



Licença Creative Atribuição–Uso Não–Comercial–Vedada
a Criação de Obras Derivadas 2.5 Brasil Commons

Fundação Universidade Federal do Rio Grande
Engenharia de Computação

Um Sistema Computacional para Extração de Características Faciais

Felipe Castro da Silva
Rodrigo Mendes Costa

Rio Grande, 7 de Maio de 2007

Fundação Universidade Federal do Rio Grande
Engenharia de Computação

Um Sistema Computacional para Extração de Características Faciais

Felipe Castro da Silva
Rodrigo Mendes Costa

Trabalho de Conclusão do Curso de Graduação em Engenharia de Computação submetido à avaliação, como requisito parcial à obtenção do título de Engenheiro de Computação.

Orientador(a): Prof. Ms. Alessandro de Lima Bicho

Co-orientador(a): Prof^a. Dra. Silvia Silva da Costa Botelho

Rio Grande, 7 de Maio de 2007

Este trabalho foi analisado e julgado adequado para a obtenção do título de Engenheiro de Computação e aprovado em sua forma final pelos orientadores.

Prof. Ms. Alessandro de Lima Bicho

Banca Examinadora:

Prof. Ms. Alessandro de Lima Bicho

DMAT – FURG (Orientador)

Prof. Ms. Glauber Acunha Gonçalves

DFIS – FURG

Eng. Ms. Rogério Schmidt Feris

Department of Computer Science – University of California, Santa Barbara

A todos que amo

F.C.S.

Àqueles que amo demais

R.M.C.

Agradecimentos

À minha mãe, por ter me educado durante toda vida com muito amor.

Ao meu pai, que sempre foi um exemplo de inteligência, dedicação e sucesso profissional.

Às minhas irmãs, pelo grande apoio dado.

À minha namorada Thiana, pelo seu carinho, amizade, companheirismo e compreensão em todos os momentos deste árduo trabalho.

Aos meus avós, que sempre vibraram com minhas conquistas pessoais e profissionais.

Ao meu orientador Prof. Alessandro de Lima Bicho, por todo seu interesse, flexibilidade e competência para conduzir da melhor maneira este trabalho.

Ao meu amigo Rodrigo Mendes Costa, por sua aplicação e brilhantismo durante todos os anos de convivência.

F.C.S.

Aos meus pais e família, por presentear-me com a vida e preenchê-la com todo amor.

À minha Analía, pelo seu amor e carinho, na alegria e na tristeza...

Ao meu orientador Prof. Alessandro de Lima Bicho, por todo seu interesse, flexibilidade e competência para conduzir da melhor maneira este trabalho.

Ao meu amigo Felipe Castro da Silva, por sua determinação e competência em cada momento necessário.

Ao meu amigo Thales Sehn Körting, por sua dedicação, lealdade e exemplo motivador.

Aos meus demais amigos, por estarem em meu coração me impulsionando e engrandecendo.

R.M.C.

Conteúdo

| | |
|---|-------------|
| Lista de Figuras | iv |
| Lista de Abreviaturas | vii |
| Resumo | viii |
| Abstract | ix |
| 1 Introdução | 1 |
| 1.1 Motivação | 2 |
| 1.2 Objetivos | 2 |
| 1.3 Organização do Texto | 3 |
| 2 Fundamentação Teórica | 5 |
| 2.1 Alguns Trabalhos Correlatos | 5 |
| 2.2 Etapas | 6 |
| 2.2.1 Aquisição | 7 |
| 2.2.2 Detecção | 8 |
| 2.2.3 Localização | 11 |
| 2.2.4 Rastreamento | 11 |
| 2.2.5 Normalização | 11 |
| 2.2.6 Segmentação | 11 |
| 2.2.7 Extração | 12 |
| 2.3 Aplicações | 12 |

| | | |
|----------|--|-----------|
| 3 | Aquisição e Localização da Face | 14 |
| 3.1 | Aquisição | 14 |
| 3.1.1 | Câmeras | 14 |
| 3.1.2 | Ambiente | 15 |
| 3.1.3 | Restrições | 16 |
| 3.2 | Localização | 17 |
| 3.2.1 | Espaços de Cores | 18 |
| 3.2.2 | Localização pelos espaços de cores YIQ e $YCbCr$ | 22 |
| 3.2.3 | Localização por médias migratórias | 24 |
| 3.2.4 | Detecção de Objetos | 26 |
| 3.2.5 | Método de ajuste de uma elipse pelos menores quadrados (<i>Ellipse-Specific</i>) | 30 |
| 3.2.6 | Simetria | 32 |
| 4 | Normalização e Segmentação | 36 |
| 4.1 | Normalização | 36 |
| 4.1.1 | Rotação | 36 |
| 4.1.2 | Ruído | 38 |
| 4.2 | Segmentação | 38 |
| 4.2.1 | Contornos | 38 |
| 4.2.2 | Métricas | 40 |
| 4.2.3 | Projeções Integrais | 41 |
| 4.2.4 | Segmentação de fundo | 41 |
| 5 | Extração das Características Faciais | 45 |
| 5.1 | Extração dos pontos característicos em imagens frontais | 45 |
| 5.2 | Extração dos pontos característicos do perfil | 47 |
| 6 | Aspectos de Implementação | 52 |
| 6.1 | Características | 52 |
| 6.2 | Síntese do Processo | 53 |
| 6.3 | Linguagem de Programação | 54 |

| | |
|---|-----------|
| <i>CONTEÚDO</i> | iii |
| 6.3.1 Histórico | 55 |
| 6.3.2 Características | 55 |
| 6.3.3 <i>Java Media FrameWork</i> (JMF) | 57 |
| 7 Considerações Finais | 58 |
| 7.1 Dificuldades Encontradas | 58 |
| 7.2 Propostas Futuras | 59 |
| Bibliografia | 60 |

Lista de Figuras

| | | |
|------|--|----|
| 1.1 | Pontos característicos considerados nesse trabalho. | 3 |
| 2.1 | Características faciais do padrão MPEG-4. Figura adaptada de [Abrantes and Pereira, 1999] | 6 |
| 3.1 | Câmeras CCD USB utilizadas nas filmagens | 15 |
| 3.2 | Ambiente e disposições utilizadas nas filmagens | 16 |
| 3.3 | Exemplo de <i>frames</i> adquiridos pelas câmeras | 16 |
| 3.4 | Cubo de cores <i>RGB</i> [Figueiró, 2003]. | 19 |
| 3.5 | Espaço de cores <i>HSV</i> como um objeto cônico [Wikipedia, 2005d]. | 20 |
| 3.6 | Sistema de coordenadas do espaço de cores <i>YIQ</i> [Ribeiro, 2004]. | 21 |
| 3.7 | Sistema de coordenadas do espaço de cores <i>YCbCr</i> [Ribeiro, 2004]. | 22 |
| 3.8 | Imagem antes da aplicação do algoritmo de médias migratórias (à esquerda). Imagem após a aplicação do algoritmo de médias migratórias (à direita). | 26 |
| 3.9 | Imagem original da face (à esquerda). Os pequenos objetos brancos dentro dos objetos pretos são os ruídos gerados pela localização da cor de pele (à direita). | 27 |
| 3.10 | Localização da cor da pele (à esquerda) e aplicação do fechamento morfológico à imagem (à direita). | 27 |
| 3.11 | Detecção de cor de pele (à esquerda). Em destaque, o objeto detectado como fundo da imagem (à direita). | 28 |
| 3.12 | Detecção de cor de pele (à esquerda). Em destaque, os objetos detectados como buracos da imagem (à direita). | 29 |

| | | |
|------|--|----|
| 3.13 | Imagem original da face (à esquerda). Objeto detectado como face na imagem (à direita). | 29 |
| 3.14 | Imagem original da face (à esquerda). Elipse aplicada ao contorno externo (à direita). | 30 |
| 3.15 | Imagem original da face (à esquerda). Elipse aplicada aos contornos interno e externo (à direita). | 31 |
| 3.16 | Elipse aplicada ao contorno externo (à esquerda). Cálculo do ângulo de inclinação da face (à direita). | 31 |
| 3.17 | Objeto detectado como face, pela cor de pele, e com a aplicação do algoritmo de detecção de objetos (em cima, à esquerda). Objeto criado a partir do contorno da primeira aplicação da elipse (em cima, à direita). Objeto resultante da operação de OR entre os dois objetos de cima (em baixo, à esquerda). Elipse gerada a partir dos contornos do novo objeto (em baixo, à direita). | 32 |
| 3.18 | Variância e Acerto para cada eixo de simetria (à esquerda). Valores de Acerto (à direita). Adaptada de [Chen et al., 2005]. | 34 |
| 3.19 | Sugestão de angulação no método da Elipse (à esquerda). Eixo de simetria detectado (à direita). | 34 |
| 4.1 | Eixo de simetria detectado (à esquerda) e a imagem frontal rotacionada (à direita) | 37 |
| 4.2 | Imagem original com ruído (à esquerda) e a imagem após aplicada a convolução de média (à direita) | 38 |
| 4.3 | Exemplo de segmentação utilizando projeções integrais | 41 |
| 4.4 | Imagem da face original (cima, à esquerda). Detecção dos contornos gerados pelo algoritmo de Canny (cima, à direita). Segmentação do fundo utilizando somente a informação da componente matiz (baixo, à esquerda). Segmentação do fundo utilizando a componente matiz e a imagem dos contornos (baixo, à direita). | 44 |
| 4.5 | Imagem com o fundo em branco (à esquerda). Linha de perfil gerada pelo algoritmo Canny (à direita). | 44 |

| | | |
|-----|---|----|
| 5.1 | Imagem original da face (à esquerda). Linhas que restringem as regiões de busca (à direita) | 46 |
| 5.2 | Pontos de perfil (vermelho e verde) [Shan et al., 2000]. | 47 |
| 5.3 | Imagem original da face (à esquerda). Ponta do nariz extraída (à direita). . | 48 |
| 5.4 | Exemplos de extração dos pontos característicos de perfil. | 50 |
| 6.1 | <i>ScreenShot</i> da tela da aplicação que valida nosso trabalho. | 53 |
| 6.2 | Etapas do processo de extração de características proposto. | 54 |

Lista de Abreviaturas

API Application Programming Interface

CCD Charge-Coupled Device

CODEC Codificador-Decodificador

FPS Frames Per Second

FTP File Transfer Protocol

GLDH Gray Level Difference Histogram

HSV Hue, Saturation e Value

HTTP HyperText Transfer Protocol

IHC Interação Humano-Computador

JVM Java Virtual Machine

RGB Red, Green e Blue

Resumo

Este trabalho propõe extrair o maior número possível de características faciais que possam descrever o movimento da face de forma a possibilitar a criação de um modelo tridimensional a fim de simular seus movimentos. Como entrada para o sistema, são aplicadas técnicas de visão computacional a imagens ortogonais de uma face humana para a geração de pontos tridimensionais que a caracterizem. O método utilizado sugere uma abordagem em várias etapas para alcançar o objetivo, a saber, os pontos característicos da face. No primeiro processo, para a imagem frontal, a face é localizada com a utilização de um método no qual a posição da face é estimada por sua cor de pele. Já na imagem de perfil, uma técnica de extração de fundo é empregada para determinar os contornos da linha da face. A partir da localização da face, o método de ajuste a uma elipse pelos menores quadrados (*Ellipse-Specific*) é utilizado. Este método tem por objetivo reconstruir, da melhor maneira possível, o formato elíptico da face, de modo a estimar sua rotação. Um método de simetria refina o ângulo de rotação da face possibilitando sua normalização. Na segmentação, os limites das características de interesse da face (sobrancelhas, olhos, nariz e boca) são delimitados. Para tal operação é utilizado o método de projeções integrais. Estes valores reduzem o espectro de análise para as etapas seguintes. A extração busca pelos pontos característicos em regiões restritas, anteriormente estimadas, tanto na imagem de perfil quanto na imagem frontal. Por fim, os pontos tridimensionais característicos da face são determinados através da correlação entre a imagem frontal e a lateral.

Abstract

This work proposes to extract the largest possible number of face features that can describe the face movement to make possible the creation of a tridimensional model to simulate its movements. As a system input, techniques of computational vision for the orthogonal images of a face are applied to the generation of tridimensional points that characterize it. The used method suggests an approach with some stages to achieve the objective, that are the features points of the face. In the first process, for the frontal image, the face is located with the use of a method where the position of the face is estimated based on its skin-color. In the profile image, a technique of background extration is used to determine the contours of the face. After the localization of the face, the method of adjustment to an ellipse (Ellipse-Specific Direct Least-Square Fitting) is used. This method reconstructs, in the best possible way, the elliptical shape of the face, to estimate its rotation. A symmetry method refines the rotation angle of the face making possible its normalization. In the segmentation, the limits of interested features of the face (eyebrows, eyes, nose and mouth) are delimited. For such operation the method of integral projections is used. These values reduce the analysis for the following stages. The extraction level searches for the feature points in restricted regions, previously estimated, bothin the profile image as in the frontal image. Finally, tridimensional feature points of the face are determined through the correlation between the frontal and profile image.

Capítulo 1

Introdução

Uma breve avaliação do histórico de pesquisas na área da Visão Computacional e da Computação Gráfica expõe uma grande evolução e uma eminente preocupação sobre o aperfeiçoamento das interfaces gráficas que interajam com os seres humanos. Dentro deste contexto, uma trajetória bastante interessante a seguir, de modo a propiciar esta melhor comunicação entre homem e máquina, seria dotar o computador com características humanas.

A ideologia de tornar o computador capaz tanto de reconhecer expressões e gestos, como também reproduzi-los de forma realística e convincente, tem sido foco de pesquisa em grupos de IHC (Interação Humano-Computador) [IHC, 2005]. Dentre as linhas de pesquisa existentes, destaca-se o interesse em modelar a face humana e determinar suas características principais.

Seguindo essa filosofia, pode-se discriminar duas categorias de modelos faciais [Garcia et al., 2001]:

- **Avatars:** modelos irrealis que produzem representações gráficas simbólicas de um usuário em realidade virtual, jogos ou outros ambientes interativos [Wikipedia, 2005a]. Segundo [Garcia et al., 2001], é a categoria que na maioria das vezes pode ser facilmente modelada e animada;
- **Clones:** modelos fidedignos que se assemelham com seres humanos [Valente and Dugelay, 2000]. Estes são considerados um dos grandes desafios para a Computação Gráfica [Lee et al., 1995].

Ambicionando que agentes virtuais interajam com múltiplos usuários em inúmeras situações completamente distintas, é inevitável considerar que eles devam alterar suas expressões faciais diante do contexto na qual se encontram [Yokoyama et al., 1996]. Este é apenas um dos propósitos do estudo de técnicas de extração de características faciais que podem ser aplicadas em outros campos de pesquisa, tais como, reconhecimento, acessibilidade, teleconferência virtual, codificação de vídeo, entre outras.

1.1 Motivação

O advento da Computação Gráfica e, paralelamente, a rápida emergência de *hardwares* próprios para a execução de aplicações tridimensionais, permitiram que novos estudos sobre a modelagem de estruturas complexas fossem realizados. Neste contexto, surgiu a proposta de modelar a face humana de forma a reconhecer e simular seus movimentos em um ambiente virtual, gerando um agente possível de utilização em inúmeras ferramentas, tornando assim a interação com o computador muito mais amigável.

Para que este objetivo seja alcançado, inicialmente é necessária a aquisição de um modelo de face humana. Utilizando técnicas de visão computacional, o intuito deste trabalho concentra-se em coletar as características estáticas e dinâmicas necessárias de um modelo humano real. Estas características são representadas através de pontos do espaço tridimensional, possibilitando a animação de uma face virtual do modelo humano analisado.

Nossa motivação alimenta-se da necessidade de intensificar pesquisas científicas sobre o processo de extração de características faciais, utilizando técnicas de processamento de sinais (imagens) aliadas à visão computacional. Seu resultado pode ser focalizado em diversos segmentos de pesquisa e desenvolvimento, gerando tanto novos trabalhos científicos como ótimas ferramentas que usufruam seus benefícios.

1.2 Objetivos

O propósito deste trabalho visa, a partir de um ambiente suficientemente controlado, extrair o maior número possível de características faciais que possam descrever o movimento

de uma face. Para tal, são utilizadas imagens ortogonais de uma face humana, onde uma conterà a vista frontal e a outra conterà o perfil da face. Deve ficar claro que apenas alguns dos pontos mais significativos da face serão utilizados, ou seja, o projeto não tem inicialmente o intuito de extrair pontos suficientes que permitam modelar geometricamente a face. Os pontos característicos que serão considerados ao longo deste trabalho estão especificados na Figura 1.1.

