

**Animação de faces tridimensionais
baseada em registro estrutural**

Andréa Britto Mattos

DISSERTAÇÃO APRESENTADA
AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA
UNIVERSIDADE DE SÃO PAULO
PARA
OBTENÇÃO DO TÍTULO
DE
MESTRE EM CIÊNCIAS

Programa: Ciência da Computação
Orientador: Prof. Roberto Marcondes Cesar Jr.

Durante o desenvolvimento deste trabalho, a autora recebeu auxílio financeiro da
FAPESP

São Paulo, Março de 2011

Animação de faces tridimensionais baseada em registro estrutural

Esta dissertação trata-se da versão original
da aluna Andréa Britto Mattos.

Agradecimentos

Em primeiro lugar, gostaria de agradecer aos meus pais, Rita e Alvaro, pelo amor e boa formação que me proporcionaram desde criança, estimulando sempre disciplina e dedicação. Sei que não poderia alcançar as minhas metas se não fossem as oportunidades que foram concedidas a mim graças ao esforço deles e dos meus avós, e por isso serei sempre grata.

Dedico também um agradecimento muito especial ao meu orientador, Roberto Marcondes Cesar Jr., por quem tenho uma grande admiração, por ter me guiado muito bem nos meus estudos desde o último ano da graduação. Suas ideias, correções, confiança, incentivo e entusiasmo permitiram com que eu pudesse crescer muito enquanto pesquisadora.

Agradeço aos membros da banca do meu exame de qualificação, os professores Carlos Hitoshi Morimoto e Marcio Lobo Netto, pelas críticas que permitiram aprimorar a qualidade do texto da minha dissertação.

Gostaria de agradecer à colaboração do professor Luiz Velho, do IMPA-RJ, e ao colega e amigo Jesús Mena-Chalco, com quem tive o imenso prazer de trabalhar durante o mestrado e cujas discussões e dicas foram de extrema importância para a nossa pesquisa. Agradeço aos demais membros e professores do grupo de Visão Computacional do IME-USP, pela ajuda e convivência ao longo destes anos na Universidade, em particular às amigas Maysa e Sílvia e ao professor Roberto Hirata Jr.

Agradeço a minha irmã Manuela pela torcida, amor e companheirismo. Agradeço ao Paulo por todo o apoio, paciência e carinho. Também gostaria de agradecer aos meus tios, primos e amigos, que fizeram com que estes anos de esforço pudessem contar também com momentos de alegria.

Finalmente, sou grata ao apoio financeiro concedido pela FAPESP, que permitiu com que eu pudesse me dedicar em tempo integral à pesquisa.

Resumo

O trabalho desenvolvido consiste em um sistema de animação facial que utiliza modelos de pessoas reais, obtidos por um escâner 3D. O banco de dados considerado é composto por modelos exibindo diferentes expressões faciais que são animados a partir de duas técnicas. Um dos métodos consiste em aplicar um processo de interpolação linear utilizando as diferentes malhas do escâner como poses-chaves, enquanto, na outra abordagem, geram-se novas expressões faciais a partir de uma face no estado de neutralidade, utilizando um modelo paramétrico baseado no padrão MPEG-4. Em ambos os casos, é necessário buscar correspondências ponto-a-ponto entre malhas faciais: para o primeiro, devem ser casadas as malhas referentes às poses-chaves interpoladas, e para o segundo, é necessário adaptar a malha neutra à um modelo paramétrico que irá definir a sua deformação durante o processo de animação.

Propomos uma alternativa para a busca de correspondências densas entre as malhas faciais, introduzindo a ideia de selecionar um pequeno conjunto de pontos e definir uma triangulação inicial refinada através de subdivisões consecutivas que computam os casamentos entre os pontos intermediários. Propomos uma abordagem para realizar de forma automática a marcação do conjunto inicial de pontos, a partir de uma face 3D na qual os marcos foram previamente selecionados. A técnica baseia-se em métodos estruturais e casamento de grafos, levando em conta informação de textura, geometria e curvatura das faces a serem casadas.

Mostraremos como o método proposto pode ser aplicado em diferentes problemas, permitindo a transferência de expressões faciais entre diferentes tipos de modelos (reais e artificiais), animação de faces obtidas via fotos 2D e geração automática de caricaturas. A metodologia proposta exibiu bons resultados que possibilitam a aplicação dos métodos estudados para a resolução de outros problemas relacionados à manipulação de faces pelo computador.

Palavras-chave: Animação facial, registro tridimensional, métodos estruturais, transferência de expressões faciais.

Abstract

This work consists of a facial animation system that uses real three-dimensional models of people, acquired by a 3D scanner. The considered database is composed by models displaying different facial expressions that are animated using two techniques. One method is to apply a linear interpolation process using the scanned meshes as keyframes, while the other approach generates new facial expressions from a neutral face, using a parametric model based on the MPEG-4 animation standard. In both cases, it is necessary to define point-to-point correspondences between the facial meshes: in the first one, the interpolated meshes should be registered and, in the second, it is necessary to adapt the neutral mesh to a parametric model that will define its deformation during the animation process.

We propose an alternative to the dense correspondence computation by introducing the idea of selecting a small set of points and setting an initial triangulation refined through a subdivision process that compute the matching between intermediate points. We propose an approach to automatically detect the initial set of points, given a 3D face in which the landmarks are previously selected. The method is based on structural methods and graph matching, taking into account texture, geometry and curvature information of the faces to be registered.

We will show how the method can be applied to different problems, enabling facial expression transfer between different model types (both real and artificial), animation of reconstructed faces from 2D photographs and automatic caricature generation. The proposed methodology has obtained good results that can be used to solve other problems related to computer facial manipulation.

Keywords: Facial animation, three-dimensional registration, structural methods, facial expression transfer.

Sumário

Lista de Abreviaturas	ix
Lista de Símbolos	xi
Lista de Figuras	xiii
Lista de Tabelas	xvii
1 Introdução	1
1.1 Motivação	1
1.2 Síntese bibliográfica	1
1.2.1 Métodos de animação facial	1
1.2.2 Busca de correspondências entre malhas faciais	4
1.2.3 Detecção automática de pontos característicos faciais	6
1.3 Objetivos	7
1.4 Organização do documento	8
2 Banco de dados de faces reais	11
2.1 Aquisição	11
2.2 Pré-processamento	12
2.3 Indivíduos	13
3 Registro estrutural de faces 3D	17
3.1 Métodos estruturais	17
3.1.1 Casamento bidimensional	18
3.2 Casamento tridimensional	19
3.2.1 Custo de aparência	22
3.2.2 Custo de estrutura	27
3.3 Avaliação do método e resultados	31
3.3.1 Resultados para o casamento entre expressões faciais diferentes	31
3.3.2 Resultados para o casamento entre indivíduos diferentes	36
3.4 Subdivisão e projeção	42

4	Método de animação por interpolação	45
4.1	Processo de animação	46
4.2	Aplicações	48
4.2.1	Sistema de reconstrução facial	49
4.2.2	Sistema de síntese de voz	50
4.2.3	Geração automática de caricaturas	51
4.3	Resultados	52
5	Método de animação por modelo paramétrico	59
5.1	Processo de animação	60
5.1.1	FAPs e FPs	60
5.1.2	Modelo	61
5.1.3	Regiões de vizinhança	62
5.2	Expressões faciais e visemas no MPEG-4	65
5.3	Resultados	66
6	Considerações finais	71
6.1	Contribuições	71
6.2	Trabalhos futuros	72
6.3	Conclusão	74
A	O Padrão MPEG-4	75
B	Construção das Expressões Faciais no MPEG-4	79
	Referências Bibliográficas	81

Lista de Abreviaturas

AAM	<i>Active Appearance Model</i> (modelo ativo de aparência)
ASM	<i>Active Shape Model</i> (model ativo de forma)
FAP	<i>Facial Animation Parameter</i> (parâmetro de animação facial)
FAPU	<i>Face Animation Parameter Units</i> (unidade paramétrica de animação facial)
FP	<i>Feature Point</i> (ponto característico)
ICP	<i>Iterative Closest Point</i>
MM	<i>Morphable Model</i>
RBF	<i>Radial Basis Functions</i> (funções de base radial)
TPS	<i>Thin Plate Spline</i>

Lista de Símbolos

c_{vec}	Custo vetorial
C_A	Constante de normalização para custo de aparência
C_S	Constante de normalização para custo modular
d_A	Distância associada à aparência
$d_{A_{HK}}$	Custo de aparência associado à imagem do mapa HK
$d_{A_{UV}}$	Custo de aparência associado à imagem de textura
d_S	Distância associada à estrutura
$d_{S_{ang}}$	Custo de estrutura angular
$d_{S_{mod}}$	Custo de estrutura modular
$d[x]$	Distância de um vértice x até o FP que define sua região de influência
e_m	Aresta do modelo
e_d	Aresta deformada
$E(\cdot)$	Função de custo para o casamento
$f_d(x)$	Função de deformação
G_d	Grafo de deformação
G_m	Grafo do modelo
G	Mapa de curvatura gaussiana
H	Mapa de curvatura média
$M_{n \times 3}^k$	Matriz de deslocamento
P_k	Pose referente à uma expressão facial
v_m	Vértice do modelo
v_i	Vertice da entrada
X, Y, Z	Dimensões das <i>bounding boxes</i> de uma face 3D em cada eixo
λ	Parâmetro para balancear os custos de aparência e estrutura em $E(\cdot)$
λ_A	Parâmetro para balancear os custos de textura e curvatura em d_A
λ_S	Parâmetro para balancear os custos angular e modular em c_{vec}
μ	Atributo dos vértices
ν	Atributo das arestas
μ_{UV}	Atributo referente às coordenadas (R,G,B) na imagem de textura
μ_{HK}	Atributo referente às coordenadas (R,G,B) na imagem do mapa HK

θ	Ângulo entre dois vetores
$(v_{xi}^k, v_{yi}^k, v_{zi}^k)$	Posição de um vértice da pose P_k
$(\Delta x_i^k, \Delta y_i^k, \Delta z_i^k)$	Deslocamento de um vértice da pose P_k

Lista de Figuras

2.1	Triangulação da nuvem de pontos obtida pelo escâner.	12
2.2	A face neutra e as seis expressões faciais consideradas.	12
2.3	Pré-processamento da malha para corrigir falhas de aquisição pelo escâner.	13
2.4	Imagens de textura para os indivíduos do banco de dados.	14
2.5	Geometria tridimensional para os indivíduos do banco de dados utilizado.	15
3.1	Casamento estrutural de imagens de eletroforese.	18
3.2	Resultado da aplicação do algoritmo ICP para alinhamento das malhas faciais.	19
3.3	Exemplo de uma relação de isomorfismo entre um grafo do modelo e um subgrafo da entrada.	20
3.4	Grafo G_m , construído a partir de um modelo de 48 marcos faciais.	20
3.5	Um exemplo de modelo e entrada a serem casados.	21
3.6	Comparação das cores dos pixels em dois pontos das faces a serem casadas.	23
3.7	Mapas de curvatura média e Gaussiana para uma face 3D.	24
3.8	Classificação HK para indivíduos e expressões faciais diferentes.	25
3.9	Comparação das cores referentes à classificação HK em dois pontos das faces a serem casadas.	26
3.10	Exemplo para o casamento do olho em duas faces diferentes.	28
3.11	Comparação entre uma aresta do modelo e arestas candidatas da entrada.	28
3.12	Esquema geral para o casamento automático entre pontos de duas faces.	30
3.13	Resultados para o casamento entre a expressão neutra e a expressão de alegria.	32
3.14	Resultados para o casamento entre a expressão neutra e a expressão de tristeza.	32
3.15	Resultados para o casamento entre a expressão neutra e a expressão de surpresa.	33
3.16	Resultados para o casamento entre a expressão neutra e a expressão de raiva.	33
3.17	Resultados para o casamento entre a expressão neutra e a expressão de nojo.	34

3.18	Resultados para o casamento entre a expressão neutra e a expressão de medo.	34
3.19	Erro obtido no casamento entre a face neutra e as demais expressões faciais.	35
3.20	Resultados para o casamento da expressão neutra entre pessoas diferentes.	36
3.21	Resultados para o casamento da expressão de alegria entre pessoas diferentes.	37
3.22	Resultados para o casamento da expressão de tristeza entre pessoas diferentes.	37
3.23	Resultados para o casamento da expressão de surpresa entre pessoas diferentes.	38
3.24	Resultados para o casamento da expressão de raiva entre pessoas diferentes.	38
3.25	Resultados para o casamento da expressão de nojo entre pessoas diferentes.	39
3.26	Resultados para o casamento da expressão de medo entre pessoas diferentes.	39
3.27	Resultados para o casamento de marcos faciais entre pessoas diferentes.	40
3.28	Visualização do marcos tridimensionais detectados e remarcação dos pontos incorretos.	42
3.29	Esquema de subdivisão da malha inicial.	43
3.30	Resultado do processo de subdivisão da malha inicial após quatro iterações.	44
4.1	Esquema do fluxo de dados do sistema de animação usando interpolação linear.	45
4.2	Geometria das expressões e vetores e mapas de cores para os deslocamentos aplicados	47
4.3	Modelo reconstruído a partir de uma fotografia 2D.	49
4.4	Faces consideradas no cálculo da face média para a geração de caricaturas.	52
4.5	Resultados dados a geometria de todas as expressões e a textura da face neutra	53
4.6	Comparação entre a animação a partir do registro semiautomático e manual.	54
4.7	Transferência de expressões levando em conta diferentes entradas.	55
4.8	Visemas utilizados, transferidos através de modelos artificiais.	56
4.9	Resultados da geração automática de caricaturas.	57
5.1	Esquema do fluxo de dados do sistema de animação usando modelo paramétrico.	60
5.2	Subconjunto dos 64 FPs utilizados no projeto	61
5.3	Modelo usado para a etapa de subdivisão na animação usando modelo paramétrico.	62

5.4	Procedimento para o cálculo das regiões de influência do FP 2.1 (centro do maxilar).	63
5.5	Cálculo da região de influência considerando mais pontos ao redor do olho.	63
5.6	Mapa de cores mostrando a intensidade de deslocamentos em pontos faciais.	65
5.7	Interface para manipular os FAPs de baixo nível e gerar a expressão de alegria.	67
5.8	Resultados para a geração de expressões a partir da face neutra de três pessoas.	68
5.9	Visemas para as vogais, gerados a partir da face neutra	69
5.10	Comparativo entre as expressões geradas e as expressões reais digitalizadas.	69
A.1	Os 84 pontos característicos faciais (FPs) descritos no padrão MPEG-4.	76
A.2	Distâncias utilizadas para computar as FAPUs.	77

Lista de Tabelas

3.1	Classificação HK.	25
3.2	Estratégia de subdivisão e projeção usando o modelo inicial definido pelo grafo G_m	44
4.1	Visemas e os fonemas relacionados para o idioma inglês, no padrão MPEG-4.	51
5.1	Estratégia de subdivisão e projeção na animação por modelo paramétrico.	61
A.1	Cálculo das FAPUs a partir de um conjunto de FPs chave.	77
B.1	FAPs de baixo nível utilizados para construir as expressões faciais.	80

Capítulo 1

Introdução

1.1 Motivação

A animação facial é um área que atrai muito interesse entre os pesquisadores, sendo amplamente estudada e possuindo muitas aplicações, especialmente no campo de entretenimento. A obtenção de uma animação facial realista é uma tarefa difícil devido à complexa estrutura da face. Além disso, os seres humanos são extremamente familiarizados com os movimentos da face e, portanto, podem facilmente identificar pequenos detalhes que sejam pouco naturais ou inconsistentes em uma animação facial.

Este estudo explora o tema da animação facial através do computador utilizando modelos tridimensionais altamente realistas. O objetivo do trabalho consiste em propor diferentes estratégias de animação baseadas em uma nova técnica para detecção de pontos correspondentes em malhas faciais. Essas estratégias permitem manipular as faces de forma a exibir diferentes expressões faciais, podendo ser aplicadas sobre diferentes tipos de modelos 3D.

A seguir, é apresentada uma introdução bibliográfica que busca sintetizar os trabalhos mais recentes realizados na área de animação facial, de modo a contextualizar e destacar a contribuição deste trabalho para a área.

1.2 Síntese bibliográfica

1.2.1 Métodos de animação facial

Para que sejam obtidos resultados realistas, os maiores desafios na produção de animações faciais através do computador são a obtenção de uma representação coerente do rosto e a modelagem da animação em si. Os resultados mais realistas em modelagem de faces geralmente são obtidos através de informações de pessoas reais, via escâneres 3D ou modelos reconstruídos a partir de fotografias [Blanz e Vetter (1999); Mena-Chalco *et al.* (2009); Pighin *et al.* (2006)]. Alguns trabalhos sugerem utilizar um banco de dados de faces realistas e modelar novas faces à partir de funções que ma-

nipulam e modificam alguns atributos faciais [Amberg *et al.* (2009); Scherbaum *et al.* (2007)]. Para aumentar o realismo na modelagem, rugas e poros podem ser produzidos em superfícies muito suaves através de técnicas estatísticas [Golovinskiy *et al.* (2006)] ou da transferência de outros modelos mais realistas adquiridos a partir de equipamentos especiais [Beeler *et al.* (2010); Bickel *et al.* (2008); Bradley *et al.* (2010); Wilson *et al.* (2010)]. Além de rugas e poros, estudos recentes visam ampliar o realismo da pele simulando também outras características e propriedades mais específicas. Jimenez *et al.* (2010) estudam como a dilatação e contração de vasos sanguíneos em cada emoção afetam a pigmentação da pele e Bickel *et al.* (2009) buscam simular a elasticidade de tecidos macios, como os que compõem a região das bochechas.

Para realizar a animação desses modelos, existem diferentes estratégias que podem ser adotadas. Esses métodos podem levar em conta poses-chaves pré-modeladas ou estimar expressões faciais a partir de uma face no seu estado de neutralidade¹. Há pelo menos quatro abordagens fundamentais para a animação facial que são amplamente utilizadas: a interpolação de poses-chaves, o modelo baseado em músculos, o modelo baseado em desempenho e a parametrização direta.

O método mais antigo, mais simples, e provavelmente também o mais utilizado, é a interpolação de quadros-chaves representados por diferentes expressões faciais. Essa estratégia gera uma transição suave entre duas poses diferentes pelo cálculo dos pontos intermediários, denominados *in-betweens*. Se as malhas que representam as poses-chaves tiverem a mesma topologia, a interpolação apenas manipula a posição dos vértices. As poses-chaves podem ser modeladas manualmente, adquiridas de um ser humano real, inferidas a partir de outros quadros em uma sequência de animação [Ma *et al.* (2009)] ou geradas através de técnicas estatísticas [Kim e Ko (2007)]. A interpolação pode ser linear ou não-linear, considerando que os movimentos físicos do rosto são normalmente realizados de forma não-linear. Também é possível aplicar a interpolação em pequenas regiões da face por vez, como propõem Joshi *et al.* (2003), a fim de combinar expressões faciais. A interpolação linear vem sendo utilizada em uma série de estudos nos quais é aplicada isoladamente ou combinada com outras técnicas como em Balci *et al.* (2007); Blanz e Vetter (1999); cai Yin *et al.* (2005); Golovinskiy *et al.* (2006); Pighin *et al.* (2006).

O modelo baseado em músculos, por sua vez, procura simular a geometria dos ossos, músculos e tecidos da face. Nessa técnica, os vértices do modelo não são alterados diretamente, devendo acompanhar uma estrutura interna à medida que esta é modificada. A abordagem de pseudomúsculos é uma variante do método, que não busca simular fielmente uma anatomia detalhada da face. Em vez disso, essa ideia busca selecionar modelos simplificados que possam ser controlados da mesma maneira. Ambas as abordagens de músculos e pseudomúsculos vem sendo exploradas em alguns

¹A face neutra possui os olhos abertos, músculos da face relaxados e lábios em contato em linha horizontal na mesma altura dos cantos da boca.

trabalhos [Choe e Ko (2005); Sifakis *et al.* (2005, 2006); Yano e Harada (2009)]. Limitações destas estratégias são a alta complexidade e custo computacional do processo de animação e a dificuldade em reproduzir movimentos sutis da superfície da pele, como rugas.

Outro método, o modelo baseado em desempenho, captura ações humanas e as reproduz para modelos sintéticos, normalmente exigindo algum equipamento adequado para a extração do movimento. Em animação facial, uma estratégia comumente usada é a de fornecer um mapeamento inicial entre a face neutra do ser humano e a face neutra do modelo alvo, e usar uma função apropriada para atualizar as correspondências entre eles à medida que a face real se move. O exemplo mais comum de animação baseada em desempenho dá-se através da técnica de *motion capture*, que tradicionalmente é feita usando marcadores especiais que são aplicados na face do ser humano [Dutrevez *et al.* (2008); Ju e Lee (2008)]. No entanto, alguns bons resultados foram alcançados utilizando abordagens sem marcadores [Chai *et al.* (2003); Vlastic *et al.* (2006); Bradley *et al.* (2010); Moreira *et al.* (2010); Weise *et al.* (2009)], mesmo que marcadores especiais possam ainda ser utilizados para enfatizar informações úteis da face, como rugas [Bickel *et al.* (2007)]. Além de transmitir ações de um ator para um modelo, também é comum obter-se uma animação a partir de uma face neutra por meio da transmissão de expressões faciais de outros modelos 3D. Este procedimento é denominado de *transferência* (ou *clonagem*) de expressões faciais e é abordado em Aubrey *et al.* (2010); Lau *et al.* (2007); Noh e Neumann (2001); Song *et al.* (2007); Theobald *et al.* (2007); Vlastic *et al.* (2006).

Por fim, na parametrização direta, a face é representada por uma malha poligonal e movimenta-se através de um conjunto de parâmetros aplicados em pontos característicos da face. Assim, essa abordagem tem como objetivo construir um modelo capaz de gerar uma grande variedade de expressões faciais com base nesse conjunto de parâmetros de controle, que deve ser o menor possível. Uma desvantagem do processo é que os parâmetros dependem da topologia da face, e devem ser reescritos para serem usados em outros modelos. Buscando superar esse problema, é comum adotar-se o modelo internacional para a animação de faces paramétricas proposto pelo padrão MPEG-4. Esse padrão é utilizado para compressão de dados digitais de áudio e vídeo, o que requer uma representação dos seus objetos de mídia. Para isso, parte do padrão está dedicada à definição da representação para corpos e faces sintéticas. O modelo para animação facial proposto pelo padrão MPEG-4 vem sendo utilizado em uma série de projetos [Balci *et al.* (2007); cai Yin *et al.* (2005); Mäkäräinen e Takala (2009); Queiroz *et al.* (2010); Tang e Huang (2008); Wu *et al.* (2002)] e será descrito com mais detalhes adiante, no Capítulo 5.

Desta forma, as técnicas que levam em conta modelos paramétricos, baseados em músculos ou em desempenho, não requerem que existam expressões pré-modeladas, estimulando-as a partir da face neutra, diferentemente do modelo de interpolação. Em-

bora poderosas, as técnicas que levam em conta apenas alterações sobre a face neutra podem gerar expressões artificiais se comparadas à geometria real que pode ser obtida através de um ser humano, por exemplo. Assim, este estudo aborda a implementação de duas estratégias de animação, de modo a comparar os resultados obtidos. Modelos de pessoas reais, obtidos através de um escâner 3D, são animados através das técnicas de interpolação e parametrização direta. Para ambos os métodos, busca-se obter uma animação realista, isto é, minimizando artefatos na animação, evitando transições bruscas e respeitando a coerência das expressões e movimentos faciais.

Para o método de interpolação, são utilizados modelos apresentando diferentes expressões faciais, obtidos pelo escâner. É possível obter uma interpolação de forma simples quando as malhas a serem interpoladas possuem o mesmo número de vértices e a mesma topologia, isto é, para cada vértice de uma malha, conhecemos o vértice correspondente da outra malha, de modo que todos os pontos entre elas respeitem uma correspondência de um-para-um. Entretanto, essa restrição não é atendida quando se lida com dados de escâner 3D, uma vez que as diferentes malhas obtidas pelo escâner representam nuvens de pontos entre as quais não há nenhum tipo de correspondência.

Da mesma forma, para a animação por parametrização, é necessário conhecer o mapeamento entre os pontos da malha a ser animada, e os pontos característicos do modelo paramétrico adotado. Não basta apenas conhecer os vértices correspondentes aos pontos característicos, sendo necessário, também, definir correspondências entre a malha e o modelo paramétrico para obter uma boa deformação da malha durante a aplicação de cada parâmetro. Assim, essa abordagem requer também, como passo inicial, a definição de correspondências entre modelos de faces.

1.2.2 Busca de correspondências entre malhas faciais

Estudos anteriores já investigaram estratégias para o cálculo de correspondências ponto a ponto em malhas faciais. A seguir, estão listados alguns trabalhos relacionados.

[Blanz e Vetter \(1999\)](#) utilizaram faces neutras de diferentes pessoas, obtidas por um escâner 3D, para construir uma estrutura denominada de *morphable model* (MM). Um MM consiste em uma representação da geometria e textura destes indivíduos em um espaço vetorial, que pode ser utilizado para reconstruir e modelar novas faces através de combinações lineares entre os indivíduos. Esse estudo introduziu um método de casamento baseado em um algoritmo de fluxo ótico para registrar faces neutras diferentes. O algoritmo busca minimizar as diferenças entre as malhas, ponderando igualmente variações de textura e geometria, podendo utilizar propriedades da superfície, como curvatura média.

Este algoritmo de fluxo foi utilizado mais tarde em outros estudos. [Basso et al. \(2006\)](#) e [Blanz et al. \(2007\)](#) estudaram estratégias para realizar o registro entre um MM e outros dados provenientes de escâner nos quais haviam regiões incompletas

devido a falhas no processo de aquisição. Os resultados exibem apenas o registro entre malhas faciais no estado de neutralidade, ou com expressões que apresentam pequenas variações com relação à face neutra. [Blanz et al. \(2003\)](#), buscaram casar faces com expressões faciais diferentes utilizando o mesmo algoritmo, mas devido às diferenças entre as malhas, foi preciso utilizá-lo de forma semiautomática, dividindo a face em porções e aplicando o algoritmo separadamente em cada parte.

Mais recentemente, nos trabalhos de [Amberg et al. \(2008\)](#) e [Paysan et al. \(2009\)](#), essa mesma linha de pesquisa introduziu uma estratégia de registro entre uma face e um MM, de maneira invariante a expressão. Entretanto, essas abordagens contam com um conjunto de treinamento consideravelmente grande, com 270 pessoas digitalizadas apresentando diferentes expressões, sob condições variadas de iluminação, e com alguns marcos faciais manualmente selecionados. Adquirir um número elevado de modelos referentes a diferentes indivíduos e expressões faciais não consiste em uma tarefa simples, e não é comum encontrar banco de dados com estas características disponíveis, de modo que estes impedimentos podem dificultar a utilização do método em outros estudos.

Com o intuito de transferir expressões faciais entre modelos artificiais, na alternativa proposta por [Noh e Neumann \(2001\)](#), pontos correspondentes são selecionados manualmente e os casamentos restantes na superfície densa são feitos via redes RBF (*Radial Basis Functions*)². Dado um modelo e a malha alvo na qual o modelo deve ser ajustado, as redes RBF são utilizadas, a partir dos pontos esparsos correspondentes, para deformar o modelo de acordo com a malha alvo. Em seguida, o modelo deformado é projetado sobre a malha alvo, para que os pontos correspondentes sejam encontrados. Para evitar a marcação manual dos pontos, tornando o processo totalmente automático, são sugeridas algumas heurísticas, supondo que a face esteja com o olhar na direção do eixo z . Por exemplo, a ponta do nariz pode ser encontrada selecionando-se o ponto com maior valor em z , e o topo da cabeça com o maior valor em y . Entretanto, não há garantia para o funcionamento das heurísticas em qualquer caso. Em particular, para malhas de escâner, nas quais é comum existir a presença de ruídos, as heurísticas dificilmente funcionariam de forma adequada.

