



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

PATRICK FRANCIS GOMES ROCHA

APLICATIVO PARA RECONHECIMENTO DE FACES HUMANAS

2012



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"
Assis**

PATRICK FRANCIS GOMES ROCHA

APLICATIVO PARA RECONHECIMENTO DE FACES HUMANAS

Trabalho de Conclusão do Programa de Iniciação Científica (PIC) do Instituto Municipal de Ensino Superior de Assis– IMESA.

Aluno: Patrick Francis Gomes Rocha

Orientadora: Profa. Dra. Marisa Atsuko Nitto

Linha de Pesquisa: Informática

**2012
Assis, SP**

FICHA CATALOGRÁFICA

ROCHA, Patrick Francis Gomes

Aplicativo Para Reconhecimento de Faces Humanas/ Patrick Francis Gomes Rocha. Fundação Educacional do Município de Assis – Fema : Assis, 2012

47p.

Orientadora: Profa. Dra. Marisa Atsuko Nitto
Projeto de Iniciação Científica (PIC) – Ciência da Computação - Instituto Municipal de Ensino Superior de Assis

1. Java 2. Reconhecimento de padrões. 3. Biometria

CDD: 001.6
Biblioteca da FEMA

RESUMO

O reconhecimento facial é uma das mais promissoras das tecnologias biométricas e tem sido um dos temas mais estudados nas áreas da visão computacional, inteligência artificial e alternativa de segurança. A biometria já é bastante utilizada, sendo que o padrão biométrico mais conhecido é a digital. Contudo, o padrão biométrico facial vem sendo muito estudado e pesquisado para a utilização comercial, pois não é necessário contato físico.

A finalidade deste projeto de pesquisa foi desenvolver um aplicativo para detecção de faces humanas, para foi utilizada a biblioteca OPENCV com JAVACV. Será feita a integração deste aplicativo de detecção de faces com outro aplicativo de tratamento de imagens. A execução deste projeto foi dividida em três etapas, sendo que duas delas foram concluídas (tratamento das imagens e detecção de faces). Este protótipo será utilizado como base para o desenvolvimento da terceira etapa que é reconhecimento de faces.

LISTA DE FIGURAS

Figura 1 – Alguns tipos de padrões biométricos	14
Figura 2 – Sistemas biométricos padrão	16
Figura 3 – resultados obtidos pelo método da escala cinza	19
Figura 4 – Detecções nas imagens com fundos simples	20
Figura 5 – Detecções nas imagens com fundos complexos.....	21
Figura 6 – Aplicação do método fuzzy	22
Figura 7 – Visão geral do sistema com relações geométricas do triângulo ...	24
Figura 8 – Resultados do sistema das relações geométricas do triângulo.....	25
Figura 9 – Tipos de reconhecimento	29
Figura 10 – Arquitetura de um sistema biométrico	30
Figura 11 – Tipos de padrões biométricos	31
Figura 12 – Arquitetura Java	32
Figura 13 – Modelagem do problema	35
Figura 14 – Modelo para armazenamentos e tratamento de imagem	37
Figura 15 – Interface principal do aplicativo	37
Figura 16 – Interface de cadastro	39
Figura 17 – Interface de busca avançada	40
Figura 18 – Interface de processamento de imagens	41
Figura 19 – Interface de detecção de faces	43

LISTA DE TABELAS

Tabela 1 – Padrões Biométricos para a identificação	13
---	----

SUMÁRIO

1 – INTRODUÇÃO	09
1.1 – OBJETIVO	11
1.2 – JUSTIFICATIVA	11
1.3 – MOTIVAÇÃO	11
1.4 – ESTRUTURA DO TRABALHO	12
2 – FUNDAMENTAÇÃO TEÓRICA BÁSICA	12
2.1 – RECONHECIMENTO DE PADRÕES	12
2.1.1 – Sistemas biométricos	15
2.2 – DETECÇÃO DE FACES	17
2.2.1 – Métodos para detecção de faces humanas.....	18
2.2.1.1 – Escala de cinza	18
2.2.1.2 – Mapa de arestas.....	20
2.2.1.3 –Cor.....	21
2.2.1.4 – Geometria da face.....	22
2.3 – BIBLIOTECA OPENCV E JAVACV.....	25
2.4 – BIOMETRIA.....	27
2.4.1 – Tipos de Reconhecimento	27
2.4.1.1 – Autenticação (um pra um)	30
2.4.1.2 – Identificação ou Busca Exaustiva (um para vários)	30
2.5 – LINGUAGEM JAVA	31
2.5.1 – Características da Linguagem	32

2.6 – BANCO DE DADOS HSQLDB	34
2.6.1 – Características do Banco	34
2.7 – BIBLIOTECA OPENCV E JAVACV	35
3 – DESENVOLVIMENTO DO PROJETO	35
3.1 – DESCRIÇÃO DO PROBLEMA	35
3.2 – MODELAGEM DO PROBLEMA	35
3.3 – IMPLEMENTAÇÃO	36
3.3.1 – Aplicativo de armazenamento e tratamento de imagens	36
3.3.2 – Aplicativo para detecção de faces	42
4 – CONCLUSÃO	43
REFERÊNCIAS	44

1 – INTRODUÇÃO

A visão computacional permite aos sistemas digitais informações de imagens. Diversas informações podem ser extraídas, as quais podem ser utilizadas para o reconhecimento de padrões complexos, como texturas, objetos, textos, padrões biométricos, entre outros (JAIN; DUIN; MAO, 2000). Essa capacidade de reconhecer padrões proporciona aos sistemas digitais um sistema de visão artificial ainda não tão eficiente quanto o humano, mas que já possui algumas características superiores, como visão noturna e *zoom*. Diversos sistemas fazem uso da visão computacional. Eles podem ser encontrados na robótica, onde a visão computacional permite que robôs decidam seus próprios movimentos, e no uso militar, permitindo que aviões de guerra e mísseis acertem seus alvos com maior precisão. Sistemas de reconhecimento de padrões biométricos são largamente utilizados na área de segurança. Esses sistemas utilizam características humanas singulares, como impressões digitais, íris, voz e face, permitindo a diferenciação entre seres humanos.

A face humana é o principal atributo, através da qual as pessoas são reconhecidas. Este fato sugere o desenvolvimento de sistemas automáticos de detecção, rastreamento e reconhecimento facial e promovem aplicações como autenticação de usuários, monitoramento de ambientes, controle de acesso, vigilância (NASCIMENTO; GONZAGA, 2008).

O reconhecimento de faces humanas é um problema bastante complexo para ser implementado de forma automática, devido a diversos fatores como: diferentes variações de orientação e tamanho da imagem, condições de iluminação do ambiente, diferenças na aparência, na expressão facial e na cor da pele, entre outros fatores que influenciam a extração de características (SUNG; POGGIO, 1998). Pesquisas também vêm sendo desenvolvidas por cientistas da computação no campo de reconhecimento de expressões faciais, para o emprego em interfaces homem x máquina. Sistemas capazes de interpretar expressões de raiva, tristeza ou alegria de seus usuários teriam grande aplicabilidade nos estudos de interação Homem x Computador (TIAN et al., 2001). Esse tipo de técnica exige algoritmos robustos para poderem

reconhecer as faces em diferentes tipos de condições de iluminação, expressões faciais e outros empecilhos que dificultam o processo.

A área de análise de faces pode ser dividida em diversas subáreas, como reconhecimento de face, detecção ou localização de face, reconhecimento de expressões faciais e análise de poses (ZHAO et al., 2003).

É importante diferenciar detecção e reconhecimento. O reconhecimento de face consiste em identificar um indivíduo por intermédio da análise de sua face, comparando-a com outras faces pré-rotuladas. A detecção ou localização de faces é a determinação da presença e posição espacial de cada face existente em uma imagem. Realizar a detecção antes de realizar a busca pelas características poupa muito trabalho e tempo, uma vez que a maioria das implementações baseia-se na procura de tais elementos em toda a imagem (LOPES, 2009).

A vantagem de se detectar a face, em um primeiro momento, é que após a fase, a procura pelas características fica limitada apenas em uma determinada região da imagem, a região detectada. A detecção de face frequentemente é utilizada como uma etapa inicial para o reconhecimento.

Os problemas da detecção de características esbarram nas mesmas dificuldades vistas na detecção de objetos. Detectar uma característica consiste em: dada uma cena encontrar onde está a face, ou várias faces (LUO e ELEFTHERIADIS, 2000).

Para o desenvolvimento do aplicativo para a detecção de faces foi utilizado OpenCV (*Open Source Computer Vision*) que é uma biblioteca de programação, de código aberto, desenvolvida inicialmente pela Intel Corporation. Este aplicativo de detecção de faces será integrado a um aplicativo de tratamento de imagens desenvolvido no projeto de iniciação científica no ano de 2011 (ROCHA, 2011; ROCHA e NITTO, 2011).