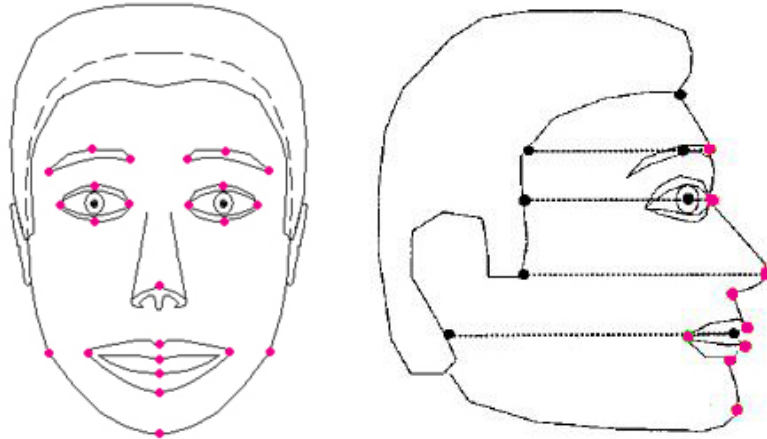


Figura 1.1: Pontos característicos considerados nesse trabalho.

Outra questão relevante à exposição dos objetivos é quanto as possíveis restrições que algumas técnicas pesquisadas impõem ao modelo analisado, descaracterizando uma postura natural e limitando o funcionamento das técnicas para apenas alguns casos. Por esse motivo, ambicionou-se superar essas limitações, escolhendo metodologias que nos permitissem alcançar o resultado sem impor tantas restrições ao modelo de face analisado.

O conceito de *software* multi-plataforma também foi considerado na elaboração desta proposta de projeto, resultando na decisão da utilização da linguagem *JAVA* como linguagem de programação. Além disso, o projeto propôs a utilização de *webcams* no papel dos sensores de entrada do sistema, o que representa uma solução de baixo custo e, ao mesmo tempo, um obstáculo a ser transposto devido à qualidade da imagem adquirida.

1.3 Organização do Texto

Após uma breve introdução sobre o contexto do trabalho, neste momento aproveita-se para orientar sobre a disposição das demais seções inclusas neste documento. Espera-se,

por meio desta, favorecer a compreensão do conteúdo abordado, além de descrever uma síntese sobre a forma como nosso trabalho foi conduzido ao longo do estudo realizado.

A organização adotada prevê uma revisão bibliográfica sobre as etapas necessárias até a extração de características faciais no Capítulo 2. Neste capítulo, uma visão geral juntamente com um histórico científico sobre o assunto é abordada. Vários conceitos são expostos para o real entendimento do processo.

O Capítulo 3 inicia a abordagem adotada nesse trabalho para a implementação de um sistema capaz de extrair características faciais. Todas as técnicas implementadas são descritas e avaliadas diante dos resultados alcançados. Nova descrição é realizada no Capítulo 4, agora sob o âmbito da segmentação de características. Finalizando a explicação do processo, o Capítulo 5 trata da extração dos pontos característicos dos elementos que constituem a face humana.

A partir do Capítulo 6, aspectos relacionados à implementação do sistema, incluindo todos os recursos utilizados, são apresentados. Já o Capítulo 7 transpõe as conclusões alcançadas e descreve as principais dificuldades encontradas, finalizando este trabalho com as perspectivas futuras.

Capítulo 2

Fundamentação Teórica

A intenção deste capítulo é realizar uma síntese superficial das etapas até a extração de características faciais e dessa forma facilitar a compreensão das escolhas e implementações presentes neste trabalho. Nas seções seguintes, serão apresentados conceitos da área que se farão importantes para a leitura. Através deste capítulo, será exposto o quanto foi pesquisado para a realização deste projeto.

2.1 Alguns Trabalhos Correlatos

Neste ponto, é prioritário que se defina exatamente o significado de características faciais que deseja-se obter. Características faciais são os elementos que compõem a face, como sobrancelhas, olhos, nariz, lábios, queixo, entre outros. Nossa intenção é extrair os pontos característicos que definem esses elementos, para dessa forma analisar o movimento e permitir a geração de um modelo para simulação. Em [Abrantes and Pereira, 1999] pode-se encontrar a especificação do padrão MPEG-4 que define pontos característicos da face que tem relevância para a definição de movimentos faciais. A Figura 2.1 mostra quais são esses pontos.

O trabalho de [Yokoyama et al., 1996] relata que muitos métodos foram propostos para extrair características faciais. Dentre eles se destacariam o casamento de padrões, projeções integrais de contornos e modelos deformáveis. O método de casamento de padrões é empregado para reconhecer as características faciais, mas é sensível a presença de transformações geométricas, distorções e deformações causadas pelas expressões faci-

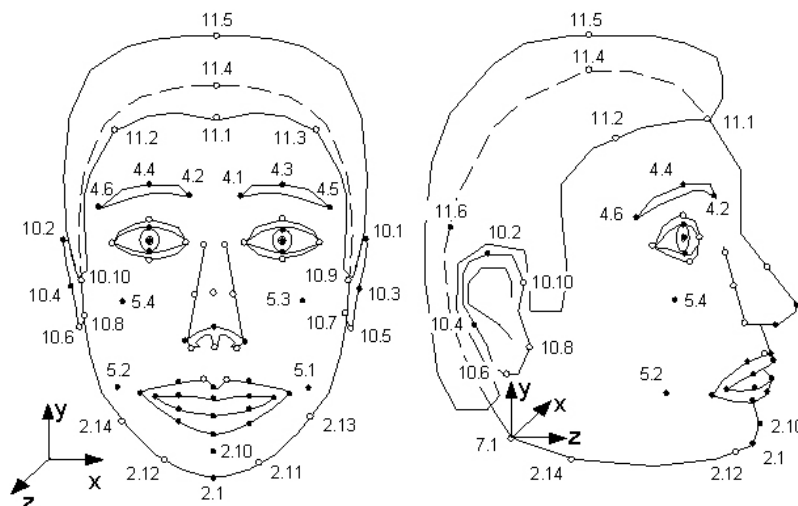


Figura 2.1: Características faciais do padrão MPEG-4. Figura adaptada de [Abrantes and Pereira, 1999]

ais. Um trabalho dentro dessa linha é o de [Lanzarotti et al., 2002]. Outros trabalhos propõem métodos heurísticos aplicados às projeções integrais e informações adicionais sobre a simetria para localizar face e características, como por exemplo, os trabalhos de [Gu et al., 2003, Shan et al., 2000, Rozinaj and Mistral, 2003]. Métodos que se baseiam apenas na análise de níveis de cinza (intensidade) são facilmente influenciados pela variação da iluminação. Técnicas que utilizam modelos deformáveis de contornos ativos, como *Snakes* [Precioso et al., 2005], detectam ou perseguem contornos evidentes. Esse último tipo modela uma função de energia que permite a adaptação da *Snake* ao contorno. No entanto, o principal problema desta técnica se deve a disposição inicial da *Snake*, juntamente com o fato das características faciais não apresentarem em grande parte das vezes contornos uniformes. Na mesma linha aparecem os padrões deformáveis [Yuille et al., 1992] que também baseiam-se em funções de energia, mas aproximam-se de outras características além de contornos, como picos e vales de intensidade.

2.2 Etapas

Após uma ampla pesquisa sobre trabalhos correlatos à extração de características faciais, foi identificado alguns conceitos que, apesar de não encontrados explicitamente com essa nomenclatura e especificidade em qualquer um dos trabalhos pesquisados, sintetizam um

consenso entre os mesmos. Esses conceitos serão apresentados aqui como uma seqüência de etapas ou tarefas que constituem um processo completo de extração de características faciais. Essas etapas seriam:

- Aquisição
- Detecção
- Localização
- Rastreamento
- Normalização
- Segmentação
- Extração

No entanto, é relevante destacar que essas etapas não são necessariamente independentes e absolutas. Tanto algumas delas podem substituir outras, quanto podem estar interligadas de forma a impossibilitar uma distinção perfeita [Zhao et al., 2003]. A percepção destas dependências ficará mais clara ao longo desse trabalho.

2.2.1 Aquisição

A indispensável etapa de aquisição é uma das mais importante no processo de extração de características faciais. Todas as demais fases dependem direta ou indiretamente das decisões tomadas para realizar o processo de aquisição. Dentre as variáveis pertinentes a esse processo, destacam-se genericamente:

- **Sensor:** o tipo de sensor utilizado para coletar as informações pertinentes sobre o modelo facial analisado;
- **Disposição dos sensores:** o arranjo adequado dos sensores para o problema proposto, dependente da técnica utilizada para a síntese da fase;
- **Ambiente:** pré-requisitos necessários para que a aquisição possa ser realizada;

- **Posição:** a posição adequada do modelo que passará pela análise;
- **Regra:** restrições as quais o modelo humano deve se submeter em relação a movimentos, a expressões, a indumentária, entre outros.

Como nossa abordagem sobre o processo de extração de características se restringe à utilização de imagens, a aplicação de câmeras como sensores de entrada foi adotada e, por esse motivo, tratar-se-á especificamente deste segmento. No entanto, algumas pesquisas na área utilizam outros tipos de equipamentos específicos como *scanners à laser*¹ capazes de gerar, através do cálculo da distância de cada ponto do objeto analisado, uma nuvem de pontos no espaço tridimensional.

Existem na literatura algumas técnicas que utilizam câmeras para a constituição de um *scanner 3D*. Algumas delas, inclusive, vinculam também a utilização de outros dispositivos para auxiliar a síntese facial, como feixes de *laser* ou luz. Uma análise dos métodos mais difundidos pode ser encontrada em [Chan et al., 2004]. Como explicado no Capítulo 1, a técnica adotada em nosso trabalho baseia-se na utilização de duas vistas ortogonais da face analisada.

Em relação ao que chama-se de ambiente, existe uma grande variedade de propostas que descrevem a estrutura necessária para que a aquisição e futura extração de características funcione adequadamente. No que diz respeito à iluminação, um dos principais problemas existentes é o de como lidar com as sombras, afinal elas são muitas vezes indesejáveis nas etapas seguintes. Por esse motivo, uma iluminação ineficiente pode comprometer todo o processo hierárquico para a extração de características.

Tanto o posicionamento do modelo quanto as possíveis deformações de sua face podem caracterizar exceções a muitos sistemas de extração de características. Sendo assim, estes cuidados também devem ser considerados durante a etapa de aquisição.

2.2.2 Detecção²

Identificada como a primeira fase de qualquer processo de localização, extração ou reconhecimento de características, a fase de detecção de faces revela os primeiros obstáculos a

¹Encontrados na literatura como *laser range scanners* ou apenas *range scanners*.

²Seção baseada no trabalho de [Feris, 2001]

serem transpostos no que diz respeito a visão computacional. Inúmeros agentes, externos e internos, afetam a eficiência dos algoritmos responsáveis pela constatação da presença ou não de uma face humana na imagem ou seqüência de imagens analisadas. Dentre os fatores externos, destacam-se ruídos na imagem, diferentes condições de iluminação e obstrução total ou parcial da face humana. Determinadas alterações e deformações na aparência da face, ocasionadas até mesmo por expressões faciais incomuns, são consideradas agentes internos e resultam em uma série de restrições impostas pelos sistemas propostos na literatura.

Imagens estáticas

Existem, basicamente, duas classificações para as técnicas utilizadas na detecção de faces em imagens estáticas: (1) abordagem baseada em características e (2) abordagem holística. A primeira, realiza a identificação da presença da face humana buscando por características faciais e possíveis relações métricas entre elas. Já a abordagem holística analisa a imagem na tentativa de detectar a face em sua íntegra.

Inicialmente, as propostas para detecção de faces basearam-se na busca por características faciais através da análise de suas formas. Essa técnica demonstrou-se sensível a interferências provenientes de ruídos. Técnicas com casamento de padrões também se mostraram ineficientes, visto que os elementos que constituem a face humana não são rígidos. Em [Yuille et al., 1992], a técnica de utilização de padrões deformáveis³ apresenta bons resultados, mas possui como principal desvantagem o elevado custo computacional envolvido.

Além destas propostas para a abordagem baseada em características, ainda se destacam métodos que utilizam modelos de contornos ativos, modelos estatísticos e técnicas baseadas em correlação.

Normalmente, abordagens holísticas realizam uma varredura na imagem por intermédio de uma janela em múltiplas escalas, extraindo pequenas regiões que são então processadas por um método específico. Técnicas como PCA (*Principal Component Anal-*

³Tradução utilizada para *deformable templates*, também conhecida na literatura brasileira como *templates* deformáveis

ysis), modelos probabilísticos utilizando *eigenfaces* apresentam resultados positivos, mas sofrem restrições quanto à variação da luminosidade e alterações da face neutra. Redes neurais apresentam resultado satisfatório na detecção de faces em imagens complexas e rotacionadas.

Outras técnicas conhecidas utilizam as cores para selecionar regiões de interesse, diminuindo significativamente o custo computacional ao varrer toda a imagem para localizar as possíveis ocorrências de uma face. A desvantagem desses métodos é também comumente relacionada a diferentes graus de iluminação. Dentre abordagens baseadas em características e abordagens holísticas, a segunda tem apresentado maior eficácia quando se trata de imagens estáticas.

Imagens dinâmicas

Quando se trata de seqüências de imagens ou vídeos, geralmente há uma perda em relação à qualidade das imagens, em contrapartida, inúmeras informações sobre movimento podem ser utilizadas. Valendo-se desse artifício, os métodos de detecção de faces em seqüências de vídeo são divididos em duas etapas: (1) redução do espaço de busca e (2) análise da ocorrência de face. A primeira etapa diz respeito a seleção eficiente de regiões candidatas à conterem a face humana, restando à segunda verificar as regiões selecionadas.

Como citado anteriormente, uma das técnicas de redução de espaços de busca é a utilização de modelos de cor da pele humana [Feris et al., 2000], pois demandam um esforço computacional muito reduzido e facilitam a detecção da face em diferentes orientações. Outras propostas interessantes utilizam a técnica de subtração de fundo como alternativa, no entanto, só apresentam bom resultados em situações de câmera imóvel e poucas alterações entre as imagens.

No que diz respeito a verificação da ocorrência ou não da face nas regiões candidatas, as principais técnicas encontradas na literatura baseiam-se basicamente em correlação com padrões, simetria, análise de textura, forma e até mesmo movimentos característicos como piscar e movimentação dos olhos.

2.2.3 Localização

A etapa de localização é muito semelhante a etapa de detecção. Elas só se diferenciam pelo fato da localização não comprovar a presença da face na região localizada. Basicamente, a detecção engloba a localização, mas a recíproca não é verdadeira. Por esse motivo, as técnicas só de localização geralmente utilizam mecanismos de avaliação menos exigentes, como por exemplo, a proposta desse trabalho que utiliza apenas a cor da pele para localizar a região onde se encontra a face.

2.2.4 Rastreamento

A etapa de rastreamento de faces, conhecida como *tracking*, é responsável pelo acompanhamento da face obtida no início de uma seqüência de vídeo até o final da mesma. Isso implica na localização constante da face em cada quadro que compõe o vídeo. Desta forma, o custo computacional é reduzido juntamente com o espaço de busca dentro do quadro. Outra vantagem do rastreamento é a continua realimentação do sistema da informações sobre o movimento que podem ser de grande valia dependendo da aplicação à qual está sendo utilizada.

2.2.5 Normalização

A etapa de normalização tem por finalidade executar as possível correções que são necessárias nas imagens ou quadros que passarão a etapa de segmentação e extração de características. Dentre os principais problemas que uma imagem possa apresentar, destacam-se a ocorrência de ruído na captura do sensor de entrada, além das possíveis rotações naturais, mesmo que pequenas, aplicadas pela face.

2.2.6 Segmentação

O processo de segmentação é a etapa onde é realizada a subdivisão de uma imagem em suas partes constituintes. Este processo é concluído quando todas os objetos de interesse tiverem sido isolados. A segmentação autônoma é uma das tarefas mais difíceis em processamento de imagens e determina o eventual sucesso ou fracasso na análise [Gonzalez and Woods, 2000].

Algoritmos de segmentação de imagens normalmente baseiam-se na análise de descontinuidades, focados na detecção de mudanças bruscas (pontos, linhas e bordas), e similaridade, dedicados a limiarização (*thresholding*) e operações entre regiões distintas. Uma vantagem específica da segmentação aplicada em seqüências de vídeo é a utilização do movimento para determinação das regiões a serem diferenciadas.

Quanto se trata especificamente da segmentação da face e características faciais, é preciso se considerar que olhos, boca, nariz e demais características são diferentes de pessoa para pessoa e, mesmo sobre a mesma pessoa, podem ser deformadas conforme a expressão facial [Yokoyama et al., 1996].

2.2.7 Extração

Por fim, a etapa de extração compreende as técnicas necessárias para extrair os pontos de controle de cada uma das características faciais segmentadas. Como os diversos elementos da face possuem características diferenciadas, geralmente técnicas distintas são aplicadas a cada segmento para realizar a tarefa de extração.