No estudo de [Vlasic et al. \(2006\)](#), que visa a transferência de expressões faciais entre modelos digitalizados, busca-se ajustar um modelo de malha na superfície proveniente do escâner. Para isso, é utilizada uma função objetivo que pondera a similaridade global das superfícies, a proximidade de pontos selecionados manualmente nas duas malhas e alguns vértices de referência. Após o alinhamento do modelo e da malha digitalizada, deforma-se o modelo sobre a malha.

Uma abordagem semelhante é utilizada por [Bickel et al. \(2008\)](#), para transferir rugas de uma malha detalhada para uma superfície suave. Novamente, uma seleção

²Redes RBF são funções suaves e contínuas que realizam interpolações entre dados fornecidos. Os valores da função são obtidos de acordo com as distâncias entre o ponto onde a função deve ser avaliada e os pontos de referência.

manual de pontos auxilia um procedimento no qual o modelo detalhado é deformado para ajustar-se na face alvo, sendo em seguida projetado sobre ela.

Já a ideia sugerida por [Hu et al. \(2009\)](#) seleciona pontos manualmente nas duas faces a serem casadas e utiliza transformações TPS (*Thin Plate Spline*)³ para criar um mapeamento suave entre os dois conjuntos de pontos. Essa etapa é responsável por alinhar os modelos a serem casados. A seguir, o casamento é feito utilizando uma estratégia que casa os pontos mais próximos em cada malha. Um problema dessa abordagem é que o casamento não preserva o mapeamento de um para um, de modo que alguns pontos de uma face podem ser mapeados no mesmo ponto da outra. Dessa forma, é necessário adicionar um algoritmo de colisão, tornando o processo computacionalmente mais caro.

Como é possível notar, nem todos os métodos citados tem a capacidade de registrar faces que possuem expressões faciais diferentes, tratando-os como casos especiais. Outros exigem um conjunto de dados de treinamento muito grande, contando ainda com uma variedade de expressões faciais. Além disso, todas essas abordagens tem em comum o foco de buscar correspondências densas entre as malhas, buscando casar todos os pontos, o que pode ocasionar alguns problemas. Destes, o principal é o número insuficiente de correspondências ou casamentos ruins em regiões planas da face, como testas e bochechas, nas quais há pouca variação de geometria e textura, demandando um pós-processamento nessas regiões para completar o processo de registro.

Além das desvantagens mencionadas, a maioria dos métodos citados requerem que um conjunto de pontos faciais correspondentes seja selecionado nas malhas faciais. Evidentemente, o processo de marcar diversos modelos pode ser exaustivo. Como será explicado mais adiante, nossa proposta depende, também, da seleção de um conjunto de pontos correspondentes. Entretanto, para minimizar a interação manual do usuário, avaliamos uma técnica automática para a detecção destes pontos, com base nos estudos descritos a seguir.

1.2.3 Detecção automática de pontos característicos faciais

Além dos estudos citados, em uma série de trabalhos é necessário também estabelecer um conjunto de pontos correspondentes em diferentes malhas faciais. O problema da detecção de pontos característicos em faces pelo computador é conhecido como *landmarking*. A definição de marcos pode ser utilizada para normalizar malhas faciais, o que costuma ser necessário para algoritmos de reconstrução ou reconhecimento de faces. Além dos problemas usuais, o método de *landmarking* pode ajudar também a resolver problemas mais específicos, como na área médica, por exemplo, para a medição tridimensional da morfologia craniofacial, identificando diferenças entre a população [[Toma et al. \(2009\)](#)].

³A transformação TPS é um caso particular de função para as redes RBF e consiste em uma técnica apropriada para a interpolação de superfícies sobre dados irregularmente espaçados.

Os algoritmos de *Active Appearance Model* (AAM) e *Active Shape Model* (ASM)⁴ vem sendo utilizados amplamente para encontrar um conjunto de marcos correspondentes em faces, sobretudo para o caso bidimensional. Essas correspondências permitem resolver problemas variados, como por exemplo, relativos à caricaturização facial [Liu *et al.* (2006); Obaid *et al.* (2009)] ou à detecção de faces em sequências de vídeo para animação baseada em desempenho [Katsamanis *et al.* (2008); Theobald *et al.* (2007); Mattheyses e Verhelst (2010); Sattar *et al.* (2009)]. Entretanto, esses algoritmos requerem um conjunto de treinamento que precisa ser manualmente rotulado, demandando um grande esforço para o caso de um conjunto de treinamento com muitos dados. Além disso, os algoritmos apresentam desempenho ruim quando consideram uma face com características que não estão presentes no banco de dados de treinamento, o que pode evidenciar pouca robustez com relação à variação de expressões faciais, tonalidade da pele, gênero, idade, pose, dentre outras características [Baltrušaitis e Robinson (2010)].

Outros métodos são também possíveis para efetuar o procedimento de *landmarking*. Lu e Jain (2006) e Whitmarsh *et al.* (2006) baseiam-se na mesma abordagem já citada para a busca de correspondências entre malhas faciais adotada em Bickel *et al.* (2008); Hu *et al.* (2009); Noh e Neumann (2001); Vlasic *et al.* (2006). Isto é, são alinhadas uma malha e um modelo que contém os marcos, utilizando uma função de deformação, e selecionam-se os pontos mais próximos da malha com relação aos marcos. Essa estratégia apresenta as mesmas desvantagens mencionadas para o caso da busca de correspondências, como por exemplo, a incapacidade de definir marcos faciais em expressões diferentes.

Por outro lado, os estudos de Çeliktutan *et al.* (2008) e Dibeklioglu *et al.* (2008), embora considerem poucos marcos (cantos dos olhos, cantos da boca e ponta do nariz), mostraram que a ideia de aliar informações estruturais a outras técnicas pode fornecer bons resultados de forma robusta.

1.3 Objetivos

Tendo em vista os problemas levantados, este trabalho possui as seguintes metas principais:

1. Evitar as desvantagens existentes na busca de correspondências densas entre as malhas faciais.

A principal diferença da abordagem adotada é que nos preocupamos apenas em estabelecer alguns pontos correspondentes que formam um modelo inicial, deixando os casamentos das regiões intermediárias para um processo de subdi-

⁴Ambos os algoritmos de AAM e ASM buscam deformar um modelo estatístico de modo a alinhá-lo em uma imagem, levando em conta aparência e forma, respectivamente.

visão que será descrito na seção 3.4. Diferente de alguns dos métodos anteriores, esta estratégia garante que modelos com expressões faciais diferentes possam ser registrados.

2. Encontrar de forma automática o conjunto inicial de pontos utilizando métodos estruturais.

Grande parte dos estudos citados até aqui utilizam a marcação manual de pontos para auxiliar no processo pela busca de correspondências. Propomos uma alternativa nova para realizar o casamento automático de um conjunto de pontos esparsos em diferentes malhas faciais, sem a necessidade de um conjunto amplo de treinamento. Para realizar o casamento, uma abordagem estrutural anteriormente estudada para imagens 2D é estendida para o caso 3D.

3. Animar as faces digitalizadas utilizando interpolação linear.

A partir do algoritmo para detecção de pontos correspondentes, utilizamos o método de interpolação para obter uma animação na qual as poses-chaves referem-se a expressões faciais diferentes. A informação obtida neste processo pode ser utilizada para animar outras faces a partir da informação de textura e geometria da face neutra, ou apenas textura.

4. Utilizar estratégias aplicadas nos problemas anteriores para propor um método de animação baseado no modelo paramétrico do MPEG-4.

A animação por face paramétrica requer uma definição de como será feita a deformação da malha ao serem aplicados os parâmetros de animação. A definição de um conjunto de pontos faciais seguida pelo processo de subdivisão é proposta como uma alternativa para definir a deformação da malha, necessária para executar a animação.

1.4 Organização do documento

O texto está organizado da seguinte forma: primeiramente, no Capítulo 2, relatamos informações sobre a aquisição e o pré-processamento das faces reais do banco de dados utilizado para os algoritmos de registro e animação propostos.

Então, no Capítulo 3, descrevemos a abordagem estrutural utilizada para encontrar um conjunto de marcos faciais correspondentes em modelos de faces. Nesse mesmo capítulo, mostramos como esse conjunto inicial de pontos esparsos é utilizado para definir correspondências densas entre as malhas, a partir de um processo de subdivisão e projeção da malha inicial.

No Capítulo 4, é descrita a estratégia de animação usada para realizar a interpolação dos modelos tridimensionais, a partir da definição da correspondência densa

entre os pontos de cada modelo. Resultados para algumas aplicações do método são apresentados.

A seguir, esses resultados são comparados com os de outra estratégia implementada, de animação baseada no modelo paramétrico do padrão MPEG-4, descrita no Capítulo 5 como uma alternativa para o método de interpolação. Detalhes adicionais referentes à animação paramétrica usando o MPEG-4 são apresentados nos Apêndices A e B.

Por fim, no Capítulo 6, encerramos o documento discutindo possíveis pontos a serem melhorados e enfatizando a contribuição do trabalho para a área.

Capítulo 2

Banco de dados de faces reais

2.1 Aquisição

O banco de dados de faces reais usado no projeto foi obtido em uma colaboração anterior entre o Grupo de Pesquisa em Visão e Processamento de Imagens do IME-USP e o Instituto Nacional de Matemática Pura e Aplicada (IMPA-RJ), relatado em [Mena-Chalco *et al.* \(2008a\)](#). Os modelos foram adquiridos no laboratório do IMPA-RJ através de um escâner 3D Konica Minolta Vivid 910. O escâner é equipado com um sensor de distância laser e uma câmera digital integrada, além de lâmpadas refletoras que fornecem condições de luz controladas. A aquisição, para cada amostra, é composta de dados de textura e geometria registrados. As imagens de textura foram adquiridas com resolução de 640×480 pixels (24 bits) e a geometria da face 3D associada a cada imagem de textura contém aproximadamente 50.000 vértices.

Diferente de alguns modelos de escâner, que são capazes de dar uma volta completa capturando diferentes pontos de vistas ao redor de um objeto, o escâner utilizado fornece a captura a partir de um único ponto de vista. Dessa forma, a geometria obtida pelo escâner consiste em uma representação da superfície 3D que associa um valor de profundidade na direção z para cada ponto no plano (x, y) . Cada vértice da geometria está também associado a um ponto na imagem de textura, permitindo, como já mencionado, o registro desses dados.

Cada aquisição do escâner gera uma nuvem de pontos com as características descritas. Essa nuvem é então triangulada gerando uma malha 3D, através da aplicação de uma triangulação de Delaunay¹, como mostra a Figura 2.1. Desta forma, quando mencionarmos neste texto a expressão *malha bruta* do escâner, estamos nos referindo à nuvem de pontos adquirida seguida do processo de triangulação.

Foram capturadas faces de indivíduos exibindo a expressão neutra e mais seis expressões faciais diferentes: alegria, tristeza, surpresa, raiva, nojo e medo. As poses di-

¹A triangulação de Delaunay conecta os pontos do escâner encontrando segmentos de reta entre eles, de forma que nenhum desses segmentos se cruzem e que cada ponto seja vértice de pelo menos um triângulo formado pelos segmentos.

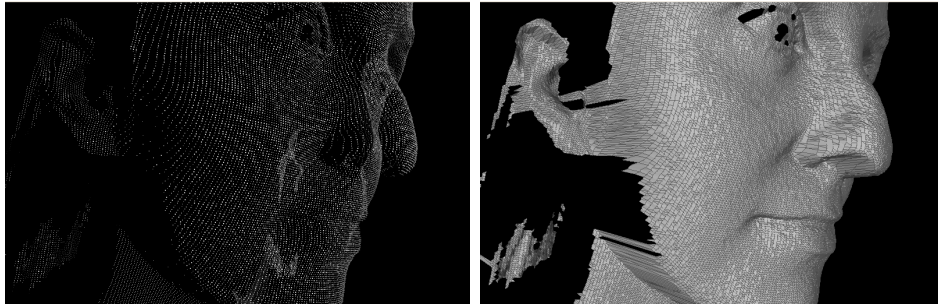
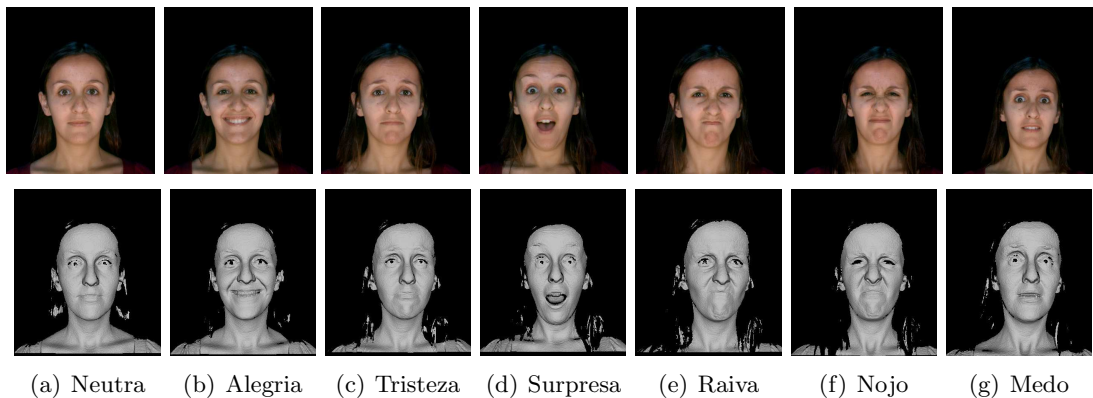


Figura 2.1: *Triangulação da nuvem de pontos obtida pelo escâner.*

gitalizadas referem-se às seis expressões universais propostas por Paul Ekman, em um estudo muito aceito e utilizado na maioria das pesquisas referentes à manipulação de expressões faciais pelo computador [Ekman e Friesen (1978)]. Ekman defende que essas seis expressões, denominadas *puras*, podem ser combinadas de modo a gerar novas expressões (conhecidas como expressões *derivadas*). A Figura 2.2 exibe as seis expressões utilizadas, além da face neutra.



(a) Neutra (b) Alegria (c) Tristeza (d) Surpresa (e) Raiva (f) Nojo (g) Medo

Figura 2.2: *A face neutra e as seis expressões faciais consideradas.*

2.2 Pré-processamento

Ao observar as figuras 2.1 e 2.2, nota-se que a aquisição feita a partir de um único ponto de vista pode conter falhas em certas regiões da face. Essas falhas ocorrem devido a presença de sombras, que impedem a captura do escâner nessas regiões. Assim, a malha bruta passa por um pré-processamento no qual esses buracos são preenchidos, aplicando-se um operador de dilatação. Essa etapa de pré-processamento também é responsável por eliminar ruídos na malha, que representam outro tipo de falha possível na aquisição dos dados. São eliminadas componentes conexas pequenas (inferiores a 1000 elementos) e suavizadas regiões nas quais os pontos dentro de uma vizinhança de

6×6 ultrapassem um dado limiar. A Figura 2.3 exibe um modelo de face antes e depois do pré-processamento descrito.

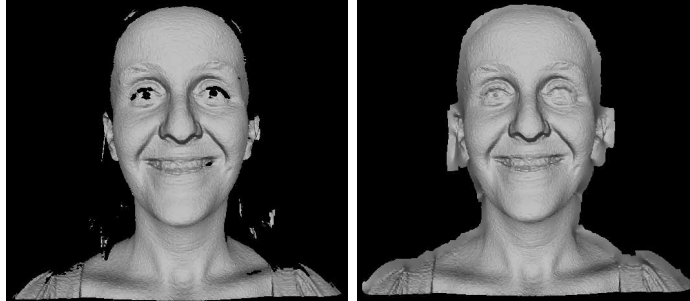


Figura 2.3: *Pré-processamento da malha para corrigir falhas de aquisição pelo escâner.*

É importante ressaltar que cada captura fornece uma nova nuvem de pontos que é triangulada, gerando um conjunto de malhas distintas, com as características descritas. As malhas não possuem nenhum tipo de correspondência, mesmo aquelas referentes a uma mesma pessoa. Isto é, as malhas possuem números diferentes de vértices e não existe nenhum tipo de relação entre os vértices de uma malha e os vértices das demais.

2.3 Indivíduos

A seguir, estão apresentadas imagens referentes à geometria e textura para 30 indivíduos que foram utilizados como dados de entrada para os algoritmos de casamento e animação que serão relatados nos próximos capítulos. As figuras mostram essas faces apenas no estado de neutralidade. Entretanto, foram feitas aquisições de todas as expressões faciais mostradas na Figura 2.2 para todos os indivíduos, totalizando 210 malhas distintas com as quais trabalhou-se durante a pesquisa.

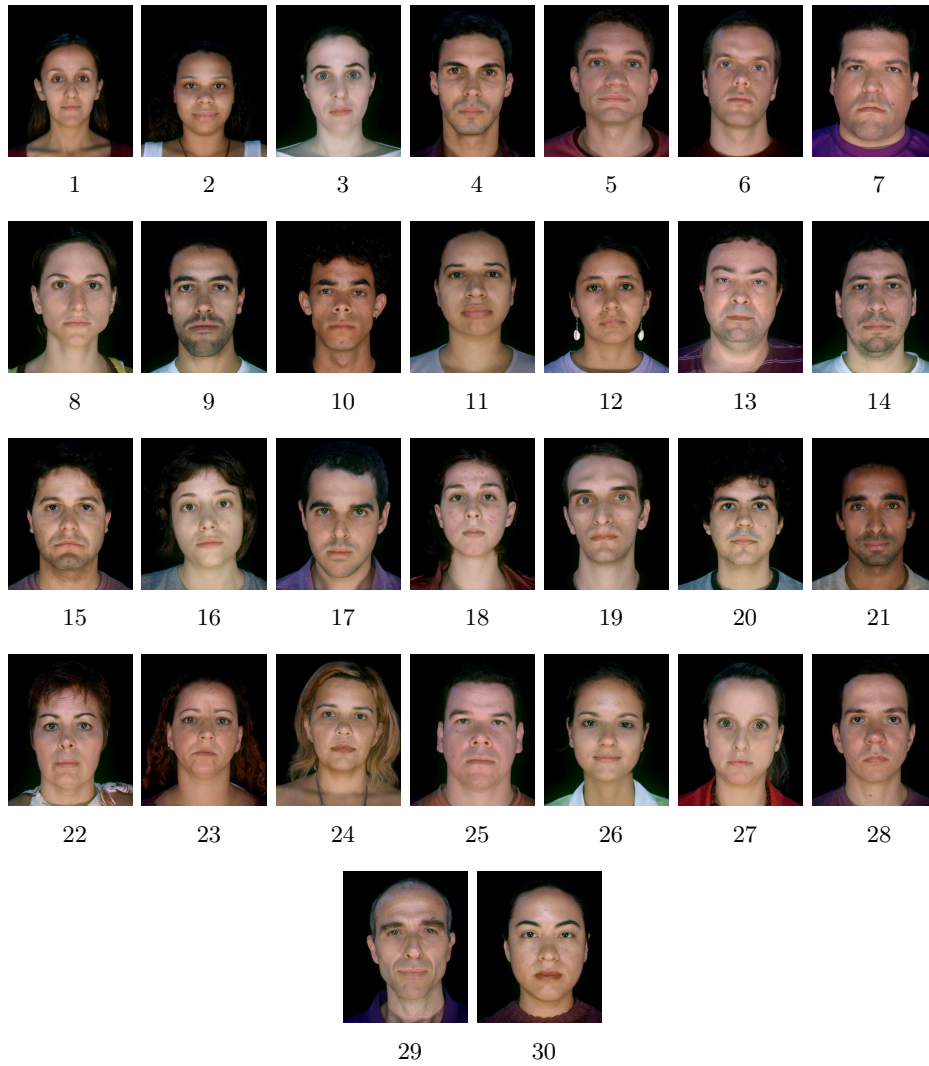


Figura 2.4: *Imagens de textura para os indivíduos do banco de dados.*

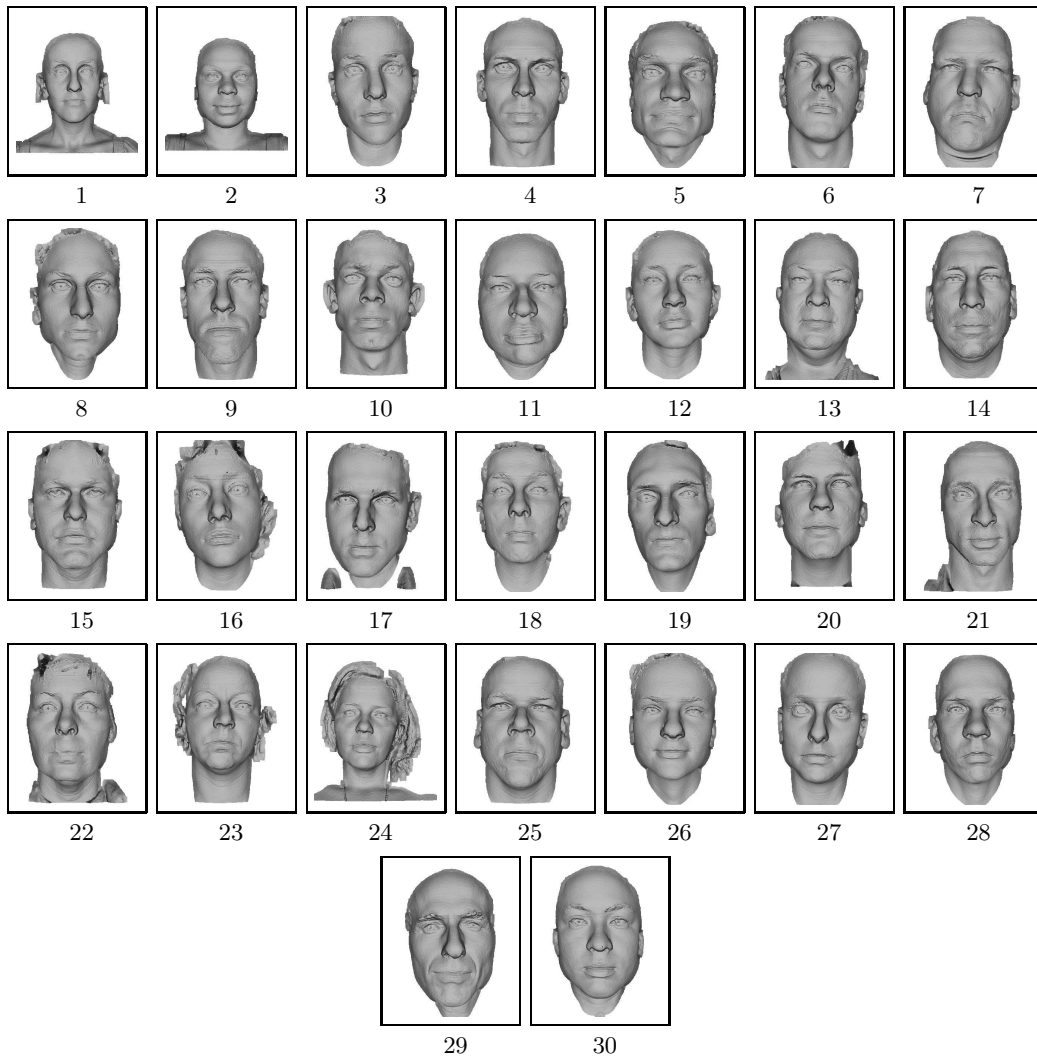


Figura 2.5: Geometria tridimensional para os indivíduos do banco de dados utilizado.

Capítulo 3

Registro estrutural de faces 3D

Neste capítulo descrevemos a abordagem proposta para detectar automaticamente um conjunto de marcos em faces 3D, sem a utilização de uma base ampla de treinamento. Descreveremos, também, como o procedimento de registro é completado a partir deste conjunto inicial de marcos.

A ideia consiste em, a partir de uma única face marcada manualmente, encontrar os pontos correspondentes em outras faces, que podem referir-se a expressões faciais diferentes ou pessoas diferentes. O método proposto baseia-se em registro estrutural e casamento de grafos. A abordagem introduz uma generalização para o caso tridimensional de um método anteriormente aplicado em imagens 2D.

A seguir, a seção 3.1 discute os conceitos relacionados aos métodos estruturais para o casamento em imagens. Então, na seção 3.2, relata-se a abordagem proposta para o casamento de marcos faciais em modelos 3D e os resultados são exibidos na seção 3.3. Finalmente, na seção 3.4, descrevemos o processo de subdivisão aplicado a partir do modelo de marcos faciais que finaliza o processo de registro.

3.1 Métodos estruturais

Os métodos estruturais em reconhecimento de padrões consideram uma imagem como sendo um conjunto de partes relacionadas entre si de maneira espacial. Não existe uma regra específica para a decomposição da imagem em partes, que podem também conter subpartes, e assim sucessivamente [da Fontoura Costa e Cesar Jr. (2009)].

Um grafo G é um par (V, E) onde V é um conjunto de vértices e E é um conjunto de pares não ordenados contidos em V , denotados como arestas. A divisão da imagem em partes permite a construção de um grafo relativo à forma analisada. Nesse grafo, os vértices representam as partes da imagem e as arestas representam as relações estruturais entre estas partes.

Esses grafos podem ser usados em uma série de problemas de reconhecimento de padrões, de forma que uma ou mais imagens possam ser comparadas encontrando

correspondências entre os grafos que as representam. Diferentes formas de mapeamento $f : V_1 \rightarrow V_2$ entre os vértices de grafos G_1 e G_2 podem ser feitas. Em particular, $f : V_1 \rightarrow V_2$ é uma relação de *isomorfismo* se $a, b \in V_1$ são adjacentes se e somente se $f(a), f(b) \in V_2$ são adjacentes. Desta forma, a comparação entre duas formas pode ser vista como um problema de otimização, no qual seus respectivos grafos devem ser casados de maneira a otimizar alguma função objetivo.

3.1.1 Casamento bidimensional

A abordagem estrutural foi utilizada em um estudo para o casamento de padrões em imagens de eletroforese em gel [Noma *et al.* (2008)]. A eletroforese consiste de um método para a separação de proteínas, através da migração das moléculas de uma amostra em um determinado gel. Desta forma, pode ser obtida uma série de imagens, uma relativa ao início do processo, e as outras, ao longo dele. Essas imagens devem ser registradas para que possa ser possível encontrar, em todas, a localização de um ponto, deslocado ao longo do tempo.

A técnica proposta para solucionar o problema leva em conta aspectos estruturais das imagens, comparando a primeira imagem obtida com as demais, através da construção de grafos relativos à cada imagem. Esses grafos são construídos conforme mencionado anteriormente, sendo os vértices os pontos da imagem analisada. O problema torna-se, então, encontrar relações de isomorfismo entre os grafos construídos. Mais precisamente, no problema particular da eletroforese, buscam-se relações entre subgrafos das imagens a serem comparadas. Isso porque, ao longo do processo, alguns pontos podem ser criados ou eliminados de uma imagem para a outra. Por isso, nem todos os pontos do modelo precisam, necessariamente, ser mapeados com algum ponto da entrada.

O estudo descrito realizou alguns testes que exibiram bons resultados, mostrando que a técnica de levar em conta informações sobre a estrutura e forma dos pontos obteve resultados satisfatórios de maneira computacionalmente eficiente. A Figura 3.1 apresenta um exemplo de casamento entre pontos de duas imagens de eletroforese.

