integer;
    w_ap, w_hz, w_vt, w_dg, w_eq, w_qa: String;
    inicio, termino: TTime;
    retorno: Variant;
begin
    screen.Cursor:= crSQLWait;
    Beep;
    Matlab.Execute('clear');
    Lista.Clear;
    edtINICIO.Text:='';
    edtINICIO.Refresh;
    edtTERMINO.Text:='';
    edtTERMINO.Refresh;
    edtTEMPOGASTO.Text:='';
    edtTEMPOGASTO.Refresh;
    w_ap:=BoolToStr(edtAPROXIMACAO.Checked);
    w_hz:=BoolToStr(edtHORIZONTAL.Checked);
    w_vt:=BoolToStr(edtVERTICAL.Checked);
    w_dg:=BoolToStr(edtDIAGONAL.Checked);
    w_eq:=BoolToStr(edtEQUALIZA.Checked);
    w_qa:=BoolToStr(chkQUALIDADE.Checked);
    Variavel := '[D, qualidade, Q] = iris(''+wavelet.Items[wavelet.ItemIndex]+
        ''', '+IntToStr(niveis.ItemIndex+1)+'+', '+w_ap+', '+w_hz+', '+w_vt+', '+w_dg+', '+
        w_eq+', '+w_qa+'');';
    inicio:=Time;
    edtINICIO.Text:=TimeToStr(inicio);
    edtINICIO.Refresh;
    Matlab.Execute(variavel);

    Tamanho:=755;
    zr := VarArrayCreate([0, Tamanho, 0, Tamanho], varDouble);
    zi := VarArrayCreate([0, Tamanho, 0, Tamanho], varDouble);
    Matlab.GetFullMatrix('D', 'base', VarArrayRef(zr), VarArrayRef(zi));
    for i := 0 to Tamanho do
        for j := 0 to Tamanho do
            Distancia[i,j]:=zr[i, j];

    zr := VarArrayCreate([0, Tamanho], varDouble);
    zi := VarArrayCreate([0, Tamanho], varDouble);
    Matlab.GetFullMatrix('qualidade', 'base', VarArrayRef(zr), VarArrayRef(zi));
    for i := 0 to Tamanho do
        Qualidade[i]:=zr[i];

    zr := VarArrayCreate([0, Tamanho, 0, 1], varDouble);
    zi := VarArrayCreate([0, Tamanho, 0, 1], varDouble);
    Matlab.GetFullMatrix('Q', 'base', VarArrayRef(zr), VarArrayRef(zi));
    for i := 0 to Tamanho do
        for j := 0 to 1 do
            Resultado[i,j]:=zr[i, j];

    termino:=Time;
    edtTERMINO.Text:=TimeToStr(termino);
    edtTEMPOGASTO.Text:=TimeToStr(termino-inicio);
    if chk_salvar.Checked then
        SalvarProcessamento;
    btnRecallPrecision.Enabled:=True;
    btnReconhece.Enabled:=True;
    btnAcertoGeral.Enabled:=True;
    Beep;
    screen.Cursor:= crDefault;
    btnAcertoGeral.Click;
end;

procedure TfrmIris.btnRecallPrecisionClick(Sender: TObject);
var Tamanho, nivel, i, j: Integer;
    Variavel: String;
    retorno: Variant;
begin
    screen.Cursor:= crSQLWait;
    Beep;
    Variavel:='ativo=exist(''+QuotedStr('D')+'')';
    retorno:=Matlab.Execute(Variavel);
    Variavel:=VarToStr(retorno);

```

```

if Variavel = '' then begin
    Tamanho:=755;
    zr := VarArrayCreate([0, Tamanho, 0, Tamanho], varDouble);
    zi := VarArrayCreate([0, Tamanho, 0, Tamanho], varDouble);
    for i := 0 to Tamanho do begin
        for j := 0 to Tamanho do begin
            zr[i, j]:=Distancia[i,j];
            zi[i, j]:=0;
        end;
    end;
    MatLab.PutFullMatrix('D', 'base', VarArrayRef(zr), VarArrayRef(zi));
end;

Variavel:='ativo=exist('+QuotedStr('quali')+');';
retorno:=Matlab.Execute(Variavel);
Variavel:=VarToStr(retorno);
if Variavel = '' then begin
    Tamanho:=755;
    zr := VarArrayCreate([0, Tamanho], varDouble);
    zi := VarArrayCreate([0, Tamanho], varDouble);
    for i := 0 to Tamanho do begin
        zr[i]:=Qualidade[i];
    end;
    MatLab.PutFullMatrix('quali', 'base', VarArrayRef(zr), VarArrayRef(zi));
end;