Segundo [Zhao et al., 2003], os métodos de extração de características podem ser diferenciados em três tipos:

- Métodos genéricos baseados em contornos, linhas e curvas;
- Métodos baseados em padrões das características;
- Métodos de correspondências estrutural que consideram geometria dos elementos;

2.3 Aplicações

O trabalho de [Tang and Huang, 1996] acrescenta que em função da tentativa de alcançar baixas taxas de codificação de vídeo, sistemas de codificação baseada em modelos têm atraído grande interesse na área de telecomunicações visual como, por exemplo, vídeofone e teleconferências onde a face humana apresenta o papel mais relevante. A principal idéia dessa abordagem é construir um modelo 3D da face humana. Apenas as partes que caracterizam o movimento são analisadas e os parâmetros da movimentação são então

transmitidos, permitindo que a expressão facial original seja sintetizada pela deformação do modelo facial.

A identificação de suspeitos a partir de câmeras de segurança é uma aplicação de grande importância da extração de características faciais. Em [Martins, 2005], é apresentado um instrumento de utilidade para órgãos policiais que aplicam recursiva e manualmente o retrato-falado como ferramenta para busca de suspeitos em arquivos fotográficos. Já em [da Silva and de Castro Santa Rosa, 2003], a busca por pessoas é feita com apenas 20% da imagem de uma face, através da técnica de *eigeneyes*.

Segundo [Lanzarotti et al., 2002] os modelos tridimensionais gerados por extração de características são usados em aplicação voltadas a medicina, video, realidade virtual e computação gráfica.

Capítulo 3

Aquisição e Localização da Face

Com uma visão mais clara do desafio de extrair características faciais a partir de uma seqüência de imagens, o trabalho se direciona a explicar as características das principais técnicas utilizadas. Em um primeiro momento, o processo de aquisição será discutido, seguindo as questões expostas na Seção 2.2.1. Logo em seguida, os métodos aplicados à localização da face serão descritos, além da apresentação do conteúdo necessário para seus entendimentos.

3.1 Aquisição

Para melhor apresentar as decisões tomadas neste projeto em relação à aquisição, esta seção comenta separadamente cada uma das escolhas que foram tomadas para definir a trajetória inicial do trabalho.

3.1.1 Câmeras

Desde o período de idealização do projeto, existiu o interesse em implementar uma técnica que realizasse a extração de características com o mínimo de recursos possíveis. Dessa forma, o custo baixo e a facilidade de obter os recursos para a sua execução tornariam o programa mais acessível para sua utilização.

Inicialmente, cogitou-se utilizar câmeras de vídeo conectadas a placas de captura de vídeo nos computadores. No entanto, essa alternativa seria tão custosa que inviabilizaria a utilização dessa pesquisa para outras aplicações.

Seguindo essa trajetória, decidiu-se aceitar o desafio de extrair características faciais de um modelo humano utilizando câmeras CCD¹ USB. Conhecidas como *webcams*, essas câmeras apresentam uma resolução muito baixa em comparação com as câmeras tradicionais, além de serem muito sensíveis as variações de luminosidade.

A Figura 3.1 mostra as câmeras utilizadas na aquisição dos vídeo utilizados na extração de características faciais. A resolução adotada foi a de 352 x 288 *pixels* com um *framerate* (taxa de *frames*) variável em função da característica CCD da câmera influenciada pela iluminação. O *framerate* médio conseguido se aproximou de 10 fps (*frames per second*).



Figura 3.1: Câmeras CCD USB utilizadas nas filmagens

3.1.2 Ambiente

Dentre as opções pesquisadas para dispor o modelo durante as filmagens, foi adotada a estrutura mostrada na Figura 3.2. Duas câmeras foram dispostas em eixos ortogonais a uma distância aproximada de 50 centímetros da cabeça do modelo.

Como fundo foi utilizado uma tecido azul, baseado na técnica de cromaqui que será explicada mais detalhadamente no Capítulo 4 sobre normalização e segmentação. A adoção de um fundo simples reduz significativamente a complexidade da etapa de localização da face, no entanto, outros trabalhos como o de [Su and Yin, 2004] dedicam-se exclusivamente a detectar a face humana em imagens complexas.

¹CCD (Charge-Coupled Device) ou Dispositivo de Carga Acoplado é um sensor para a gravação de imagens formado por um circuito integrado contendo um conjunto de capacitores ligados (acoplados). Sob o controle de um circuito externo, cada capacitor pode transferir sua carga elétrica para um outro capacitor vizinho) [Wikipedia, 2005c].



Figura 3.2: Ambiente e disposições utilizadas nas filmagens

Quanto à iluminação, nenhum padrão mais rigoroso foi utilizado, apenas tentou-se dispor lâmpadas suficientes ao redor do modelo de forma a evitar ao máximo o aparecimento de sombras sobre sua face. Inicialmente, apenas lâmpadas fluorescentes foram consideradas, no entanto, uma lâmpada frontal incandescente fez-se necessária para manter o tom da pele mais perceptível, já que o mesmo se apresentava muito pálido sob a iluminação fluorescente. A Figura 3.3 mostra um exemplo de aquisição realizado.



Figura 3.3: Exemplo de *frames* adquiridos pelas câmeras

3.1.3 Restrições

Como foi colocado na Seção 1.2, um dos principais objetivos desse trabalho é evitar impor restrições ao modelo facial analisado. Isso porque o projeto propõe acompanhar e reproduzir o movimento facial. No entanto, infelizmente algumas restrições tiveram que ser impostas para que o projeto pudesse ser implementado em tempo. Mesmo assim, como serão descritas nos capítulos seguintes, algumas restrições encontradas em trabalhos correlatos puderam ser superadas.

Inevitavelmente, para a técnica de correlação, entre imagem frontal e imagem de perfil, utilizada até o atual estado do projeto, a face não pode realizar rotações significativas sobre o eixo das profundidades, pois isso resultará em erros na aproximação da região segmentada, especificamente a da boca. Essa restrição será melhor detalhada no Capítulo 4 sobre normalização e segmentação.

3.2 Localização

A próxima etapa, após se realizar a aquisição de um vídeo da face, através das câmeras frontal e lateral, será a localização da face em cada imagem (ou *frame*). Localizar a face, antes de extrair cada característica facial em particular, facilitará o trabalho, pois a procura por elementos não será em toda imagem, mas sim em um espaço reduzido.

Contudo, a maioria das técnicas utilizadas para localizar faces tem por objetivo tentar suprir alguns problemas comuns como [Yang et al., 2002]:

- **Pose:** as imagens de face variam de acordo com a posição da câmera. Conseqüentemente, algumas características faciais, tais como um olho ou o nariz, podem estar parcialmente ou totalmente oclusos.
- **Presença ou não de componentes estruturais:** características faciais tais como barbas, bigodes e até mesmo adereços, como, por exemplo, óculos, podem estar presentes, o que resulta em uma grande variabilidade entre esses componentes incluindo suas formas, cores e tamanhos.
- **Expressão Facial:** a expressão da face influencia diretamente na aparência e na forma da imagem.
- **Oclusões:** a face pode estar obstruída parcialmente por outros objetos.
- **Condições da imagem:** quando uma imagem é obtida, fatores como iluminação e características da câmera afetam a aparência da face.

Existem várias técnicas utilizadas para localizar a face em uma imagem. Por exemplo, em imagens coloridas, pontos de cor de pele podem ser encontrados pelo método baseado

em características invariantes², que é um processo interessante para se obter uma nuvem de pontos que represente a face.

A utilização da cor da pele para determinar a localização da face é uma sub-área dos métodos baseados em características invariantes. Dentro desse espectro de pesquisa, foram estudados alguns métodos, sendo que um deles foi utilizado como uma alternativa para a localização da face (Seção 3.2.3).

A opção pelo método baseado em características invariantes, deu-se por sua capacidade de localizar a face independentemente de algum conhecimento sobre sua forma e posição na imagem.

Os métodos baseados na cor da pele para localização de uma face geralmente apresentam ruído devido ou a um fundo complexo ou a objetos (que possuem cor semelhante à da pele humana) que estejam na imagem. Para resolver este problema, utilizou-se um algoritmo de detecção de objetos (Seção 3.2.4). Esse algoritmo mantém na imagem apenas o maior objeto. Como o enfoque deste trabalho é adquirir imagens de uma face, considera-se, por definição, que o maior objeto conexo localizado na imagem é a face.

Todos os métodos que serão apresentados fazem, em algum momento, referência a um ou mais espaços de cores. Por esse motivo, será feita uma breve revisão sobre os espaços de cores necessários para a idealização da aplicação.

3.2.1 Espaços de Cores

O propósito de um espaço de cores é estabelecer um método formal, necessário para se quantificar as sensações visuais das cores que podem ser mais precisamente especificadas. Este método explica as propriedades ou o comportamento das cores em um contexto particular. Basicamente, um espaço de cor possui uma especificação em um sistema de coordenadas tridimensionais. Neste sistema de coordenadas, cada cor é representada por um único ponto desse espaço tridimensional.

Os espaços de cores mais frequentemente utilizados no processamento e análise de imagens são o *RGB*, *HSV*, *YIQ* e *YCbCr*. A seguir, serão apresentados cada um desses espaços baseado em [Gonzalez and Woods, 2000].

²O conceito de características invariantes refere-se à premissa na qual a cor da pele de uma pessoa se mostra homogênea ao longo da face em condições ideais de iluminação

RGB

No espaço *RGB*, cada cor aparece nos seus componentes espectrais primários de vermelho, verde e azul. Baseado num sistema de coordenadas cartesianas, tem como subespaço de cores de interesse um cubo. Na Figura 3.4 está representado o cubo *RGB*, no qual os valores de vermelho, verde e azul estão nos três cantos, levando-se em consideração que o preto é a origem do sistema de coordenadas cartesianas; ciano, magenta e amarelo estão nos outros três cantos. A escala de níveis de cinza (componente *V*, ver Seção 3.2.1 (*HSV*)) está na linha que liga o ponto preto ao ponto branco. Essa informação é muito utilizada no processamento e análise de imagens.

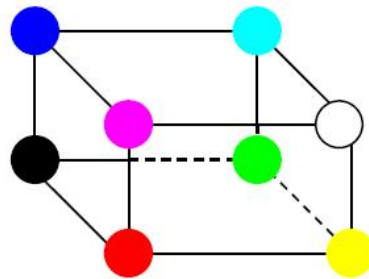


Figura 3.4: Cubo de cores *RGB* [Figueiró, 2003].

Imagens no espaço de cores *RGB* consistem em três planos de imagem independentes. Um para cada cor primária. Quando alimentadas em um monitor *RGB*, essas três imagens combinam-se sobre a tela fosfórea para produzir uma imagem de cores compostas. Assim, o uso do modelo *RGB* para processamento de imagens faz sentido quando as imagens são naturalmente expressas em termos de planos de três cores. Alternativamente, a maioria das câmeras coloridas usadas para aquisição de imagens digitais utilizam o formato *RGB*, o que por si só torna este modelo importante em processamento de imagens.

HSV

A matiz (componente *H*) é um atributo que descreve uma cor pura, enquanto a saturação (componente *S*) dá uma medida do grau de diluição de uma cor pura por luz branca. O espaço *HSV* de cores deve sua utilidade a dois fatores principais: o componente de intensidade *V* é desacoplado da informação de cor da imagem e os componentes de matiz e saturação são intimamente relacionados a percepção humana de cores. Essas carac-

terísticas tornam o espaço de cores HSV uma ferramenta ideal para o desenvolvimento de algoritmos de processamento de imagens baseados em alguma das propriedades do sistema visual humano.

Exemplos da utilização do espaço de cores HSV têm-se desde sistemas de inspeção automática de amadurecimento de frutas a sistemas para correspondência de amostras de cores ou para inspeção de qualidade de produtos coloridos. Nessas aplicações, a chave consiste em basear a operação do sistema sobre propriedades de cores na forma em que uma pessoa usaria aquelas propriedades para desempenhar a tarefa em questão. A Figura 3.5 mostra a representação do espaço de cores HSV em uma forma cônica, bem como a localização de suas componentes.

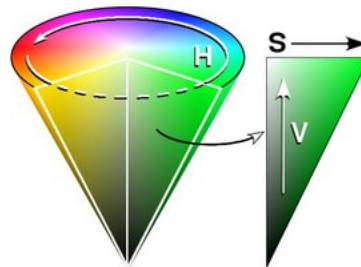


Figura 3.5: Espaço de cores HSV como um objeto cônico [Wikipedia, 2005d].

YIQ

O espaço de cores YIQ é usado na transmissão comercial de televisão colorida. Basicamente, YIQ é uma recodificação de RGB para eficiência da transmissão e para manutenção da compatibilidade com os padrões monocromáticos de televisão. De fato, o componente Y do sistema YIQ fornece todas as informações de vídeo necessárias para um aparelho de televisão monocromático.

O modelo YIQ foi projetado para tirar vantagem da maior sensibilidade da visão humana a mudanças na luminância do que as mudanças de matiz e saturação. Assim, o padrão YIQ possui uma banda mais larga (ou mais bits no caso de imagens digitais) para a componente Y e possui uma banda mais estreita (ou menos bits no caso de imagens digitais) para as componentes I e Q . Em [Pritchard, 1977, Smith, 1978] podem-se achar maiores detalhes sobre essa importante propriedade.

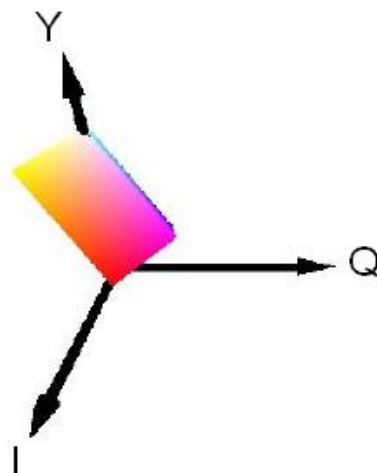


Figura 3.6: Sistema de coordenadas do espaço de cores YIQ [Ribeiro, 2004].

Além de ser um padrão de aceite geral, a principal vantagem do espaço YIQ em processamento de imagens é que a luminância (Y) e a informação de cores (I e Q) são desacopladas. Sabendo-se que a luminância é proporcional à quantidade de luz percebida pelo olho, a importância desse desacoplamento é que pode-se mexer na luminância sem alterar a cor da imagem.

$YCbCr$

O $YCbCr$ é um espaço de cores utilizado em sistemas de televisão e vídeo. O Y é a componente de luminância, ou seja, contém as informações sobre os tons de cinza, no entanto, Cb e Cr são as componentes cromáticas. A componente Cb contém a informação relativa a cor azul, ao passo que a componente Cr contém a informação da cor vermelha [Agostini et al., 2004].

O espaço $YCbCr$, além da sua aplicação à televisão e ao vídeo, é também empregado por alguns formatos digitais de imagem, dos quais o mais importante é o formato JFIF (também, incorretamente, designado por JPEG) [Ribeiro, 2004]. A vantagem do emprego deste modelo é, tal como em televisão, a separação da cor em componentes de luminância e de crominância. Como a vista humana é muito mais sensível à luminância do que à crominância, a informação sobre as componentes de crominância de uma dada cor no espaço $YCbCr$ não necessita de uma gama de valores (e precisão) igual à da sua luminância. Se as componentes $YCbCr$ ocuparem originalmente 8 bits cada, é possível então reduzir

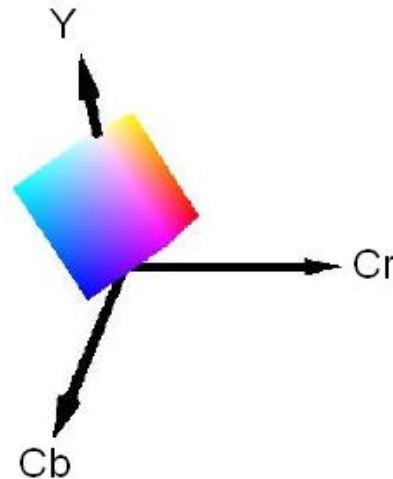


Figura 3.7: Sistema de coordenadas do espaço de cores $YCbCr$ [Ribeiro, 2004].

a representação das componentes de crominância a apenas 2 bits, ou seja, um quarto do espaço inicialmente ocupado e a informação sobre uma cor passa a ocupar apenas 12 bits em vez dos 24 bits iniciais. Uma imagem pode assim ser reduzida a metade do seu tamanho original. Esta técnica é designada por quantização da cor.

3.2.2 Localização pelos espaços de cores YIQ e $YCbCr$

A localização de cores, utilizando o espaço de cores YIQ , é muito utilizada por sua característica de independência entre suas componentes. Em [Buhiyan et al., 2003], baseado no conceito de que a cor da pele tende a se aglomerar numa região do espaço de cores, um *threshold* sobre o espaço YIQ é utilizado para selecionar, na imagem, quais *pixels* são ou não cor de pele. Como a imagem é adquirida no sistema de cores RGB , os valores de cada *pixel* devem ser convertidos para o espaço YIQ . A transformação, segundo [Gonzalez and Woods, 2000], é dada pela Equação 3.1.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.320 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (3.1)$$

Com os valores no espaço de cores YIQ , segundo a condição abaixo, pode-se binarizar a imagem.

$$\begin{aligned} SE (0.234 < Y < 0.781) \text{ E } (0.078 < I < 0.195) \text{ ENTÃO} \\ PRETO \\ SENÃO \\ BRANCO \end{aligned}$$

Após a limiarização binarizada dos *pixels*, tem-se na cor preta as áreas da imagem onde possivelmente seja uma face, e em branco o restante da imagem. O algoritmo de detecção de objetos (Seção 3.2.4) é utilizado para manter na imagem somente o maior objeto preto, retirando os buracos contidos neste objeto.