A implementação do aplicativo foi desenvolvido utilizando tecnologias Java com banco de dados HSQLDB (*Hypersonic SQL Database*).

1.1 - OBJETIVO

O objetivo neste projeto de pesquisa é adquirir habilidades em desenvolvimento de detecção de faces humanas, utilizando tecnologias sobre visão computacional, processamento digital de imagens, aprendizagem automática, OPENCV e JAVACV. Neste protótipo serão utilizadas técnicas e métodos amplamente divulgados na literatura, as quais têm a finalidade de manipular e processar imagens capturadas por uma câmera de forma automática.

1.2 - JUSTIFICATIVA

O desenvolvimento deste projeto deu-se pelo fato da biometria ser um dos métodos mais eficazes para a segurança nos tempos de hoje, com isso a tecnologia vem crescendo e a demanda para o mercado de trabalho na área da segurança, ser cada vez mais prospera principalmente na área de tecnologia da informação. A escolha do padrão facial biométrico vem pelo fato de ser um dos mais eficazes, entre os tipos de *padrões biométricos*. Ele vem sendo muito estudado e pesquisado, sendo padrão biométrico usado “como protótipo” na próxima Copa do Mundo de 2014, aqui no Brasil, e também nas olimpíadas de 2016, aqui no Brasil, no Rio de Janeiro.

Outro ponto significativo para a escolha deste tipo de biometria foi a gama de possibilidades de pesquisa e estudos, aprofundando em outras áreas, como armazenamento de imagens em banco de dados, processamento digital de imagens, entre outros.

1.3 - MOTIVAÇÃO

A motivação para desenvolver este projeto de pesquisa, consiste no fato da biometria ser uma área que está em crescimento. E o reconhecimento facial é um dos padrões utilizados na biometria. Tendo em vista, que o projeto foi

iniciado no segundo ano do curso bacharelado em ciência da computação, a preocupação inicial foi adquirir conhecimento em processamento digital de imagens e banco de dados. Este enfoque é de suma importância, pois servirá de base para o desenvolvimento de aplicativos nesta área. A possibilidade de adquirir conhecimento sobre tecnologias Java também contribuiu para o interesse no desenvolvimento deste projeto.

1.4 – ESTRUTURA DO TRABALHO

A estrutura do trabalho foi dividida em capítulos:

Capítulo 1: Introdução

Capítulo 2: Fundamentação Teórica Básica

Capítulo 3: Desenvolvimento do Projeto

Capítulo 4: Conclusão

Capítulo 5: Referências Bibliográficas

2 - FUNDAMENTAÇÃO TEÓRICA BÁSICA

Neste capítulo será abordada toda a fundamentação teórica das tecnologias e ferramentas utilizadas para o desenvolvimento do projeto.

2.1 - RECONHECIMENTO DE PADRÕES

O Reconhecimento de Padrões (RP) é o estudo das classificações de objetos em categorias ou classes. Há tempos existem pesquisas feitas para a implementação e elaboração de algoritmos que resolvam esses estudos. (SERRANO, 2010).

Esta área apresenta inúmeras opções de aplicações em varias áreas, principalmente na área tecnológica. Como exemplo, pode ser citada a classificação de documentos na *internet* (buscadores), sendo classificados por semântica, por categorias; classificação de dados multimídia (por exemplo, o *Windows Media Player*), que classifica as mídias por gênero, título, autor; e classificações biométricas, que são características físicas ou comportamentais, sendo individuais e pessoais que são capturadas de cada indivíduo e usadas também como forma classificatória. A tabela 1 mostra alguns tipos de reconhecimento de padrões.

Domínio do Problema	Aplicação	Padrão de Entrada	Classes de Padrão
Bioinformática	Análise de Sequência	DNA/Sequência de proteínas	Tipos conhecidos de genes/padrões
Mineração de dados	Busca por padrões significantes	Pontos em um espaço multidimensional	Compactar e bem separar grupos
Classificação de documentos	Busca na Internet	Documento texto	Categorias semânticas (negócios, esportes e etc.)
Análise de documentos de imagem	Máquinas de leitura para cego	Documento de imagem	Palavras e caracteres alfanuméricos
Automação industrial	Inspeção de circuito impresso em placas	Intensidade ou alcance de imagem	Produto defeituoso/não defeituoso
Recuperação de base de dados multimídia	Busca na Internet	Vídeo clipe	Gêneros de vídeos
Reconhecimento biométrico	Identificação pessoal	Face, íris, impressão digital	Usuários autorizados para controle de acesso
Sensoriamento remoto	Prognóstico da produção de colheita	Imagem multiespectral	Categorias de aproveitamento de terra, desenvolvimento de padrões de colheita
Reconhecimento de voz	Inquérito por telefone sem assistência de operador	Voz em forma de onda	Palavras faladas

Tabela 1 – Padrões Biométricos para Identificação (Adaptado de SERRANO, 2010)

Os sistemas de reconhecimento, usando padrões biométricos, vêm alavancando e motivando pesquisadores para estudo de algoritmos e formulas, para a melhora na logica e desempenho destes sistemas. Conseqüentemente iniciou-se um novo estudo de padrões biométricos, chamado computação afetiva, que dá a um computador a capacidade de reconhecer e expressar emoções e empregar mecanismos que contribuem para a tomada de decisão racional (JAIN; DUIN; MAO, 2000).

Para um ser humano reconhecer algo, é muito simples e automático. O individuo captura características e compara com características armazenadas no cérebro, sendo assim retorna e identifica que tipo de objeto esta tratando, ou observando. Computacionalmente é da mesma maneira, captura-se características, do individuo em questão, armazena-se em um banco de dados todas as características, e ao receber outras características, realiza comparações, retornando satisfatório se coincidirem as características ou insatisfatório se não coincidirem as características. A figura 1 mostra alguns tipos de padrões.

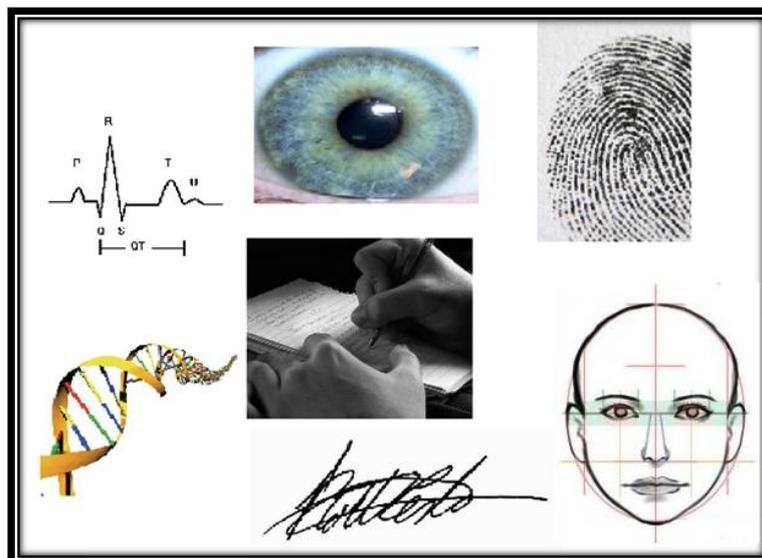


Figura 1 – Alguns tipos de padrões biométricos

2.1.1 Sistemas Biométricos

Um sistema biométrico é, essencialmente, um sistema de reconhecimento de padrões que busca extrair o padrão mais distintivo de uma pessoa, armazená-lo para depois comparar com novas amostras e determinar a identidade de cada amostra dentro de uma população (CANEDO, 2011). Assim sendo, um sistema biométrico típico tem pelo menos cinco componentes principais:

- **componente de apresentação e captura de dados biométricos:** o traço biométrico precisa ser apresentado a um sensor, que irá transformar a informação em uma representação digital (foto, vídeo, áudio, etc);
- **componente de processamento do dado biométrico e extração do *template*:** esse componente transforma a representação digital do traço biométrico em um *template*. Normalmente isso é feito através da segmentação da representação digital, avaliação (e melhoria) da qualidade e extração de características únicas. O processo usado para extração é muitas vezes patenteado e sempre mantido em segredo pelas empresas;
- **componente de armazenamento do *template*:** o armazenamento do *template* é um componente que apresenta crescente desafio e muitas opções de implementação. Os templates podem ser armazenados de três formas em sistemas biométricos: no sensor ou computador local, repositório central e cartão.
- **componente de comparação de *templates* e decisão:** na comparação de *templates* os valores matemáticos das características extraídas dos traços biométricos são comparados para determinar o seu grau de correlação ou similaridade, num processo chamado de *matching* (comparação);
- **canal de transmissão:** o canal de transmissão é um componente comum a todos os outros sistemas de informação. E assim como em

outros sistemas, nos sistemas biométricos ele é muito importante para definir a arquitetura do sistema, a política de segurança e privacidade.