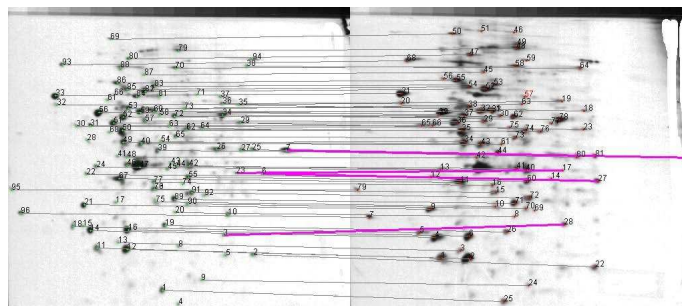


Figura 3.1: Casamento estrutural de imagens de eletroforese (Figura adaptada de Noma *et al.* (2008)).

3.2 Casamento tridimensional

A ideia desta parte do projeto é estender a técnica descrita para o caso tridimensional. Inicialmente, são marcados manualmente pontos em uma malha, denominada de *modelo*. Então, o algoritmo busca os pontos correspondentes nas demais malhas a serem casadas (denotadas como *entradas*). Uma vez que estão disponíveis textura e geometria registradas para cada face, o método utiliza informação 2D e 3D para o casamento.

Como primeiro passo, o algoritmo de casamento requer que sejam alinhadas as malhas de entrada com relação à malha modelo. Esse passo é importante pois o método estrutural é sensível a variações de posição entre os dados que serão casados. Para isso, utilizamos o algoritmo ICP (*Iterative Closest Point*) [Besl e McKay (1992)]. O algoritmo é capaz de alinhar de forma rígida duas nuvens de pontos minimizando a distância entre eles, através do cálculo de matrizes de rotação e translação (R e t , respectivamente). O ICP é inicializado atribuindo uma estimativa grosseira para R e t , que vão sendo recalculados, iterativamente, de modo a minimizar o erro de ajuste. A Figura 3.2 ilustra a aplicação do algoritmo ICP, alinhando duas malhas faciais.

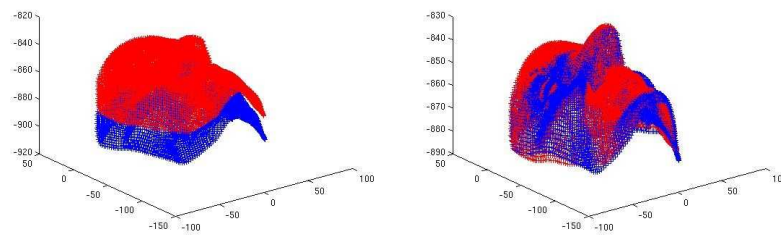


Figura 3.2: Resultado da aplicação do algoritmo ICP para alinhamento das malhas faciais.

Diferente do problema da eletroforese, citado na seção anterior, é necessário encontrar um correspondente para cada ponto do modelo, dentre um conjunto de pontos da entrada. Dessa forma, é necessário estabelecer uma relação de isomorfismo entre o grafo do modelo e um subgrafo da entrada. A Figura 3.3 ilustra um exemplo simplificado do problema que deve ser resolvido.

A ideia geral do algoritmo consiste em construir um grafo cujos vértices são os pontos marcados no modelo e selecionar, em cada entrada, os pontos que produzem um casamento sub-ótimo com o grafo do modelo. O grafo do modelo é denominado de G_m . O algoritmo não requer um padrão específico de marcos faciais a serem casados. Para esta pesquisa, adotou-se um padrão de 48 pontos, mostrado na Figura 3.4, juntamente com o grafo construído a partir desses pontos.

A decisão de adotar esse modelo de pontos deveu-se ao fato de que as faces do banco de dados utilizado já haviam sido previamente marcadas, para o estudo relatado

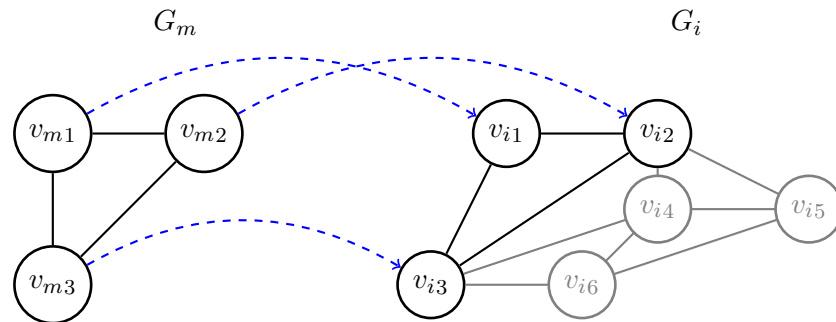


Figura 3.3: Exemplo de uma relação de isomorfismo entre um grafo do modelo (G_m) e um subgrafo da entrada (G_i).

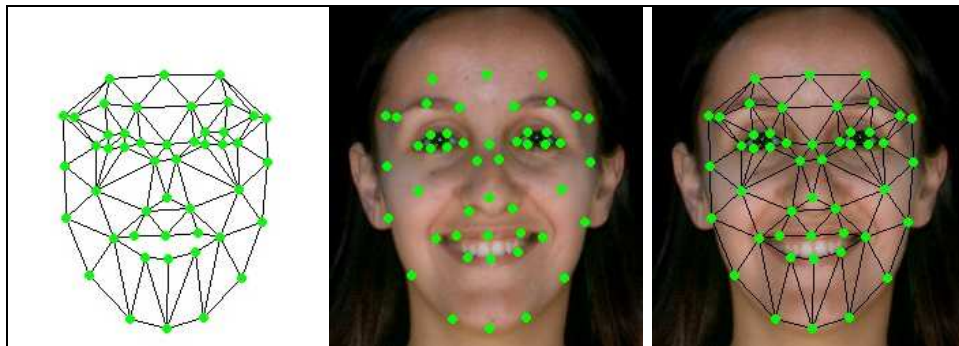
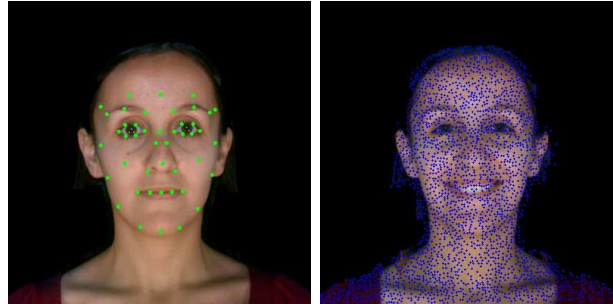


Figura 3.4: Grafo G_m , construído a partir de um modelo de 48 marcos faciais.

em Mena-Chalco *et al.* (2008b). Os 48 pontos foram selecionados manualmente em cada face adquirida pelo escâner, com o objetivo de normalizar esses dados. Na nossa pesquisa, utilizamos esses pontos como um gabarito para estimar o erro obtido no processo de casamento.

Após estabelecidos os marcos a serem casados, definimos um conjunto de pontos candidatos na entrada e buscamos o subconjunto que fornece o melhor casamento com o modelo, estabelecendo uma relação de isomorfismo entre o grafo do modelo e um subgrafo da entrada. O conjunto de candidatos pode ser selecionado de diversas formas, por exemplo, a partir de uma amostra igualmente espaçada de pontos da malha, ou mesmo considerando todos os pontos dela. Entretanto, como o procedimento de analisar todos os pontos da entrada possui custo muito alto, optou-se por selecionar de forma automática um conjunto de candidatos para o casamento. Para essa detecção de pontos chaves, utilizamos o detector *good features to track*, proposto por Shi e Tomasi (1994). O detector computa, para cada pixel, o menor auto-valor - que esteja acima de um limiar l - dentro da vizinhança do pixel, selecionando pontos candidatos. Então, os n candidatos de maior auto-valor são selecionados como pontos chaves, desde que respeitem uma distância mínima d entre eles. O detector permite ajustar os seus parâmetros de modo a detectar quantidades variáveis de pontos, estabelecendo valores para l , d e n .

Para os testes realizados, utilizamos a implementação do detector disponibilizada pela biblioteca OpenCV, para processamento de imagens, selecionando um total de 1500 candidatos. O detector opera sobre a imagem de textura 2D e os pontos selecionados são projetados na geometria 3D. A Figura 3.5 exibe um exemplo de pontos candidatos detectados automaticamente para uma entrada.



(a) Modelo, com pontos selecionados manualmente. (b) Entrada, com pontos candidatos selecionados através do detector.

Figura 3.5: Um exemplo de modelo e entrada a serem casados.

Uma vez tendo sido selecionado o conjunto de pontos candidatos para o casamento, é preciso avaliar quais desses pontos fornece o melhor casamento com o grafo do modelo. Isto é feito atribuindo custos para o casamento entre os pontos do modelo e cada candidato da entrada, e selecionando, para cada ponto do modelo, o ponto da entrada com menor custo. Essa seleção é feita de maneira gulosa, isto é, uma vez estabelecido o casamento entre o ponto do modelo analisado no momento e um candidato da entrada, este candidato é removido do espaço de busca para os próximos pontos.

Embora rápida, a solução gulosa pode apresentar uma desvantagem. Suponha que seja atribuído um casamento incorreto entre um ponto v_m^1 do modelo e um ponto v_i da entrada. Caso um próximo ponto v_m^2 pudesse ser casado v_i , este casamento já não seria possível, uma vez que o ponto v_i teria sido previamente retirado da solução. Entretanto, para o problema considerado, essa situação não fornece um problema grave, pois não existe um mapeamento único entre os pontos do modelo e da entrada que pode ser considerado aceitável. Isto é, no problema da definição de marcos 3D em uma malha densa, para cada marco facial, existe um conjunto de pontos dentro de uma certa região que pode ser considerado correto. Mesmo com a marcação manual, não é possível obter uma solução totalmente precisa. Desta forma, ainda com a remoção do ponto v_i , o ponto v_m^2 poderia ser mapeado corretamente com outros pontos candidatos próximos de v_i .

Para realizar o mapeamento de pontos entre o grafo do modelo e os candidatos da entrada, a abordagem estrutural requer a definição de um critério que estabelece custos nos vértices e nas arestas do grafo. O custo do vértice adotado leva em conta a informação local do ponto, relativa às coordenadas (R,G,B) na imagem de textura,

e à curvatura do ponto, como será descrito mais adiante. O custo das arestas leva em conta a dimensão e rotação de um vetor que representa a aresta. Formalmente, para cada par (v_m, v_i) , onde v_m corresponde aos vértices do modelo, e v_i corresponde aos vértices da entrada, deseja-se minimizar a função:

$$E(v_m, v_i) = \lambda d_A(v_m, v_i) + (1 - \lambda) d_S(v_m, v_i) \quad (3.1)$$

onde d_A pondera a aparência e d_S pondera a estrutura dos padrões a serem casados. O parâmetro $\lambda \in [0, 1]$ é usado como um fator de peso que define com qual intensidade aparência e estrutura devem ser levadas em conta para o casamento.

Em resumo, o casamento entre pontos de duas faces é realizado minimizando a equação 3.1 através do algoritmo guloso listado a seguir:

Algoritmo 1 Computa pares (v_m, v_i) de custo mínimo

```

1:  $P \leftarrow \emptyset$ 
2: para todo vértice  $v_m \in V_m$  faça
3:      $c_{min} \leftarrow \infty$ 
4:      $v_{min} \leftarrow \text{NULL}$ 
5:     para todo vértice  $v_i \in V_i$  faça
6:          $c \leftarrow E(v_m, v_i)$ 
7:         se  $c < c_{min}$  então
8:              $c_{min} \leftarrow c$ 
9:              $v_{min} \leftarrow v_i$ 
10:     $P \leftarrow P \cup \{(v_m, v_{min})\}$ 
11:     $V_i \leftarrow V_i - \{v_{min}\}$ 
12: devolva  $P$ 

```

As próximas seções detalham como é feito o cálculo dos custos relativos à aparência (d_A) e à estrutura (d_S) dos padrões a serem casados.

3.2.1 Custo de aparência

Como os modelos utilizados contém informações de geometria e textura registradas, para cada ponto tridimensional (x, y, z) , é atribuída uma coordenada (u, v) na imagem 2D correspondente. Dessa forma, todo ponto a ser casado possui um atributo $\mu_{UV}(v)$ referente a cor na coordenada (R,G,B) da imagem de textura.

Como mencionado, o fator $d_A(v_m, v_i)$, que faz parte da equação 3.1, diz respeito ao custo calculado levando em conta a aparência dos pontos v_m e v_i . Neste trabalho, a *aparência* é definida como um atributo local de cada ponto, que diz respeito, basicamente, à distância Euclidiana entre as cores de v_m e v_i , calculada a partir das coordenadas (R,G,B) desses pontos¹.

¹A utilização da distância Euclidiana como função de custo entre cores para o casamento de grafos foi retirada de Noma (2010).

Quando buscamos casar expressões faciais diferentes de uma mesma pessoa, é importante levar em conta a cor dos pontos na imagem de textura, pois é esperado que pontos correspondentes possuam cores similares (olhos, boca, etc.). Por outro lado, quando tentamos realizar casamentos entre indivíduos diferentes, essa similaridade não é respeitada, uma vez que indivíduos diferentes podem possuir tonalidades de pele bastante diferentes, assim como variações de cor dos olhos, sobrancelhas, barba etc. A Figura 3.6 ilustra um exemplo para essa situação.

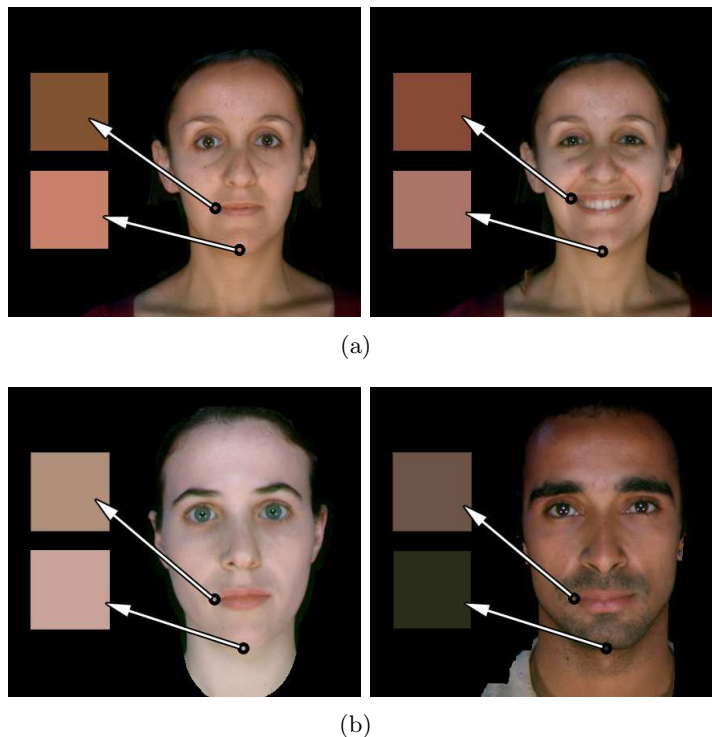


Figura 3.6: Comparação das cores dos pixels em dois pontos das faces a serem casadas. Em (a), mostra-se um exemplo para expressões de um mesmo indivíduo, e a cor dos pontos correspondentes é semelhante. Em (b), mostra-se um exemplo para faces de indivíduos diferentes, e não existe a mesma relação entre a cor dos pontos correspondentes, devido à tonalidade diferente de pele entre os indivíduos e à presença de barba.

Desta forma, nota-se que levar em conta a cor da imagem de textura por si só não oferece um parâmetro de comparação suficientemente bom para o casamento. Por isso, além da textura, o custo de aparência leva em conta, também, um outro tipo de imagem, relacionado a uma classificação da curvatura da malha.

A abordagem usada é denominada de *classificação HK*, sendo computada a partir dos mapas de curvatura média (H) e Gaussiana (K) de um modelo tridimensional, conforme descrito em Colombo *et al.* (2006). Seja S a superfície definida pela face 3D de modo que:

$$S = \{(x, y, z) | f(x, y) = z\}.$$

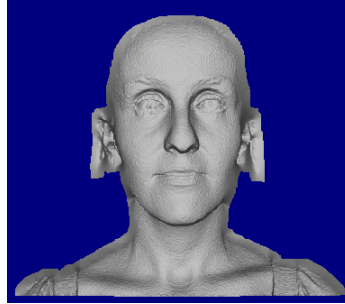
Então, para todo $(x_0, y_0, f(x_0, y_0)) \in S$, a curvatura média (H) e Gaussiana (K) são computadas como:

$$H(x_0, y_0) = \frac{(1 + f_y^2)f_{xx} - 2f_x f_y f_{xy} + (1 + f_x^2)f_{yy}}{2(1 + f_x^2 + f_y^2)^{\frac{3}{2}}}$$

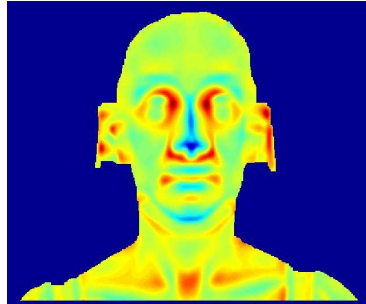
$$K(x_0, y_0) = \frac{f_{xx}f_{yy} - f_{xy}^2}{(1 + f_x^2 + f_y^2)^2}$$

onde $f_x, f_y, f_{xy}, f_{xx}, f_{yy}$ correspondem à primeira e à segunda derivadas de f em (x_0, y_0) .

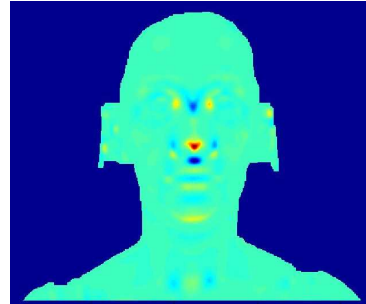
Dado que a segunda derivada é muito sensível a ruído, antes de calcular a curvatura, um filtro Gaussiano é aplicado sobre superfície da face, suavizando-a. Veja na Figura 3.7 os mapas de curvatura média e Gaussiana para a face de uma das pessoas do banco de dados.



(a) Face 3D



(b) Curvatura média



(c) Curvatura Gaussiana

Figura 3.7: Mapas de curvatura média e Gaussiana para uma face 3D.

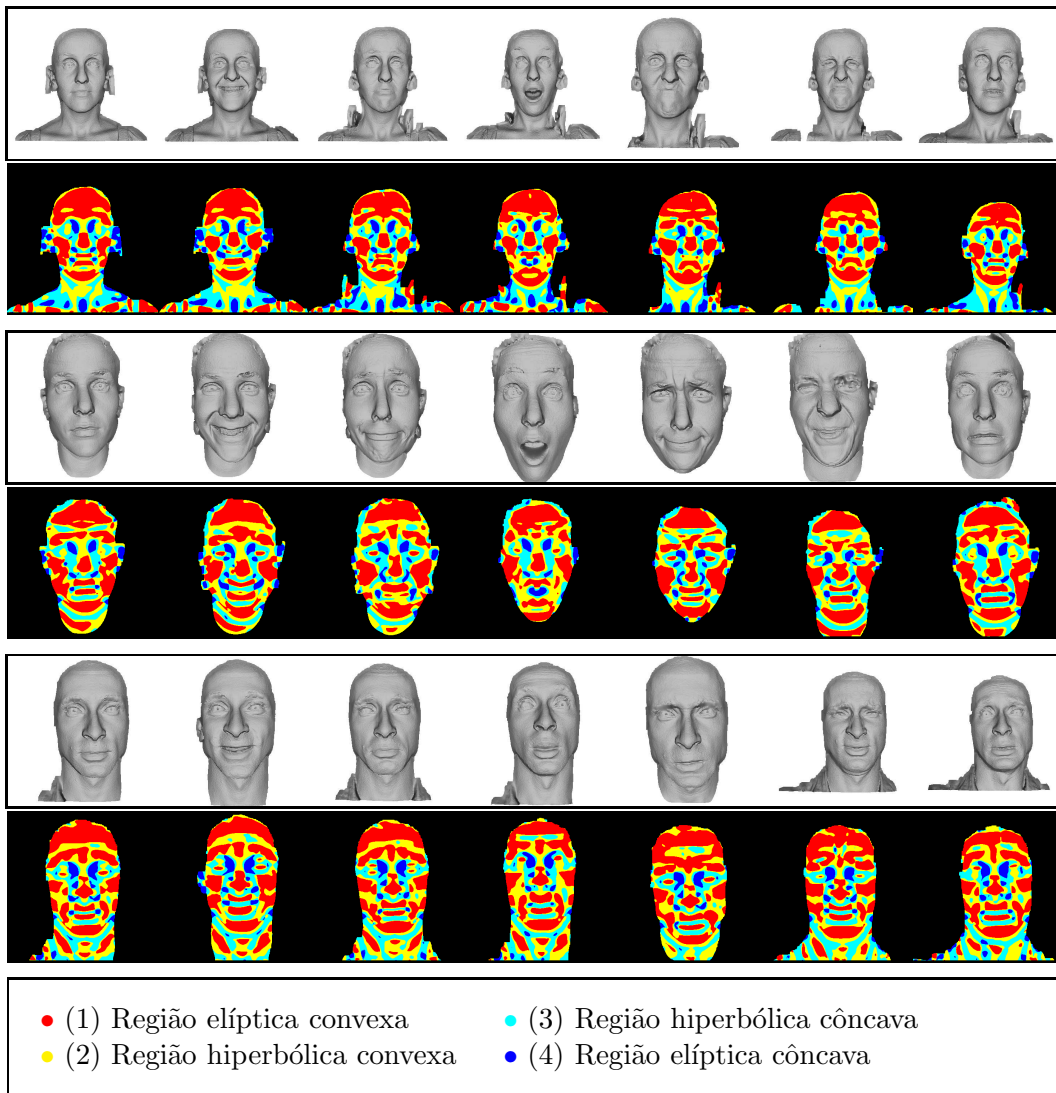
Após calcular os mapas de curvatura, a classificação HK cria um mapeamento de regiões hiperbólicas côncavas, hiperbólicas convexas, elípticas côncavas e elípticas convexas da face. O comportamento local da região é dado através do sinal dos mapas H e K em cada ponto da superfície, como mostra a Tabela 3.1².

²Quando $K = 0$, a classificação HK também especifica regiões cilíndricas convexas ($H < 0$), cilíndricas côncavas ($H > 0$) e planares ($H = 0$), entretanto, essas classificações não ocorrem na prática para o problema considerado, uma vez que é muito difícil encontrar um ponto no qual a curvatura vale exatamente 0.

Tabela 3.1: *Classificação HK.*

	$K < 0$	$K > 0$
$H < 0$	Região hiperbólica convexa	Região elíptica convexa
$H > 0$	Região hiperbólica côncava	Região elíptica côncava

A ideia de utilizar a classificação HK para o casamento baseia-se no fato de que diferentes pessoas e diferentes expressões faciais costumam possuir a mesma classificação em regiões correspondentes. A Figura 3.8 exibe a geometria para os três indivíduos exibidos na Figura 3.6 e a classificação HK correspondente.

**Figura 3.8:** *Classificação HK para indivíduos e expressões faciais diferentes.*

Note que, usualmente, pontos correspondentes costumam estar relacionados às mesmas regiões classificadas. Por exemplo, a maior parte da testa, bochechas, queixo e nariz

estão classificadas como pertencentes à região 1, cantos internos dos olhos e cantos da boca são normalmente classificados como pertencentes à região 4, etc. A Figura 3.9 exibe, para o mesmo exemplo da Figura 3.6, a cor de pixels correspondentes segundo a imagem da classificação HK. Diferente da situação que leva em conta o valor dos pixels na imagem de textura, pontos correspondentes tomados neste segundo exemplo possuem cores correspondentes na classificação HK.

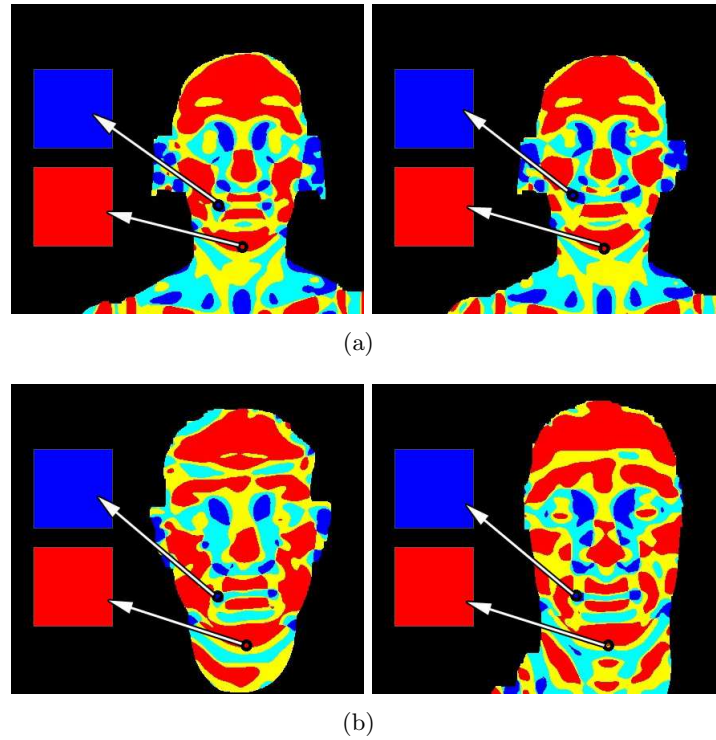


Figura 3.9: Comparação das cores referentes à classificação HK em dois pontos das faces a serem casadas.

Assim, além do atributo $\mu_{UV}(v)$, referente às coordenadas (R,G,B) da imagem de textura, todo ponto a ser casado possui também um atributo $\mu_{HK}(v)$, referente à imagem da classificação HK. Dessa forma, para um dado par (v_m, v_i) , o custo de aparência d_A é formulado da seguinte forma:

$$d_A(v_m, v_i) = \lambda_A d_{A_{UV}}(v_m, v_i) + (1 - \lambda_A) d_{A_{HK}}(v_m, v_i) \quad (3.2)$$

onde

$$d_{A_{UV}}(v_m, v_i) = \frac{\text{distânciaEuclidiana}(\mu_{UV}(v_m), \mu_{UV}(v_i))}{C_A},$$

$$d_{A_{HK}}(v_m, v_i) = \frac{\text{distânciaEuclidiana}(\mu_{HK}(v_m), \mu_{HK}(v_i))}{C_A}.$$

Nas equações anteriores, C_A é uma constante para normalizar os valores entre 0 e 1 e $\lambda_A \in [0, 1]$ é um fator de peso que pondera com que intensidade cada imagem deve ser levada em conta para o casamento. A distância Euclidiana entre atributos $\mu(v_1) = (R_1, G_1, B_1)$ e $\mu(v_2) = (R_2, G_2, B_2)$ vale $distânciaEuclidiana(\mu(v_1), \mu(v_2)) = \sqrt{(R_2 - R_1)^2 + (G_2 - G_1)^2 + (B_2 - B_1)^2}$.

Note que as cores utilizadas para a classificação HK estabelecem que existe semelhança máxima entre pontos pertencentes à mesma região, uma vez que a distância computada entre seus componentes (R,G,B) vale zero. Da mesma forma, pontos entre regiões côncavas e entre regiões convexas possuem ainda componentes em comum, demonstrando alguma similaridade.