Variavel:='iris_recall_precision('+edtSEARCH.Text+', D, quali, '+QuotedStr(
    'Wavelet' +wavelet.Items[wavelet.ItemIndex]+' -
'+niveis.Items[niveis.ItemIndex])+');';
Matlab.Execute(Variavel);
Beep;
screen.Cursor:= crDefault;
end;

procedure TfrmIris.Reconhecimento(indice: Integer);
var
    idx:integer;
    fpath:string;
    i,j : Integer;
    ind: Integer;
begin
    ImageEnMView2.Clear;
    ImageEnMView2.LockPaint;
    for i:=0 to 755 do begin
        ind:=Distancia[indice,i]-1;
        fpath:=ImageEnMView1.ImageFileName[ind];
        idx:=ImageEnMView2.AppendImage;
        ImageEnMView2.ImageFileName[idx]:=fpath;
        if Qualidade[ind] = 0 then
            ImageEnMView2.ImageBottomText[idx].Caption:=extractfilename(fpath)
        else
            ImageEnMView2.ImageBottomText[idx].Caption:=extractfilename(fpath)+'*';
        end;
    end;
    ImageEnMView2.UnlockPaint;
    ImageEnMView2.SelectedImage:=0;
    ImageEnMView2.Sort(xcompare);
end;

procedure TfrmIris.btnReconheceClick(Sender: TObject);
var quali: Integer;
begin
    AcertoGeral(ImageEnMView1.SelectedImage, ImageEnMView1.SelectedImage);
    quali:=Qualidade[ImageEnMView1.SelectedImage];

    if quali <> 0 then begin
        ShowMessage('Imagem
'+extractfilename(ImageEnMView1.ImageFileName[ImageEnMView1.SelectedImage])+
        '
' Sem Qualidade Para Processamento !'+#10+IntToStr(quali));
        exit;
    end;
    Reconhecimento(ImageEnMView1.SelectedImage);
end;

procedure TfrmIris.AcertoGeral(posicao, maximo: Integer);
var
    i, j: Integer;
    total, acerto: Integer;

```

```

    dist_orig, dist_procura: Integer;
    tot: Integer;
    arq1, arq2: String;
begin
    screen.Cursor:= crSQLWait;
    acerto:=0;
    total:=0;
    for i:=posicao to maximo do begin
        if Qualidade[i] = 0 then begin
            j:=0;
            tot:=6;
            dist_orig:=Distancia[i,0]-1;
            dist_orig:=(dist_orig div 7);
            while (j <= tot) and (tot <= 755) do begin
                dist_procura:=Distancia[i,j]-1;
                if Qualidade[dist_procura] = 0 then begin
                    Inc(total);
                    dist_procura:=(dist_procura div 7);
                    if (dist_orig = dist_procura) and (j < 7) then
                        Inc(acerto);
                end
            end;
            else
                Inc(tot);
                Inc(j);
            end;
        end;
    end;
    if total > 0 then begin
        edtacerto.Text:=IntToStr(acerto)+'/'+IntToStr(total);
        porc_acerto.Value:=(acerto/total)*100;
    end;
    ImageEnMView2.UnlockPaint;
    ImageEnMView2.SelectedImage:=0;
    ImageEnMView2.Sort(xcompare);
    screen.Cursor:= crDefault;
end;

procedure TfrmIris.btnAcertoGeralClick(Sender: TObject);
begin
    AcertoGeral(1, Length(Distancia));
end;

procedure TfrmIris.LocalizaRegiaoDeInteresse;
var lin_ini, lin_fim, col_ini, col_fim, i: Integer;
begin
    with iris_segmentada do begin
        lin_ini:=1;
        lin_fim:=64;
        col_ini:=1;
        col_fim:=64;

        for i:=0 to 7 do begin
            Picture.Bitmap.IgnorePalette:=True;
            Canvas.Pen.Width := 1;
            Canvas.Pen.Color:= clRed;
            Canvas.Brush.Style:=bsClear;
            Canvas.Rectangle(col_ini, lin_ini, col_fim, lin_fim);
            col_ini:=col_ini+64;
            col_fim:=col_fim+64;
        end;
        Canvas.Pen.Width := 2;
        Canvas.Pen.Color:= clYellow;
        Canvas.Brush.Style:=bsClear;
        Canvas.Rectangle(lin_fim, lin_ini, col_fim, col_ini);
    end;
end;

procedure TfrmIris.ImageEnMView1DbClick(Sender: TObject);
var
    bmp: TBitmap;
    img: string;
    idx: Integer;
begin
    idx:=ImageEnMView1.SelectedImage;
    bmp:=TBitmap.Create;
    bmp.LoadFromFile(ImageEnMView1.ImageFileName[ImageEnMView1.SelectedImage]);
    formZoom.iris.Canvas.Draw(0,0,bmp);
end;

```