A figura 2 mostra um sistema biométrico padrão.

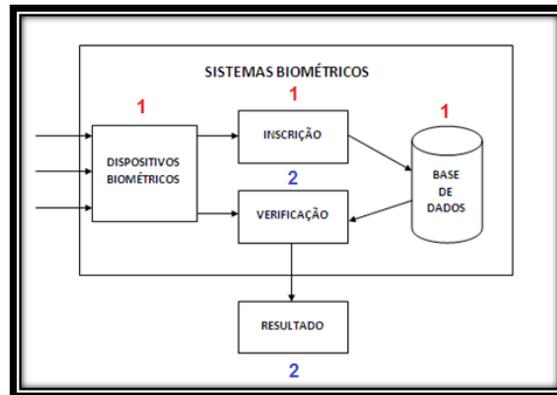


Figura 2 – Sistema Biométrico Padrão

Esse mesmo sistema tem dois processos importantes: processo de cadastro e tratamento de imagens e o processo de identificação. Na figura 4, o processo de cadastro e tratamento de imagens pode ser identificado pelo número 1 e o processo de identificação pelo número 2.

Os dispositivos biométricos são sensores que capturam e digitalizam a informação biométrica. Para que a pessoa faça parte do sistema, inicialmente a informação é processada na etapa de inscrição (ou registro), por uma ou mais vezes, e armazenada na base de dados. Na etapa de verificação, a mesma pessoa deve fornecer sua informação Biométrica através dos mesmos dispositivos biométricos para que seja comparada com a armazenada na base de dados e, assim, possa fornecer o resultado (identificação).

2.2 – DETECÇÃO DE FACES

Um das tarefas que devem ser realizadas na maioria dos Sistemas de Reconhecimento de Faces (SRF) são detectar a presença da face em uma determinada imagem. Detectar a face antes de detectar cada característica em particular poupa muito trabalho, uma vez que a maioria dos algoritmos se baseia na procura por tais elementos em toda a imagem. A vantagem de se detectar a face, em um primeiro momento, é que após esta fase a procura pelas características fica limitada apenas em uma determinada região da imagem.

Como em qualquer problema a serem resolvidas, as técnicas de detecção de faces, apesar de suas diferentes abordagens, esbarram em alguns problemas comuns (YANG et al., 2002). Esses problemas são:

- **Pose:** as imagens de face variam de acordo com a posição da câmera que registrou a imagem;
- **Expressão Facial:** a expressão da face influencia diretamente na aparência da imagem de face;
- **Presença de Elementos Estruturais:** a presença de elementos como barba, bigode e óculos que podem modificar as características em termos de tamanho, luminosidade, etc...;
- **Ocultação:** no caso de imagens feitas em ambientes não controlados as faces podem aparecer, parcial ou totalmente sobrepostas, por objetos ou até mesmo por outras faces.

Na tarefa de processamento de faces humanas, o problema de detecção de faces é um dos mais importantes a serem solucionados. A detecção de faces merece especial estudo, uma vez que é o pré-processamento necessário para as áreas de reconhecimento automático e análise de expressões faciais.

A localização de faces humanas em imagens digitais é uma tarefa importante em diversas aplicações. A indexação e a recuperação de imagens de vídeo contendo atividades humanas requerem a detecção automática da localização das faces dentro da cena. As técnicas que reconhecem faces ou analisam expressões faciais também requerem conhecimento sobre a sua localização dentro da imagem. Aplicações em tempo real, tal como a visão de robôs, devido à necessidade de resposta rápida, podem utilizar técnicas de detecção de faces humanas para que se possam tomar decisões num curto espaço de tempo (BRAS FILIPE, 2008).

2.2.1 – Métodos para detecção de faces humanas

Os métodos de detecção de faces humanas baseiam-se em algumas informações prévias. Alguns métodos de detecção de faces usam a informação de imagens em nível de cinza. Outros métodos utilizam as informações das arestas contidas numa imagem. Existe um número considerável de métodos que usam a informação de cor para detectar faces. Também existem métodos que utilizam a informação da geometria da face para detectar faces.

2.2.1.1 – *Escala de cinza*

Os métodos baseados em escala de cinza usam características pré-definidas da imagem, tanto para treinar o sistema como para criar o modelo. O método examina pequenas subjanelas da imagem e decide que cada janela contém uma face, usando redes neuronais artificiais (ROWLEY et al., 1998).

Ao ser usado o algoritmo de *bootstrap* durante o treino da rede neuronal, este irá adicionar imagens que não contêm faces no conjunto de treino, eliminando assim a difícil tarefa manual de selecionar os exemplos de não caras no treino. O sistema pode detectar entre 77.9% e 90.3% das faces num conjunto de 130 imagens, com um número aceitável de falsas detecções. Embora projetado para detectar faces frontais, a rede pode ser também treinada para detectar

faces de perfil. A figura 3 mostra alguns resultados obtidos pelo método proposto por Rowley et al.



Figura 3 – Resultados obtidos pelo método da escala cinza (ROWLEY et al, 1998)

Os métodos de Viola e Jones são sistemas de detecção de faces capaz de processar imagens de uma forma rápida, alcançando uma taxa de detecção bastante aceitáveis. Este método é um dos mais usados e citados pela comunidade científica (VIOLA e JONES, 2004).

Há três contribuições fundamentais a introdução de uma nova representação de imagem, chamada imagem integral, que permite que as características usadas pelo detector sejam processadas através de uma única passagem pela

imagem. A segunda é um classificador simples e eficiente que é construído usando o *AdaBoost*. Este algoritmo seleciona um pequeno número de características visuais críticas de um conjunto muito grande de potenciais características. A terceira consiste num método para combinar os classificadores numa cascata, que permite descartar as regiões de fundo da imagem de forma rápida, focando-se assim apenas nas regiões de maior interesse.

2.2.1.2 – Mapa de arestas

Para detectar faces humanas através do mapa de arestas, primeiramente terá que se detectar as arestas e, a seguir, relacioná-las com um modelo de face para verificar as corretas detecções (HJELMAS e LOW, 2001).

O método baseado na informação do formato da face. A imagem de entrada é realçada por meio da equalização do histograma, de seguida, é realiza uma detecção de arestas utilizando o filtro da mediana. As arestas extraídas são ligadas, usando um método baseado numa função de energia. O contorno da face é depois extraído utilizando a informação da direção da ligação das arestas (WANG e TAN, 2000). As figuras 4 e 5 mostram, respectivamente, as detecções nas imagens com fundos simples e fundos complexos.



Figura 4 – Detecções nas imagens com fundos simples (BRAS FILIPE, 2008)

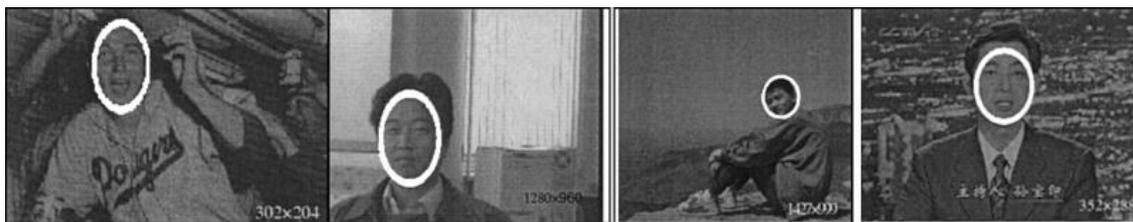


Figura 5 – Detecções nas imagens com fundos complexos (BRAS FILIPE, 2008)

2.2.1.3 – Cor

Um número considerável de técnicas utiliza a informação da cor para detectar faces. Essas técnicas, primeiro, selecionam as regiões de imagem mais prováveis de serem faces e só depois as tentam detectar nas regiões selecionadas, usando padrões faciais.

Este método isola a região próxima do laranja no espaço de cor YIQ como região semelhante à pele humana e eliminaram as regiões remanescentes (DAI e NAKANO, 1996). A partir daí, empregaram características de textura em imagens com nível de cinza para identificar faces nas regiões da pele. No primeiro teste foram selecionadas aleatoriamente 10 pessoas diferentes numa base de dados de faces e foi realizado um conjunto de cinco ou seis testes por pessoa. Nestes seus testes, estavam incluídas faces com rotação, inclinação e diferentes expressões. As taxas de corretas detecções verificadas foram de 98%. Contudo, este sistema não pode detectar faces com oclusão parcial, faces usando óculos e faces de perfil.