Outro espaço de cores que apresenta a propriedade de independência entre suas componentes é o $YCbCr$. Sendo conhecido como um método rápido, genérico e robusto para localização de faces, as distribuições cromáticas de cor de pele, Cb e Cr , são utilizadas para localizar a face. Pelo espaço de cores $YCbCr$, a distribuição de cor de pele sempre tem valores que satisfazem o teste lógico abaixo, segundo [Sheng, 2003]. Em seu trabalho, *Sheng* diz que esta característica é independente de raça humana e cor de pele. Entretanto, esta propriedade somente pode ser aplicada a cabelos escuros.

$$\begin{aligned} SE (0.301 < Cb < 0.496) \text{ E } (0.520 < Cr < 0.676) \text{ ENTÃO} \\ PRETO \\ SENÃO \\ BRANCO \end{aligned}$$

Para a aplicação do método de localização proposto por [Sheng, 2003], é preciso converter a cor do *pixel*, do espaço de cores RGB , para o espaço de cores $YCbCr$. Tal transformação é dada pela equação 3.2.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.301 & 0.586 & 0.113 \\ -0.172 & -0.340 & 0.512 \\ 0.512 & -0.430 & -0.082 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 0 \\ 0.500 \\ 0.500 \end{bmatrix} \quad (3.2)$$

Com o valor de cada *pixel* no espaço de cores $YCbCr$, pode-se limiarizar o que é cor de pele e o que não é cor de pele, através da condição definida por [Sheng, 2003]. Valores de *pixel* que estiverem dentro do limiar, passam a ser pretos na imagem. Valores que não estiverem dentro do limiar, passam a ser brancos na imagem.

Após, a limiarização binarizada dos *pixels*, tem-se na cor preta as áreas da imagem onde possivelmente estará a face e em branco o restante da imagem. O algoritmo de detecção de objetos (Seção 3.2.4) é utilizado para manter na imagem somente o maior objeto preto, bem como retirar os buracos existentes neste objeto.

Em nosso trabalho, optou-se por utilizar um método que une os dois espaços de cores (*YIQ* e *YCbCr*). Para cada *pixel* da imagem, se verifica se ele está dentro de algum dos intervalos definidos. Se estiver, este *pixel* é localizado como um ponto da face (cor de pele), caso contrário é descartado. Um método similar foi utilizado em [Singh et al., 2003]. Além disso, o método de detecção de objetos é aplicado para segmentar da região da face.

3.2.3 Localização por médias migratórias

Este método de localização baseia-se na característica de cromaticidade das cores dos *pixels*. Utiliza a informação de que as componentes *H* (matiz) de todos os *pixels* pertencentes à pele humana, do espaço de cores *HSV*, são iguais ou muito próximas. Isso pode ser comprovado pela análise do cone de cores da Figura 3.5. Se seccionar-se o cone perpendicularmente ao seu eixo, se obterá uma circunferência. Nos pontos que formam esta circunferência estão representados os valores da componente *H*.

Como cada imagem (*frame*) adquirida pela câmera está representada no espaço de cores *RGB*, transforma-se cada pixel do espaço de cores *RGB* para o espaço de cores *HSV*, pelas equações seguintes:

$$H = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R - G) + (R - B)]}{[(R - G)^2 + (R - B)(G - B)]^{\frac{1}{2}}} \right\} \quad (3.3)$$

$$S = 1 - \frac{3}{R + G + B} [\min(R, G, B)] \quad (3.4)$$

$$V = \frac{1}{3}(R + G + B) \quad (3.5)$$

Na análise do cone de cores do espaço *HSV*, nota-se que à medida que se desloca para baixo (diminuindo a componente de luminância *V*), a circunferência gerada pela secção, citada anteriormente, diminui. Por esse fato, conclui-se que, quanto menor for a circunferência, mais difícil será a distinção das cores. Todavia, se a luminância é muita alta, *i.e.*, se o

pixel é muito claro, a distinção entre as cores também fica complicada. Portanto, valores de *pixel* com V muito baixos e muito altos são descartados, visto que, quando o V é pequeno, as cores são todas escuras (próximas a preto) e, quando o V é grande, as cores são todas claras (próximas a branco), dificultando a detecção da matiz. O limiar que se utiliza para desconsiderar os valores de V pequenos foi de 50 (na escala de 0 a 255). Para valores grandes de V , considere-se o limiar de 230 (na escala de 0 a 255).

Outra análise que deve ser levada em consideração é em relação a componente de saturação S . Quando se considerar as circunferências com seus menores raios, nota-se que a saturação impede a distinção entre as cores. Portanto, valores com S muito pequenos também são descartados, pois quando o S é pequeno, as cores são pouco saturadas (acinzentadas) e a matiz não poderá ser distinguida. O limiar que utiliza-se para desconsiderar os valores de S pequenos foi de 30 (na escala de 0 a 255).

O ruído gerado pelos *pixels* que foram desconsiderados será tratado quando se aplicar o método de ajuste por uma elipse e o método de detecção de objeto.

A idéia de aplicar um método de agrupamento de cores, para a localização da face, surgiu da análise sobre imagens de face. Notou-se que as imagens possuíam três grandes grupos de cores (fundo, cabelo e face) e que, a partir dessas características, pode-se implementar um algoritmo que agrupe os *pixels* por sua semelhança na componente H .

O algoritmo abaixo mostra o funcionamento deste método:

1. Sorteiam-se três valores iniciais, que representarão cada uma das médias dos três grupos de cor de *pixel*.
2. Para cada *pixel* da imagem, calcula-se a distância³ entre seu valor de H e os valores das médias de cada um dos três grupos. O *pixel* fará parte do grupo em que a distância for menor.
3. Calcula-se a nova média de cada grupo, somando todos os valores de H de cada *pixel* pertencentes ao grupo e dividindo pelo número de *pixels* pertencentes ao grupo.
4. Se a média de todos os grupos não variou com relação ao passo anterior, o processo de iteração pára, senão volta ao passo 2.

³Como a representação geométrica da componente H no cone é uma circunferência, faz-se necessário o cálculo da distância nos dois sentidos; a menor entre elas é considerada.

5. Ao final do processo, cada *pixel* recebe o valor de H correspondente a média do grupo ao qual ele faz parte.

Com esse método conseguiu-se dividir a imagem em três grandes grupos de cor, como mostra a Figura 3.8. Porém, não se utilizará este método para nossa aplicação, pelo fato de ser um processo lento, apresentando resultados semelhantes ao do método anterior (Seção 3.2.2).



Figura 3.8: Imagem antes da aplicação do algoritmo de médias migratórias (à esquerda). Imagem após a aplicação do algoritmo de médias migratorias (à direita).

3.2.4 Detecção de Objetos

Como o método que gera os pontos pertencentes à face (cor de pele) é sensível as condições de iluminação (seção 3.2.2), surge a necessidade de eliminar pontos que foram localizados ocasionalmente, mas que não fazem parte da pele humana. Esse ruído pode ser por um fundo complexo ou por objetos que possuem uma cor semelhante a da face. Para descartar o ruído gerado pela localização de cor de pele (Figura 3.9), utiliza-se um algoritmo que detecta todos os objetos conexos da imagem.

Baseado no trabalho de [Feris et al., 2000], foi aplicada uma operação morfológica de fechamento sobre a imagem para possibilitar a junção de áreas da face que possam estar separadas por algum motivo. Essa técnica se mostrou interessante para casos em que o modelo possui barba, visto que consegue em alguns casos unificar as duas regiões definidas pela barba. A Figura 3.10 mostra um exemplo do efeito resultante do fechamento aplicado a imagem.



Figura 3.9: Imagem original da face (à esquerda). Os pequenos objetos brancos dentro dos objetos pretos são os ruídos gerados pela localização da cor de pele (à direita).



Figura 3.10: Localização da cor da pele (à esquerda) e aplicação do fechamento morfológico à imagem (à direita).

Para detectar objetos dentro da imagem, foi desenvolvido um algoritmo que funciona da seguinte maneira:

- Para cada *pixel* $[i, j]$ da imagem de dimensões $w \times h$, faz-se:
- Analisa-se⁴ os *pixels* $[i+1, j]$ (à direita), $[i, j+1]$ (a baixo) e $[i+1, j+1]$ na diagonal direita para baixo; porém, só considera-se os que possuem a mesma cor do *pixel* $[i, j]$.
- Se algum desses *pixels* já fizer parte de algum objeto, o *pixel* $[i, j]$ fará parte desse mesmo objeto; caso nenhum *pixel* vizinho faça parte de um objeto, todos eles receberão um novo objeto.

⁴A análise dos *pixels* que estão na borda da imagem deve ser tratada como uma exceção. Caso isso não seja feito, pode-se acessar um *pixel* que esteja fora da imagem, o que provocará um erro.

- Pode acontecer de quando ocorrer a análise acima, detectar-se a união de dois objetos até então distintos. Quando isso ocorrer, deve-se torná-los um único objeto.

Este método detecta tanto os objetos pretos (gerados pelos métodos de localização de cor de pele), quanto os objetos brancos (formados pelos *pixels* que não fazem parte da cor da pele).

A ordem, descrita abaixo, de detecção de tais objetos tem que ser seguida, caso contrário o resultado final não fica o esperado.

1. Achar os objetos brancos da imagem.
2. Com exceção do maior objeto branco⁵, todos os outros devem ser “pintados” de preto (retirar os buracos da imagem).
3. Achar os objetos pretos da imagem.
4. Manter na imagem apenas o objeto preto com maior número de *pixels* conexos. Este objeto é considerado, por definição, como sendo a melhor representação da face. Em [Sirohey, 1993] esta definição também é considerada.



Figura 3.11: Detecção de cor de pele (à esquerda). Em destaque, o objeto detectado como fundo da imagem (à direita).

Baseado na heurística de que o maior objeto branco da imagem é o fundo, somente o maior objeto branco é mantido na imagem. Todos os outros objetos brancos (buracos) são pintados de preto. A figura 3.11 mostra, em destaque, o que foi detectado como fundo.

⁵Como a aquisição dos vídeos é feita em um ambiente controlado, por definição, considera-se o maior objeto branco da imagem como o fundo.

Já na figura 3.12, estão ressaltados os buracos da imagem. Para o método de ajuste de uma elipse pelos menores quadrados (Seção 3.2.5), a remoção dos buracos da imagem é de fundamental importância para melhores resultados.

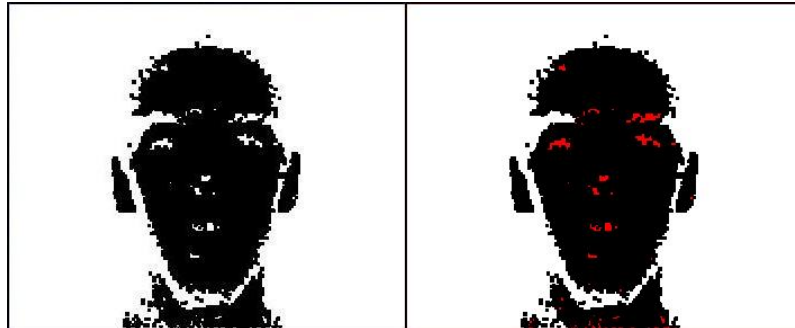


Figura 3.12: Detecção de cor de pele (à esquerda). Em destaque, os objetos detectados como buracos da imagem (à direita).

Após a detecção dos objetos brancos, partiu-se para a localização dos objetos pretos da imagem. Considera-se que o maior objeto preto encontrado na imagem é a face. A figura 3.13 mostra a segmentação do objeto detectado como face na imagem.



Figura 3.13: Imagem original da face (à esquerda). Objeto detectado como face na imagem (à direita).

Uma vez que a face já está localizada, todas as operações feitas sobre a imagem são limitadas por um retângulo (Figura 3.13) que abrange o objeto detectado pelo método de detecção de objetos.

3.2.5 Método de ajuste de uma elipse pelos menores quadrados (*Ellipse-Specific*)

A limiarização binarizada dos pontos que representam a cor da pele (Seções 3.2.2, e 3.2.3), e a detecção do maior objeto representado por esses pontos (Seção 3.2.4), são métodos imprecisos, pois podem não identificar com exatidão a face, como mostra a Figura 3.13. Segundo [Hadid et al., 2002], todas as faces podem ser aproximadas por uma forma elíptica. Portanto, para minimizar o erro, adotou-se a técnica de aproximar uma elipse à face.

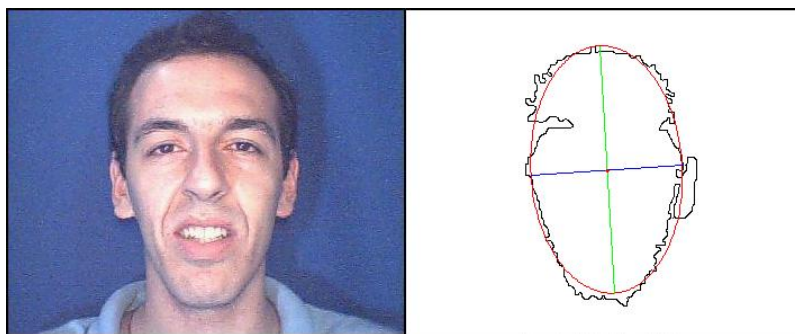


Figura 3.14: Imagem original da face (à esquerda). Elipse aplicada ao contorno externo (à direita).

Para este propósito, usa-se o algoritmo de ajuste de uma elipse pelos menores quadrados (*Ellipse-Specific*) proposto por [Pilu et al., 1996]. Este algoritmo considera uma única elipse que se ajusta a uma nuvem de pontos de face. Para a geração da nuvem de pontos, utiliza-se o maior objeto (Seção 3.2.4) que representa a face. Porém não utiliza-se todo o objeto, mas seus contornos. Os contornos são gerados pelo algoritmo de Canny (Seção 4.2.1). É importante ressaltar que só utiliza-se os contornos externos deste objeto (Figura 3.14), pois se considerar-se os contornos internos (contornos dos buracos), a elipse tenderá a estes pontos (Figura 3.15). E portanto, seus diâmetros irão diminuir. Essa diminuição da elipse não é satisfatória ao problema que foi posta à prova.

Outra informação importante que a elipse pode fornecer é o quão inclinada está a face. Esse dado pode ser adquirido calculando-se o menor e o maior diâmetros da elipse. Com estes dois valores, pode-se calcular o ângulo de inclinação da elipse em relação ao eixo vertical da imagem (Figura 3.16). Tendo o ângulo de inclinação da face, pode-se



Figura 3.15: Imagem original da face (à esquerda). Elipse aplicada aos contornos interno e externo (à direita).

normalizá-la através da rotação de seus *pixels*.

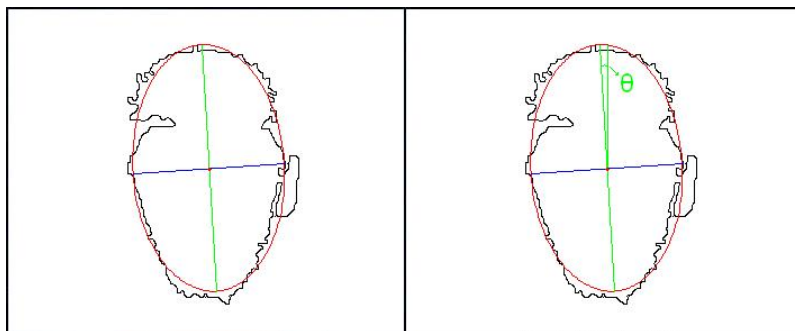


Figura 3.16: Elipse aplicada ao contorno externo (à esquerda). Cálculo do ângulo de inclinação da face (à direita).

Em [Pilu et al., 1996], apresentado no capítulo 3, está demonstrado como funciona o método, e toda sua fundamentação matemática, bem como, no capítulo 4 estão seus testes, que demonstram sua satisfatoriedade para tal aplicação.

Contudo, após a implementação desse método e depois de vários testes, percebeu-se que a elipse não estava retornando uma informação apropriada para nosso problema, não pelo fato do método em si não ser capaz, mas porque nosso método de localização da face (Seções 3.2.2), não retorna exatamente todos os *pixels* da face, o que compromete muito o ajuste da elipse.