3.2.2 Custo de estrutura

Quando selecionamos um ponto v_m do modelo e buscamos o correspondente v_i dentre os candidatos da entrada, levando em conta as informações de cor e curvatura, estamos apenas medindo a similaridade local desses pontos. Entretanto, é importante avaliar também a informação entre as estruturas a serem casadas. Isto é, quando buscamos casar o ponto v_m com algum candidato v_i , além dos atributos de cores similares, desejamos que o candidato possua uma posição relativa próxima de v_m . Dado que as faces foram previamente alinhadas, espera-se que o par v_i não esteja muito distante de v_m , e nem muito acima, abaixo, à esquerda ou à direita de v_m , por exemplo.

Para levar em conta a informação estrutural entre os grafos a serem casados, não basta minimizar o custo relativo aos vértices. É necessário, também, minimizar o custo relativo às arestas do grafo. Para isto, utiliza-se uma estrutura auxiliar denominada *grafo de deformação* (G_d). Inicialmente, o grafo de deformação é simplesmente uma cópia do grafo do modelo. Então, para todos os pontos candidatos da malha de entrada, testa-se o quanto a inserção do ponto na solução deforma o grafo do modelo, selecionando os pontos que minimizam essa deformação. A Figura 3.10 mostra um exemplo de como é realizado um casamento a partir do grafo de deformação.

Para medir a *deformação* do grafo G_d , é necessário definir atributos para as arestas do grafo. Na seção anterior, estabelecemos que cada vértice do grafo possui um atributo $\mu(v)$, que diz respeito às coordenadas (R,G,B) na imagem de textura ou de curvatura. Da mesma forma, cada aresta possui também um atributo $\nu(e)$. O atributo da aresta $\nu(e)$ diz respeito ao vetor que representa a aresta, cujas extremidades são os vértices que definem e . O custo das arestas para o casamento entre um ponto v_m e um candidato v_i , por sua vez, está relacionado à similaridade que cada aresta adjacente a v_m possui com relação às arestas adjacentes a v_i no grafo de deformação. Esta medida de similaridade leva em conta dois fatores: o ângulo e o comprimento dos vetores que representam as arestas a serem casadas, como exemplifica a Figura 3.11

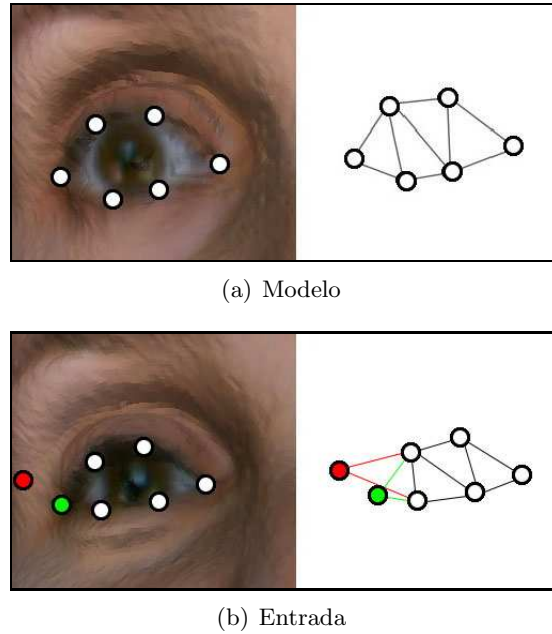


Figura 3.10: Exemplo para o casamento do olho em duas faces diferentes. Em (a) os pontos do olho são selecionados no modelo. Para encontrar o correspondente do canto externo do olho, são avaliados todos os pontos candidatos na entrada. Em particular, no exemplo em (b) testa-se a inclusão de dois pontos na solução final (ponto verde e ponto vermelho). Avalia-se, então a deformação que esses pontos geram no grafo à direita, selecionando o ponto verde como aquele que executa uma menor deformação sobre o grafo.

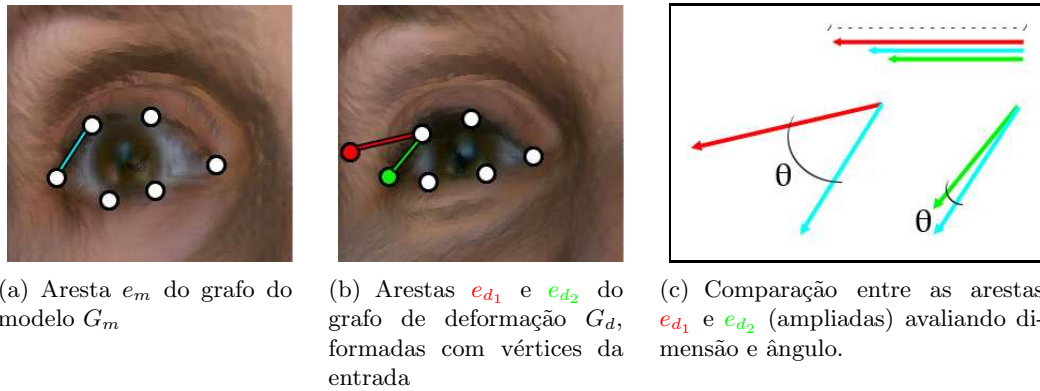


Figura 3.11: Comparação entre uma aresta do modelo e arestas candidatas da entrada. O ponto a ser casado é o mesmo do exemplificado na Figura 3.10, isto é, o canto externo do olho. Nota-se que, embora as dimensões de e_m , e_{d_1} e e_{d_2} sejam similares, o ângulo entre e_m e e_{d_1} é bem maior do que o ângulo entre e_m e e_{d_2} , representando uma solução mais custosa.

Note que a Figura 3.11 exibe a comparação para apenas uma aresta adjacente ao ponto referente ao canto externo do olho. Entretanto, o custo do casamento leva em conta todas as arestas adjacentes ao ponto a ser casado, extraindo uma média do custo segundo o número de arestas adjacentes. Assim, para o caso do custo relativo às arestas, a equação que deve ser minimizada é a seguinte:

$$d_S(v_m, v_i) = \frac{1}{|E(v_d)|} \sum_{e_d \in E(v_d)} c_{vec}(\nu(e_m), \nu(e_d)) \quad (3.3)$$

Na equação 3.4, c_{vec} representa custo vetorial, referente aos vetores que representam as arestas comparadas. Por sua vez, a informação que mede o quanto a estrutura do grafo é modificada é dada em termos de ângulo e módulo:

$$c_{vec}(\vec{v}_1, \vec{v}_2) = \lambda_S d_{S_{ang}}(\vec{v}_1, \vec{v}_2) + (1 - \lambda_S) d_{S_{mod}}(\vec{v}_1, \vec{v}_2) \quad (3.4)$$

em que $d_{S_{ang}}$ representa o custo angular e $d_{S_{mod}}$ representa o custo modular, calculados como:

$$d_{S_{ang}}(\vec{v}_1, \vec{v}_2) = \frac{|\cos\theta - 1|}{2},$$

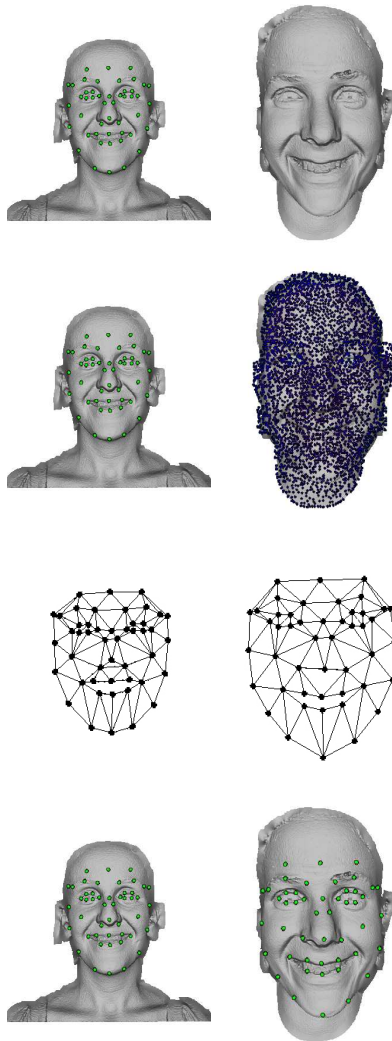
$$d_{S_{mod}}(\vec{v}_1, \vec{v}_2) = \frac{||\vec{v}_1| - |\vec{v}_2||}{C_S}.$$

Nas equações anteriores, C_S é uma constante para normalizar os valores entre 0 e 1 e $\lambda_S \in [0, 1]$ é um fator de peso que pondera a intensidade com a qual o custo angular e o custo modular devem ser levados em conta no casamento. θ representa o ângulo entre \vec{v}_1 e \vec{v}_2 e $|\vec{v}_1|$, $|\vec{v}_2|$ representam as dimensões de \vec{v}_1 e \vec{v}_2 , respectivamente.

De forma resumida, então, o casamento entre os pontos do grafo do modelo G_m e os pontos da entrada leva em conta informações de aparência e estrutura, ponderadas pelo parâmetro λ . O custo de aparência d_A pondera a distância entre atributos locais dos vértices a serem casados, como mostrado na equação 3.2. Já o custo estrutural d_S , formulado na equação 3.4, avalia a similaridade entre a média das arestas das quais fazem parte, no grafo G_d , os vértices a serem casados. O custo de aparência leva em conta a distância Euclidiana entre os componentes (R,G,B) na imagem de textura e de curvatura, e o peso das distâncias $d_{A_{UV}}$ e $d_{A_{HK}}$ é dado conforme o parâmetro λ_A . O custo de estrutura também pondera dois fatores, distância angular $d_{S_{ang}}$ e modular $d_{S_{mod}}$, através de λ_S . Um esquema resumido para o algoritmo é exibido na Figura 3.12.

No algoritmo proposto, conforme mencionado, o grafo G_d é iniciado como sendo uma cópia de G_m , e à medida que os pontos de menor custo vão sendo selecionados, o grafo de deformação vai sendo atualizado com estes novos pontos. Para os próximos pontos a serem casados, os cálculos levam em conta o grafo já atualizado. O problema desta abordagem é que, se for atribuído um casamento incorreto, os próximos cálculos passarão a considerar um grafo deformado de maneira não adequada, e os resultados seguintes poderão ficar comprometidos.

Por outro lado, o trabalho de Colombo *et al.* (2006) utilizou a classificação HK, também implementada no nosso estudo, para encontrar de forma robusta três marcos faciais: a ponta do nariz e os cantos internos dos olhos. O nariz é detectado como o



(a) O problema consiste em, dados pontos marcados na face modelo (face à esquerda), encontrar os pontos correspondentes na face de entrada (à direita).

(b) Um conjunto de pontos característicos na face de entrada é detectado de forma automática.

(c) Para todos os pontos detectados, buscam-se relações de isomorfismo entre o grafo do modelo e um subgrafo dos pontos característicos detectados na entrada, levando em conta textura, classificação HK e informação estrutural.

(d) Finalmente, o casamento ótimo calculado no passo anterior permite obter os pontos correspondentes na face de entrada.

Figura 3.12: Esquema geral para o casamento automático entre pontos de duas faces.

ponto de maior curvatura Gaussiana pertencente à região 1 (convexa elíptica). O canto do olho esquerdo é detectado como o ponto de maior curvatura média dentro da maior componente conexa da região 4 (côncava elíptica), à esquerda do nariz. O canto do olho direito é encontrado analogamente. Assim, optou-se por buscar os casamentos a partir de uma solução inicial, encontrada também de forma automática, a partir dos mapas de curvatura computados. A partir desses três pontos, os demais marcos foram encontrados seguindo a solução estrutural descrita.

3.3 Avaliação do método e resultados

Foram aplicados diferentes testes para avaliar o desempenho do casamento automático, comparando os erros encontrados em diferentes situações. Como dados de entrada, foram tomadas as 210 malhas referentes às sete expressões faciais dos 30 indivíduos apresentados na Figura 2.4.

Para avaliar o comportamento do algoritmo no casamento entre expressões faciais diferentes e pessoas diferentes, foram considerados dois tipos de testes. No primeiro, buscou-se casar os pontos marcados da face neutra com os pontos das demais expressões faciais, sempre para a mesma pessoa. No segundo, casaram-se os pontos marcados de uma mesma expressão, mas entre pessoas diferentes.

Para todos os testes, considerou-se $\lambda_S = \lambda_A = 0.5$, isto é, no custo de aparência, as imagens de textura e curvatura foram consideradas com o mesmo peso e, no caso da estrutura, distância angular e modular influenciaram igualmente no cálculo do custo. Para o parâmetro λ , que pondera aparência e estrutura, iterou-se valores entre $[0, 1]$ com um passo de 0.1, selecionando o melhor λ a partir do menor número de erros computado. Os resultados exibidos a seguir, portanto, foram calculados a partir do melhor λ , cujo valor varia para cada experimento.

3.3.1 Resultados para o casamento entre expressões faciais diferentes

As Figuras de 3.13 à 3.18 mostram os resultados obtidos para as duas primeiras pessoas do banco de dados, para o casamento entre a face neutra marcada e as demais expressões faciais. Em cada caso, a foto da pessoa à esquerda apresenta os pontos que foram manualmente selecionados na expressão neutra e a foto da pessoa à direita mostra os pontos correspondentes, automaticamente selecionados, em cada expressão, usando a abordagem proposta. Os pontos casados na cor verde são considerados como acerto por estarem próximos dos pontos determinados no gabarito. Já pontos em vermelho, apresentam casamentos considerados incorretos, por estarem distantes dos pontos do gabarito. Lembramos que, para o problema considerado, não existe uma solução única de correspondência, e por isso são aceitos pontos determinados dentro de uma vizinhança para os marcos do gabarito. A distância aceitável para os pontos do gabarito é normalizada em relação às dimensões da face e possui um valor variável de acordo com os pontos do modelo. Linhas entre as duas fotos mostram a correspondência entre os pontos que foram casados incorretamente.

Para os exemplos apresentados, nota-se que os resultados são bons quando existe pouca variação entre as expressões a serem casadas. Por outro lado, em expressões como alegria ou surpresa, nas quais a boca está aberta, ainda que alguns pontos consigam ser casados corretamente, ocorrem erros na região.

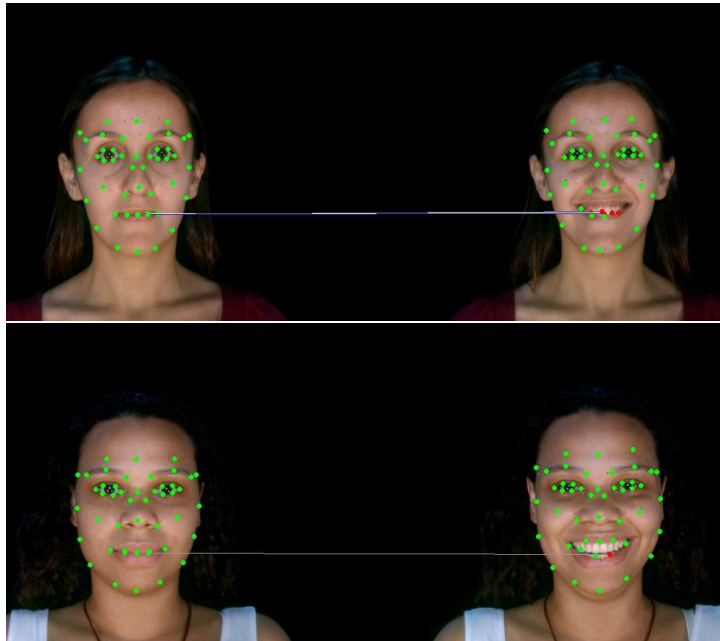


Figura 3.13: Resultados para o casamento entre a expressão neutra e a expressão de alegria.

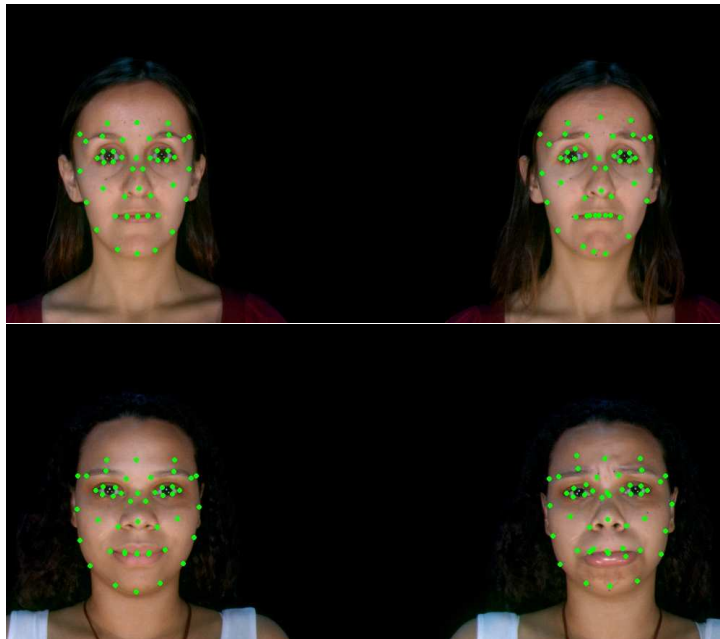


Figura 3.14: Resultados para o casamento entre a expressão neutra e a expressão de tristeza.

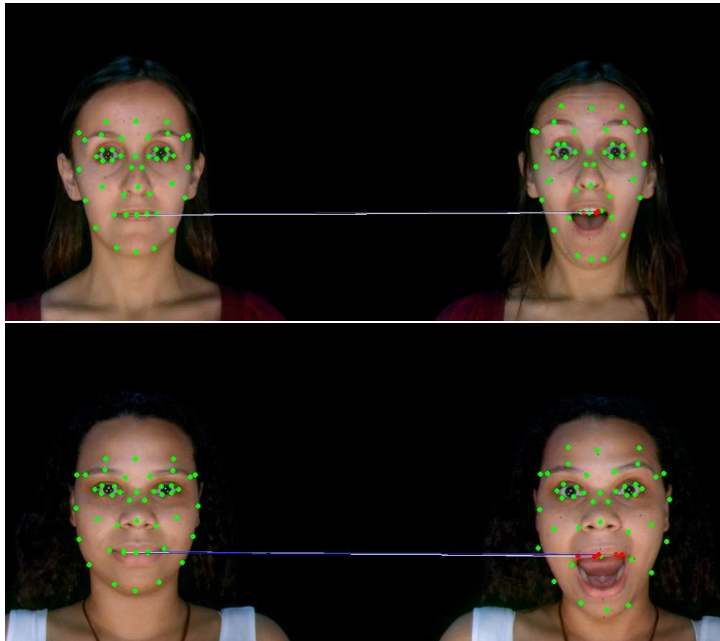


Figura 3.15: Resultados para o casamento entre a expressão neutra e a expressão de surpresa.

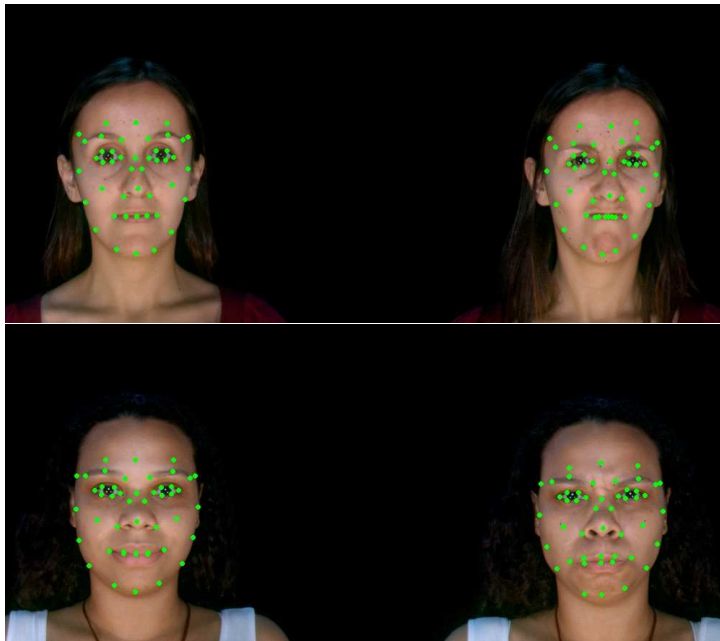


Figura 3.16: Resultados para o casamento entre a expressão neutra e a expressão de raiva.

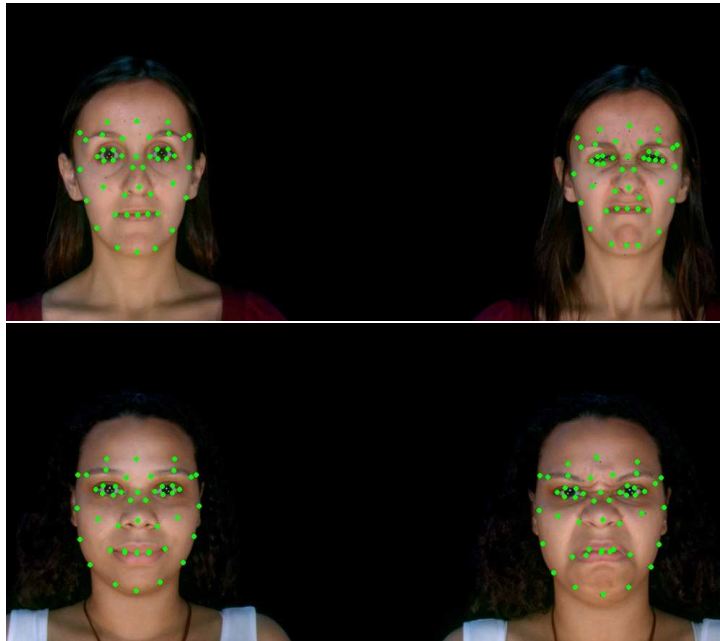


Figura 3.17: Resultados para o casamento entre a expressão neutra e a expressão de nojo.

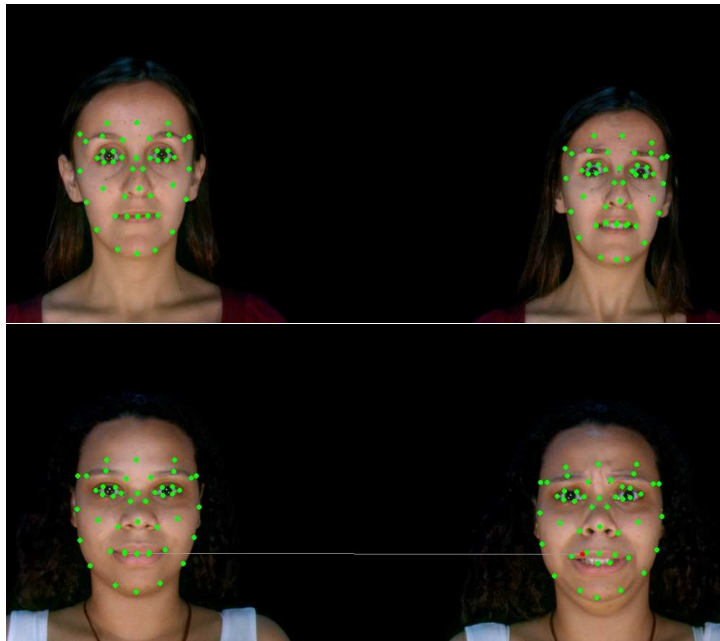
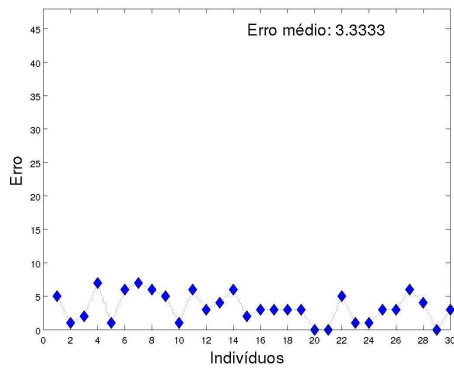


Figura 3.18: Resultados para o casamento entre a expressão neutra e a expressão de medo.

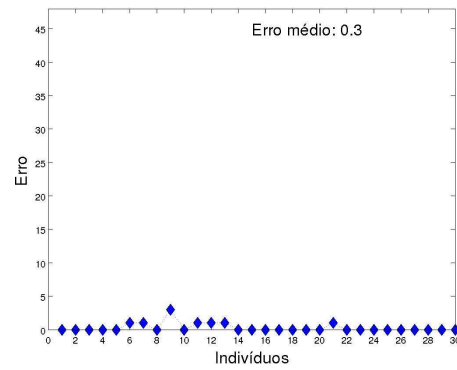
O casamento entre a expressão neutra e as demais expressões foi testado para todos os 30 indivíduos do banco de dados e este comportamento se manteve para todas as pessoas. Na maior parte dos casos, o algoritmo se comportou bem para o casamento entre olhos, sobrancelhas, nariz, pontos ao redor do rosto e situações nas quais a boca está parcialmente aberta. Entretanto, quando a boca está muito aberta, são encontrados

erros nessa área.

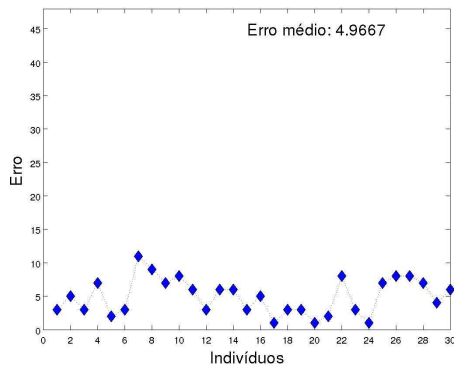
A Figura 3.19 apresenta gráficos que mostram o comportamento do algoritmo no casamento entre a face neutra e cada expressão, para as 30 pessoas. O índice dos indivíduos no gráfico corresponde ao mesmo apresentado na Figura 2.4.



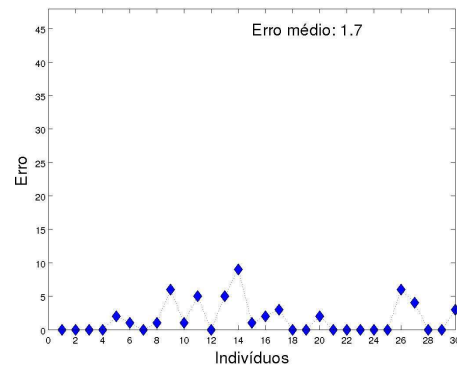
(a) Alegria



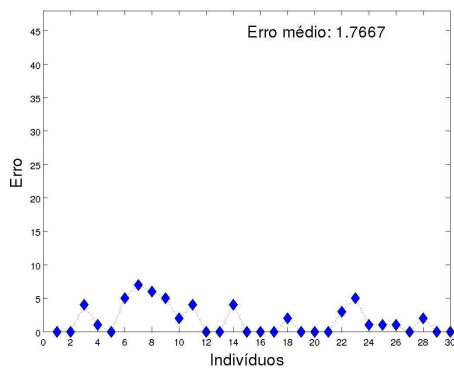
(b) Tristeza



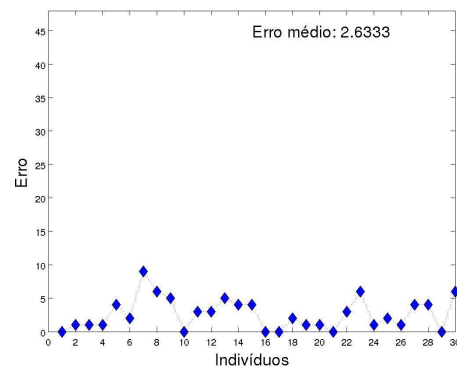
(c) Surpresa



(d) Raiva



(e) Nojo



(f) Medo

Figura 3.19: Erro obtido no casamento entre a face neutra e as demais expressões faciais.