Outro método para detectar faces em imagens coloridas é baseado na teoria *fuzzy*. Esse método trabalha com dois modelos *fuzzy*: um para descrever a cor da pele e outro para descrever a cor do cabelo, utilizando um espaço de cor percentual para aumentar a precisão do método (CHEN; WU e YACHIDA, 1999). Foi criado um modelo para extrair as regiões da cor da pele e outro para extrair as regiões da cor do cabelo. Comparando estes dois modelos com um

modelo de *head-shape*, e utilizando o método de fusão do modelo baseado na teoria *fuzzy*, tentam-se detectar as faces candidatas.

No teste realizado pelos autores foi usada uma base de dados com 233 faces, em que 186 eram faces asiáticas e as outras caucasianas. O tamanho das faces varia entre 20 x 24 a 200 x 240 pixels. O índice de acerto nas detecções foi de 97% em imagens com tamanho de faces superiores a 50 x 60 pixels. As falhas desse método ocorrem devido a fatores, tais como: a variação da iluminação, a oclusão facial, as faces adjacentes (se duas ou mais caras estiverem muito próximas, os modelos que descrevem a cor da pele e do cabelo podem ser fundidos, resultando numa forma bem diferente de uma única cabeça) e o estilo do cabelo (por exemplo, faces com um penteado especial, faces de pessoas carecas ou usando chapéus).

A figura 6 mostra o resultado da aplicação do método baseado em *fuzzy*.



Figura 6 – Aplicação do método *fuzzy* (CHEN; WU e YACHIDA, 1999)

2.2.1.4 – Geometria da face

Em muitas técnicas de detecção de face, o conhecimento da geometria da face tem sido empregue para caracterizar e, posteriormente, verificar várias características faciais nos seus estados de incerteza (HJELMAS e LOW, 2001).

Nesse sistema, tentam-se estabelecer as possíveis localizações dos olhos em imagens binarizadas. Para cada possível par de olhos, o algoritmo irá fazer uma busca à procura de um nariz, uma boca e sobrancelhas. Cada característica facial tem uma função de avaliação associada, que é utilizada para determinar a face candidata.

Em 2001, Lin e Fan propõem a detecção de faces humanas utilizando as relações geométricas do triângulo. Este sistema possui duas partes principais. A primeira consiste na procura de regiões que possam possuir faces. A segunda executa a verificação da face (LIN e FAN, 2001).

A primeira parte do processo é constituída por quatro passos:

- Ler a imagem e convertê-la para uma imagem binária;
- Rotular os quatro componentes conectados na imagem para formarem vários blocos e encontrar o centro de cada um deles;
- Detectar quaisquer três centros, de três diferentes blocos, para formarem um triângulo isóscele (imagem frontal) ou um triângulo retângulo (imagem de perfil);
- Agrupar os blocos que satisfazem o critério de triângulo como possível face.

A segunda parte é constituída de três etapas:

- Normalizar o tamanho de todas as possíveis regiões;
- Fornecer a cada região de possível face normalizada um peso na função máscara;
- Executar a verificação por *thresholding* do peso obtido na etapa anterior.

A figura 7 mostra uma visão do sistema para detecção de faces utilizando as relações geométricas do triângulo.

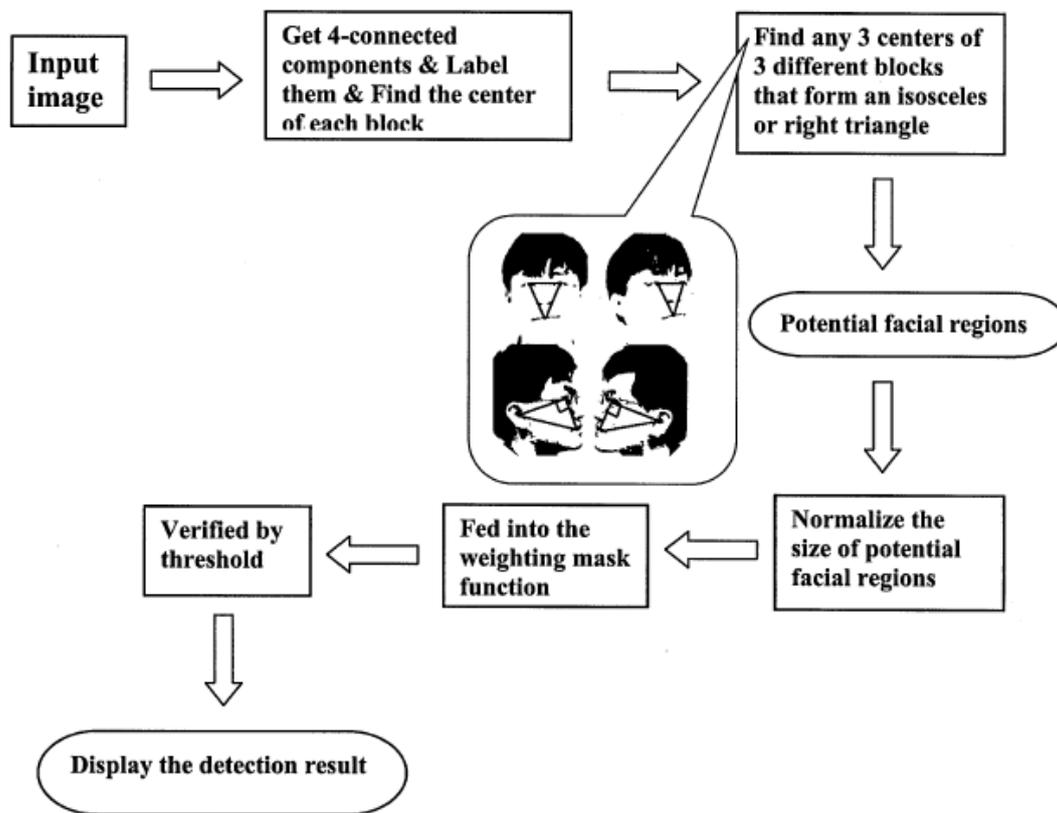


Figura 7 – Visão geral do sistema com relações geométricas do triângulo (LIN e FAN, 2001)

Este método pode lidar com diferentes tamanhos de faces nas imagens, diferentes condições de iluminação, ruído, problema de desfocagem, variação de pose e de expressão. O sistema pode também detectar faces de perfil, faces com problema de oclusão parcial da boca e faces com óculos de sol.

A figura 8 mostra alguns resultados obtidos pelo método das relações geométricas do triângulo.



Figura 8 – Resultados do sistema das relações geométricas do triângulo (LIN e FAN, 2001)

2.3 – BIBLIOTECA OPENCV E JAVACV

A biblioteca OpenCV foi desenvolvida pela Intel e possui mais de 500 funções (WILSON, 2012). Foi idealizada com o objetivo de tornar a visão computacional acessível a usuários e programadores em áreas tais como a interação humana-computador em tempo real e a robótica. A biblioteca está disponível com o código fonte e os executáveis (binários) otimizados para os processadores Intel. Um programa OpenCV, ao ser executado, invoca automaticamente uma DLL (*Dynamic Linked Library*) que detecta o tipo de processador e carrega, por sua vez, a DLL otimizada para este. Juntamente com o pacote OpenCV é oferecida a biblioteca IPL (*Image Processing Library*), da qual a OpenCV depende parcialmente, além de documentação e um conjunto de códigos exemplos. A biblioteca foi desenvolvida nas linguagens de programação C/C++, sendo que a primeira versão foi lançada no final de 2006 e a 2.0 foi lançada em setembro de 2009.

A biblioteca está dividida em cinco grupos de funções:

- Processamento de imagens;
- Análise estrutural;
- Análise de movimento e rastreamento de objetos;
- Reconhecimento de padrões;
- Calibração de câmera e reconstrução 3D.

De acordo com (MARENGONI e STRINGHINI, 2009) as principais características da biblioteca OpenCV são:

- Manipulação de dados de imagem (alocação, release, cópia configuração, conversão);
- Imagem e vídeo I / O (ficheiro e câmera de entrada com base imagem, / saída de arquivo de vídeo);
- Matrix e rotinas vetor álgebra linear e de manipulação (produtos, solvers, eigenvalues, SVD);
- Diversas estruturas de dados dinâmicas (listas, filas, conjuntos, árvores, gráficos);
- Processamento de imagem básico (filtragem, detecção de bordas, detecção de canto, de amostragem e interpolação, conversão de cores, operações morfológicas, histogramas, pirâmides de imagem);
- Análise estrutural (componentes conectados, contorno de processamento, distância transformar, vários momentos, combinando modelo, Hough transformar, aproximação poligonal, montagem linha, montagem elipse, a triangulação de Delaunay);
- Câmara de calibração (encontrar e rastrear os padrões de calibração, calibração, estimativa da matriz fundamental, a estimativa homografia, correspondência estéreo);
- De análise de movimento (fluxo óptico, segmentação de movimento, tracking).
- Objeto de reconhecimento (eigen-métodos, HMM);

- Básicos GUI (exibição de imagem / vídeo, teclado e manipulação de mouse, scroll-bars);
- Rotulagem de imagem (linha, cônica, polígono de desenho, texto).