Para resolver o problema da elipse não estar retornando uma forma aproximada da face, optou-se por desenvolver uma técnica de união entre o objeto detectado como face e a primeira elipse ajustada a face. Este método faz uma operação de *OR* entre a imagem que contém o objeto face, e uma imagem criada com uma forma elíptica. Após a união das

imagens, recalcula-se os contornos desse novo objeto, a fim de gerar uma nuvem de pontos mais dispersa e com uma forma mais semelhante às linhas da face. A idéia de se fazer essa operação parte do princípio que com a união, os contornos desse novo objeto serão mais condizentes com o da face original. A Figura 3.17 mostra o processo de união, bem como o resultado final da aplicação do método de ajuste de uma elipse (*Ellipse-Specific*).

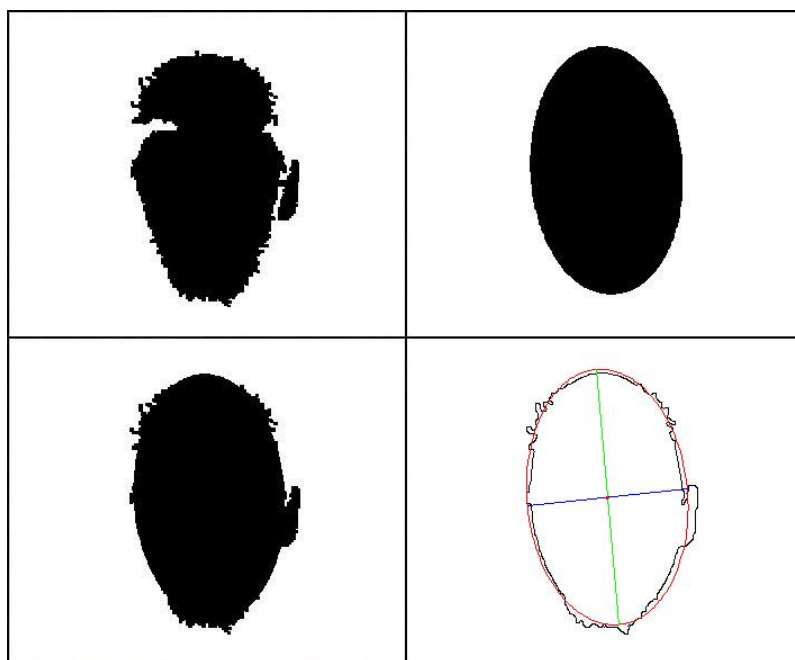


Figura 3.17: Objeto detectado como face, pela cor de pele, e com a aplicação do algoritmo de detecção de objetos (em cima, à esquerda). Objeto criado a partir do contorno da primeira aplicação da elipse (em cima, à direita). Objeto resultante da operação de *OR* entre os dois objetos de cima (em baixo, à esquerda). Elipse gerada a partir dos contornos do novo objeto (em baixo, à direita).

3.2.6 Simetria

Até este momento, na etapa de localização, foi identificada a região correspondente a face através do método de detecção de objetos apresentado na Seção 3.2.4. Essa informação define a região onde a face se encontra dentro da imagem. No entanto, para que se possa seguir adiante no processo de segmentação das características, é preciso garantir que esta região esteja devidamente normalizada.

Os resultados obtidos através do método de ajuste de elipse descrito na Seção 3.2.5

deixam explícita a possível rotação em profundidade que a face pode apresentar. Essa rotação no eixo da profundidade é tradicional do comportamento natural humano, inclusive sendo foco de pesquisa sobre o realismo de animações faciais em [Watt and Watt, 1992] e [Parke and Waters, 1996]. Infelizmente, o cálculo apresentado na técnica da elipse gera apenas uma resposta aproximada e imprecisa do ângulo correspondente à rotação da face.

A fim de superar esse contratempo e garantir a obtenção de informações suficientes para a etapa de normalização, novas formas de localização da face e suas características foram pesquisadas. Considerando a ideologia descrita na Seção 1.2, as técnicas que mais se aplicavam ao curso que o trabalho se propôs a seguir foram baseadas em Biometria⁶.

O estudo sobre a simetria facial tem sido uma das pesquisas mais ativas em trabalhos sobre processamento de faces humanas [Ming et al., 2000]. Essa alternativa se mostrou muito interessante, visto que nossa etapa de localização já possuía a região da face, um dos pré-requisitos para a aplicação do método. A avaliação da região da face é realizada em função da análise da característica estatística de dependência dos níveis de cinza. Na maioria das regiões da face, dado um eixo de simetria correto, a diferença entre os níveis de cinza entre *pixels* equidistantes bilateralmente do eixo deve se manter aproximadamente a mesma *pixel à pixel* [Chen et al., 2005].

Nossa abordagem baseia-se no trabalho de [Chen et al., 2005], onde dois indicadores são utilizados para identificar o melhor eixo de simetria. Para tal, um histograma da diferença em níveis de cinza (GLDH)⁷ é calculado para as possíveis posições do eixo de simetria dentro da região da face. A medida que o eixo iterativo se afasta do verdadeiro eixo de simetria, o GLDH exibe uma alta variância. Desta forma, a variância que é o primeiro indicador utilizado para detecção do eixo de simetria é definida como:

$$\sigma_i^2 = \frac{\sum_{j=1}^h (d_{i,j} - \bar{d}_i)^2}{h - 1} = \frac{\sum_{j=1}^h (d_{i,j}^2) - h\bar{d}_i^2}{h - 1} \quad (3.6)$$

onde h é a altura da região do rosto, d_i é o GLDH do eixo i em questão e \bar{d}_i é a média do GLDH do eixo.

⁶Biometria é o estudo estatístico das características físicas ou comportamentais dos seres vivos. Os sistemas chamados biométricos podem basear seu funcionamento em características de diversas partes do corpo humano, por exemplo: os olhos, a palma da mão, as digitais do dedo, a retina ou íris dos olhos. [Wikipedia, 2005b]

⁷Sigla de *Gray Level Difference Histogram*

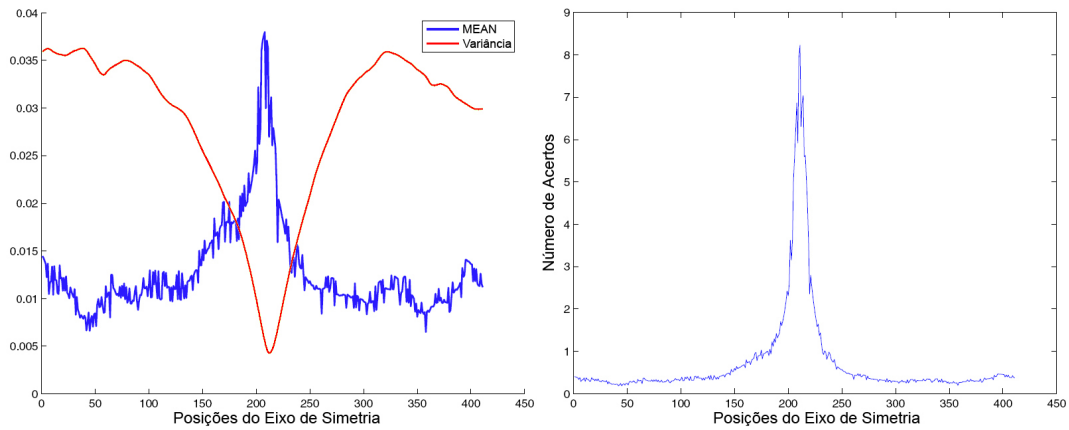


Figura 3.18: Variância e Acerto para cada eixo de simetria (à esquerda). Valores de Acerto (à direita). Adaptada de [Chen et al., 2005].

O segundo critério que [Chen et al., 2005] utiliza para avaliar qual o melhor eixo de simetria é denominado MEAN. MEAN é um quantificador de acerto para um determinado eixo. Assim, a definição do eixo é feita sobre a localização daquele que possui a maior taxa de acerto, ou seja, o que acumular maior número de MEANs. Para isso, utiliza-se um limiar (*threshold*) para definir o grau de semelhança entre os pares de *pixels*. Esse limiar é aplicado sobre a diferença entre os tons de cinza dos dois *pixels* analisados. A Figura 3.18 apresenta os gráficos dos dois indicadores definidos, além de sua combinação definida como

$$Y_i = \frac{MEAN_i}{\sigma_i^2} \quad (3.7)$$

e denominada como *Y score*, que é a utilizada em nossa implementação. Quanto maior o *Y*, maior o grau de simetria encontrado.

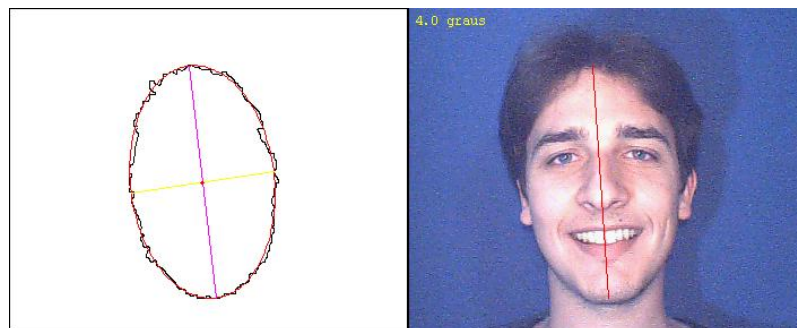


Figura 3.19: Sugestão de angulação no método da Elipse (à esquerda). Eixo de simetria detectado (à direita).

Com relação à rotação da imagem, o trabalho de [Chen et al., 2005] vincula a solução de aplicar a busca pela melhor simetria nas diversas rotações de uma imagem dentro de um intervalo definido como, por exemplo, $[-15^\circ, 15^\circ]$. A abordagem apresentada nesse trabalho é um pouco diferenciada, uma vez que o intervalo não é pré-definido estaticamente. Para tal, faz-se valer a informação adquirida no método da elipse aplicado na Seção 3.2.5. Apesar de impreciso, este método retorna uma boa referência sobre a angulação da face na imagem. Dessa forma, essa informação pode ser utilizada para especificar um conjunto de ângulos a serem testados dentro de seu entorno. Essa alternativa mostrou-se bastante eficiente, uma vez que reduziu significativamente o espaço de busca e por sua vez o custo computacional, além de ampliar o intervalo de rotações aceitáveis da face. No entanto, um entorno de busca considerável deve ser adotado para superar uma possível diferença entre a sugestão de angulação passada da elipse e o eixo de simetria real. A Figura 3.19 mostra um exemplo de como a angulação sugerida pela elipse auxilia no cálculo da simetria.

Capítulo 4

Normalização e Segmentação das Características Faciais

Até o momento foram apresentadas as etapas responsáveis pela localização de uma face e obtenção de algumas informações através de processamento de imagens. Neste capítulo serão demonstradas as soluções implementadas para resolver duas questões fundamentais dentro do processo de extração de características: a normalização e a segmentação.

4.1 Normalização

Conforme foi comentado na Seção 2.2.5, o processo de normalização é responsável pelo realce e restauração de imagens. Nessa seção, serão apresentados dois problemas, normalizados ao longo da implementação do sistema de extração de características. Inicialmente, será tratado o problema das rotações naturais da face, seguido da filtragem necessária para a redução de ruídos inerente a captura de quadros de vídeo.

4.1.1 Rotação

Para a maioria das técnicas de segmentação e extração de características faciais, a face analisada deve se encontrar na posição vertical, não podendo apresentar rotação sobre nenhum eixo cartesiano. No entanto, é natural que mesmo que involuntariamente o modelo humano analisado realize rotações sobre o eixo ortogonal a imagem frontal (profundidade). Assim, após a localização da região da imagem que contém a face, é preciso

buscar informações para que se possa tentar corrigir possíveis rotações indesejadas que inviabilizem o resto do processo.

Pelo fato deste trabalho considerar duas vistas ortogonais da imagem (vista frontal e perfil), as únicas rotações que podem ser compensadas pelo sistema proposto são as realizadas sobre o eixo perpendicular a imagem frontal (profundidade), pois estas não comprometem significativamente a imagem frontal, podendo ser então normalizada. Para este trabalho, a vista frontal tem um papel mais relevante à extração de características do que a vista de perfil, podendo as informações de profundidade serem interpoladas entre cada quadro do vídeo analisado. As rotações sobre os demais eixos são prejudiciais à análise porque podem ocultar características faciais necessárias da imagem frontal, dificultando ou até mesmo inviabilizando o processo de normalização.

Para que se possa rotacionar as imagens obtidas é preciso encontrar alguma informação dentro da região localizada e/ou detectada como face, permitindo identificar o ângulo de inclinação da mesma. Muitos trabalhos, como o de [Rozinaj and Mistral, 2003], utilizam os olhos como referencial para realizar a normalização. Isso implica obrigatoriamente na existência de uma etapa de detecção, onde a localização específica dos olhos seja obtida, além de implicitamente impor possíveis restrições ao modelo analisado.



Figura 4.1: Eixo de simetria detectado (à esquerda) e a imagem frontal rotacionada (à direita)

A utilização do eixo de simetria calculado na etapa de localização, para a definição do ângulo de inclinação da face, foi identificada com a melhor opção para este processo de normalização. A Figura 4.1 expõe o resultado da normalização referente à rotação do eixo de simetria.

4.1.2 Ruído

Durante a análise feita sobre questões relacionadas à aquisição das imagens através de câmeras de vídeo (Seção 3.1), fica clara a necessidade da realização de tratamento para a redução do ruído contido nas imagens adquiridas. A Figura 4.2 expõe o resultado da normalização referente ao ruído encontrado nas imagens. Para corrigir esse problema é utilizada uma convolução cujo *kernel* 3×3 aplica a média da vizinhança-9 sobre a imagem.



Figura 4.2: Imagem original com ruído (à esquerda) e a imagem após aplicada a convolução de média (à direita)

4.2 Segmentação

Uma vez normalizada a imagem e localizada a região que compreende a face, a próxima tarefa a ser realizada é a de segmentar as características faciais internas à ela. A complexidade da execução automática dessa etapa já foi brevemente discutida na Seção 2.2.6.

Neste ponto, deve-se ressaltar que a informação sobre o eixo de simetria da face já é conhecida e que pode ser útil as técnicas aqui empregadas para delimitar as regiões que contém cada elemento da face. As subseções seguintes relatarão as técnicas empregadas nesta etapa.

4.2.1 Contornos

Segundo [Seara, 1998], desde que uma borda é definida por uma mudança no nível de cinza, quando ocorre uma descontinuidade na intensidade, ou quando o gradiente da imagem tem uma variação abrupta, um operador que é sensível a estas mudanças operará

como um detector de bordas. Um operador de derivada faz exatamente esta função. Uma interpretação de uma derivada seria a taxa de mudança de uma função, e a taxa de mudança dos níveis de cinza em uma imagem é maior perto das bordas e menor em áreas constantes. Ao pegarem-se os valores da intensidade da imagem e achar-se os pontos onde a derivada é um ponto de máximo, ter-se-á marcado suas bordas. Dado que as imagens são em duas dimensões, é importante considerar mudanças nos níveis de cinza em muitas direções.

Dentre as informações que podem ser extraídas de uma imagem, a detecção de bordas é a que mais se destaca na tentativa de extrair características faciais, apesar de outros indicadores como cor, intensidade, picos e vales, também terem sua inegável utilidade. Na implementação deste trabalho, foram utilizadas duas técnicas de extração de contornos, ambas baseadas na análise dos gradientes da região localizada como face.

Sobel

Segundo [Gonzalez and Woods, 2000] a maioria das técnicas de detecção de borda baseiam-se na computação de um operador local diferencial. Os operadores de Sobel possuem a vantagem de fornecer, a um só tempo, os efeitos de diferenciação e de suavização. Uma vez que a derivação aumenta o ruído, o efeito de suavização é uma característica particularmente atrativa dos operadores de Sobel.

Uma vez que o procedimento tenha sido completado para todas as posições possíveis, aplicando-se a convolução da máscara, o resultado é uma imagem gradiente do mesmo tamanho que a original. Abaixo as máscaras freqüentemente denominadas operadores de Sobel.

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \text{ e } \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

A aplicação da detecção de contorno utilizando os operadores de Sobel foi necessária, dentro deste trabalho, como passo anterior ao método de projeção integral que será detalhado na Seção 4.2.3.

Algoritmo Canny

Na criação do algoritmo Canny [Canny, 1986], definiu-se um conjunto de requisitos que um detector de bordas deveria satisfazer. São eles:

- **Taxa de erro:** o detector de bordas deveria detectar e achar somente bordas, nenhuma borda deveria faltar;
- **Localização:** a distância entre os pixels de borda encontradas pelo detector de bordas e a borda atual deveriam ser o menor possível;
- **Resposta:** o detector de bordas não deveria identificar múltiplos pixels de borda onde somente exista um único pixel.

Para a aplicação do algoritmo Canny é preciso seguir os seguintes passos:

1. Aplicação de um filtro de suavização Gaussiano;
2. Cálculo do gradiente (magnitude e orientação) sobre a imagem suavizada;
3. Aplicação do método de *Non-maximal Suppression*: percorre o contorno utilizando sua direção e elimina pontos que não são contornos verdadeiros;
4. Aplicação do método de *Hysteresis Thresholding*: aplica dois níveis de limiarização, um superior e outro inferior.