A partir dos dados mostrados nos gráficos, nota-se que os melhores resultados foram obtidos para o casamento entre a face neutra e a expressão de tristeza, na qual a maioria das pessoas apresenta a boca fechada. Da mesma forma, os maiores erros foram obtidos para a expressão de surpresa, na qual as pessoas costumam apresentar a boca bastante aberta. Para as expressões de nojo e raiva, existe uma certa variação nos resultados, uma vez que, no momento da aquisição, certas pessoas optaram por mostrar os dentes, por exemplo, enquanto outras preferiram manter a boca fechada.

3.3.2 Resultados para o casamento entre indivíduos diferentes

O segundo teste avaliou o casamento da mesma expressão facial entre indivíduos diferentes. As Figuras 3.20 à 3.26 mostram alguns resultados para diferentes pessoas do banco de dados. Novamente, em cada teste, a foto da esquerda corresponde aos pontos marcados manualmente, enquanto a foto da direita apresenta os pontos detectados automaticamente a partir deles.

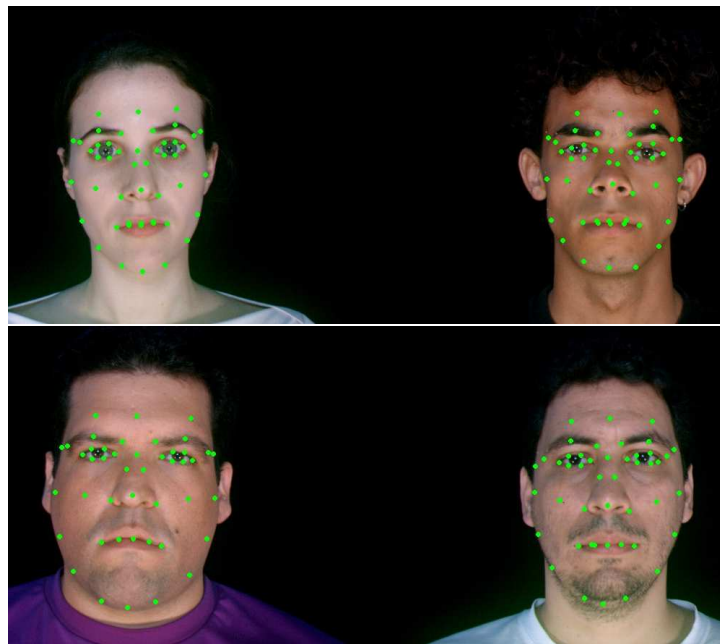


Figura 3.20: Resultados para o casamento da expressão neutra entre pessoas diferentes.

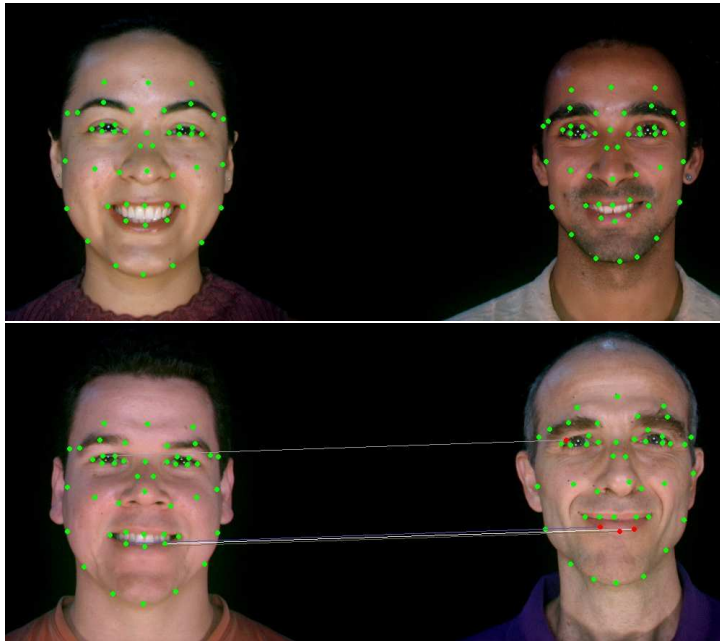


Figura 3.21: Resultados para o casamento da expressão de alegria entre pessoas diferentes.



Figura 3.22: Resultados para o casamento da expressão de tristeza entre pessoas diferentes.

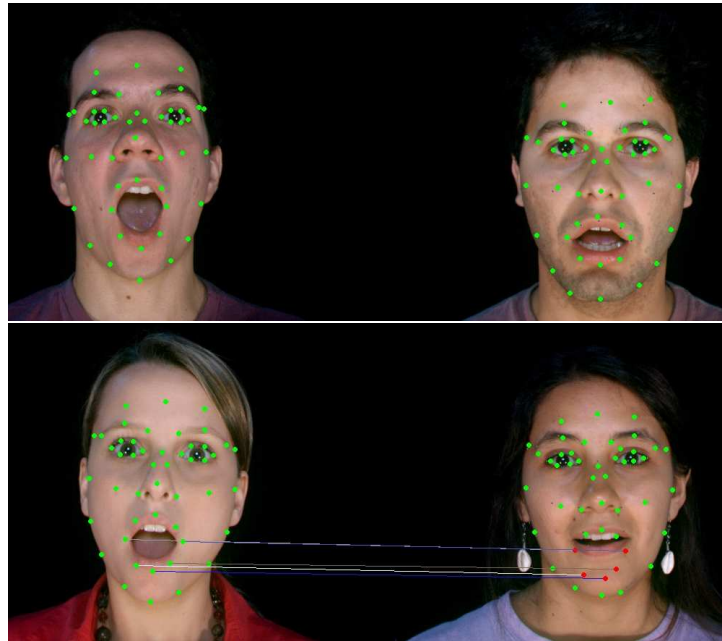


Figura 3.23: Resultados para o casamento da expressão de surpresa entre pessoas diferentes.

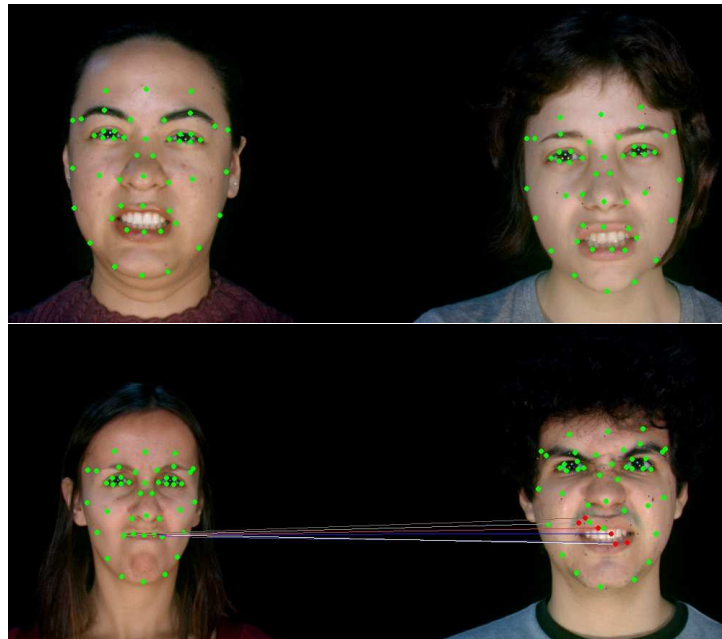


Figura 3.24: Resultados para o casamento da expressão de raiva entre pessoas diferentes.

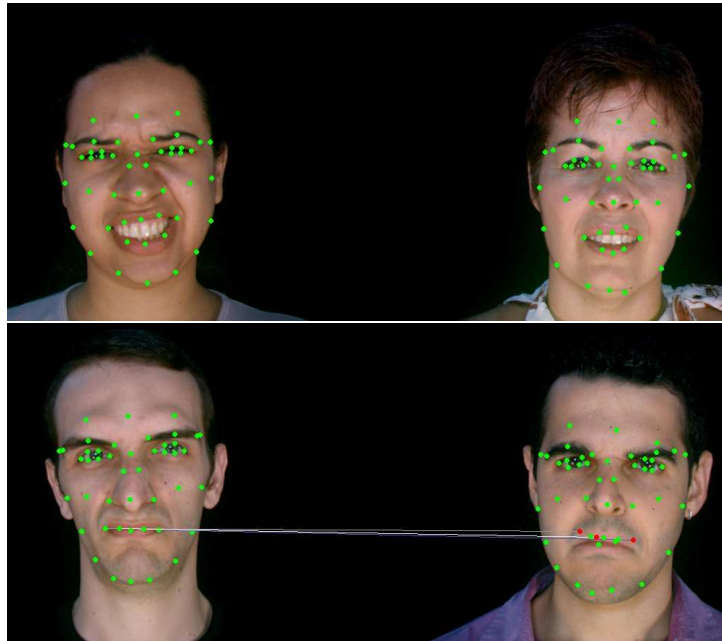


Figura 3.25: Resultados para o casamento da expressão de nojo entre pessoas diferentes.

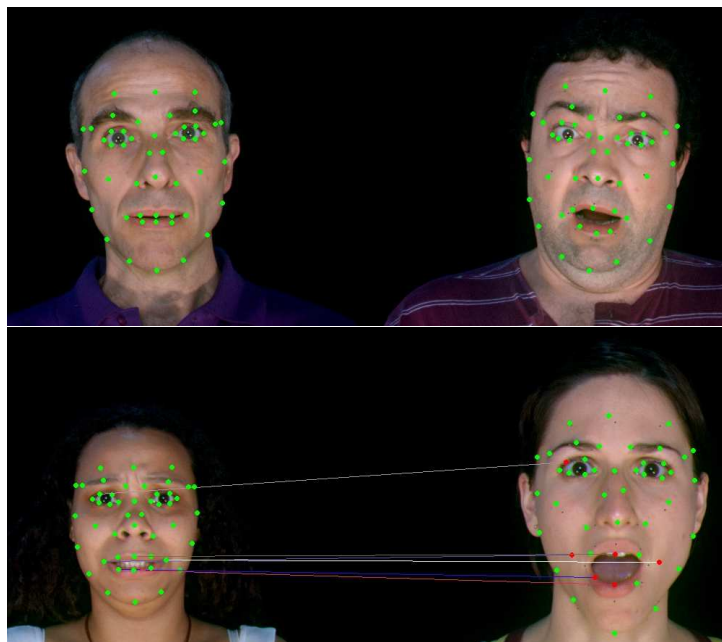


Figura 3.26: Resultados para o casamento da expressão de medo entre pessoas diferentes.

Como mostra o exemplo, o algoritmo funcionou bem para o casamento entre expressões neutras, ainda que considere indivíduos com características faciais diferentes (cor da pele, formato do rosto, etc.). Para as demais expressões, assim como ocorreu no teste anterior, alguns erros foram obtidos, sobretudo na região da boca, quando as expressões a serem casadas são muito diferentes. Para cada expressão, o primeiro exem-

plo (na linha superior) exibe bons resultados do programa, que conseguiu encontrar um casamento correto para todos os pontos, mesmo com expressões e características faciais razoavelmente distintas. O segundo exemplo (na linha inferior), por outro lado, exibe falhas de correspondência, em sua maioria, na região da boca da entrada, quando a sua estrutura difere muito da boca do modelo.

Novamente, o teste foi aplicado para todas as expressões faciais de todos os indivíduos do banco de dados. Para cada expressão e cada indivíduo, foi casada a expressão correspondente de todas as demais pessoas e o erro médio desse processo foi computado. Assim, os gráficos da Figura 3.27 exibem, para cada expressão facial, o erro médio de cada indivíduo em questão, ao ter a sua face casada com todas as demais entradas possíveis.

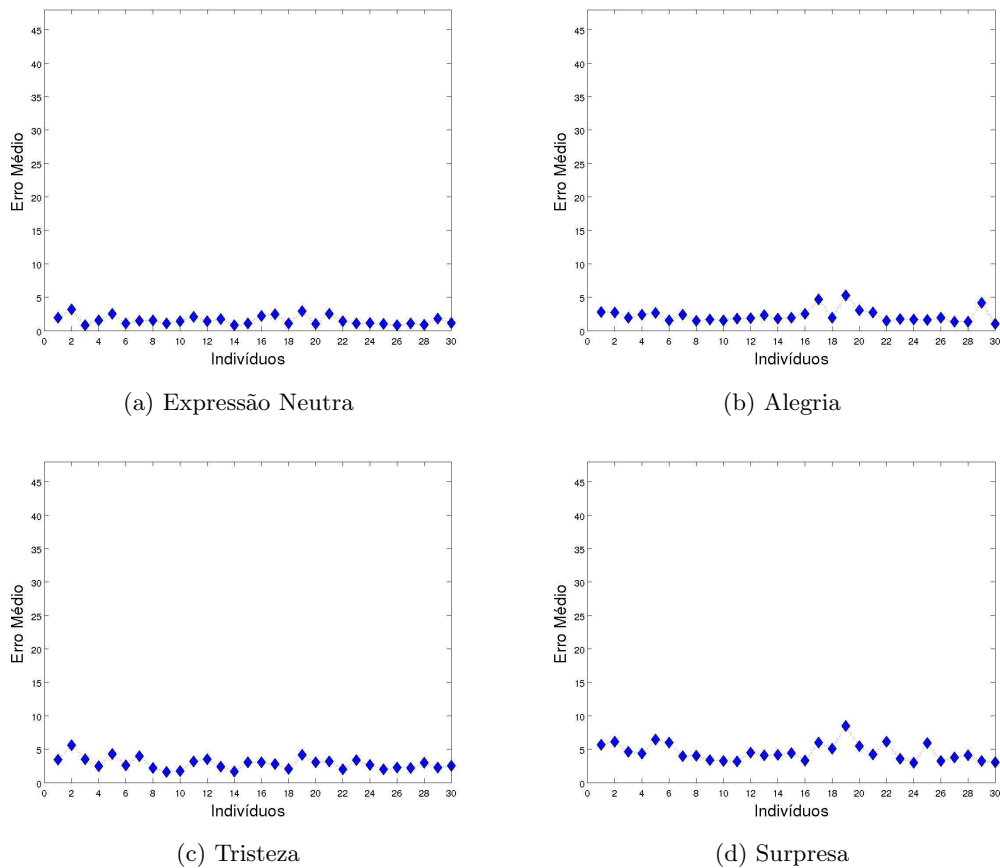


Figura 3.27: Resultados para o casamento de marcos faciais entre pessoas diferentes.

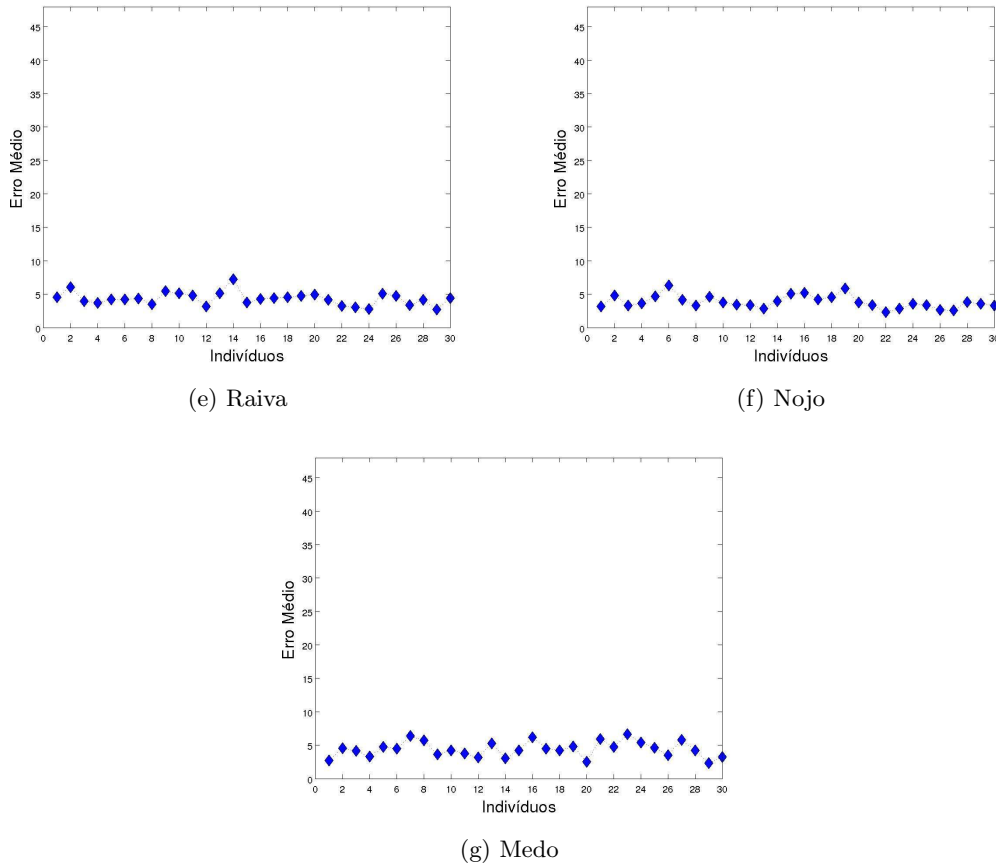


Figura 3.27 (continuação)

No banco de dados considerado, as expressões variam de forma significativa de uma pessoa para outra, tanto em termos de intensidade (por exemplo, abertura grande ou pequena da boca na expressão de surpresa) quanto das próprias características exibidas (por exemplo, a opção de abrir ou não a boca na expressão de raiva ou nojo). Dessa forma, certos problemas são ocasionados no casamento. Entretanto, para o casamento da face neutra, na qual existe um padrão de comportamento (boca fechada, olhos abertos e músculos relaxados) resultados com poucos erros são obtidos.

De modo geral, nota-se que o algoritmo é capaz de realizar bons casamentos, e, ainda nos piores casos, cerca de 90% dos pontos conseguem ser casados corretamente. Dessa forma, o procedimento de *landmarking* pode ser completado manualmente nas regiões nas quais houveram falhas no casamento. Para completar o processo, foi implementada uma pequena interface para a seleção manual dos pontos incorretos. Assim, ainda que não represente um processo totalmente automático, o método exige uma interação mínima do usuário, que consiste em um procedimento muito mais rápido e menos exaustivo do que a etapa original de selecionar manualmente todos os 48 pontos em cada uma das malhas.

Finalmente, é importante esclarecer que, ainda que os pontos casados sejam exibi-

dos em 2D nas figuras anteriores, a saída para o programa consiste em um conjunto de pontos tridimensionais. A Figura 3.28 exibe o resultado tridimensional dos pontos detectados no exemplo da Figura 3.15, visualizados a partir da interface implementada para a seleção manual dos pontos incorretos.

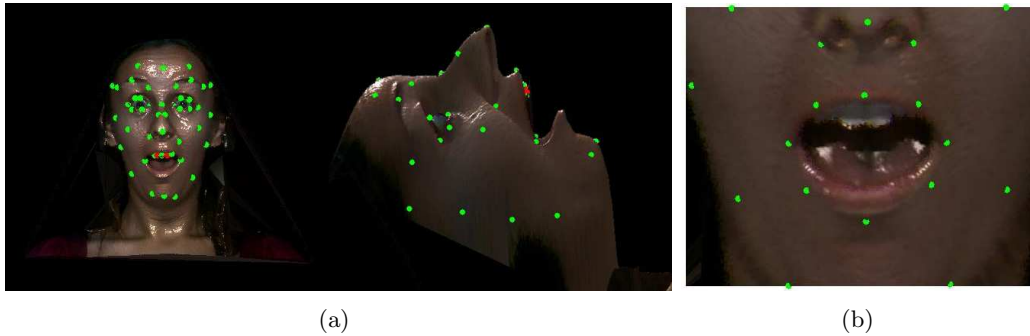


Figura 3.28: Interface para visualização do marcos tridimensionais detectados (a) e re-marcação dos pontos incorretos (b).

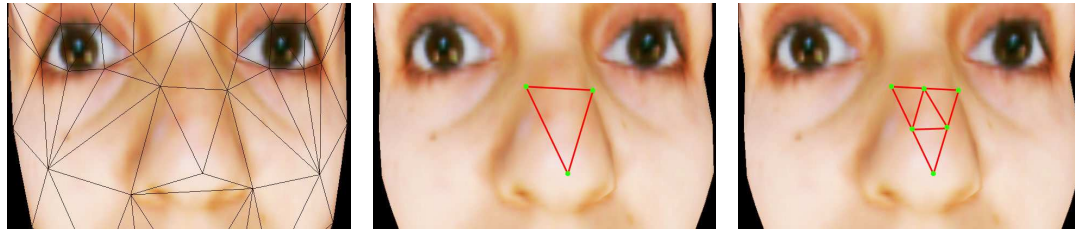
3.4 Subdivisão e projeção

Até o momento, este capítulo descreveu o procedimento proposto para encontrar um conjunto esparsos de marcos correspondentes em faces 3D. Para completar o processo de registro necessário para a animação, devemos encontrar, também, os correspondentes dos pontos intermediários. Como mencionado na seção 1.3, um dos objetivos deste trabalho consiste em evitar a busca densa de correspondências faciais devido às desvantagens mencionadas na seção 1.2.2. Para isso, a partir dos pontos esparsos selecionados, calculamos os pontos intermediários correspondentes aplicando um procedimento de subdivisão e projeção que é executado separadamente em cada malha.

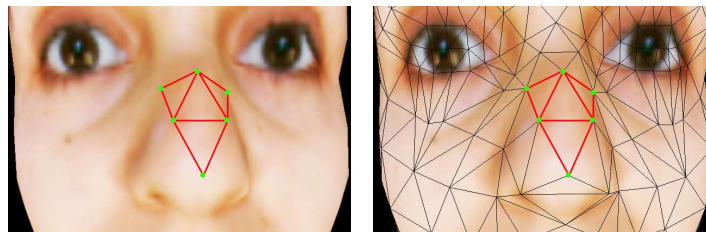
Para que o processo de subdivisão seja aplicado, é necessário um modelo inicial de pontos e triangulação. Usamos o modelo definido pelo grafo G_m , resultando em modelo inicial de 48 vértices e 74 triângulos conforme foi mostrado na Figura 3.4. O modelo utilizado não considera a região interna da boca, uma vez que a geometria da língua e de dentes superiores e inferiores não está disponível em todas as expressões, dado que essas estruturas não são obtidas pelo escâner 3D. Além disso, nos métodos de animação que serão descritos mais adiante, utiliza-se apenas a imagem da face neutra para o mapeamento de textura, na qual essa região não é mostrada. Assim, é possível animar outros modelos usando somente as informações da face neutra.

Para processar os dados brutos obtidos pelo escâner 3D e gerar modelos com geometrias equivalentes, seguimos a abordagem de Guskov *et al.* (2000) e Golovinskiy *et al.* (2006), na qual o modelo inicial de 48 pontos é subdividido e reprojetoado sucessivamente. A malha bruta é refinada para 9648 vértices através da aplicação de duas subdivisões lineares seguidas de duas subdivisões de Loop [Loop (1987)]. Ambos os

esquemas de subdivisão permitem dividir cada triângulo do modelo inicial em quatro triângulos menores, acrescentando novos vértices no ponto médio de cada aresta. Depois de cada subdivisão, os vértices novos são reprojitados na malha bruta, utilizando a direção de seus vetores normais, como mostra a Figura 3.29.



(a) Triângulos da malha inicial (b) Seleciona-se um triângulo (c) Cria-se novos vértices no ponto médio das arestas



(d) Projeta-se os novos vértices na malha (e) Repete-se o processo para cada triângulo inicial

Figura 3.29: *Esquema de subdivisão da malha inicial.*

Em cada face do banco de dados, o processo de subdivisão e projeção é aplicado usando o mesmo modelo inicial e o mesmo número de etapas. Assim, as malhas resultantes possuem o mesmo número de vértices, pois cada triângulo inicial é dividido considerando a mesma estratégia. Além disso, o processo permite gerar malhas cujos vértices possuem correspondência total entre si, gerando, em cada iteração, modelos que apresentam aproximações cada vez mais refinadas da malha bruta.

Note que cada malha é processada de forma independente, evitando o problema de casar regiões faciais planas, mencionado na seção 1.2.2. Veja, na Figura 3.30, o resultado das subdivisões e projeções em cada iteração, e na Tabela 3.2 a estratégia considerada no sistema de animação. Poderia se obter uma malha com mais pontos correspondentes aplicando-se mais procedimentos de subdivisão. Entretanto, para o caso da animação, não há uma diferença visual perceptível utilizando uma malha mais densa.

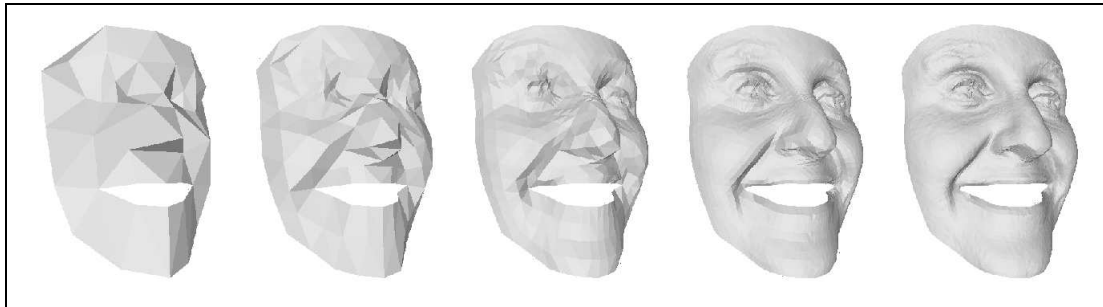


Figura 3.30: Resultado do processo de subdivisão da malha inicial após quatro iterações.

Tabela 3.2: Estratégia de subdivisão e projeção usando o modelo inicial definido pelo grafo G_m .

Descrição	Vértices	Triângulos
Modelo inicial	48	74
Subdivisão linear e projeção	170	296
Subdivisão linear e projeção	636	1184
Subdivisão de Loop e projeção	2456	4736
Subdivisão de Loop e projeção	9648	18944

Capítulo 4

Método de animação por interpolação

Como mencionado na seção 1.2.2, os métodos estudados para a interpolação de malhas faciais baseiam-se em encontrar correspondências entre todos os pontos das malhas, o que possui desvantagens. A abordagem proposta Capítulo 3 sugere que poucos pontos correspondentes sejam indicados, deixando as demais correspondências para uma etapa de subdivisão aplicada em cada malha, de forma independente.

Após gerar os modelos subdivididos, é possível, para cada expressão, calcular o deslocamento de cada vértice em relação à face neutra. Essa informação é mantida em um conjunto de matrizes utilizadas para animar os outros modelos a partir da informação de textura e geometria para a face neutra, ou apenas geometria. O fluxograma para o método é exibido na Figura 4.1.