Já a biblioteca JavaCV, é uma biblioteca de *link*, onde o usuário deve ter uma versão da OpenCV em sua máquina e um JAR (Java Archive, ou arquivo em Java) para fazer a programação usando os algoritmos da biblioteca da OpenCV em Java, esta ideia foi amplamente utilizada neste projeto.

2.4 – BIOMETRIA

A Biometria é um estudo que está dentro do Reconhecimento de Padrões, é o termo utilizado para designar a ciência que estuda as características físicas ou comportamentais dos seres vivos, a fim de que estes possam ser manuseados (SERRANO, 2010).

Na área de segurança da informação, é o estudo para a utilização de métodos automatizados para identificação de pessoas com base em suas características biométricas (físicas ou aspectos comportamentais) a fim de identificá-las e abolir o uso de senhas ou cartões magnéticos (MAGALHÃES, SANTOS, 2003).

A Biometria tem a característica de dispensar o uso de qualquer objeto, ou que memorize algo para identificar-se. Partindo do princípio de que cada indivíduo tem sua característica biométrica única, a identificação do indivíduo é feita através dessas características em questão.

2.4.1 – Tipos de Reconhecimento

Os sistemas biométricos realizam o reconhecimento de forma automática e pessoal de duas maneiras: a Autenticação e a Identificação (conhecida como Busca Exaustiva).

2.4.1.1 - Autenticação (um para um)

É um tipo de comparação que é feito a partir da imagem processada com somente uma imagem de dentro do banco. Este processo é muito utilizado para pontos eletrônicos, pelo qual o usuário antes de realizar a comparação digita seu código. Ao digitar seu código, se previamente cadastrado no sistema, já retornará todos seus dados, inclusive a imagem a ser comparada. Em seguida, o usuário fornece seu padrão biométrico (digital, face, íris, entre outros) capturado por algum dispositivo, esse padrão que o usuário forneceu, passa por todo processamento e é comparado somente com o padrão biométrico do código que ele forneceu (somente com o seu próprio padrão).

2.4.1.2 - Identificação ou Busca Exaustiva (um para vários)

É o tipo de comparação em que o usuário não fornece nenhum código antes da comparação, fornecendo somente o padrão biométrico. A partir daí, realiza-se uma busca entre o padrão biométrico fornecido, com todos os existentes e armazenados dentro do banco. A busca é feita de duas formas: insatisfatório (se chegar ao fim e não encontrar nenhuma característica do padrão fornecido, com os salvos em banco) ou satisfatório (se encontrar alguma característica semelhante).

A figura 9 mostra os dois tipos de reconhecimento.



Figura 9 – Tipos de Reconhecimento (VALOIS e NUNES, 2007)

O processo de um sistema biométrico consiste em duas fases principais: a Coleta de Dados (ou registro) e o Reconhecimento. A coleta de dados é um processo minucioso que não pode haver erros, pois será com os dados coletados e armazenados no banco que serão feitas as comparações. A coleta é feita com o preenchimento dos dados do usuário (nome, idade, sexo, telefone), juntamente com a captura do padrão biométrico feita por algum sensor. A fase do Reconhecimento é feita, quando o usuário passar outra vez pelo sistema e se identificar com seu padrão biométrico, fazendo a comparação com o padrão armazenado em banco.

A figura 10 ilustra a arquitetura de um sistema biométrico.

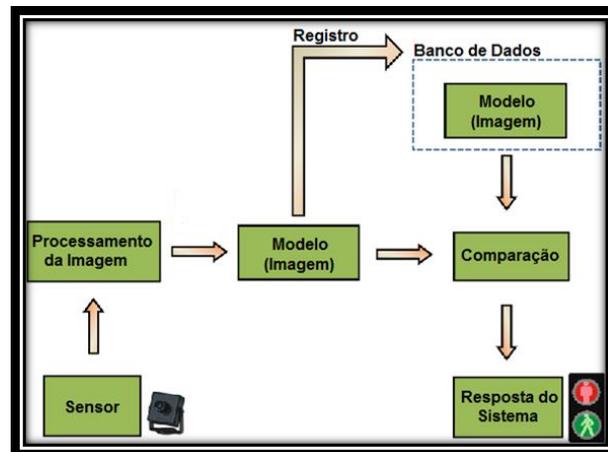


Figura 10 – Arquitetura de um sistema Biométrico

Teoricamente, quaisquer características humanas, comportamentais ou físicas, podem ser utilizadas para a identificação de pessoas, desde que satisfaçam os cinco principais requisitos:

- **Universalidade:** significa que todas as pessoas devem possuir a característica;
- **Unicidade:** indica que a característica não pode ser igual em pessoas diferentes;
- **Permanência:** significa que a característica não deve variar com o tempo;
- **Coleta:** indica que a característica pode ser medida quantitativamente;
- **Aceitação:** significa que o padrão biométrico a ser usado pelo sistema, deve ser de fácil captura e aceitação pelos usuários.

Existe também outra característica, que devido à evolução tecnológica, ela vem sendo vista como mais um requisito: a *Proteção*, referindo-se que o padrão biométrico e o sistema a serem utilizados, dificultem a forma de fraudar. Na prática nenhum padrão, atende com perfeição todos os requisitos, mas

consegue se resultados positivos e vem sendo muito utilizado para dar mais segurança, em todos os setores que o utilizam.

Dentre as características biométricas existentes algumas são mais utilizadas devido ao melhor custo/benefício e também pela facilidade de utilização devido a vários aplicativos no mercado. A figura 11 mostra um gráfico dos tipos de padrões biométricos mais utilizados.

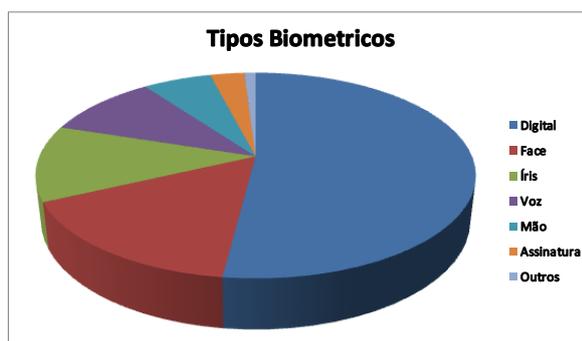


Figura 11 – Tipos de Padrões Biométricos (SERRANO, 2010).

2.5 - LINGUAGEM JAVA

Desenvolvida por uma equipe de programadores na década de 90 e chefiada por James Gosling, da empresa Sun Microsystems, Java é uma linguagem orientada a objeto, tendo uma sintaxe similar a linguagem C/C++, e diferente de outras linguagens convencionais, que são compiladas para código nativo “.exe”. A linguagem Java é compilada para um bytecodes “.class” que é totalmente interpretado e em seguida executado pela máquina virtual, chamada de *Java Virtual Machine* (JVM). A linguagem é executada em mais de 850 milhões de computadores pessoais e em bilhões de dispositivos por todo mundo, bem como, *smartphones*, telefones celulares e dispositivos embarcados.

A figura 12 mostra a arquitetura da linguagem Java.



Figura 12 – Arquitetura Java

A arquitetura ou plataforma Java tem basicamente dois componentes: a JVM e a interface de programação de aplicações (*API – Application Programming Interface*). A JVM emula o ambiente computacional e é a responsável pela portabilidade da linguagem, pois, uma vez instalado na máquina, pode-se executar qualquer sistema em Java, independente de sistema operacional. Os “*bytecodes*” são uma espécie de código *assembler* para a JVM. Este código é otimizado pela JVM, que o interpreta, gerando e passando ao *hardware* em que esta instalada, os comandos necessários. Por isso que se diz que os sistemas Java são compilados e interpretados (GRADVOHL, 2008).

Existe ainda a possibilidade de executar junto ao código Java códigos nativos, como mostra a figura 14, mais esse tipo de programação não é muito recomendado, pois tiraria a principal característica Java que é a *portabilidade*.

2.5.1 - Características da Linguagem

Segundo (H.M, 2003), a linguagem Java foi arquitetada tendo em vista as seguintes características e vantagens:

- **Orientação a objeto:** baseado no modelo de Simula67;
- **Portabilidade:** independência de plataforma, *writeonce, run anywhere*, ou seja, independe de sistema operacional, implementou-se uma vez, roda em qualquer lugar;
- **Recursos de Rede:** possui inúmeras bibliotecas que interagem por cooperação com protocolos TCP/IP, como HTTP e FTP;
- **Segurança:** pode executar programas via rede com restrições de execução.