A detecção de contorno utilizando o algoritmo Canny foi empregada principalmente para a segmentação do perfil. Para isso, este foi executado sobre a extração do fundo realizada na etapa que será descrita na Seção 4.2.4.

4.2.2 Métricas

Ao longo do processo de segmentação contido neste trabalho, em vários momentos surgiu a necessidade de adotar métricas que auxiliassem na restrição do espaço de busca pelas características. No caso dos olhos e sobrancelhas, por exemplo, sua região de busca foi restringida a parte superior da área localizada como face. Do mesmo modo, foi utilizado métricas no auxílio à localização do nariz e extração dos pontos de perfil. Alguns trabalhos encontrados na literatura consideram o uso de métricas uma ferramenta útil para a segmentação da face. O trabalho de [Gu et al., 2003] sugere esta possibilidade.

4.2.3 Projeções Integrais

Trabalhos como [Gu et al., 2003, Rozinaj and Mistral, 2003, Lanzarotti et al., 2002, Shan et al., 2000] utilizam projeções integrais sobre contornos, cores e intensidades da imagem para segmentar cada uma das características faciais. Essa técnica mostrou-se bastante interessante pois não apresenta restrições quanto as deformações causadas pelas expressões faciais.

O primeiro passo para aplicação da técnica de projeções integrais é escolher qual informação será integrada e projetada sobre uma determinada direção. No caso específico deste trabalho, foram utilizados operadores de Sobel para calcular os gradientes nos sentidos de interesse para realizar a projeção integral e localizar as características, como mostra a Figura 4.3.

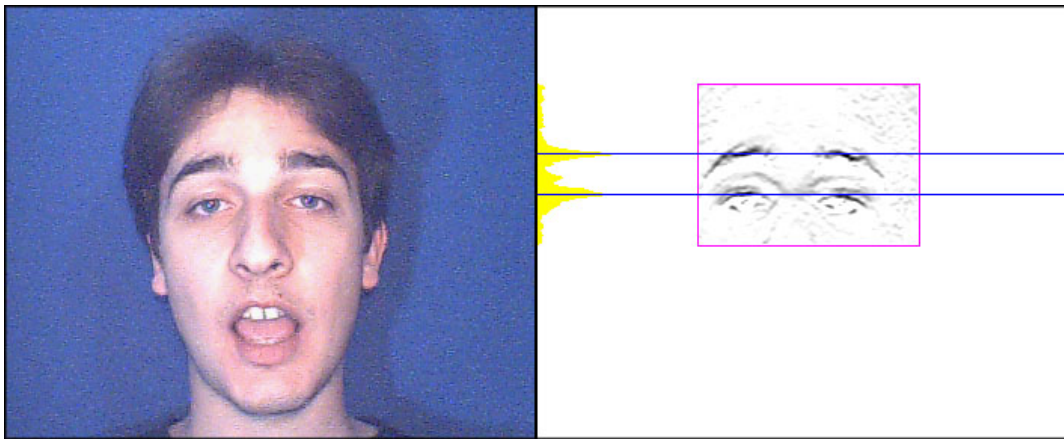


Figura 4.3: Exemplo de segmentação utilizando projeções integrais

4.2.4 Segmentação de fundo

Como o objetivo do trabalho é extrair os pontos da face, para que esta seja, possivelmente, animada em um modelo tridimensional, não basta somente encontrar os pontos de uma imagem bidimensional. Surge a necessidade da extração da informação de profundidade (coordenada Z) de cada ponto da face analisada.

Para tal extração, utiliza-se uma técnica chamada de cromaqui na imagem de perfil. O cromaqui é uma técnica que consiste em filmar uma cena em que o cenário de fundo é totalmente pintado de uma só cor, a qual depois será removida e substituída por uma

imagem, animação gráfica ou outro vídeo¹ [Battaiola et al., 2001]. No caso, não se precisa colocar outro fundo, mas sim segmentar completamente o fundo azul utilizado. A cor azul foi escolhida pela sua grande diferença com a cor da pele humana. Em [Shan et al., 2000], utilizou-se um fundo azul para segmentação do perfil da face.

Como abordagem inicial para excluir o fundo azul, utiliza-se a análise da componente matiz (H) no espaço de cores HSV . O processo de segmentação dá-se sob uma imagem de perfil da face, cujo seu fundo é um pano azul. Um *threshold* foi definido, na condição abaixo, experimentalmente, para separar o fundo azul da face. Esse processo de separação, só é feito do lado onde o nariz da face está (o outro lado não serve como informação relevante para nossa aplicação).

$$(120 < H < 200)$$

Como o equipamento utilizado para a captura das imagens não retorna uma imagem com qualidade, precisa-se, antes de mais nada, aplicar na imagem original uma convolução, expressa pela matriz 4.1.

$$\begin{bmatrix} \frac{2}{115} & \frac{4}{115} & \frac{5}{115} & \frac{4}{115} & \frac{2}{115} \\ \frac{4}{115} & \frac{9}{115} & \frac{12}{115} & \frac{9}{115} & \frac{4}{115} \\ \frac{5}{115} & \frac{12}{115} & \frac{15}{115} & \frac{12}{115} & \frac{5}{115} \\ \frac{4}{115} & \frac{9}{115} & \frac{12}{115} & \frac{9}{115} & \frac{4}{115} \\ \frac{2}{115} & \frac{4}{115} & \frac{5}{115} & \frac{4}{115} & \frac{2}{115} \end{bmatrix} \quad (4.1)$$

O algoritmo abaixo descreve, de maneira resumida, o método de segmentação do fundo azul.

1. Tendo uma imagem auxiliar toda preta.
2. Analisa-se a linha da imagem a procura de um *pixel* que não seja verdadeiro para o limiar definido. Caso seja verdadeiro, “pinta-se” o *pixel* de branco e incrementa a coluna.
3. Quando encontrar um *pixel* diferente de azul o processo pára e pula para a próxima linha (passo 1).

¹Tradicionalmente, a técnica utilizava equipamentos ópticos especiais para a eliminação do fundo. Atualmente, placas gráficas permitem a captura do vídeo por um computador e a sua edição através de programas específicos que possibilitam também a realização do cromaqui.

4. Ao final de todas as linhas, ter-se-á em branco, o fundo, e o resto da imagem auxiliar em preto.

Porém, ao longo dos testes que se realizou, viu-se que a informação de matiz é incerta para baixas e altas luminância e para baixas saturações (mesmo problema da Seção 3.2.3). Portanto, utiliza-se, como informação adicional, os contornos gerados pelo algoritmo de Canny (Seção 4.2.1). Contudo, a área que se considera para a interferência dos contornos é somente os limites superior e inferior da região segmentada como cor de pele da imagem de perfil da face.

O novo método utiliza o mesmo algoritmo apresentado, porém reformula a condição anterior para usar a informação do contorno. Para isso, aplica-se o algoritmo de Canny na imagem de perfil, o que resultará nos contornos da mesma. Agora, além da informação da matiz, de o *pixel* ser azul, também analisa-se a existência de um contorno na imagem correspondente aos contornos gerados pelo Canny. Isso resulta em um perfil muito mais definido e fidedigno da imagem original. A figura 4.4 mostra o comparativo dos dois métodos.

Após ressaltar-se o fundo com a cor branca, aplica-se o algoritmo de Canny para gerar a linha do perfil da face. A figura 4.5 mostra a linha de perfil gerada após a aplicação do método de segmentação do fundo.

Quanto maior for a precisão ao segmentar-se o fundo, maior será a definição da linha do perfil. Para a seção 5.2, a segmentação do fundo é fundamental para seu sucesso na extração dos pontos característicos.

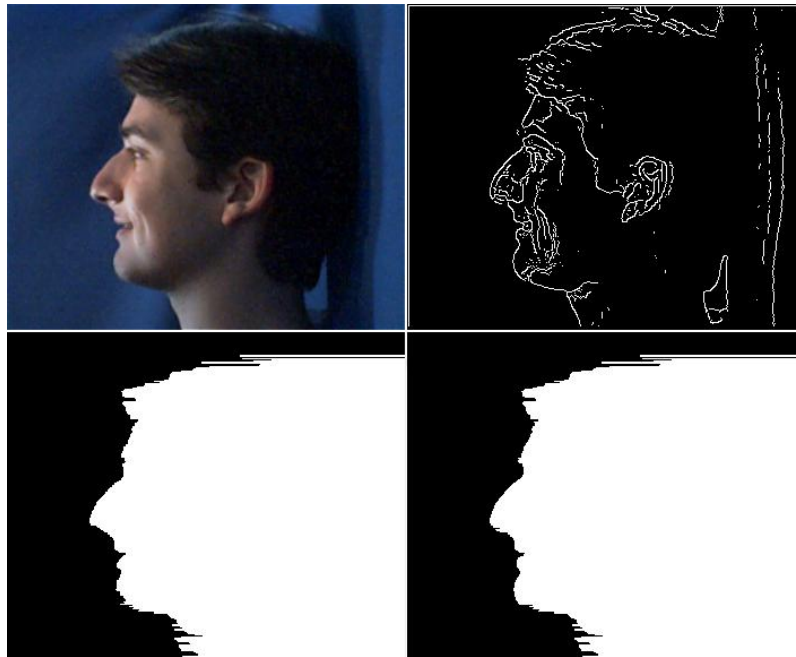


Figura 4.4: Imagem da face original (cima, à esquerda). Detecção dos contornos gerados pelo algoritmo de Canny (cima, à direita). Segmentação do fundo utilizando somente a informação da componente matiz (baixo, à esquerda). Segmentação do fundo utilizando a componente matiz e a imagem dos contornos (baixo, à direita).



Figura 4.5: Imagem com o fundo em branco (à esquerda). Linha de perfil gerada pelo algoritmo Canny (à direita).

Capítulo 5

Extração das Características Faciais

5.1 Extração dos pontos característicos em imagens frontais

Uma vez segmentada a região dos olhos e sombrancelhas através das projeções integrais dos gradientes, a etapa de extração destina-se a refinar a busca internamente as regiões delimitadas. Novas projeções podem ser aplicadas, além de outras técnicas que utilizem, por exemplo, contornos ativos e padrões deformáveis. Por uma restrição de tempo durante a implementação, até o momento atual, o projeto apenas define os limites dessas regiões como sendo as coordenadas dos pontos característicos da face.

Para a extração da ponta do nariz, percorre-se o eixo de simetria da face, da altura inferior dos olhos para baixo. O ponto médio da região (segmento) que tiver a maior luminância, dada pela componente V do espaço de cores HSV , é a ponta do nariz. Em [Shan et al., 2000], este método é utilizado, demonstrando bons resultados, pois a região de maior iluminação utilizado para a aquisição das imagens era frontal à face.

Como se possui o ponto médio horizontal dos olhos e a ponta do nariz, pode-se calcular a distância (como feito na parte de extração das características na imagem de perfil) entre eles e correlacionar com a distância calculada para estes mesmos pontos na extração dos pontos de perfil. Com isso, ter-se-á um fator de multiplicação entre as duas imagens.

Para colocá-las em escala, basta multiplicar uma delas por este fator.

$$D_{O_p N_p} = O_p - N_p$$

$$D_{O_f N_f} = O_f - N_f$$

$$FATOR = \frac{D_{O_p N_p}}{D_{O_f N_f}}$$

Na localização dos pontos característicos da boca, a região de busca é estimada por métricas. Essas medidas são calculadas a partir dos pontos extraídos na imagem de perfil. Uma vez restringidas as regiões de busca pelos pontos característicos da boca, pode-se aplicar em cada uma delas o método de integração.

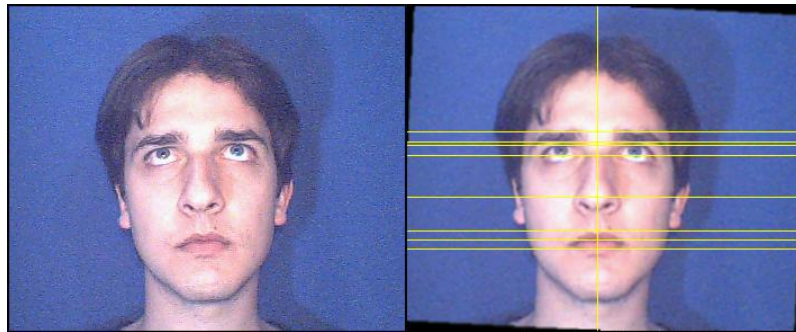


Figura 5.1: Imagem original da face (à esquerda). Linhas que restringem as regiões de busca (à direita)

Contudo, na localização dos pontos correspondentes aos lábios superior e inferior, utiliza-se o método de intersecção entre a linha do eixo de simetria e os contornos gerados (Seção 4.2.1), adaptado de [Zhang, 1997], na região segmentada pelas métricas. A estimação dos pontos candidatos a contorno do lábio consiste em duas partes: a detecção dos contornos (utiliza-se o algoritmo de Canny) e a seleção dos candidatos a pontos labiais.

Este método não obteve bons resultados até o momento, pois os contornos gerados pelo algoritmo de Canny nem sempre geram informação sobre o eixo de simetria. Outra questão é a flexibilidade nas expressões faciais permitidas ao modelo analisado. Estas deformações possibilitam a variação no número de contornos sobre o eixo, o que compromete este método de extração. Por isso, optou-se por mudar a abordagem a ser utilizada na extração dos pontos característicos da boca para o método que será referenciado como uma proposta futura, na Seção 7.2, deste trabalho.

5.2 Extração dos pontos característicos do perfil

Na seção 4.2.4 descreveu-se a técnica utilizada de segmentação do fundo para extração da linha do perfil da face. O presente método depende integralmente do método de extração da linha de perfil. Quanto mais preciso for, maior será a taxa de acerto desse método. Segundo [Shan et al., 2000], todos os pontos (pretos e vermelhos), exceto o ponto verde, dados pela figura 5.2, são pontos fundamentais para o perfil. Utilizando a imagem de perfil, se extrairá somente os pontos vermelhos e o ponto verde inserido na nossa aplicação para obter a informação do canto da boca. Em [Harmon and Hunt, 1978], são utilizados, na extração do perfil, nove pontos. Em nosso trabalho extraem-se oito desses pontos e, por necessidade, desloca-se o nono ponto para a região entre boca e o queixo.

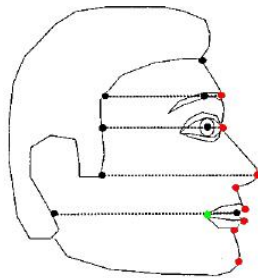


Figura 5.2: Pontos de perfil (vermelho e verde) [Shan et al., 2000].

Para a análise do perfil, cria-se um vetor contendo o valor da coluna onde está o primeiro *pixel* de cor branca. Esse vetor conterá os pontos de uma curva que define o perfil da face. Toda a análise para a extração dos pontos característicos do perfil é feita através desta curva.

Inicialmente, procura-se pela ponta do nariz, considerando que este é o máximo global desta curva e, portanto, o ponto mais à esquerda, como mostra a Figura 5.3.

Após encontrar-se a ponta do nariz, surge o desafio de encontrar os demais pontos do perfil que, ao contrário da ponta do nariz, não são tão triviais de identificar. Partindo da hipótese de que os máximos locais e mínimos locais são pontos de picos e vales do perfil, respectivamente, criou-se um método que pudesse encontrá-los de maneira seqüencial. Este método funciona da seguinte maneira:

- Do nariz pra baixo, procura-se por três picos e três vales, correspondentes ao ponto

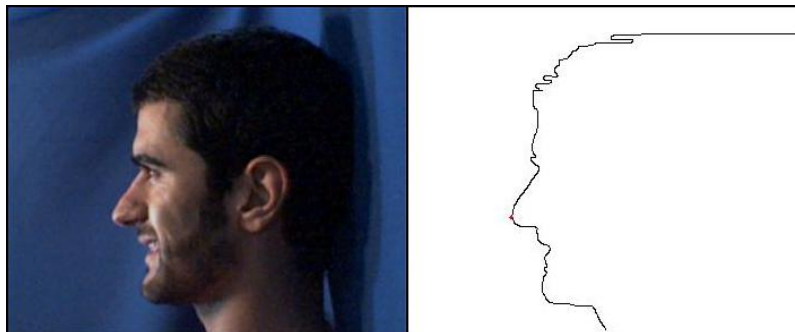


Figura 5.3: Imagem original da face (à esquerda). Ponta do nariz extraída (à direita).

abaixo do nariz, ao lábio superior, ao canto da boca, ao lábio inferior, ao ponto entre a boca e ao queixo, e o queixo.

- Do nariz pra cima, procura-se por um pico e um vale, correspondentes ao meio dos olhos e à sobrancelha.