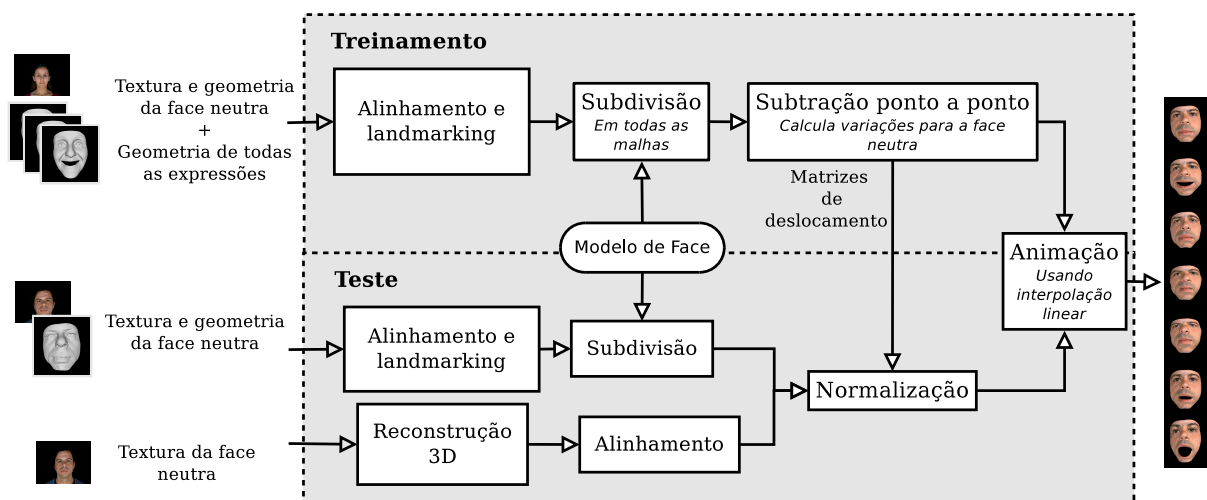


Figura 4.1: Esquema do fluxo de dados do sistema de animação. Cada bloco representa um processo e cada seta representa o fluxo de informação entre os processos. Na fase de teste, o sistema recebe como entrada um modelo de textura e geometria ou uma foto 2D.

Neste capítulo, a seção 4.1 explica como os dados subdivididos podem ser interpolados e utilizados para a transferência de expressões faciais. Na seção 4.2, são exibidas algumas aplicações para o método de animação e o resultado dessas aplicações são exibidos e avaliados na seção 4.3.

4.1 Processo de animação

O procedimento de subdivisão para a animação deve ser executado a partir de um conjunto de pontos faciais correspondentes. Esses pontos podem ser manualmente selecionados ou encontrados usando a técnica de casamento descrita na seção anterior. Para a estratégia de animação descrita neste capítulo, não existe uma restrição sobre o modelo inicial utilizado. Adotamos o modelo de 48 marcos faciais com o qual trabalhou-se no Capítulo 3.

Após os modelos serem subdivididos, para cada vértice da face neutra é possível localizar o vértice correspondente em todas as expressões, uma vez que todos recebem o mesmo índice no processo de subdivisão. Portanto, uma vez definidas as correspondências ponto-a-ponto entre cada modelo, é trivial calcular a variação da posição de um vértice com relação à face neutra. Considerando que existem m modelos a serem interpolados, cada um contendo n vértices, denotamos esses modelos por

$$P_k, \quad k = 1, \dots, m$$

onde P_k representa as expressões faciais (P_1 é a face neutra), e a posição de um vértice da pose P_k por

$$v_{xi}^k, \quad v_{yi}^k, \quad v_{zi}^k, \quad i = 1, \dots, n$$

Assim, a diferença linear entre cada vértice de uma determinada pose e seu vértice correspondente na face neutra é representada por

$$\Delta x_i^k = v_{xi}^k - v_{xi}^1 \tag{4.1}$$

$$\Delta y_i^k = v_{yi}^k - v_{yi}^1 \tag{4.2}$$

$$\Delta z_i^k = v_{zi}^k - v_{zi}^1 \tag{4.3}$$

A Figura 4.3 exhibe as diferenças entre a face neutra e as demais expressões consideradas.

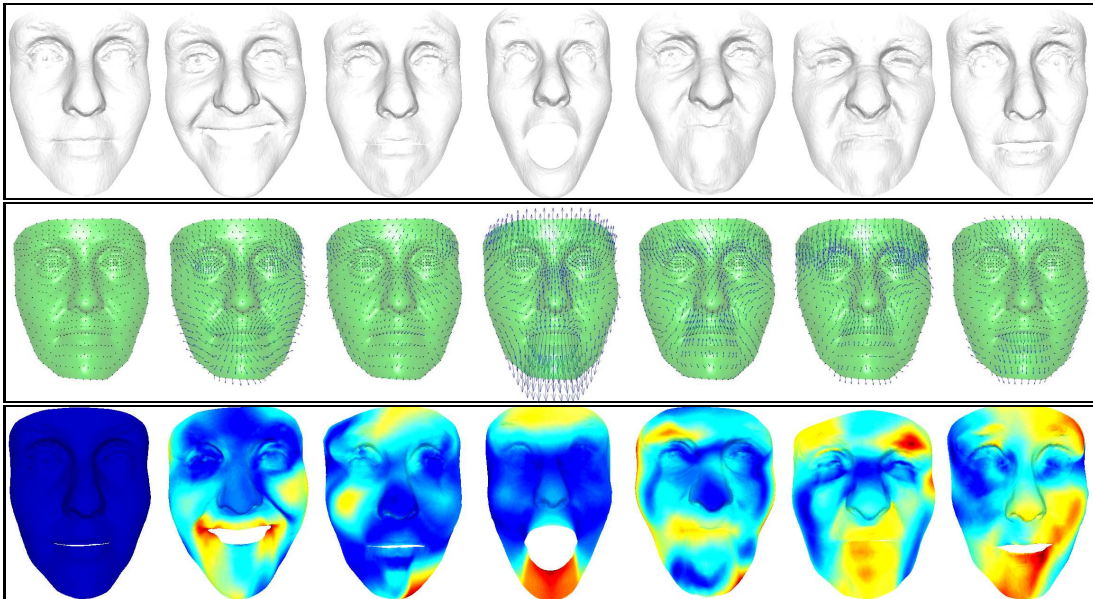


Figura 4.2: Geometria das expressões do conjunto de dados (acima), vetores denotando os deslocamentos calculados a partir da face neutra (centro) e um mapa de cores para a aplicação de deslocamentos (abaixo). No mapa, cores frias indicam regiões da face que são pouco modificadas, enquanto cores quentes indicam regiões de alta mudança em relação à face neutra.

É comum que o rosto da pessoa não fique estático no momento da aquisição das expressões faciais. Portanto, antes de computar a diferença com relação à face neutra, é necessário alinhar todos os modelos a serem interpolados. No casamento automático, os modelos tridimensionais são previamente alinhados utilizando o algoritmo ICP. Caso a animação seja feita a partir de pontos manualmente selecionados, é necessário aplicar o ICP, então, para alinhar os modelos, antes de computar a diferença entre cada ponto.

Após computar os deslocamentos de todos os vértices, é possível construir uma matriz $M_{n \times 3}^k$ para cada uma das k poses, com as translações que devem ser aplicadas nos vértices da face neutra. O processo de interpolação gradualmente aumenta essa variação até o seu valor final, gerando uma animação. Também é possível combinar várias poses em uma única expressão, através da aplicação de parcelas dos deslocamentos das poses a serem combinadas. Há uma variedade de maneiras de como as seis expressões universais podem ser combinadas gerando novas poses [Krumhuber *et al.* (2010); Pelachaud (2009); Rodrigues (2007)]. Essa característica pode ser aplicada no nosso sistema gerando expressões que originalmente não fazem parte do banco de dados.

Ao final do processo, uma animação é obtida dada a geometria de todas as expressões de uma única pessoa do banco de dados. O conhecimento obtido nesse processo pode ser usado para animar outras pessoas do banco de dados fornecendo-se somente a geometria da face neutra. Considere que todos os passos descritos anteriormente foram aplicados sobre uma pessoa arbitrária denominada *base*. Se o mesmo processo de subdivisão for aplicado à face neutra de uma outra pessoa do banco de dados, a malha

resultante será equivalente à malha neutra anteriormente subdividida da base.

Uma vez que o conjunto de matrizes M^k é calculado para a base, os seus valores podem ser usados para transferir as expressões da base para outro modelo. No entanto, deve-se levar em conta que as dimensões do modelo podem ser significativamente diferentes das dimensões da base. Assim, a influência dos deslocamentos transferidos poderia ser muito pequena, no caso em que o modelo é muito maior do que a base, ou muito grande, caso contrário. Portanto, uma normalização é aplicada, levando em conta as dimensões das *bounding boxes* do modelo e da base. Os deslocamentos feitos pelo vértice i do modelo considerando a pose k são calculados como:

$$\Delta x_i^k = \Delta x_{i (base)}^k \times \frac{X_{modelo}}{X_{base}} \quad (4.4)$$

$$\Delta y_i^k = \Delta y_{i (base)}^k \times \frac{Y_{modelo}}{Y_{base}} \quad (4.5)$$

$$\Delta z_i^k = \Delta z_{i (base)}^k \times \frac{Z_{modelo}}{Z_{base}} \quad (4.6)$$

onde X , Y e Z representam as dimensões das *bounding boxes* em cada eixo, para a base e o modelo.

Além disso, deve-se considerar que, além das dimensões, a posição e rotação do modelo e da base podem ser diferentes. O método ICP é aplicado para alinhar o modelo com relação à base a partir da qual as animações serão transferidas.

Na verdade, não é relevante se o novo modelo a ser animado faz parte do banco de dados ou não. Podemos simplesmente considerar qualquer modelo que contenha as informações de geometria da face neutra para produzir uma animação semelhante. Uma vez que a busca de correspondências entre as malhas baseia-se na seleção de pontos e etapa de subdivisão, é possível gerar modelos equivalentes a partir de malhas que possuam topologias consideravelmente distintas.

4.2 Aplicações

No final da seção 4.1 mencionou-se que a transferência de expressões conforme a metodologia proposta pode ser realizada entre modelos que possuem topologias diferentes. Esse fato nos permite incluir duas aplicações adicionais ao trabalho: usar e animar o resultado de modelos reconstruídos a partir de fotografias 2D e gerar novas expressões para os modelos de pessoas reais do banco de dados, calculando as matrizes de deslocamento de outros modelos que possuam poses diferentes.

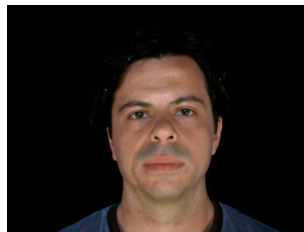
Além disso, uma estratégia que permite calcular as diferenças entre diferentes malhas pode fornecer outras aplicações além da animação. Ainda neste capítulo, mostraremos como o método foi usado para gerar caricaturas automáticas.

4.2.1 Sistema de reconstrução facial

Além da capacidade de gerar uma animação tendo como entrada a geometria de um modelo exibindo a expressão neutra, buscou-se uma situação ainda mais elaborada: obter um modelo animado dado uma única foto 2D como entrada.

Um sistema de reconstrução facial baseado em Análise de Componentes Principais (PCA) é usado para a obtenção de um modelo 3D a partir de uma fotografia 2D [Mena-Chalco (2010)]. Nesse projeto, foram capturadas as faces dos indivíduos que constituem o banco de dados usado atualmente no nosso estudo. Uma vez coletados, os dados são submetidos a um processo de treinamento e marcadores são alinhados na forma resultante. Dada como entrada uma imagem frontal da face a ser reconstruída, um conjunto de pontos característicos é selecionado manualmente na imagem 2D. O conjunto de pontos é usado para normalizar a textura de entrada. A geometria facial 3D é produzida projetando a textura normalizada sobre o espaço de geometria obtido na etapa de treinamento. A projeção é realizada usando uma base vetorial PCA e uma função de otimização linear para relacionar a informação de textura 2D com a geometria 3D, como descrito em Mena-Chalco *et al.* (2009). A reconstrução 3D, por fim, é dada através do mapeamento da textura normalizada sobre a geometria.

É importante ressaltar que a face foi reconstruída utilizando o mesmo modelo proposto no nosso trabalho, eliminando a necessidade de aplicar sobre ela o procedimento de subdivisão. Entretanto, poderíamos facilmente utilizar faces reconstruídas a partir de outros modelos. Usando um sistema de reconstrução genérico, bastaria selecionar-se, de forma manual ou automática, os 48 pontos na face reconstruída, efetuar a subdivisão como descrito na seção 3.4 e a animação como descrito na seção 4.1.



(a) Fotografia 2D de entrada



(b) Modelo tridimensional resultante

Figura 4.3: Modelo reconstruído a partir de uma fotografia 2D usando a abordagem proposta por Mena-Chalco (2010).

4.2.2 Sistema de síntese de voz

O banco de dados de pessoas reais utilizado neste trabalho contém modelos para a face neutra e para as seis expressões propostas por Ekman, como indicado na Figura 4.3. Para adicionar mais expressões à esses modelos, o procedimento descrito no capítulo anterior foi aplicado a um outro conjunto de malhas tridimensionais, que contém a geometria de outras expressões faciais não consideradas no banco de dados. Esse conjunto de modelos foi criado pelo software FaceGen Modeler (www.facegen.com), que gera modelos com outras expressões faciais além das seis usadas, como, por exemplo, expressões referentes a visemas¹. Os modelos gerados são artificiais, cuja geometria difere bastante dos modelos reais utilizados. A face neutra criada pelo software foi exportada, juntamente com as expressões de visemas e a expressão do modelo com os olhos fechados, e todas essas malhas foram fornecidas como entrada para o procedimento descrito na seção 4.1, calculando a matriz de deslocamento para cada pose. Uma vez que, após o processo de subdivisão, o modelo neutro artificial é equivalente ao modelo neutro real, as matrizes podem ser utilizadas para transferir as expressões de um para o outro.

Os modelos foram, então, usados por um sistema de síntese de voz que manipula a face, sincronizando o movimento dos lábios com um texto sintetizado. O texto é digitado pelo usuário e pode ser falado nos idiomas inglês e português. O sistema desenvolvido utiliza uma biblioteca de *text-to-speech* para reconhecer os fonemas contidos no texto, e uma *callback* é registrada para notificar o sistema quando um evento fonema ocorre. Em seguida, o visema relacionado é exibido, de acordo com um mapeamento previamente definido de fonemas e visemas proposto pelo padrão MPEG-4. Um visema pode corresponder a mais de um fonema e não existe ainda um consenso de quantos visemas existem ou de quantos visemas são necessários para uma animação facial consistente. O MPEG-4 define 15 visemas (incluindo o silêncio) para o inglês, que são mostrados na Tabela 4.1².

Como não existe, até o momento, um padrão oficial de mapeamento de visemas para o português, uma aproximação é realizada, considerando-se apenas um subconjunto de visemas do MPEG-4 que correspondem a fonemas presentes na língua portuguesa³. Durante a fala humana, o visema exibido no momento atual é afetado pelo visema apresentado anteriormente, e também a expressão da face no momento da fala [Rodrigues (2007)]. No nosso sistema, esses detalhes são levados em consideração, usando o fato de que várias poses podem ser facilmente combinadas. A expressão do modelo com os olhos fechados é usada como um movimento de piscar, para proporcionar maior realismo à animação.

¹Um visema é o correspondente visual de um fonema, isto é, o formato da boca quando a face exhibe um fonema.

²Ao longo deste texto, os fonemas estão representados segundo a codificação estabelecida pelo Alfabeto Fonético Internacional (AFI).

³O visema referente aos fonemas /θ/, /ð/ (veja Tabela 4.1) não faz parte do português.

Tabela 4.1: *Visemas e os fonemas relacionados para o idioma inglês, no padrão MPEG-4.*

Id	Fonemas	Exemplos
0	nenhum	-
1	/p/, /b/, /m/	put, bed, mill
2	/f/, /v/	far, voice
3	/θ/, /ð/	think, that
4	/t/, /d/	Tip, doll
5	/k/, /g/	call, gas
6	/ʃ/, /ʒ/	chair, join
7	/s/, /z/	Sir, zeal
8	/l/, /n/	Lot, not
9	/r/	red
10	/a/	car
11	/ε/	bed
12	/i/	Tip
13	/ɔ/	top
14	/u/	book

4.2.3 Geração automática de caricaturas

Uma caricatura é uma representação da face na qual a forma de alguns atributos faciais aparece exagerada. Comumente, esses atributos referem-se a partes da face nas quais já existe uma pequena disparidade em comparação com uma face média. A capacidade de realizar interpolações pelo cálculo de distâncias lineares é usada para gerar caricaturas automáticas. A ideia da abordagem é sugerida por [Blanz e Vetter \(1999\)](#) e consiste em caricaturizar o rosto de um indivíduo aumentando a sua distância em relação a uma face média.

Em nosso trabalho, a face média foi obtida pelo cálculo da média de cada vértice das faces neutras pertencentes ao banco de dados utilizado, exibido previamente na Figura 2.5. Para computar a média dos vértices, todas as faces foram subdivididas usando o procedimento descrito na Seção 3.4. A Figura 4.4(a) mostra a geometria dos 30 indivíduos a partir dos quais a face média foi computada, após o procedimento de subdivisão que permitiu gerar modelos topologicamente equivalentes. A face média resultante é mostrada na Figura 4.4(b).

Os modelos de face são alinhados e a diferença linear entre cada vértice do modelo e o vértice correspondente na face média é calculada. Os valores obtidos desses deslocamentos são aplicados sobre o modelo de face para criar caricaturas de uma forma simples. Ao aumentar a sua distância em relação à face média, é possível aumentar a disparidade de atributos nos quais já havia contrastes em relação à face média. É importante notar que os deslocamentos aplicados de forma reversa no modelo de face permitem obter uma deformação que aproxima o modelo da face média.

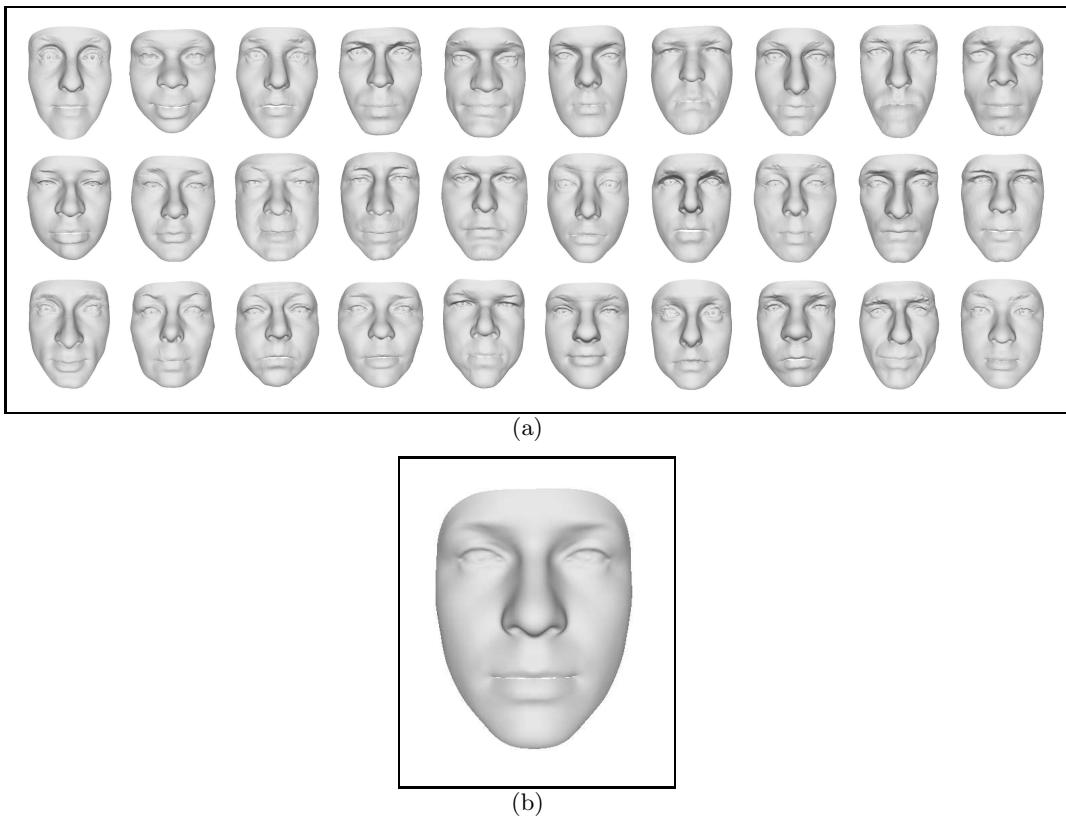


Figura 4.4: Faces consideradas no cálculo da face média para a geração de caricaturas. (a) Geometria facial 3D dos 30 indivíduos após a subdivisão. (b) Face neutra média.

4.3 Resultados

O sistema foi testado com dados reais tendo sido avaliado em alguns experimentos. Como ponto de partida, tomou-se a face do primeiro indivíduo do banco de dados. A partir dos pontos manualmente selecionados na face neutra, os marcos correspondentes foram detectados de forma automática nas demais expressões, como mostraram os resultados exibidos na seção anterior. Os pontos casados incorretamente foram remarcados manualmente usando a interface implementada, e cada modelo referente a uma expressão facial foi subdividido.

A Figura 4.5 apresenta os resultados da interpolação das expressões faciais para essa pessoa do banco de dados, a partir da sua face neutra e das seis expressões faciais. Esses dados foram considerados na fase de treinamento para o cálculo das matrizes de deslocamento aplicadas nos experimentos que serão descritos a seguir.

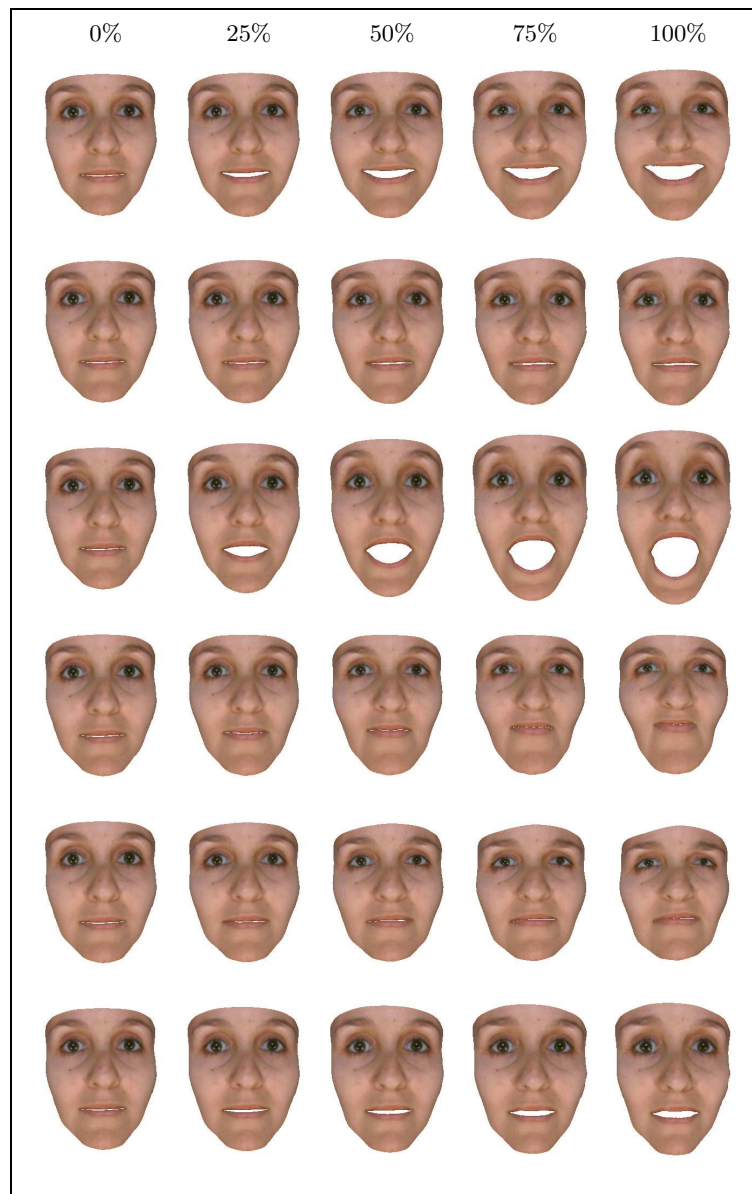


Figura 4.5: Resultados para uma pessoa do banco de dados, fornecendo como entrada a geometria de todas as expressões e a textura da face neutra. As interpolações das seis expressões são apresentadas gradativamente em cada linha da imagem. De cima para baixo, são exibidas as expressões de alegria, tristeza, surpresa, raiva, nojo e medo.

Como mencionado, esses resultados foram obtidos a partir do casamento estrutural dos pontos, tendo sido manualmente marcada apenas a face neutra. A fim de comparar esse resultado com o produto obtido usando uma marcação manual dos 48 marcos faciais em cada expressão, a Figura 4.6 exhibe os resultados para a animação de cada expressão nessas duas situações. Devido à alta densidade da malha, a posição dos pontos na marcação semiautomática não corresponde à posição exata dos pontos marcados manualmente. Nota-se que a diferença de precisão para a definição dos marcos iniciais altera, ainda que pouco, o resultado final da animação. Entretanto, ainda que a

discussão sobre a qualidade dos resultados da animação consista em um julgamento subjetivo, ambos os resultados correspondem a soluções viáveis no sentido de que é possível obter-se animações que estão de acordo com as expressões de uma pessoa real.

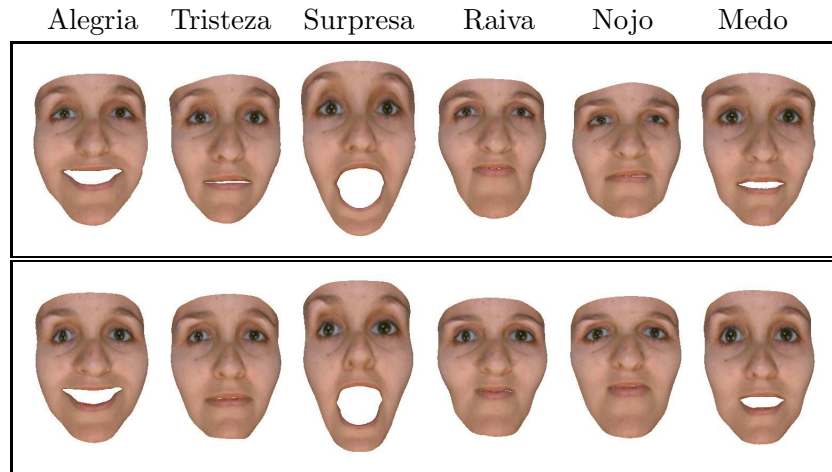


Figura 4.6: Comparação entre a animação produzida a partir do registro semiautomático (topo) e manual (abaixo). No primeiro caso, apenas a face neutra foi manualmente marcada, e no segundo, o modelo inicial foi também manualmente selecionado em cada uma das seis expressões.

Tomando como base os resultados previamente exibidos para a primeira pessoa do banco de dados, alguns experimentos foram realizados para avaliar a transferência das expressões do modelo base para outros modelos. A matriz de deslocamento considerada foi calculada a partir das malhas subdivididas através do método de marcação semiautomático. Um experimento foi considerado com o intuito de investigar a transferência de expressões para um outro modelo, dada a sua informação de textura e geometria para a face neutra. O modelo alvo, para qual as expressões faciais da base foram transferidas, não faz parte do banco de dados exibido na Figura 2.4 e nem do banco de treinamento para o algoritmo de reconstrução relatado em [Mena-Chalco et al. \(2009\)](#). Entretanto, esse foi adquirido juntamente com os outros modelos, possuindo uma estrutura geométrica similar. Os resultados para a animação desse indivíduo tomando como entrada apenas a sua expressão neutra são apresentados na Figura 4.7(a).

Um segundo experimento foi realizado para animar uma face 3D reconstruída tendo como entrada apenas uma fotografia 2D. Neste experimento, utilizamos, novamente, o mesmo indivíduo apresentado na Figura 4.7(a), entretanto, fornecendo como entrada apenas a sua imagem de textura. Enfatizamos que a fotografia para este indivíduo foi tirada sob condições de luz controladas, como o restante dos indivíduos do banco de dados. O resultado do experimento é mostrado na Figura 4.7(b).