Além dessas quatro principais características da linguagem, existem outras que não deixam de ser menos importantes:

- **Sintaxe:** similar a Linguagem C/C++.
- **Facilidades de Internacionalização:** suporta nativamente caracteres Unicode;
- **Simplicidade na especificação:** tanto da linguagem como do "ambiente" de execução (JVM);
- **Distribuição:** é distribuída com um vasto conjunto de bibliotecas (ou APIs);
- **Multitarefa:** possui facilidades para criação de programas distribuídos e multitarefa (múltiplas linhas de execução num mesmo programa);
- **Desalocação:** possui um sistema de desalocação automática de memória por processo de coletor de lixo (*garbage collector*);
- **Carga Dinâmica de Código:** programas em Java são formados por uma coleção de classes armazenadas independentemente e que podem ser carregadas no momento de utilização.

2.6 - BANCO DE DADOS HSQLDB

Criado em 1998, o *Hypersonic SQL Database* (HSQLDB), é um projeto de banco de dados livre, escrito totalmente em Java, que permite a manipulação de banco de dados em uma arquitetura cliente-servidor ou *standalone*. Uma das vantagens de utilizar o HSQLDB é a possibilidade de agregar o banco de dados ao pacote das aplicações.

É uma tecnologia muito flexível, que manipula muito bem os dados, e ainda tem a possibilidade de manipulação em disco, em memória ou em formato texto, além de ocupar um pequeno espaço em disco.

2.6.1 - Características do Banco

Segundo (SEVERO, 2008), as principais características do banco de dados são:

- Suporte à linguagem SQL básica, incluindo junções, *triggers* e visões;
- Portabilidade em virtude de sua implementação ser feita em Java;
- Repositórios acessíveis através de tecnologia JDBC;
- Criação de bancos de dados em arquivo texto, banco de dados e em memória;
- Recurso de *dump* para *backups* facilitados;
- Ocupa pouco espaço em disco;
- Praticamente dispensa configurações para operar.

3 - DESENVOLVIMENTO DO PROJETO

Neste capítulo serão apresentados a definição e modelagem do problema, bem como a sua implementação. A implementação foi feita utilizando as tecnologias Java e os conceitos básicos de processamento digital de imagens.

3.1 – DESCRIÇÃO DO PROBLEMA

O objetivo deste projeto é realizar o estudo aprofundado dos métodos para desenvolvimento de um aplicativo para detecção de faces humanas e integrando com outro aplicativo para tratamento de imagens. Para a detecção de faces foram utilizadas as bibliotecas OpenCV e JavaCV. A implementação do aplicativo foi feita em Java.

3.2 – MODELAGEM DO PROBLEMA

A figura 13 mostra a modelagem do problema que será desenvolvido no projeto.

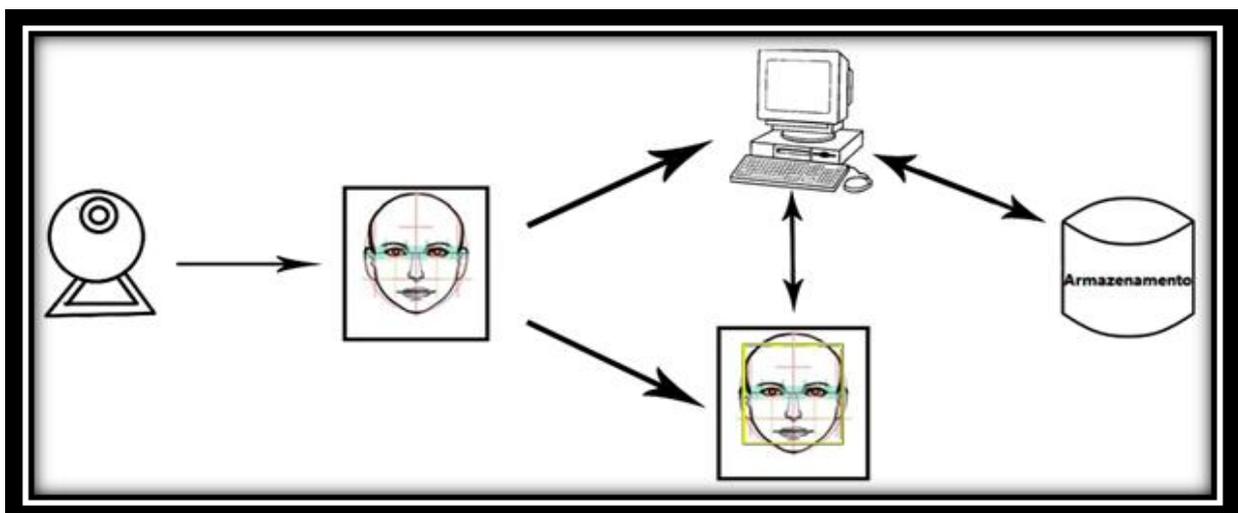


Figura 13 – Modelagem do problema.

O desenvolvimento do projeto foi dividido em três módulos:

- Módulo 1: Criação do banco de Dados;
- Módulo 2: Desenvolvimento do Aplicativo de Tratamento das Imagens;
- Módulo 3: Desenvolvimento do Aplicativo de Detecção de Faces.

Os módulos 1 e 2 foram desenvolvidos no projeto de iniciação científica de 2001 e será integrado com o aplicativo de detecção de faces desenvolvido neste projeto.

3.3 – IMPLEMENTAÇÃO

Nesta seção será feita a descrição da implementação do projeto. Foi escolhida a linguagem Java por ter muitos recursos e ser totalmente livre (*open source*), ou seja, sem nenhum custo. O protótipo foi criado no ambiente integrado de desenvolvimento (IDE – *Integrated Development Environment*) NetBeans 7.0.1. O banco HSQLDB foi escolhido por ser um banco de dados *open source*, totalmente feito em Java e atrativo para armazenamento e manipulação de imagens.

3.3.1 – Aplicativo de armazenamento e tratamento de imagens

Na primeira fase foi desenvolvido um aplicativo de armazenamento e tratamento de imagens capturadas por uma *web cam*, e que na figura 3 é representado pelo processo descrito com o número um (1). Esta fase foi desenvolvida no programa de iniciação científica em 2011, e o resultado obtido foi satisfatório (ROCHA, 2011; ROCHA e NITTO, 2011). Será feita uma explanação dos resultados obtidos neste projeto. O modelo adotado para o armazenamento e tratamento da imagem é mostrado na figura 14.

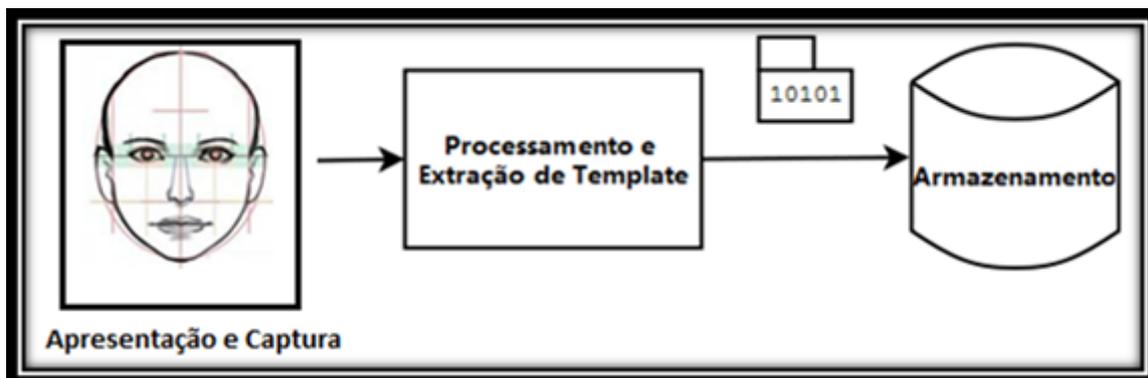


Figura 14: Modelo para armazenamento e tratamento de imagem (ROCHA, 2011)

A implementação do aplicativo foi feita utilizando as tecnologias Java e o banco de dados utilizado foi o HSQLDB.

A figura 15 mostra a interface principal do aplicativo.



Figura 15 – Interface principal do aplicativo

As opções disponíveis ao usuário são:

- Cadastrar: para cadastrar um novo indivíduo ou usuários;
- Buscar: realiza uma busca para verificar os indivíduos cadastrados pelo nome ou código;
- Proc. das Imagens: realiza o processamento e tratamento das imagens cadastradas;
- Autenticar: realiza a autenticação do indivíduo ou usuários já cadastrado;
- Sair: realiza a saída do aplicativo.