A busca por picos e vales foi implementada da seguinte maneira, considerando que a face está virada para a esquerda (se a face estiver vira pra direita, troca-se pico por vales e vice-versa, e maior por menor e vice-versa):

- Para cada ponto j da curva de perfil
- Se o ponto anterior $[j-1]$, for menor que o ponto atual $[j]$, se está na iminência de um vale. Se nesse ponto houver uma eminência anterior de pico, esta irá se confirmar.
- Se o ponto anterior $[j-1]$, for maior que o ponto atual $[j]$, se está na iminência de um pico. Se nesse ponto houver uma eminência anterior de vale, esta irá se confirmar.
- Se o ponto anterior $[j-1]$, for igual ao ponto atual $[j]$, computa-se a tamanho desta “reta”.

O problema deste método é que, se por uma imprecisão do processo de segmentação do fundo, o perfil fique ruidoso, tendo picos e vales em locais que não deveriam ser ou até poder-se-ia não achar os pontos do perfil. Isso também pode ocorrer se a imagem estiver com uma qualidade ruim. Para resolver este problema, foi criado um novo método que ordena a busca pelos pontos, baseado na busca do ponto anterior. Portanto, criam-se

regiões de busca pelos pontos. A busca pelos picos e vales é feita da mesma maneira, porém com uma precisão maior.

As regiões de busca não podem ser impostas a região da boca, pois esta contém pontos que tem grande mobilidade (movimento de abrir e fechar a boca).

O método é implementado com uma ordem pré-estabelecida, definida para achar os pontos do perfil.

1. Procura-se pela ponta do nariz (ponto N), da mesma maneira como relatado anteriormente.
2. A partir do ponto N , para cima, procura-se o primeiro vale (ponto O), considerando que na região entre os olhos existe um vale. Calcula-se a distância entre as linhas dos pontos N e O (Equação 5.1), sendo usada como uma métrica para determinar as demais regiões de busca por pontos. Se a remota possibilidade de não se encontrar o vale entre os olhos acontecer, seta-se como tal ponto o meio da maior reta¹ encontrada nessa busca.

$$D_{O_p N_p} = O_p - N_p \quad (5.1)$$

3. Com o cálculo de $D_{O_p N_p}$, procura-se pelo primeiro vale abaixo do nariz (ponto B). Essa busca é feita dentro do intervalo dado pela equação 5.2. Caso não se encontre um vale nesse intervalo, seta-se como o ponto B o início (de cima pra baixo) da maior reta encontrada nessa região (essa reta tem que ter, no mínimo, 4 *pixels*). Se a reta for pequena (menos de 4 *pixels*), o ponto B é definido pela equação 5.3.

$$N \leq B \leq (N + 0,6 * D_{O_p N_p}) \quad (5.2)$$

$$B = N + 0,4 * D_{O_p N_p} \quad (5.3)$$

4. A seguir, parti-se para a busca do pico correspondente a sobrancelha (ponto S). Essa busca é feita dentro do intervalo dado pela equação 5.4. Caso não se encontre um pico nesse intervalo, seta-se como o ponto S o início (de baixo para cima) da maior reta encontrada nessa região.

$$O \geq S \geq (O - 0,5 * D_{O_p N_p}) \quad (5.4)$$

¹A maior reta é a maior dentre as retas encontradas no item 4 da descrição da busca de picos e vales feita acima.

5. O último ponto fixo a ser buscado é o lábio superior (ponto U). A busca do pico correspondente ao lábio superior é feita dentro do intervalo dado pela equação 5.5. Caso não se encontre um pico nesse intervalo, seta-se como o ponto U o fim (de baixo pra cima) da maior reta encontrada nessa região (essa reta tem que ter no mínimo 3 *pixels*). Se a reta for pequena (menos de 3 *pixels*), o ponto U é definido pela equação 5.6.

$$S \leq U \leq (S + 0,35 * D_{O_p N_p}) \quad (5.5)$$

$$U = S + 0,3 * D_{O_p N_p} \quad (5.6)$$

6. Agora, os pontos abaixo do lábio superior podem ter uma mobilidade que não pode ser estimada por métricas. Por isso, procura-se, de maneira seqüencial, até encontrar dois picos e 2 vales, que correspondem ao canto da boca, lábio inferior, ponto entre a boca e o queixo, e o queixo.

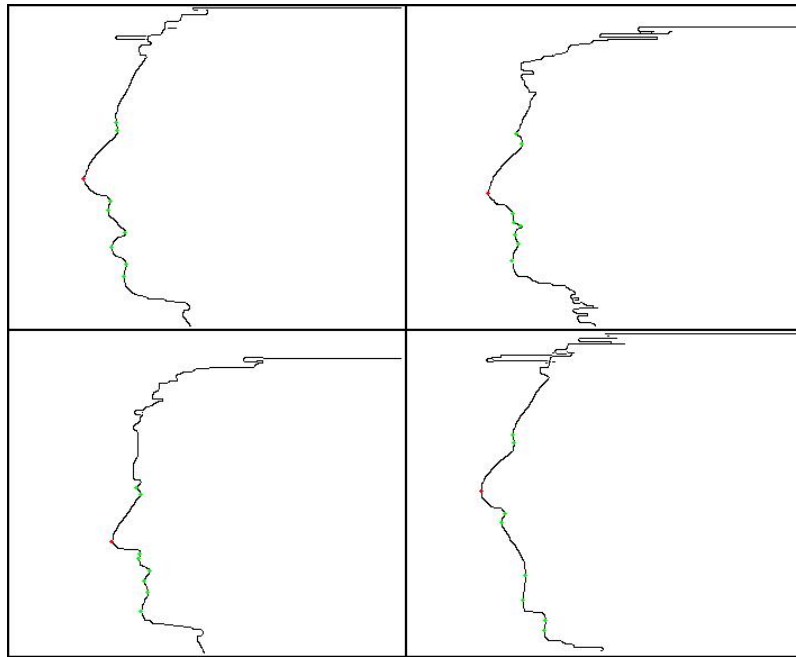


Figura 5.4: Exemplos de extração dos pontos característicos de perfil.

As constantes utilizadas nas equações 5.2, 5.3, 5.2, 5.5 e 5.6, foram estimadas segundo as métricas definidas por [Alattar and Rajala, 1999] e [Gu et al., 2003]. Em [Alattar and Rajala, 1999], utilizou-se um modelo elíptico de uma face com métricas baseadas no tamanho do maior diâmetro desta elipse. Já em [Gu et al., 2003], é citado

um método muito comum para extrair as formas dos olhos, nariz, boca e queixo, pela distância e escala entre estes órgãos. No nosso trabalho, todas as métricas são proporcionais à distância entre a ponta do nariz e os olhos ($D_{O_p N_p}$). Em [Liposcak and Loncaric, 1999], um método semelhante, baseado em escalas, foi utilizado.

Para comprovar os bons resultados adquiridos na aplicação desse método, apresenta-se na Figura 5.4 alguns exemplos da extração dos pontos, em faces distintas, com expressões variadas.

Capítulo 6

Aspectos de Implementação

Nesse capítulo apresentar-se-á nossa aplicação. Aspectos como *interface*, características, recursos e linguagem de programação foram pontos desafiadores e decisivos ao longo deste processo.

6.1 Características

Primeiramente, quando se idealizou os aspectos que se gostaria de abordar, e os problemas que se teria que resolver, surgiu o questionamento sobre qual linguagem de programação utilizar. Nesta escolha, não estava-se interessado em optar por uma linguagem que se dominasse, mas por uma que proporcionasse toda a estrutura e recursos onde se pudesse fazer a aquisição dos vídeos e processá-los.

A alternativa escolhida foi a linguagem de programação JAVA. Na Seção 6.3 se apresentará a estrutura do interpretador JAVA, bem como todas as características que a credenciam como uma das linguagens mais aceitas no âmbito acadêmico.

Na presente seção, se discutirá as características da *interface* e do funcionamento da aplicação como um todo, bem como, apresentar-se-á toda a estrutura criada para dar suporte a aquisição e processamento das imagens (ou *frames*).

Para se criar um ambiente amigável, sugeriu-se a estruturação por projetos. Um projeto é uma abstração do ambiente restrito a um par de vídeos (frontal e lateral), que funciona, basicamente, como uma chave de acesso a estes vídeos e, por consequência a seus *frames*.

Após a criação do projeto, podem-se importar os vídeos. O processo de importação dos vídeos é explicado na Seção 6.3.3. A partir deste momento, pode-se começar a realizar o processamento de cada uma das etapas do método de extração de características faciais.

Para cada projeto, a execução da aplicação retornará todas as etapas envolvidas no processo de extração de características faciais (vídeos (aquisição), localização, normalização, segmentação e extração). Portanto, pode-se visualizar cada uma das etapas em separado. Esta é uma grande vantagem do ponto de vista didático do trabalho.

Abaixo, na Figura 6.1, apresenta-se a *interface* da aplicação.

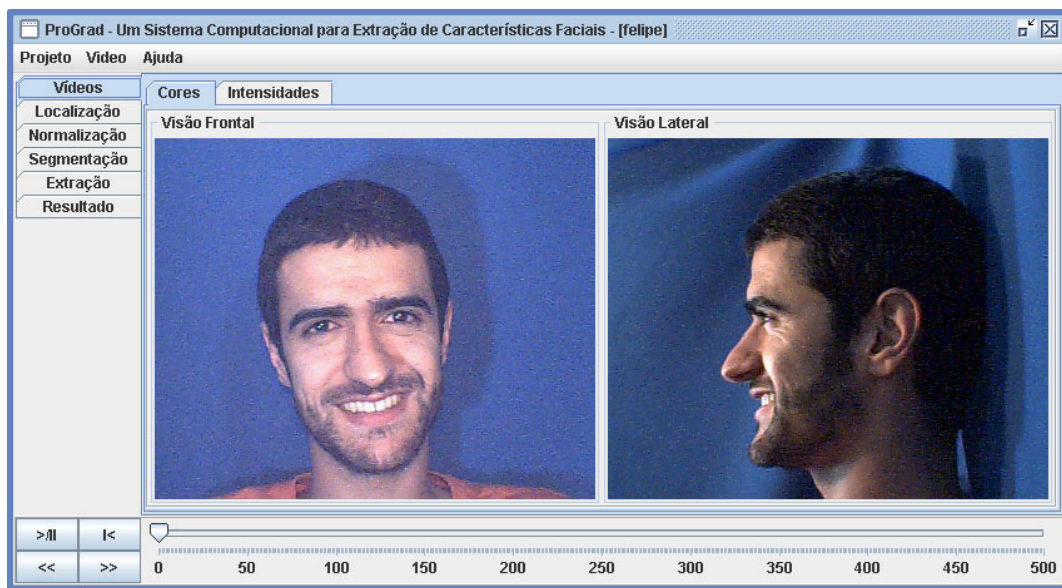


Figura 6.1: *ScreenShot* da tela da aplicação que valida nosso trabalho.

Na esquerda, pode-se ter acesso a cada uma das etapas. À direita, estão as imagens que comprovam as aplicações dos métodos descritos anteriormente.

6.2 Síntese do Processo

Para propiciar um melhor entendimento do sistema para extração de características proposto, a Figura 6.2 expõe cada etapa do processo.

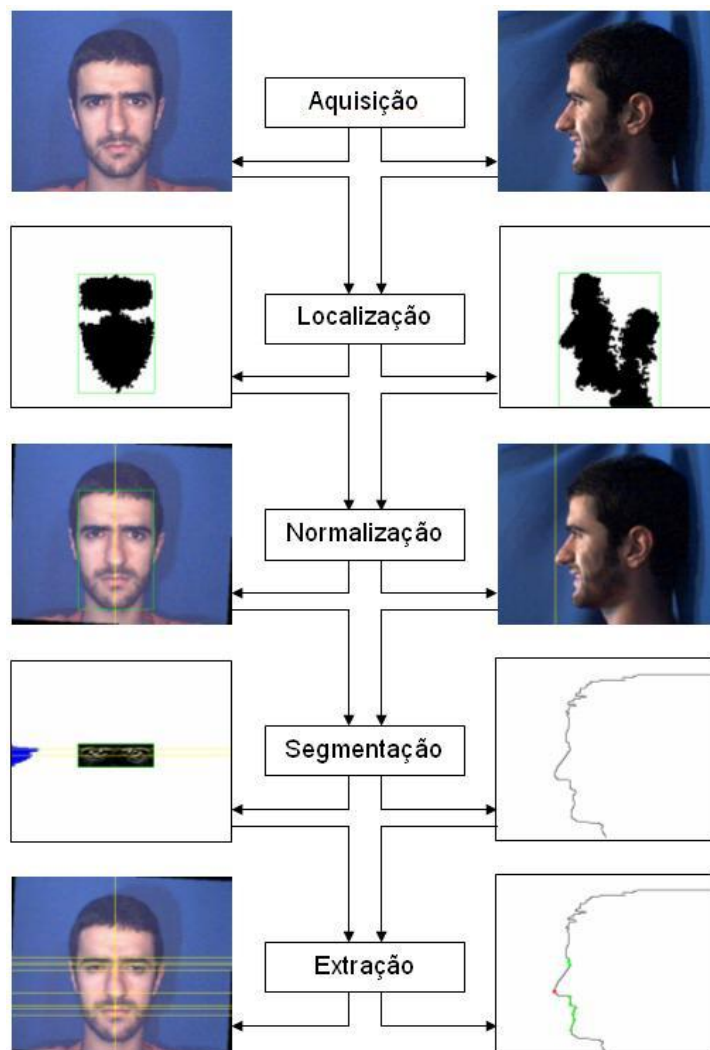


Figura 6.2: Etapas do processo de extração de características proposto.

6.3 Linguagem de Programação

O Java é uma linguagem de programação orientada a objetos que foi desenvolvida para ser suportada em diferentes plataformas e sistemas operacionais [Java, 2005].

Desenvolvida pela Sun Microsystems Java possui um modelo conforme a linguagem de programação C++ e possui recursos especiais que a tornam ideal para o uso na internet. Com Java é possível colocar gráficos interativos e outros efeitos especiais em páginas da Internet utilizando programas especiais em Java chamados Applets, que são executados dentro de um “ambiente hospedeiro” (o browser) [Bicho et al., 2002].

6.3.1 Histórico

Inicialmente o Java foi desenvolvido para programar *chips* de aparelhos domésticos e de uso pessoal como, PDA's, forno de microondas, etc.

A equipe que desenvolvia o projeto na Sun, liderado por James Gosling, tinha inicialmente pensado em utilizar C ou C++, mas descobriu logo cedo que a linguagem não se adequava ao projeto. Como resultado, em 1990, Gosling começou a projetar uma nova linguagem que ele batizou de Oak, segundo ele, por causa da árvore que ele via da sua janela. O nome posteriormente teve que ser mudado por que já existia uma outra linguagem registrada com o mesmo nome.

O projeto avançou, mas o mercado decepcionou e a Sun decidiu redirecionar o rumo do projeto investindo no ramo da TV interativa. O programa durou dois anos e consumiu milhões de dólares.

Consistia no desenvolvimento de um protótipo de controle para TV interativa, usando uma linguagem desenvolvida especialmente para este fim (Oak). Quando o projeto finalmente foi apresentado em 1993, descobriu-se que o mercado de TV interativa não existia, e não havia uma previsão para quando iria deslançar.

O financiamento para o projeto estava para ser cortado e a equipe transferida para outros projetos, quando a Sun decidiu abandonar o hardware do projeto, e focar o seu desenvolvimento na Web. A partir daí, a equipe trabalhou em um ritmo alucinante e em segredo até liberar a primeira versão Alfa da linguagem, em maio de 1995.

6.3.2 Características

Java é uma linguagem de alto nível, com sintaxe similar a do C++. É antes de tudo uma linguagem simples, fortemente tipada, independente de arquitetura, robusta, segura, extensível, bem estruturada, distribuída, multithreaded e com garbage collection [Wester, 2003]. Abaixo está uma descrição de cada uma das características mencionadas:

- **Simplicidade:** Java é muito parecida com C++, mas muito mais simples. Java não possui varias facilidades que possibilitam uma maior flexibilidade na hora de se desenvolver. Alguns exemplos disso são: sobrecarga de operadores, structs, unions, aritmética de ponteiros, herança múltipla, diretivas de pré-processamento

e a memória alocada dinamicamente é gerenciada pela própria linguagem, que usa algoritmos de coleta de lixo (*garbage collection*) para desalocar regiões de memórias que não estão mais sendo usadas.