Para completar os testes referentes à transferência de expressões faciais, avaliou-se, também, a animação de uma face reconstruída a partir de uma fotografia arbitrária. A imagem usada não pertence ao banco de dados e conta, inclusive, com a presença de

sombras na face. Os resultados são mostrados na Figura 4.7(c).

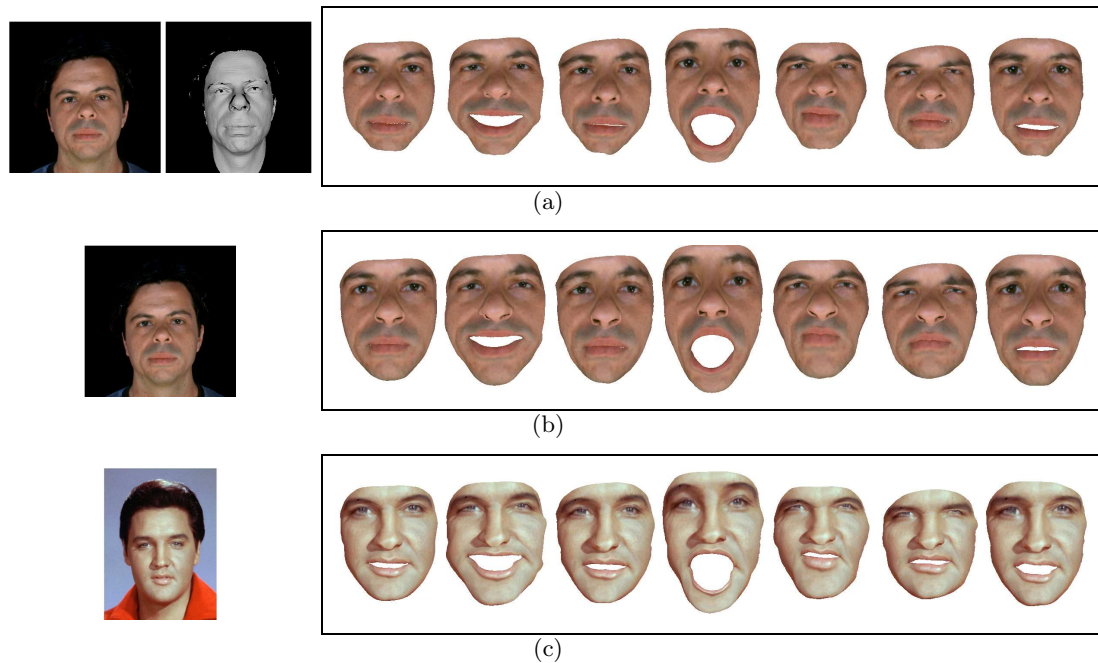


Figura 4.7: Resultados para diferentes tipos de entrada (a) textura e geometria da face neutra, (b) apenas a textura da face neutra de uma foto tirada sob condições controladas, (c) apenas a textura de uma foto arbitrária. Cada imagem está associada a uma expressão facial com 100% de deslocamento.

Em outro experimento, cujos resultados são exibidos na Figura 4.8, mostrou-se como as expressões de modelos artificiais foram aplicadas aos modelos reais utilizados no trabalho, permitindo que estes fossem usados como parte do sistema de animação e síntese de voz desenvolvido. O modelo artificial contém cerca de 6.000 vértices, tendo uma topologia que difere de forma significativa da dos modelos reais. Note que nem todos os métodos de transferência de expressão funcionam quando os modelos têm topologias muito diferentes. Alguns métodos de interpolação que dependem da busca de correspondências densas entre as malhas poderiam falhar nessa situação.

Finalmente, no último experimento, caricaturas automáticas de duas pessoas foram geradas a partir da face média exibida na Figura 4.4.

O efeito de aumentar e diminuir os deslocamentos dos dois modelos com relação à face média é mostrado na Figura 4.9. A Figura 4.9(a) mostra a caricatura de uma pessoa que também faz parte do cálculo da face média e a caricatura mostrada na Figura 4.9(b) considera uma pessoa que não faz parte desse cálculo.

Vídeos apresentando os resultados para todos os experimentos mencionados estão disponíveis em www.vision.ime.usp.br/~dedea/videos.

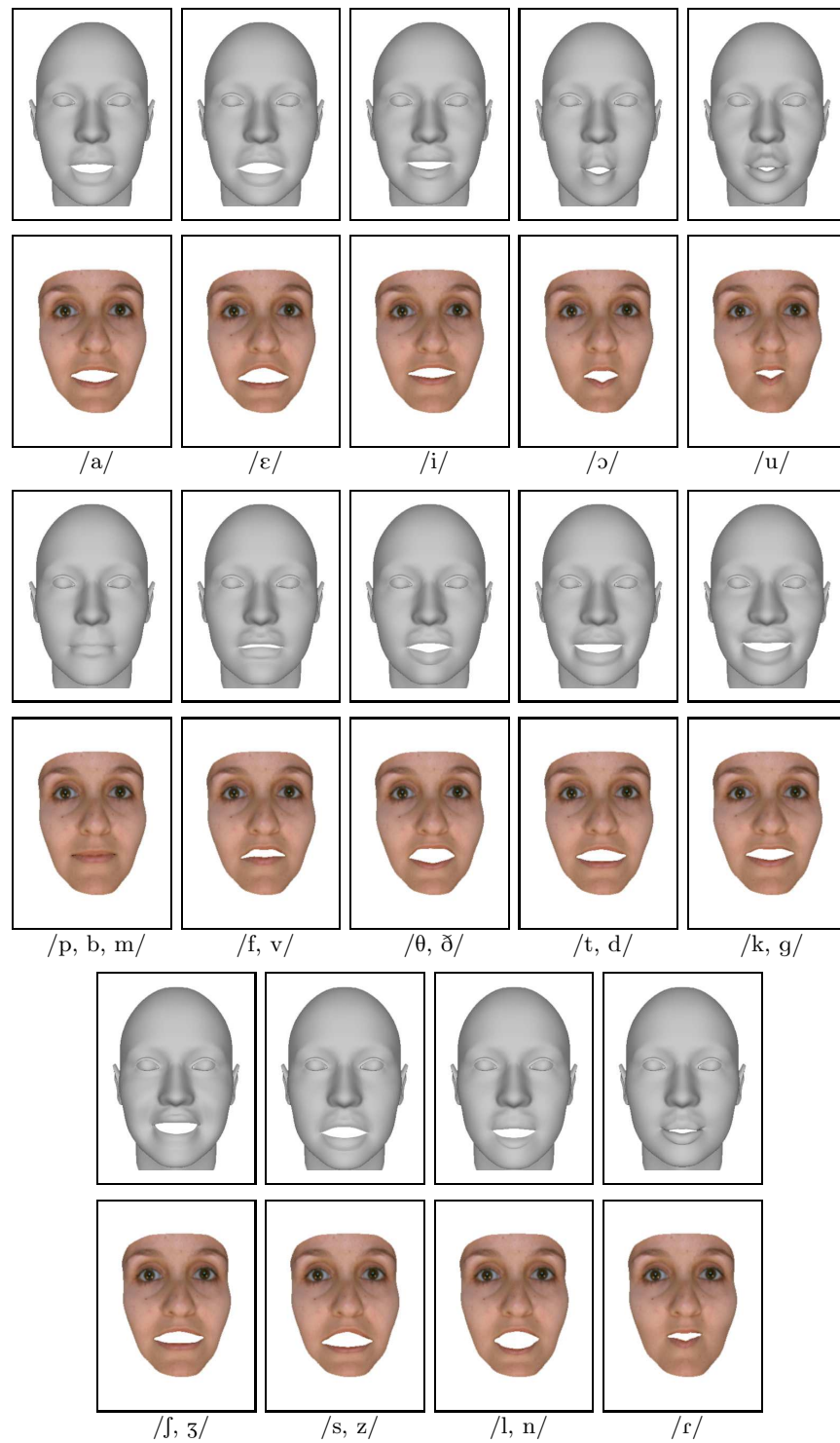


Figura 4.8: Visemas utilizados no sistema de animação, transferidos através de modelos artificiais. As duas primeiras linhas mostram a transferência dos visemas relativos as vogais.

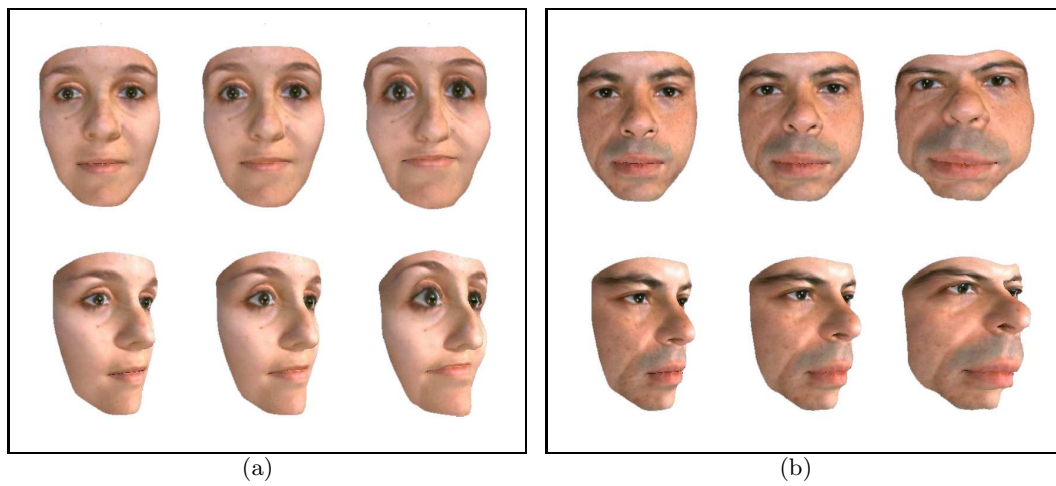


Figura 4.9: Resultados da geração automática de caricaturas. Modelo original (centro), modelo transformado diminuindo os deslocamentos com relação à face média (à esquerda), e o modelo caricaturizado pelo aumento dos deslocamentos (direita).

Capítulo 5

Método de animação por modelo paramétrico

Na seção 1.2.1, comentou-se sobre as diferentes técnicas existentes que são comumente aplicadas em animação facial. Embora este trabalho tenha sido iniciado abordando o método de interpolação, os procedimentos de marcação de pontos e subdivisão puderam também ser utilizados para a implementação de um outro método, baseado em um modelo paramétrico. Assim, além de fornecer mais uma aplicação para a estratégia de animação elaborada no estudo, pode-se também comparar os resultados das duas abordagens

Neste método, exploramos a técnica de parametrização direta utilizando o modelo do MPEG-4, aplicando-a ao mesmo conjunto de modelos reais. O método é usado para que, a partir de uma face no estado de neutralidade, possam ser geradas poses-chaves referentes às seis expressões faciais propostas por Ekman e aos 15 visemas descritos no MPEG-4.

A seção 5.1.1 descreve a teoria necessária para entender o funcionamento do padrão MPEG-4, relatando os pontos faciais alterados e os parâmetros de animação usados. A seção 5.1.3 relata como esses parâmetros foram aplicados em regiões da face vizinhas aos pontos característicos para obter uma animação consistente. Então, a seção 5.2 descreve como os parâmetros de baixo nível obtidos na fase anterior foram usados para gerar as expressões de emoções e visemas. O fluxograma para todo o processo é exibido na Figura 5.1.

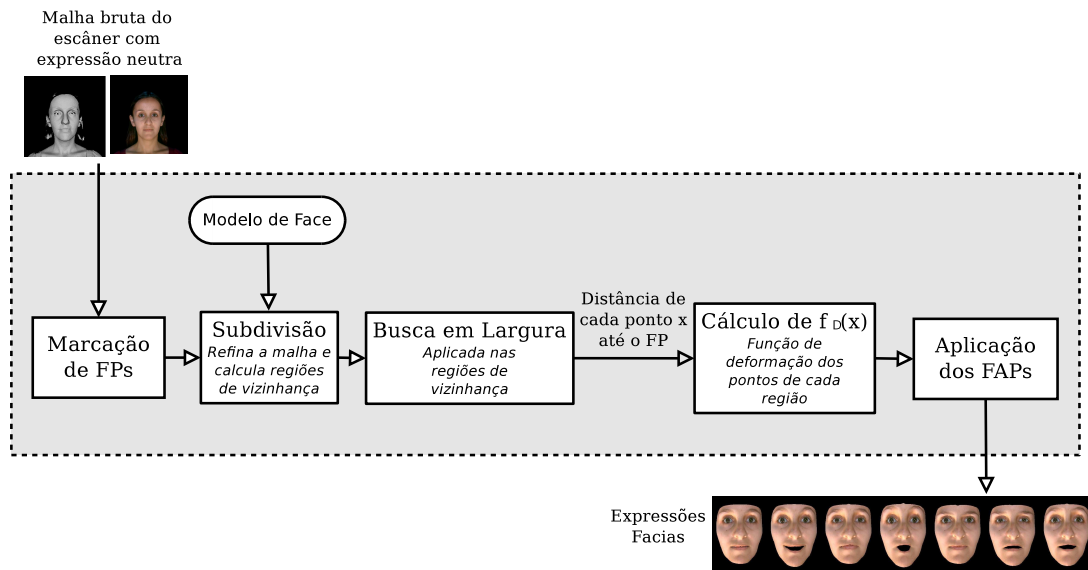


Figura 5.1: Fluxograma contendo os passos realizados para gerar uma animação facial a partir da malha neutra obtida pelo escâner.

5.1 Processo de animação

5.1.1 FAPs e FPs

O padrão MPEG-4 define 84 pontos característicos faciais denominados FPs (*Feature Points*) e 68 parâmetros de animação denominados FAPs (*Facial Animation Parameters*). Dos 68 parâmetros, existem dois FAPs de alto nível: o FAP 1, que representa os visemas e o FAP 2 que representa as expressões faciais. Todos os demais parâmetros são denominados de baixo nível e modificam pequenas porções da face, de modo que, sendo aplicados em conjunto, são capazes de gerar as expressões de alto nível. Para exemplificar, tomamos o movimento de levantar o meio da sobrancelha esquerda. Esse movimento é dado pelo FAP de baixo nível *raise_Lm_eyebrow* que atua sobre o centro da sobrancelha movendo-o para cima. A intensidade desse movimento deve estar entre um intervalo especificado pelo padrão, levando em conta, também, uma unidade de normalização.

Dos 84 pontos definidos no MPEG-4, alguns deles atuam sobre dentes, língua, etc. No caso deste projeto, uma vez que se trabalha com dados digitalizados, não é obtida a região interna da boca e portanto desconsidera-se esses pontos. Dessa forma, trabalha-se apenas com um subconjunto de FPs e FAPs, no caso, 64 pontos e 46 parâmetros de baixo nível. Foram descartados os pontos referentes à região interna da boca, orelhas, região atrás da cabeça e pupila, trabalhando-se apenas com a região central da face. O subconjunto de FPs usado está exibido na Figura 5.2.

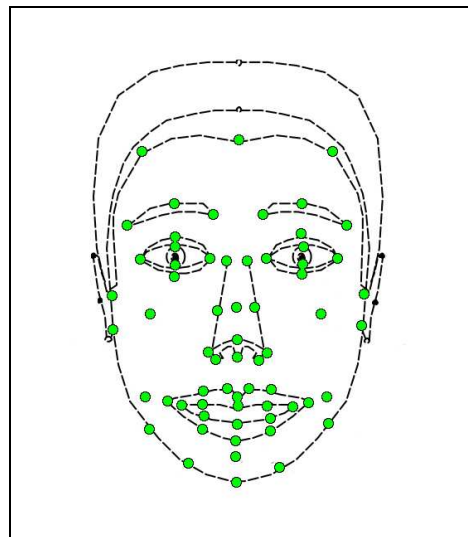


Figura 5.2: Subconjunto dos 64 FPs utilizados no projeto, que consideram apenas a região central da face.

O Apêndice A deve ser consultado para mais detalhes sobre o padrão e o funcionamento dos parâmetros. Nesse apêndice, também é descrita a normalização aplicada para animar faces de diferentes dimensões.

5.1.2 Modelo

Assim como no método descrito no Capítulo 4, os dados foram submetidos ao mesmo processo de subdivisão e projeção. Entretanto, para esta estratégia de animação, foi necessário utilizar um modelo diferente daquele previamente mostrado na Figura 3.4, utilizado para os testes referentes ao casamento automático e à animação por interpolação linear. Utilizou-se um modelo inicial contendo os 64 FPs do MPEG-4 e, além desses, foi necessário adicionar mais 8 pontos em cada olho (a razão será explicada mais adiante). Assim, utilizou-se um modelo de 80 pontos e 140 triângulos, que foi aplicado manualmente em cada malha a ser processada. Veja na Figura 5.3 o modelo utilizado e na tabela 5.1 a estratégia de subdivisões considerada nesta etapa.

Tabela 5.1: Estratégia de subdivisão e projeção usando o modelo inicial para a animação usando modelo paramétrico.

Descrição	Vértices	Triângulos
Modelo inicial (malha base)	80	140
Subdivisão linear e projeção	300	560
Subdivisão linear e projeção	1160	2240
Subdivisão de Loop e projeção	4560	8960

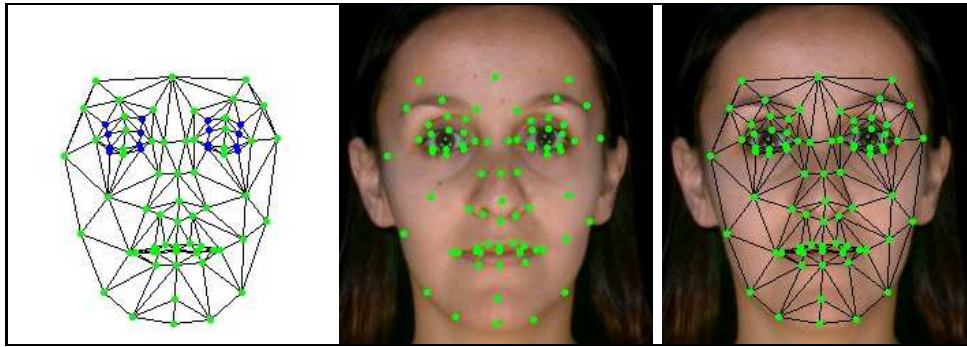


Figura 5.3: Modelo usado para a etapa de subdivisão e projeção. Na imagem à esquerda, os vértices em verde são os mesmos pontos do subconjunto mostrado na Figura 5.2 e os vértices em azul são os demais pontos adicionados, totalizando um modelo de 80 vértices e 140 triângulos.

5.1.3 Regiões de vizinhança

Como foi mencionado na seção 5.1.1, o padrão MPEG-4 define um conjunto de pontos que são deslocados de acordo com alguns parâmetros. Entretanto, os pontos característicos do MPEG-4 por si só não oferecem qualidade suficiente para a animação, isto é, em uma malha complexa, alterar um único ponto não fornece um bom efeito para a animação. Desta forma, é necessário definir, também, quais vértices ao redor do FP serão modificados. Além disso, aplicar o mesmo valor de deslocamento em todos vértices vizinhos não causaria um bom resultado e, por isso, também é necessário especificar qual parcela do deslocamento total será aplicada em cada um dos vértices vizinhos.

O padrão MPEG-4 não define como são calculadas as regiões de vizinhança e nem a função de deformação aplicada nos vértices da região. Os detalhes de implementação de como será realizada a deformação facial da malha dado um parâmetro varia conforme o modelo a ser animado, sendo deixada para o desenvolvedor do sistema de animação. O MPEG-4 propõe o uso de uma tabela denominada FAT (*Facial Animation Table*) para determinar quais vértices são afetados por um parâmetro particular e como estes são modificados.

Em alguns trabalhos, as regiões de influência são definidas manualmente [Balci *et al.* (2007); Cosker *et al.* (2010); Garchery e Magnenat-Thalmann (2001); Kim *et al.* (2010); Queiroz *et al.* (2010)], podendo também ser estabelecidas através de abordagens que estimam um casamento entre a malha a ser animada e um modelo que contém o mapeamento destas zonas [Fratarcangeli e Schaerf (2005); Tang e Huang (2008); cai Yin *et al.* (2005)]. Neste projeto, as regiões de influência foram calculadas usando a triangulação inicial do modelo da Figura 5.3, utilizado no processo de subdivisão.

Basicamente, para cada um dos pontos que são modificados por algum parâmetro, são indicados, no modelo inicial, os triângulos que irão delimitar a região de influência correspondente. Como cada triângulo é subdividido em quatro novos triângulos, ao final dos procedimentos de subdivisão realizados, cada triângulo inicial irá gerar um conjunto

de novos triângulos. Entretanto, todos esses novos vértices gerados são conhecidos no processo de subdivisão, ou seja, para cada triângulo da malha inicial, é fácil localizar todos os pontos que estão dentro dele. Assim, indicando poucos triângulos na malha inicial, o algoritmo identifica todos os vértices que compõem a região de vizinhança delimitados pelos triângulos. O procedimento é ilustrado na Figura 5.4.

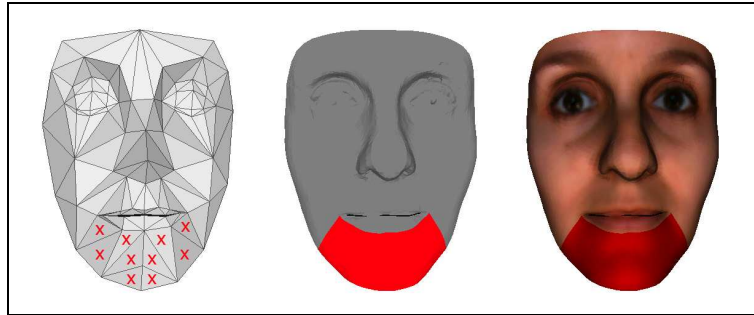


Figura 5.4: Procedimento para o cálculo das regiões de influência do FP 2.1 (centro do maxilar). Um conjunto de triângulos é indicado na malha inicial e esses triângulos são subdivididos, gerando os vértices que irão compor a região.

A ideia apresentava bons resultados tomando como malha inicial o subconjunto de FPs mostrado na Figura 5.2 que é considerado neste projeto, exceto para a região dos olhos. A razão disto é que os pontos definidos nos olhos e nas pálpebras eram insuficientes para o cálculo, levando em conta a curvatura dos olhos, e portanto essa região não era satisfatoriamente delimitada. Dessa forma, o processo de subdivisão foi feito considerando o modelo da Figura 5.3, no qual foram adicionados mais pontos ao redor dos olhos e das pálpebras. A melhoria obtida na adição desses pontos está mostrada na Figura 5.5. Para todos os demais FPs, o algoritmo funcionou bem.

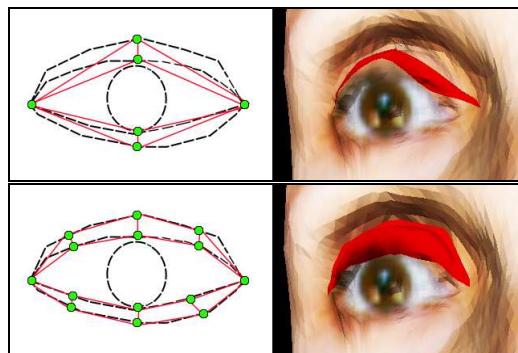


Figura 5.5: Comparação do cálculo da região de influência do FP 3.2 (olho direito) considerando o modelo contendo apenas os FPs do MPEG-4 (topo), e considerando mais pontos ao redor do olho (abaixo).

Após especificar as regiões de vizinhança dos FPs, foi necessário estabelecer uma função de deformação para transmitir de forma adequada o deslocamento do FP para os

seus vizinhos. Neste quesito, era desejado que os vértices que estivessem mais próximos ao FP sofressem uma parcela grande do valor de deslocamento, enquanto os vértices mais distantes sofressem um valor pequeno, aproximando-se de zero. Naturalmente, todos os outros pontos que não fizessem parte da região de vizinhança não poderiam ser alterados.

Neste trabalho, isso foi feito da seguinte forma: primeiro, para cada região de vizinhança, calculou-se a distância de cada vizinho para o FP em questão. Isto foi feito segundo uma estratégia de grafos, usando busca em largura a partir do FP, para definir as distâncias dos demais pontos. Assim, para cada um dos x vértices grafo G_i , ($i = 1, \dots, 64$) relativo à uma região de influência, foram calculados os valores de $d[x]$, que especificam a distância de x até o FP. Os vértices adjacentes ao FP possuem $d = 1$, os adjacentes à estes possuem $d = 2$, e assim sucessivamente. Evidentemente, $d_{FP} = 0$. Neste processo é calculado também o valor de d_{max} que denota a máxima distância computada no processo, isto é:

$$d_{max} = \max\{d[x]\}, x \in G_i$$

Uma vez tendo sido calculados $d[x]$ e d_{max} , esses valores foram usados na função $f_D(x)$ que especifica o deslocamento de cada ponto x quando sujeito, em um dado parâmetro, à uma transformação de valor V_{FAP} :

$$f_D(x) = \left(e^{-\frac{d[x]^2}{2d_{max} \times w}} \right) \times V_{FAP}$$

onde w representa um fator de peso. Note que, no FP, $f_D(x)$ vale zero, e portanto o parâmetro é aplicado com influência total.

A função $f_D(x)$ representa um caso particular da função Gaussiana onde a curva possui altura 1, está centrada na origem e possui largura d_{max} .

O fator w determina o quão acentuada é a queda do valor de deslocamento à medida que se aproxima da borda da região de influência. Alguns resultados para a aplicação da função $f_D(x)$ são exibidos na Figura 5.6.

Uma vez tendo sido calculados os pontos para as regiões de vizinhança e a função de deformação para esses pontos, esses valores foram adicionados à FAT. O sistema, então, interpola e combina os valores da FAT, que são considerados como poses-chaves no processo. Note que, desta forma, combina-se as estratégias de modelo paramétrico e interpolação descritas na Seção 1.2.1. De fato, é comum combinar as diferentes estratégias de animação. Por exemplo, uma opção que não foi utilizada neste projeto, mas que também poderia ser viável, seria preencher os campos da FAT atrelando os vértices vizinhos de um FP à uma estrutura de pseudomúsculo.

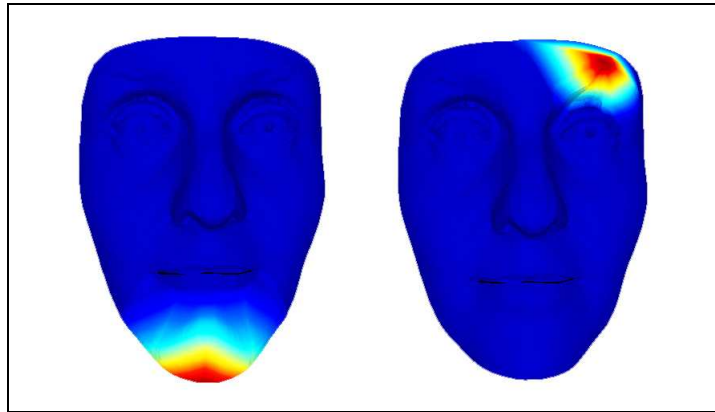


Figura 5.6: Mapa de cores mostrando a intensidade de deslocamentos em pontos faciais referentes ao centro da sobrancelha (esquerda) e centro do maxilar (direita). Note que o valor dos deslocamentos é alto nas proximidades dos FPs modificados e diminui conforme afasta-se desses pontos.