```

formZoom.iris.Stretch:=True;
formZoom.ShowModal;
FreeAndNil(bmp);
ImageEnMView1.SelectedImage:=idx;
end;

procedure TfrmIris.ImageEnMView2BeforeImageDraw(Sender: TObject; idx, Left,
Top: Integer; Canvas: TCanvas);
begin
  if Pos('*', ImageEnMView2.ImageBottomText[idx].Caption) <> 0 then
    begin
      ImageEnMView2.ImageBottomText[idx].Background:=clRed;
      ImageEnMView2.ImageTopText[idx].Background:=clRed;
      ImageEnMView2.ImageTopText[idx].Caption:='Iris Não Localizada';
    end
  else begin
    if
Copy(extractfilename(ImageEnMView1.ImageFileName[ImageEnMView1.SelectedImage]),1,3)
<>
      Copy(extractfilename(ImageEnMView2.ImageFileName[idx]),1,3) then
      ImageEnMView2.ImageBottomText[idx].Background:=clBlue
    else
      ImageEnMView2.ImageBottomText[idx].Background:=clYellow;
    end;
  end;
end;

procedure TfrmIris.SalvarProcessamento;
var
  F1, F2, F3: TextFile;
  i, j, tam: Integer;
  Lista: TStringList;
begin
  AssignFile(F1, 'iris_distancia.iris');
  Rewrite(F1);
  tam:=Length(Distancia)-1;
  Lista:=TStringList.Create;
  for i:=0 to tam do begin
    Lista.Clear;
    for j:=0 to tam do begin
      Lista.Add(Format('%3.3d',[Distancia[i,j]]));
    end;
    Writeln(F1, Lista.CommaText);
  end;
  CloseFile(F1);

  AssignFile(F2, 'iris_qualidade.iris');
  Rewrite(F2);
  tam:=Length(Qualidade)-1;
  for i:=0 to tam do begin
    Lista.Clear;
    Lista.Add(Format('%3d',[Qualidade[i]]));
    Writeln(F2, Lista.CommaText);
  end;
  CloseFile(F2);

  AssignFile(F3, 'iris_resultado.iris');
  Rewrite(F3);
  tam:=Length(Resultado)-1;
  for i:=0 to tam do begin
    Lista.Clear;
    for j:=0 to 1 do begin
      Lista.Add(FloatToStr(Resultado[i,j]));
    end;
    Writeln(F3, Lista.CommaText);
  end;
  FreeAndNil(Lista);
  CloseFile(F3);
end;

procedure TfrmIris.btnAbrirClick(Sender: TObject);
var
  F1, F2, F3: TextFile;
  Linha: String;
  i, j, tam, dist, pos1, pos2: Integer;
  Lista: TStringList;
begin
  // Abrir arquivo com a última distância Euclidiana calculada

```

```

AssignFile(F1, 'iris_distancia.iris');
Reset(F1);
tam:=Length(Distancia)-1;
i:=0;
Lista:=TStringList.Create;
while not Eof(F1) do begin
  Readln(F1, linha);
  Lista.CommaText:=Linha;
  for j:=0 to tam do
    Distancia[i,j]:=StrToInt(Lista[j]);
  Inc(i);
end;
CloseFile(F1);

AssignFile(F2, 'iris_qualidade.iris');
Reset(F2);
tam:=Length(Qualidade)-1;
Lista.Clear;
i:=0;
while not Eof(F2) do begin
  Readln(F2, linha);
  Lista.CommaText:=Linha;
  Qualidade[i]:=StrToInt(Lista[0]);
  inc(i);
end;
CloseFile(F2);

AssignFile(F3, 'iris_resultado.iris');
Reset(F3);
tam:=Length(Resultado)-1;
i:=0;
Lista.Clear;
while not Eof(F3) do begin
  Readln(F3, linha);
  Lista.CommaText:=Linha;
  for j:=0 to 1 do
    Resultado[i,j]:=StrToFloat(Lista[j]);
  Inc(i);
end;
CloseFile(F3);
btnRecallPrecision.Enabled:=True;
btnReconhece.Enabled:=True;
btnAcertoGeral.Enabled:=True;
end;

procedure TfrmIris.iris_segmentadaDbClick(Sender: TObject);
var
  imgWindow: TBitmap;
  recorta: TBitmap;
  d,s:TRect;
begin
  imgWindow := GetFormImage;
  recorta := TBitmap.Create;
  recorta.width := 512;
  recorta.Height := 64;

  s.Left:=360;
  s.Top:=24;
  s.Right:=513;
  s.Bottom:=88;
  d.Top:=24;
  d.Left:=360;
  d.Right:=512;
  d.Bottom:=64;
  try
    recorta.Canvas.CopyRect(d, imgWindow.Canvas, s);
    Clipboard.Assign(recorta);
  finally
    imgWindow.Free;
    recorta.free;
  end;
  beep;
end;

end.

```