- **Orientação a Objetos:** Com a exceção dos tipos básicos da linguagem (int, float, double, etc.), a maior parte das estruturas do Java são objetos. O código fica bem organizado na forma de classes, que podem estabelecer relacionamentos de herança simples entre si. Para utilizar os métodos e atributos de uma classe não necessariamente ela precisa ser instanciada, outra facilidade encontrada.
- **Multithreading:** Sistemas mais modernos dão suporte à multitarefa, ou seja, o computador é capaz de executar diversas tarefas ao mesmo tempo. Cada um desses sistemas tem o seu tipo de multitarefa, preemptiva ou não, com ou sem multiprocessamento. Java tem o suporte a multitarefa embutido na linguagem: um programa Java pode possuir mais de uma linha de execução (thread).
- **Exceções:** A máquina virtual Java faz uma verificação em tempo de execução quanto aos acessos de memória, abertura de arquivos e uma série de eventos que podem “travar” em outras linguagens, mas que geram exceções em programas Java.
- **Processamento Distribuído:** Possui várias API¹s que dão suporte a acesso remoto (sockets) e a protocolos utilizados na internet (HTTP, FTP, Telnet, etc.). Isso facilita demais a elaboração de aplicativos baseados em arquiteturas cliente-servidor.
- **Independência de Máquina:** O Java é multiplataforma, *i. e.*, pode ser compilado em uma máquina e rodar em outra, inclusive com sistemas operacionais distintos. O código-fonte em Java é pré-compilado em bytecodes, que são conjuntos de instruções semelhantes ao código de máquina, mas sem serem específicos de qualquer plataforma [Bicho et al., 2002]. O código Java compilado, pode ser executado por um interpretador chamado Máquina Virtual Java (JVM), que é um emulador de processador.

¹Application Programming Interface (API) - conjunto de rotinas e padrões estabelecidos por um software para utilização de suas funcionalidades

- *Garbage Collector*: Em Java, os programadores não necessitam preocupar-se com o gerenciamento de memória. Java, ao contrário, utiliza-se de um conceito de *garbage collection* (coleta de lixo). Sua função é a de varrer a memória de tempos em tempos, liberando automaticamente os blocos que não estão sendo utilizados.

6.3.3 *Java Media Framework (JMF)*

Uma das principais justificativas para a escolha da linguagem JAVA para a implementação do sistema aqui apresentado foi exatamente a existência de uma API própria para a manipulação de vídeos. A JMF permitiu que fossem criados CODECs para acessar *frame* e *frame* os vídeos selecionados dentro do sistema. Dessa forma, cada *frame* pôde ser armazenado vinculado ao projeto que o importou. Essa disposição se mostrou bastante útil para a finalidade buscada, uma vez que possibilitou o acesso incondicional a cada *frame* salvo. Para que as etapas do processo de extração de características fossem processadas em etapas e não necessariamente em tempo real.

Capítulo 7

Considerações Finais

Os objetivos traçados na proposta deste trabalho foram parcialmente alcançados. Um grande apanhado de métodos foi estudado para tornar viável esta implementação, visto que o prévio conhecimento da linha de pesquisa deste trabalho era muito superficial. A seguir se relatará as dificuldades encontradas e as propostas futuras para este trabalho.

7.1 Dificuldades Encontradas

Desde o início, encontraram-se dificuldades na realização de nosso trabalho. Com o total desconhecimento da área que se optou por desenvolver nossa pesquisa, começou-se sempre dos aspectos mais básicos até propostas mais elaboradas para a resolução do problema. Esse foi o motivo pelo qual se demorou muito para começar a implementação propriamente dita.

Utilizou-se, para a aquisição dos vídeos, recursos simples e com qualidade limitada, sem nenhum tipo de calibração, o que provocou a inserção de diversos ruídos em nossas imagens.

A iluminação do ambiente de aquisição dos vídeos foi feita de maneira simples e não parametrizada. Ao realizarem-se os primeiros testes com os vídeos que se adquiriu utilizando lâmpadas de luz fluorescente, os métodos de localização de cor de pele não funcionavam. Depois de um tempo, detectou-se que o problema era a luz fluorescente. Adicionou-se uma iluminação com lâmpadas incandescentes e o método passou a funcionar.

Na aquisição dos vídeos, utilizaram-se duas *webcams* independentes. Porém, por restrições de *software*, os vídeos foram adquiridos em computadores distintos, o que acarretou na não sincronização de alguns vídeos, agravada pela variação dos *framerates* de cada câmera. Dessa forma, a sincronização teve de ser feita manualmente.

A linguagem de programação (JAVA) utilizada para o desenvolvimento da aplicação, não era de nosso domínio. Por isso, foi desafiante o aprendizado da linguagem ao longo do projeto. Em contra partida, sua escolha foi de total satisfatoriedade para o sucesso de nosso sistema.

7.2 Propostas Futuras

Atualmente, a aquisição dos vídeos é feita em um ambiente propício, mas não totalmente controlado. Nossa perspectiva é dar maior flexibilidade ao sistema para ser aplicado em ambientes complexos.

Para isso, deve-se aprimorar o método de localização dos pontos de cor de pele. Atualmente, esta localização é feita segundo um limiar nos espaços de cores *YIQ* e *YCbCr*, porém em [Feris et al., 2000, Wang and Sung, 1999, Saber and Tekalp, 1998] foi utilizado uma distribuição gaussiana para modelar a região de cor de pele humana.

Por se tratar de uma aplicação que visa extrair pontos característicos de uma face, para que estes sejam animados em um modelo virtual tridimensional em [Körting, 2005], o aperfeiçoamento do processo de extração dos pontos característicos da região da boca é de fundamental importância para dar maior realidade ao processo. A aplicação de *deformable templates* proposta em [Yuille et al., 1992], apresentou grandes resultados para a estimação dos pontos característicos da boca em [Zhang, 1997, Reinders et al., 1993, Rudianto and Ngan, 1996]. Este é o método que se gostaria de, futuramente, implementar em nosso trabalho.

Nosso objetivo final é integrar completamente nosso sistema ao trabalho de [Körting, 2005] que faz a animação dos pontos características extraídos *frame a frame*.

Bibliografia

- [Abrantes and Pereira, 1999] Abrantes, G. A. and Pereira, F. (1999). MPEG-4 Facial Animation Technology: Survey, Implementation and Results. In *IEEE Transactions on Circuits and Systems for Video Technology*, volume 9, pages 290–305.
- [Agostini et al., 2004] Agostini, L. V., Silva, I. S., and Bampi, S. (2004). Parallel color space converters for JPEG image compression. 44(4):697–703.
- [Alattar and Rajala, 1999] Alattar, A. M. and Rajala, S. A. (1999). Facial features localization in front view head and shoulders images. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 6, pages 3557–3560.
- [Battaiola et al., 2001] Battaiola, A. L., de G. Domingues, R., Feijó, B., Swarcman, D., Clua, E. W. G., Kosovitz, L. E., Dreux, M., Pessoa, C. A., and Ramalho, G. (2001). Desenvolvimento de Jogos em Computadores e Celulares. *RITA*, VIII(2):7–46.
- [Bicho et al., 2002] Bicho, A. L., da Silveira Jr., L. G., da Cruz, A. J. A., and Raposo, A. B. (2002). Programação Gráfica 3D com OpenGL, Open Inventor e Java 3D. In *Revista Eletrônica de Iniciação Científica (REIC)*.
- [Buhiyan et al., 2003] Buhiyan, M. A. A., Ampornaramveth, V., and Ueno, S. Y. H. (2003). Face Detection and Facial Feature Localization for Human-machine Interface. *Nll*, (5).
- [Canny, 1986] Canny, J. (1986). A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698.

- [Chan et al., 2004] Chan, M., Chia-Yen, C., Barton, G., Delmas, P., Gimel'farb, G., Leclercq, P., and Fischer, T. (2004). Evaluation of 3D face analysis and synthesis techniques. In *ICME '04 – IEEE International Conference on Multimedia and Expo*, volume 2, pages 1203–1206.
- [Chen et al., 2005] Chen, X., Flynn, P. J., and Bowyer, K. W. (2005). Fully Automated Facial Symmetry Axis Detection in Frontal Color Images. In *Fourth IEEE Workshop on Automatic Identification Advanced Technologies*, pages 106–111.
- [da Silva and de Castro Santa Rosa, 2003] da Silva, P. Q. and de Castro Santa Rosa, A. N. (2003). Face Recognition Based on Eigeneyes. 13(2):419–423.
- [Feris, 2001] Feris, R. S. (2001). Rastreamento Eficiente de Faces em um Subespaço Wavelet. Master's thesis, IME-USP.
- [Feris et al., 2000] Feris, R. S., Campos, T. E., and Cesar-Jr., R. M. (2000). Detection and Tracking of Facial Features in Video Sequences. In *Lecture Notes in Artificial Intelligence*, volume 1973, pages 129–137.
- [Figueiró, 2003] Figueiró, T. (2003). Conversão RGB-HSV-GrayScale. disponível em <http://www.lapsi.eletro.ufrgs.br/lili/doc/image-processing/conversao.pdf>. acessado em 15/12/2005.
- [Garcia et al., 2001] Garcia, E., Dugelay, J.-L., and Delingette, H. (2001). Low cost 3D face acquisition and modeling. In *ITCC 2001, IEEE International Conference on Information Technology: Coding and Computing, April 2-4, 2001, Las Vegas, USA*.
- [Gonzalez and Woods, 2000] Gonzalez, R. C. and Woods, R. E. (2000). *Processamento de Imagens Digitais*. Editora Edgard Blücher Ltda.
- [Gu et al., 2003] Gu, H., Su, G., and Du, C. (2003). Feature Points Extraction from Faces. In *Image and Vision Computing NZ, Palmerston North*, pages 154–158, Beijing, China.
- [Hadid et al., 2002] Hadid, A., Pietikainen, M., and Martinkauppi, B. (2002). Color-based face detection using skin locus model and hierarchical filtering. pages IV: 196–200.

- [Harmon and Hunt, 1978] Harmon, L. D. and Hunt, W. F. (1978). Automatic recognition of human face profiles. *Computer Graphics and Image Processing*, 6(2):135–156.
- [IHC, 2005] IHC (2005). *IHC Brasil*. Disponível em <http://www.serg.inf.puc-rio.br/ihc/>. Acessado em 13/12/2005.
- [Java, 2005] Java, P. (2005). O que é Java? disponível em <http://www.portaljava.com.br/>. acessado em 26/12/2005.
- [Körting, 2005] Körting, T. S. (2005). Animador Tridimensional de Faces. In *Projeto de Engenharia de Computação (FURG)*, Rio Grande, RS, Brasil.
- [Lanzarotti et al., 2002] Lanzarotti, R., Borghese, N. A., and Campadelli, P. (2002). Robust identification and matching of fiducial points for the reconstruction of 3D human faces from raw video sequences. In *First International Symposium on 3D Data Processing Visualization and Transmission*, pages 690–693.
- [Lee et al., 1995] Lee, Y., Terzopoulos, D., and Walters, K. (1995). Realistic modeling for facial animation. In *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 55–62, New York, NY, USA. ACM Press.
- [Liposcak and Loncaric, 1999] Liposcak, Z. and Loncaric, S. (1999). A Scale-Space Approach to Face Recognition from Profiles. In *CAIP '99: Proceedings of the 8th International Conference on Computer Analysis of Images and Patterns*, pages 243–250, London, UK. Springer-Verlag.
- [Martins, 2005] Martins, I. (2005). Pesquisa em bases fotográficas, a partir do retrato falado - uma solução para integração através da antropometria de Bertillon.
- [Ming et al., 2000] Ming, Y., Yingchun, G., and Youan, K. (2000). Human face orientation estimation using symmetry and feature points analysis. In *WCCC-ICSP 2000. 5th International Conference on Signal Processing Proceedings*, volume 2, pages 1419–1422.
- [Parke and Waters, 1996] Parke, F. I. and Waters, K. (1996). *Computer facial animation*. A. K. Peters, Ltd., Natick, MA, USA.

- [Pilu et al., 1996] Pilu, M., Fitzgibbon, A., and Fisher, R. (1996). Ellipse-specific direct least-square fitting.
- [Precioso et al., 2005] Precioso, F., Barlaud, M., and Blu, T. (2005). Robust Real-Time Segmentation of Images and Videos Using a Smooth-Spline Snake-based algorithm. volume 14.
- [Pritchard, 1977] Pritchard, D. H. (1977). U.S. Color Television Fundamentals. *IEEE Trans. Consumer Electronics*, CE-23(4):467–478.
- [Reinders et al., 1993] Reinders, M., Odijk, F., van der Lubbe, J., and Gerbrands, J. (1993). Tracking of global motion and facial expressions of a human face in image sequences. In *Visual Communications and Image Processing'93*, volume SPIE Vol. 2904, pages 1516–1527, Cambridge.
- [Ribeiro, 2004] Ribeiro, N. (2004). *Multimédia e Tecnologias Interactivas*. Fca - Editora Informática.
- [Rozinaj and Mistral, 2003] Rozinaj, G. and Mistral, F.-L. (2003). Facial Features Detection For 3D Face Modeling. *4th International Conference on Industrial Tools*, 03:951–954.
- [Rudianto and Ngan, 1996] Rudianto, R. and Ngan, K. (1996). Automatic 3D wireframe model fitting to frontal facial image in model-based video coding. In *Picture Coding Symposium (PCS'96)*, volume SPIE Vol. 2904, pages 585–588.
- [Saber and Tekalp, 1998] Saber, E. and Tekalp, A. M. (1998). Frontal-view Face Detection and Facial Features extraction Using Color, Shape and Symetry Based Cost Functions. pages 669–680. Ed. Elsevier.
- [Seara, 1998] Seara, D. M. (1998). Algoritmos para Detecção de Bordas. disponível em <http://www.inf.ufsc.br/~visao/1998/seara/>.
- [Shan et al., 2000] Shan, S., Gao, W., Yan, J., Zhang, H., and Chen, X. (2000). Individual 3d face synthesis based on orthogonal photos and speech-driven facial animation. In *ICIP*.

- [Sheng, 2003] Sheng, Y. (2003). Fast and Automatic Facial Feature Extraction for 3-D Model-Based Video Coding. *Research Excellence Awards Competition 2003*.
- [Singh et al., 2003] Singh, S. K., Chauhan, D. S., Vatsa, M., and Singh, R. (2003). A Robust Skin Color Based Face Detection Algorithm. In *Tamkang Journal of Science and Engineering*, volume 6, pages 227–234.
- [Sirohey, 1993] Sirohey, S. A. (1993). Human Face Segmentation and Identification. Technical Report CS-TR-3176.
- [Smith, 1978] Smith, A. R. (1978). Color gamut transform pairs. In *SIGGRAPH '78: Proceedings of the 5th annual conference on Computer graphics and interactive techniques*, pages 12–19, New York, NY, USA. ACM Press.
- [Su and Yin, 2004] Su, S. and Yin, B. (2004). A robust face detection method. In *Third International Conference on Image and Graphics*, pages 302–305.
- [Tang and Huang, 1996] Tang, L.-A. and Huang, T. S. (1996). Automatic construction of 3D human face models based on 2D images. In *International Conference on Image Processing*, volume 3, pages 467–470.
- [Valente and Dugelay, 2000] Valente, S. and Dugelay, J.-L. (2000). Face Tracking and Realistic Animations for Telecommunicant Clones. *IEEE MultiMedia*, 7(1):34–43.
- [Wang and Sung, 1999] Wang, J. G. and Sung, E. (1999). Frontal-view Face Detection and Facial Features extraction Using Color and Morphological Operations. *Pattern Recognition Letters* 20, pages 1053–1068.
- [Watt and Watt, 1992] Watt, A. and Watt, M. (1992). *Advanced animation and rendering techniques : theory and practic*. Addison-Wesley.
- [Wester, 2003] Wester, I. (2003). Linguagem JAVA. disponível em <http://www.infowester.com/>. acessado em 26/12/2005.
- [Wikipedia, 2005a] Wikipedia (2005a). Avatar (Realidade Virtual). disponível em [http://pt.wikipedia.org/wiki/Avatar_\(Realidade_Virtual\)](http://pt.wikipedia.org/wiki/Avatar_(Realidade_Virtual)). acessado em 28/12/2005.

- [Wikipedia, 2005b] Wikipedia (2005b). Biometria. disponível em <http://pt.wikipedia.org/wiki/Biometria>. acessado em 28/12/2005.
- [Wikipedia, 2005c] Wikipedia (2005c). CCD. disponível em <http://pt.wikipedia.org/wiki/CCD>. acessado em 31/12/2005.
- [Wikipedia, 2005d] Wikipedia (2005d). HSV Color Spaces. disponível em http://en.wikipedia.org/wiki/Image:HSV_cone.jpg. acessado em 14/12/2005.
- [Yang et al., 2002] Yang, M.-H., Kriegman, D. J., and Ahuja, N. (2002). Detecting Faces in Images: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(1):34–58.
- [Yokoyama et al., 1996] Yokoyama, T., Wu, H., and Yachida, M. (1996). Automatic detection of facial feature points and contours. In *Robot and Human Communication, 1996., 5th IEEE International Workshop*, pages 335–340.
- [Yuille et al., 1992] Yuille, A. L., Hallinan, P. W., and Cohen, D. S. (1992). Feature extraction from faces using deformable templates. *Int. J. Comput. Vision*, 8(2):99–111.
- [Zhang, 1997] Zhang, L. (1997). Estimation of the mouth features using deformable template matching. pages 328–335.
- [Zhao et al., 2003] Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. (2003). Face recognition: A literature survey.