Neste projeto não foram utilizadas bibliotecas para a implementação dos métodos e técnicas para o tratamento de imagens. Eles foram implementados passo-a-passo para adquirir conhecimento do funcionamento desses métodos e técnicas.

O dispositivo utilizado para a captura biométrica foi uma *web cam*. A inscrição é feita pelo sistema, o usuário cadastra todos seus dados, informa o padrão biométrico e ao salvar é feito todo o processamento de imagens, armazenando os dados e as imagens processadas no banco de dados.

Na interface inicial (MENU), o usuário terá opções de realizar um cadastro, fazer uma busca dos usuários previamente cadastrados, e visualizar o resultado das fases do processamento de Imagens. As opções oferecidas pelo usuário são: Cadastrar, Buscar, Processamento das Imagens, Autenticar e Sair.

A figura 16 mostra a interface de cadastro ao selecionar a opção Cadastrar na interface principal.

Figura 16 – Interface de Cadastro

Na interface de cadastro o usuário tem uma opção de Busca Simples, no qual ao optar por ela, informará um código que deseja buscar, existindo o código digitado o sistema retorna o usuário com aquele código, não existindo o sistema retorna uma mensagem informando que não existe usuário com o código digitado. A busca sendo satisfatória, o usuário poderá fazer alteração de dados na opção Alterar, ou a exclusão do usuário da opção Deletar.

Ainda na interface de cadastro, ao optar em Inserir, o usuário automaticamente ativará todos os campos para serem preenchidos com seus dados. O próprio sistema informa o código do usuário, para não haver erro de duplicação de chaves. Logo após fornecer todos os dados o usuário tem a opção de Abrir Face, no qual deverá previamente já ter em disco uma foto de sua face (somente da face), para armazená-la em banco. Essa opção de Abrir Face, o usuário terá que informar o caminho por onde está armazenada sua face, para processá-la e salvá-la em banco.

Após todos os dados e faces informados, o usuário deverá clicar no botão Salvar. Essa opção realizará todo processamento da imagem e armazenará em banco, o processamento e a imagem que o usuário forneceu. A opção de Busca Avançada, no qual o usuário optando por este tipo de busca abrirá outra interface, como mostra a figura 17.

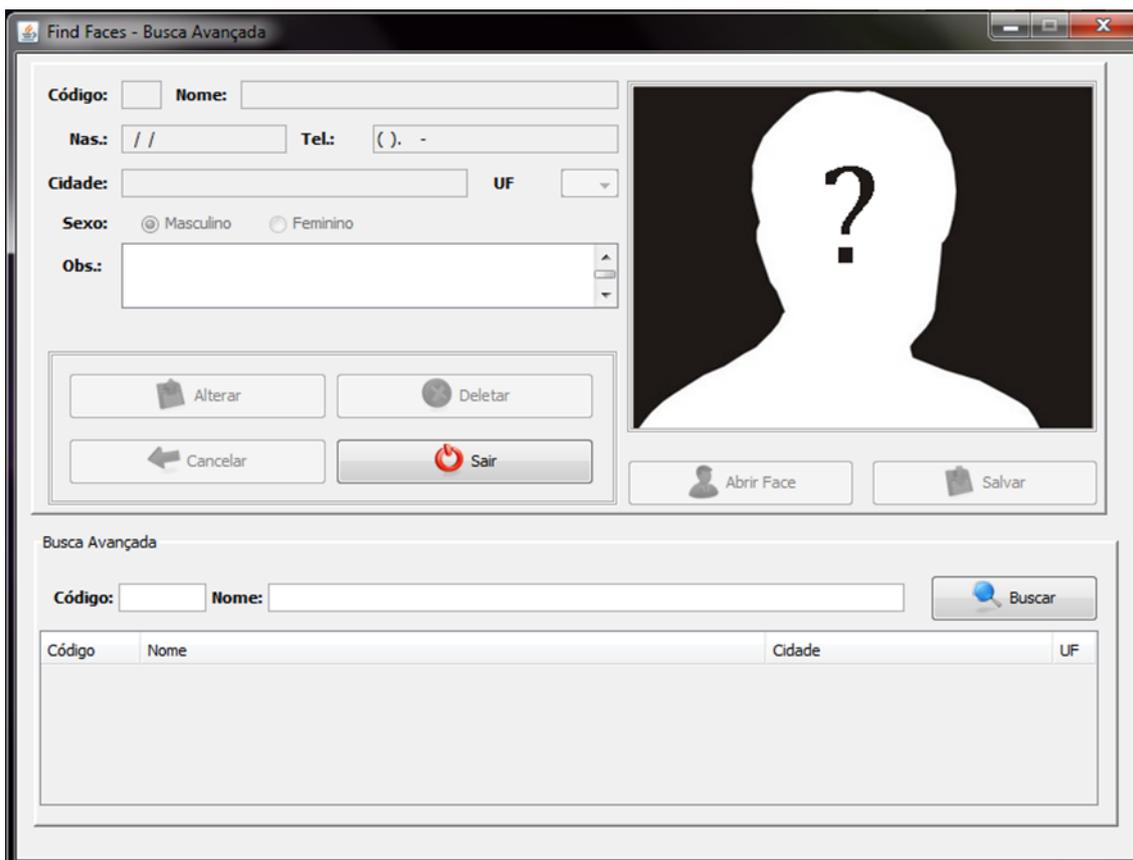


Figura 17 – Interface de Busca Avançada

A interface de Busca Avançada retorna todos os usuários, podendo ser feita uma filtragem, por código ou nome, para melhorar a busca. Essa opção foi criada a fim de manutenção, isto é, caso o usuário queira realizar alguma alteração de dados e não sabe o código de alguém armazenado, esta opção retornara a ele todos os usuários, sendo assim é possível logo após a busca, é possível a alteração dos dados da face, ou a exclusão do usuário. Lembrando

também que esta opção de Busca Avançada, em um software comercial, somente o administrador do sistema terá acesso a essa opção.

A figura 18 a interface do processamento de imagens.

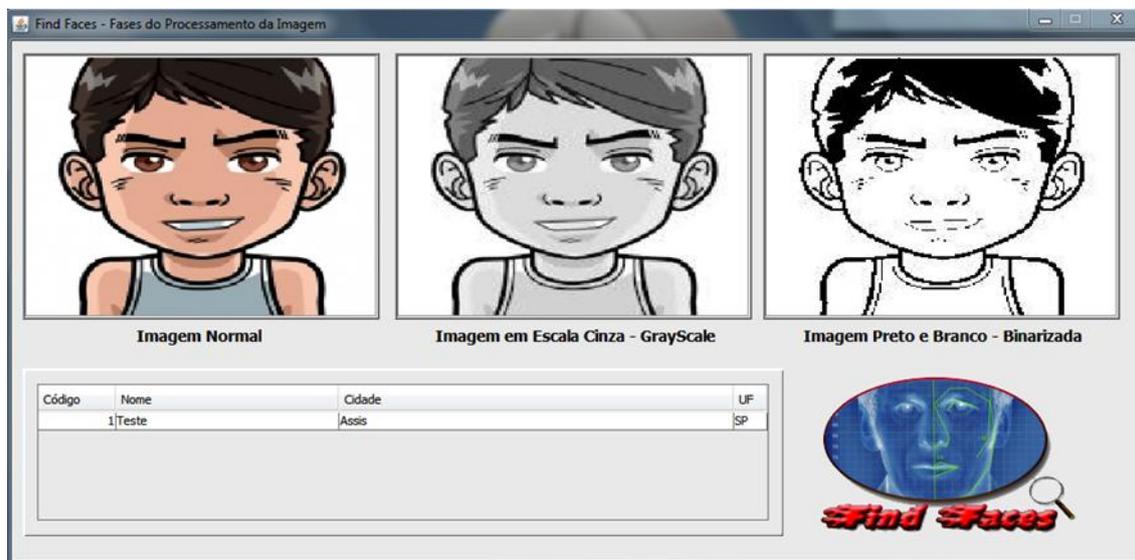


Figura 18– Interface de Processamento de Imagens

Os métodos utilizados para realização do Processamento Digital das Imagens foram: Conversão em tons de Cinza, e a Binarização (Imagem em Preto e Branco). Foi implementado uma interface para ilustrar o resultado das imagens após cada processo. Lembrando que essa interface não realiza o processamento digital da imagem, somente demonstra o resultado de cada filtro logo após o seu processo. Num sistema comercial, não existe essas opção, a mesma foi implementada somente a fins acadêmicos. Lembrando também, que para a continuação deste projeto, deverá ser feito mais filtros, como detecção de bordas, extração de ruídos, entre outros.

3.3.2 – Aplicativo para detecção de faces

Para o desenvolvimento do aplicativo de detecção de faces foram utilizadas as bibliotecas OpenCV com *link* da JavaCV. O aplicativo realiza a detecção de faces em tempo real com as imagens capturadas por uma *web cam* ou também por uma imagem armazenada em um banco de dados. Para a implementação do aplicativo foram utilizadas as funções existentes na biblioteca OpenCV para o processamento de imagens.

A figura 19 mostra o resultado obtido pela aplicação. Ao abrir a interface do aplicativo, o usuário terá a opção de abrir sua *web cam* e automaticamente o sistema detecta a face do usuário a frente da câmera.

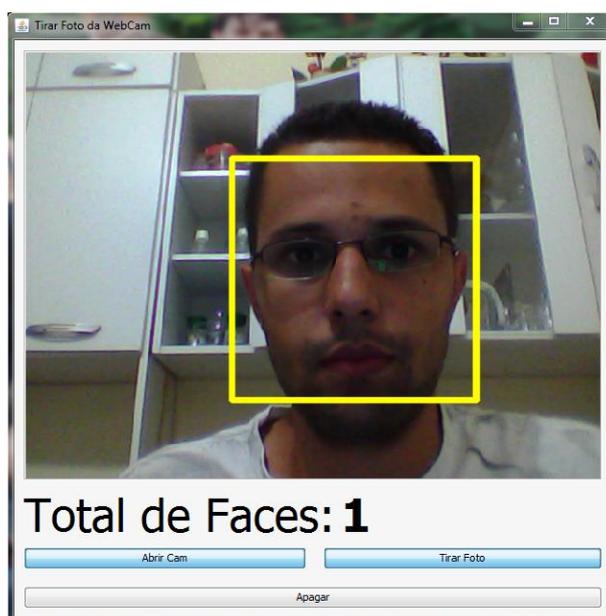


Figura 19 – Interface de detecção de faces

Deve-se lembrar, para realizar a detecção de faces, o algoritmo da biblioteca da OpenCV utiliza por base, todos os estudos, métodos e formulas que foram estudadas na primeira fase deste projeto, mas, desta vez, de forma automática e recursiva, com todos os filtros que baseou-se a fase inicial e muitos outros.

4 – CONCLUSÃO

No início do desenvolvimento do projeto, devido ao pouco conhecimento na área de processamento de imagens e linguagem Java, as dificuldades foram muitas e com o decorrer do tempo foi possível solucioná-las.

O desenvolvimento deste projeto trouxe uma experiência muito grande na formação acadêmica e uma visão de multidisciplinaridade, já que envolveram muitos conceitos vistos no primeiro, segundo e terceiro ano de curso.

Finalizado a segunda fase do projeto, pode-se observar que todo o estudo da primeira fase foi essencial, para que pudesse ser entendida toda a lógica da detecção e conseqüentemente reconhecimento de padrões.

O desenvolvimento do aplicativo mostrou satisfatório, pois os estudos efetuados na primeira fase de desenvolvimento do aplicativo para armazenamento e tratamento de imagens auxiliaram num melhor entendimento do problema. Por fim, esta pesquisa mostrou o nível de detalhamento e eficácia de um algoritmo bem implementado, utilizando combinações de bibliotecas e filtros, e pode afirmar que no futuro os sistemas biométricos substituirão totalmente as senhas, os cartões, pois, o nível de segurança é incomparavelmente mais alto que o convencional.

A próxima fase a ser desenvolvida será o aplicativo de reconhecimento de faces propriamente dito, onde serão integrados todos os aplicativos desenvolvidos até o presente.

REFERÊNCIAS

ALBUQUERQUE, Márcio Portes de; ALBUQUERQUE, Marcelo Portes de. **Processamento de Imagens: Métodos e Análises**, Centro Brasileiro de Pesquisas Físicas CBPF, 2004.

BARBALHO, José ValneyMelo. **NetBeans: Java, Swing**

BASTOS, Vanessa Padores. **Técnicas de Segmentação de Imagens para Recuperação de Informações Visuais**. Disponível em: <<http://paginas.ucpel.tche.br/~vbastos/index.htm>> Acesso em Julho de 2011.

BRAS FILIPE, S.: **Detecção de faces humanas em tempo real**, Universidade da Beira Interior, Covilhã, 2008.

CAELLUUM, Ensino e Inovação. **Algoritmos e Estruturas de Dados Java** Disponível em: <www.caelum.com.br/apostilas> Acesso em Maio de 2011.

CANEDO, J. A. Visão Geral de Um Sistema Biométrico. Disponível em <http://www.forumbiometria.com/fundamentos-de-biometria/129-visao-geral-de-um-sistema-biometrico.html>. Acesso em julho de 2011.

CHEN, Q.; WU, H. e YACHIDA, T.: **Face detection by fuzzy pattern matching**. In ICCV '95: Proceedings of the Fifth International Conference on Computer Vision, page 591, Washington, DC, USA, 1999.

DAI, Y. e NAKANO, Y.: **Face-texture model based on sglD and its application in face detection in a color scene**. Pattern Recognition, 29(6):1007–1017, 1996.

DEITEL, H.M.: **Java como Programar**, Bookman, 2003.

FILHO, Ogê Marques; NETO, Hugo Vieira. **Processamento Digital de Imagens**, Brasport, 1999.

GONZALES, R. e WINITZ, P., **Digital Image Processing**. Addison- Wesley Publishing Company, 1992.

GRADVOHL, André Leon S. **Introdução a Linguagem de Programação JAVA**, Universidade Estadual de Campinas UNICAMP, 2008.

HJELMAS, E. e LOW, B.K.: **Face detection: A survey**. Computer Vision and Image Understanding, 83(3):236–274, September 2001

JUNIOR, Ademar de Souza Reis; FILHO, Milton Soares. **Aplicações de Processamento de Imagens a Sistemas de Segurança**, Universidade Federal do Paraná, 2002.

LIN, C. e FAN, Kuo-Chin: **Triangle-based approach to the detection of human face**. Pattern Recognition, 34(6):1271–1284, 2001.

LOPES, E, C. - **Detecção de faces e características faciais**, Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS) – Departamento de pós Graduação em Ciência da Computação, 2009

LUO, H. e ELEFTHERIADIS, A.: **On Face Detection in the Compressed Domain**, International Multimedia Conference, Proceedings of the Eight ACM International Conference on Multimedia, 2000.

MAGALHÃES, Paulo; SANTOS, Henrique. **Biometria e Autenticação**, Universidade do Minho, Portugal, 2003.

MARIN, Luciene de Oliveira; BARRETO, Jorge Muniz. **Reconhecimento de Faces**, Universidade Federal de Santa Catarina, 2003.

MÜLLER, Daniel Nehme; DARONCO, Everaldo Luis. **Operações Aritméticas em Imagens**, Universidade Federal do Rio Grande do Sul, 2000.

NASCIMENTO, A. V.; GONZAGA, A. - **Detecção de faces humanas em Imagens digitais: Um algoritmo baseado em Lógica Nebulosa**, Departamento de Engenharia Elétrica - USP, 2008

ROCHA, P.F.G. e NITTO, M.A.: **Aplicativo Para Reconhecimento de Faces Humanas**, IV Fórum de Ciência e Tecnologia, Fema-Imesa, Assis, SP, 2011.

ROCHA, P.F.G.: **Aplicativo Para Armazenamento e Tratamento de Imagens de Faces Humanas**, Relatório do Programa de Iniciação Científica, Fema-Imesa, Assis, SP, 2011.

ROWLEY, H.; BALUJA S. and KANADE, T.: **Neural network based face detection**. In Computer Vision and Pattern Recognition '96, June 1996.

SERRANO, Tiago Barquilha. **Padrões Biométricos Para Identificação**, Fundação Educacional do Município de Assis FEMA Assis, 2010.

SEVERO, Carlos Emilio Padilla. **HSQLDB: um banco de dados livre escrito em Java**, Grupo de Usuários Java GUJ, 2008.

SUNG, K. K. e POGGIO, T.: **Example-based learning for view based human face detection**. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(1):39–51, 1998.

VASCONCELOS, C. M.: **Introdução ao Processamento Digital de Imagens**, Centro Brasileiro de pesquisas Físicas, 2001

VIOLA, P. e JONES, M.J.: **Robust real-time face detection**. International Journal of Computer Vision, 57(2):137–154, 2004.

WANG, J. e TAN, T.: **A new face detection method based on shape information**. Pattern Recogn. Lett., 21(6-7):463–471, 2000.

WILSON, G.: **Programmer's Tool Chest – The OpenCV Library**. Disponível em <http://www.ddj.com/architect/184404319?pgno=1>. Acesso em outubro de 2012.

YANG, Ming-Hsuan, David J. Kriegman, Narendra Ahuja, **Detecting Faces in Images: A Survey**, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.24, No.1, Janeiro, 2002.