5.2 Expressões faciais e visemas no MPEG-4

Na seção anterior, foi descrito como os parâmetros de baixo nível do MPEG-4 foram utilizados no projeto, permitindo que a face exibisse pequenos movimentos. Esses parâmetros, então, foram usados em conjunto para gerar as expressões faciais e os visemas, que compõem os FAPs de alto nível FAP 1 e FAP 2, respectivamente.

O nome de cada parâmetro de baixo nível descreve o movimento que deve ser feito (abaixar, levantar, esticar, etc) e a parte da face sobre a qual atua (boca, sobrancelha, olhos, etc). Os parâmetros usados foram escolhidos conforme as características de cada expressão facial. Por exemplo, para gerar uma expressão de alegria, a face deve realizar os seguintes movimentos:

1. Levantar os cantos direito e esquerdo, interno e externo da boca

FAPs:

- *raise_l_cornerlip*, *raise_l_cornerlip_o*
- *raise_r_cornerlip*, *raise_r_cornerlip_o*

2. Esticar os cantos direito e esquerdo, interno e externo da boca

FAPs:

- *stretch_l_cornerlip*, *stretch_l_cornerlip_o*
- *stretch_r_cornerlip*, *stretch_r_cornerlip_o*

3. Abaixar o maxilar

FAP: *open_jaw*

4. Abrir a boca

FAPs (aplicados no sentido inverso):

- *raise_b_midlip*, *raise_b_midlip_o*
- *raise_b_lip_lm*, *raise_b_lip_lm_o*
- *raise_b_lip_rm*, *raise_b_lip_rm_o*

5. Levantar as bochechas

FAPs:

- *lift_L_cheek*
- *lift_r_cheek*

6. Levantar o meio das sobrancelhas direita e esquerda

FAPs:

- *raise_L_m_eyebrow*
- *raise_r_m_eyebrow*

O Apêndice B descreve mais detalhes sobre a aplicação destes parâmetros (valor, direção, etc) e como as demais expressões faciais foram construídas a partir dos parâmetros de baixo nível do MPEG-4.

5.3 Resultados

O sistema foi testado com dados reais e, para avaliar a animação conforme o modelo do MPEG-4, construiu-se uma interface com um menu contendo *scrollbars* para os 46 FAPs de baixo nível utilizados, de modo a manipular uma face combinando os parâmetros na direção positiva e negativa, como mostrado na Figura 5.7. Os modelos obtidos, apresentando as expressões faciais, são mostrados na Figura 5.8 e os visemas obtidos para as vogais são mostrados na Figura 5.9.

Finalmente, a Figura 5.10 permite fazer uma comparação entre as seis expressões geradas com o modelo de face paramétrica do MPEG-4 com as mesmas expressões digitalizadas de uma pessoa real, utilizadas no método de interpolação descrito no capítulo anterior. Nota-se que as expressões do modelo paramétrico são ainda um pouco artificiais se comparadas às expressões reais digitalizadas, nas quais rugas, por exemplo, são mais evidentes.

A página www.vision.ime.usp.br/~dedea/videos também contém vídeos com os resultados para a estratégia de animação utilizando o modelo paramétrico.

Nota-se que as expressões do modelo paramétrico são ainda um pouco artificiais se comparadas às expressões reais digitalizadas, nas quais rugas, por exemplo, são mais evidentes. Por outro lado, ao analisar a expressão de nojo, por exemplo, percebe-se

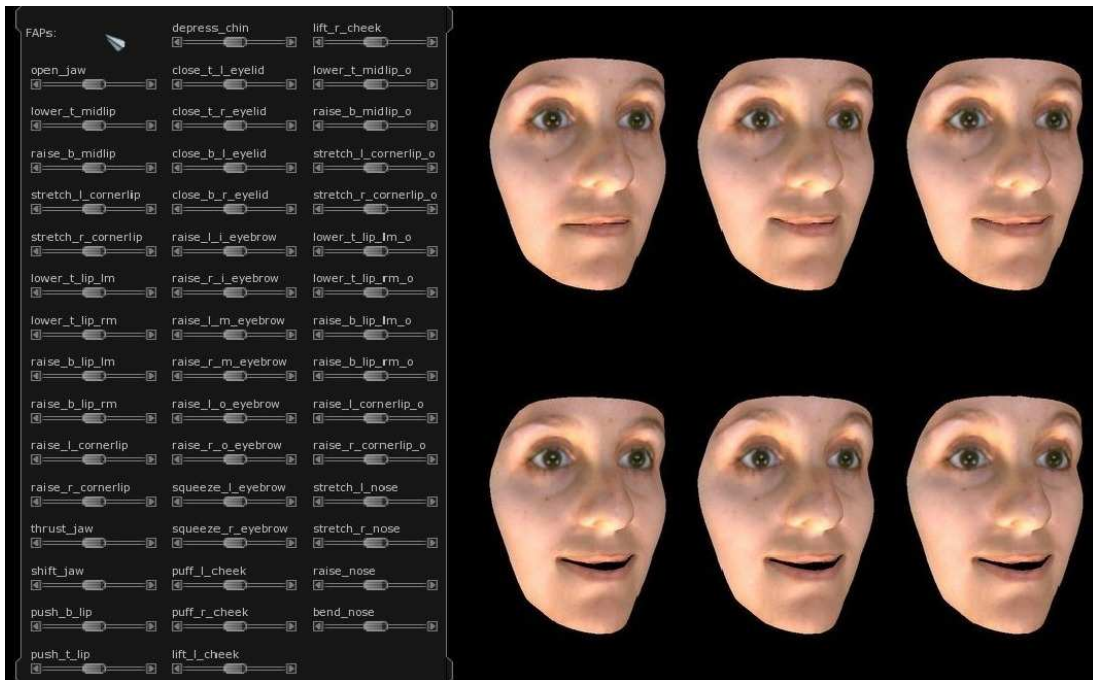


Figura 5.7: Interface criada para manipular os FAPs de baixo nível (menu à esquerda) e o efeito de aplicar a sequência de parâmetros usados para gerar a expressão de alegria a partir da face neutra: levantar os cantos da boca, esticar os cantos da boca, abrir a boca e o maxilar, levantar as sobrancelhas, levantar as bochechas.

que o modelo paramétrico é capaz de gerar uma deformação mais consistente na região dos olhos. Isso ocorre pois, enquanto a animação paramétrica manipula diretamente a região das pálpebras, deixando os olhos inalterados, o modelo de interpolação deforma toda a região correspondente¹. De fato, a diferença fundamental entre as abordagens é baseada no fato de que o modelo de interpolação gera uma deformação global na malha, enquanto o modelo paramétrico gera pequenas deformações locais.

¹Por representar uma região complexa para a animação, certos estudos desconsideram a região dos olhos, substituindo-as por um modelo artificial de globo ocular.

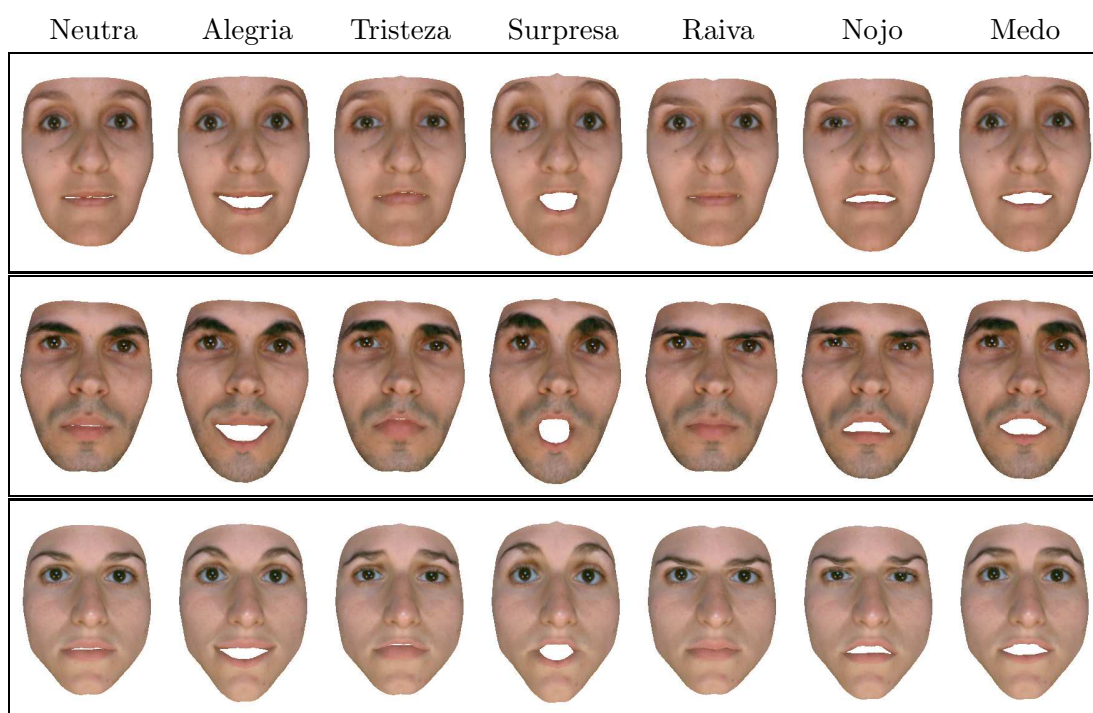


Figura 5.8: Resultados para três pessoas diferentes do banco de dados. A primeira coluna exibe a face neutra a partir da qual as expressões foram geradas. Nas demais colunas estão as expressões resultantes.

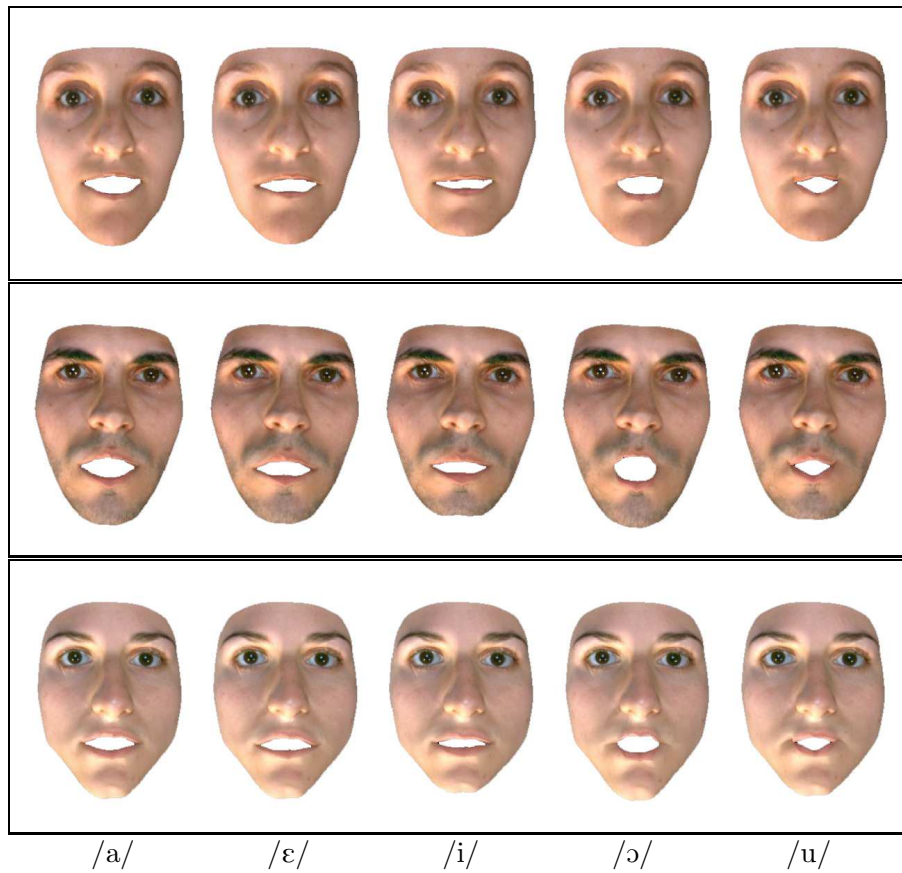


Figura 5.9: *Visemas para as vogais, gerados a partir da face neutra*

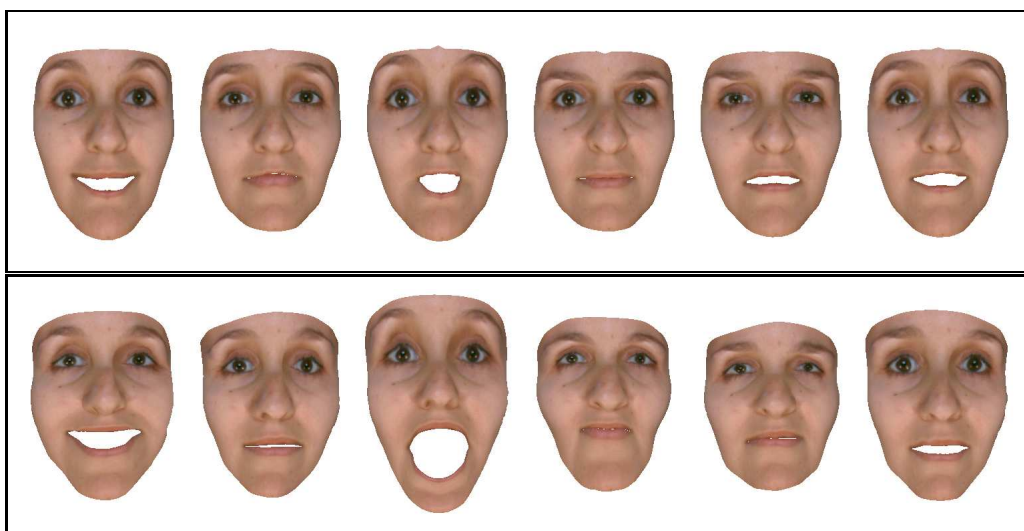


Figura 5.10: *Comparativo entre as expressões geradas no trabalho (topo) e as expressões reais digitalizadas (abaixo).*

Capítulo 6

Considerações finais

6.1 Contribuições

Além das atividades mencionadas, foram publicados dois trabalhos relatando os estudos na área. O trabalho [Mattos e Cesar-Jr. \(2009\)](#) foi apresentado como poster no SIBGRAPI 2009 (Brazilian Symposium on Computer Graphics and Image Processing) descrevendo a ideia geral da proposta de utilizar o registro estrutural para a animação de faces tridimensionais. Também foi publicado o artigo [Mattos *et al.* \(2010\)](#) no SIBGRAPI 2010, relatando o processo de animação descrito na seção 4.1. Os documentos podem ser encontrados em www.vision.ime.usp.br/~dedea/research.

Uma das contribuições deste estudo está relacionada à nova proposta para realizar o casamento automático de marcos faciais, fornecendo-se como entrada apenas uma face marcada. O estudo mostrou que bons resultados podem ser obtidos e, ainda nos piores casos, o procedimento pode ser completado demandando pouco esforço manual do usuário.

Através da definição desses poucos marcos faciais relacionados, nós mostramos uma estratégia para o cálculo da correspondência entre um número elevado de pontos da face, completando o processo de registro. Esse passo é baseado em um processo de subdivisão, que evita os problemas existentes na busca de correspondências densas entre malhas faciais.

A partir do procedimento de registro, este trabalho descreve um sistema de animação facial 3D usando faces reais. A animação é realizada por meio de interpolações lineares, que exigem o cálculo de correspondências ponto-a-ponto entre as faces. Matrizes de deslocamento são calculadas, sendo posteriormente usadas para executar animações faciais a partir de outros tipos de modelos de face.

Como alternativa, levando em conta as mesmas ferramentas utilizadas, descreveu-se também a implementação de um sistema de animação facial usando um subconjunto de pontos característicos e parâmetros do padrão MPEG-4. O trabalho introduziu uma maneira de calcular as regiões de influência para os pontos característicos, baseando-se

no mesmo processo de subdivisão do modelo inicial, fornecendo, assim, uma alternativa de como estabelecer a deformação da malha em cada parâmetro.

6.2 Trabalhos futuros

Uma vez exibidos os resultados alcançados durante o mestrado, foram identificadas algumas melhorias que poderiam ser acrescentadas, além de problemas que poderiam ser resolvidos utilizando as abordagens já implementadas. Abaixo, são listadas algumas possíveis continuidades que poderiam ser consideradas para o desenvolvimento do projeto.

1. Combinar os métodos de animação por interpolação e modelo paramétrico.
Buscando ressaltar os pontos fortes de cada estratégia, uma possibilidade seria aplicar os parâmetros do MPEG-4 baseando-se nas expressões capturadas. Isto é, poderia calcular-se a distância entre pontos chaves nas expressões capturadas e deformar a face conforme esse valor.
2. Unificar o modelo inicial de subdivisão para as duas abordagens.
A ideia proposta no item anterior requer que as faces sejam subdivididas utilizando o mesmo modelo inicial. Assim, seria necessário unificar o modelo paramétrico para as duas abordagens. Para executar de forma eficiente a remarcação dos pontos para a interpolação, seria interessante avaliar a estratégia do casamento automático utilizando o modelo baseado no padrão MPEG-4.
3. Transferir detalhes faciais entre malhas, ampliando o realismo na animação.
As abordagens implementadas permitem propor uma ideia para que detalhes capturados pelo escâner na aquisição das expressões faciais, como rugas, possam ser transferidos para o modelo de animação paramétrico. Utilizando o mesmo modelo para subdivisão, é possível definir uma correspondência de um-para-um entre a malha neutra a ser animada e as malhas das expressões digitalizadas. Assim, pode-se delimitar uma região na qual existe a presença de rugas (testa, por exemplo), utilizando as zonas de influência descritas em 5.1.3. Da mesma maneira como as expressões podem ser subtraídas da face neutra e transferidas para outros modelos, pode-se subtrair uma região com rugas de uma região suavizada, e transferir os vetores que modificam a região da mesma forma.
4. Utilizar aquisições de movimentos faciais para realizar animações não lineares.
As expressões faciais (capturadas ou geradas parametricamente) são interpoladas de forma simples, linearmente. O IMPA-RJ, onde existem colaborações entre o grupo de pesquisa do IME-USP e o Professor Luiz Velho, possui um equipamento para captura de movimentos faciais. Os valores provenientes dessa aquisição po-

deriam ser utilizados para animar as faces tridimensionais mantendo uma maior coerência com os movimentos humanos, isto é, de forma não linear.

5. Utilizar modelos de outros bancos de dados disponíveis.

Embora os resultados mostrados para animação e registro utilizem o banco de dados construído por [Mena-Chalco et al. \(2008a\)](#), existem também outras alternativas de bancos de faces 3D disponíveis. Em particular [Savran et al. \(2008\)](#) e [Yin et al. \(2006\)](#) apresentam uma boa variedade de faces que poderiam ser usadas para validar os resultados obtidos durante a pesquisa, utilizando outros tipos de dados.

6. Realizar investigações sobre o modo de ponderar estrutura e aparência no casamento estrutural.

No casamento estrutural para faces 3D, seria interessante fazer uma análise mais fina sobre a ponderação entre aparência e estrutura, investigando, inclusive, pesos variados dependendo dos pontos a serem casados, ou outros fatores que poderiam ser levados em conta para o casamento. Os testes realizados mostram que a região da boca apresenta-se como uma região problemática, uma vez que, entre certas expressões, existe uma grande variação dos parâmetros considerados (curvatura, geometria, etc.). Assim, para obter melhores casamentos nessa região, seria ideal considerar um modelo diferenciado para a boca.

7. Avaliar variações que permitam resolver o problema do casamento estrutural.

Assim como o método de [Noma et al. \(2008\)](#), outra abordagem bem-sucedida de casamento estrutural em 2D poderia ser considerada. Na ideia estrutural proposta por [Hashimoto e Cesar \(2009\)](#), no lugar de basear-se na detecção utilizando apenas pontos chaves, essa abordagem busca construir um conjunto de grafos cujos vértices são os pontos detectados. Assim, é possível extrair vetores de atributos dos pontos que levam em conta informações estruturais em cada região.

8. Analisar o casamento estrutural como alternativa para a resolução de outros problemas.

Embora esta pesquisa tenha adotado o casamento estrutural com foco para o registro tridimensional, poderíamos avaliá-lo, também, para imagens 2D, o que permitiria auxiliar na resolução de problemas como reconhecimento de faces ou reconhecimento de expressões faciais para sequências de vídeo, por exemplo. A rotulação de atributos faciais poderia ser utilizada como passo fundamental para geração de bases de dados necessárias para treinamento em diversos algoritmos de reconhecimento facial. Para o caso de sequências de vídeo, em particular, existe grande interesse em reconhecimento de faces para o campo de segurança e vigilância eletrônica.

6.3 Conclusão

A criação de faces animadas realistas permanece sendo um dos maiores desafios em computação gráfica, motivando a existência de diversas abordagens, com diferentes níveis de complexidade, para produzir uma animação facial. Paralelamente, o avanço dos escâneres 3D tornou possível capturar com precisão uma boa geometria facial para a animação. Desta forma, este estudo pretende contribuir para a utilização desses modelos, propondo uma técnica nova de registro entre as diversas poses capturadas, que permite realizar a animação desses modelos de diferentes formas.

Bons resultados foram obtidos e uma série de continuidades para o projeto foram listadas, de modo que espera-se que esta pesquisa possa auxiliar para a resolução de outros problemas. Além de contribuir para a área de animação facial, a definição automática de pontos faciais poderia ser utilizada em uma série de projetos que realizam essa etapa de forma manual, para, por exemplo, buscar correspondências densas entre faces, normalizar modelos de faces, etc. Assim, o estudo realizado é capaz de contribuir também com outros trabalhos recentes que envolvem a manipulação de faces 3D, em áreas como reconhecimento de faces, reconhecimento de expressões faciais, transferência de expressões faciais, transferência de detalhes faciais e reconstrução facial.

Apêndice A

O Padrão MPEG-4

O padrão MPEG-4 define um conjunto de pontos de controle que forma um modelo facial, sendo a primeira proposta para uma malha facial poligonal considerada universal. Conforme mencionado na seção 5.1.1, o padrão define 84 pontos característicos (FPs) e 68 parâmetros de animação (FAPs). Cada FAP atua em um único FP, embora mais de um FAP possa atuar em um mesmo FP. É importante ressaltar que nem todo FP é alterado por um FAP. A Figura A.1 mostra os 84 FPs definidos no MPEG-4, destacando aqueles que são alterados na animação. Cada FAP representa um deslocamento no FP no qual atua, e este deslocamento deve ocorrer em um único eixo (x , y ou z). O FAP de baixo nível *raise_lm_eyebrow*, tomado como exemplo na seção 5.1.1, atua sobre o FP 4.3 movendo-o na direção y para cima.

Existem FAPs unidirecionais, que atuam em apenas uma direção, e bidirecionais, que atuam também no sentido inverso. Por exemplo, o FAP *stretch_lm_cornerlip* estica o canto esquerdo da boca, movendo-o na direção x , e pode ser aplicado nas duas direções uma vez que a face é capaz de esticar e encolher os cantos da boca. Já o FAP *open_jaw*, que desloca o maxilar para baixo movendo-o na direção y , é um FAP unidirecional, uma vez que a face humana não é capaz de, a partir da face neutra, levantar o maxilar na direção vertical.

Em cada parâmetro de animação, os FPs sofrem um deslocamento de um certo valor. Entretanto, estes valores devem ser normalizados para que possa se obter uma padronização dos movimentos de acordo com cada face a ser animada. Para normalizar os parâmetros, o padrão define unidades denominadas de FAPUs (*Face Animation Parameter Units*) que são computadas a partir de distâncias espaciais entre pontos chave da face. A Figura A.2 e a Tabela A.1 exibem como este cálculo é feito, levando em conta os FPs correspondentes às distâncias mencionadas. Todos os FPs, FAPs e FAPUs são definidos na face no seu estado de neutralidade.

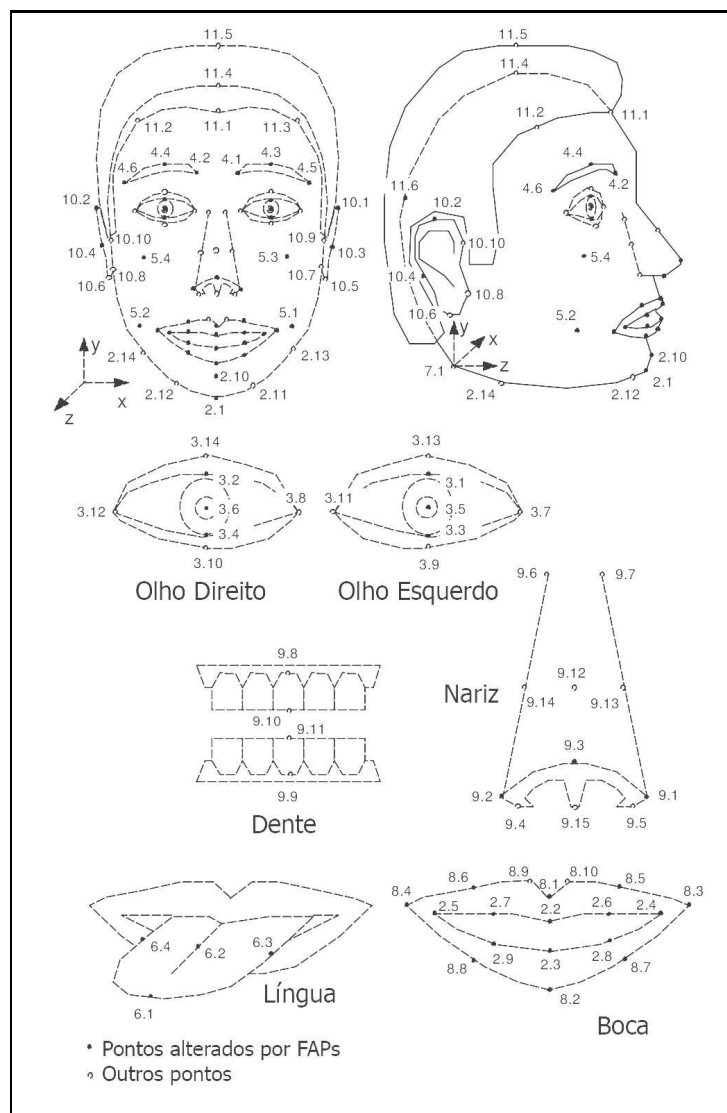


Figura A.1: Os 84 pontos característicos faciais (FPs) descritos no padrão MPEG-4.

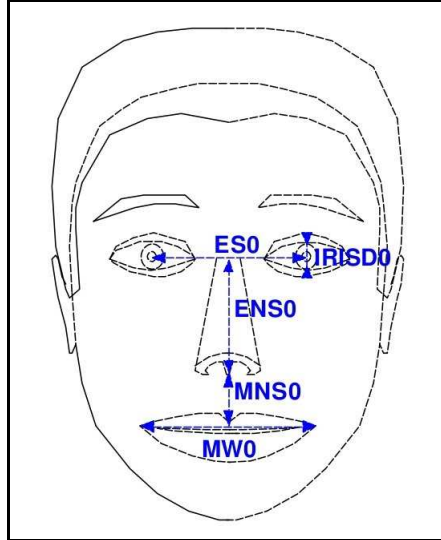


Figura A.2: Distâncias utilizadas para computar as FAPUs, unidades usadas para normalizar o valor dos deslocamentos de cada ponto nos movimentos faciais.

Tabela A.1: Cálculo das FAPUs a partir de um conjunto de FPs chave.

IRISD0	Diâmetro da íris	$IRISD = \frac{IRISD0}{1024}$
ESO	Separação entre os olhos	$ES = \frac{ESO}{1024}$
ENSO	Separação entre nariz e olho	$ENS = \frac{ENSO}{1024}$
MNSO	Separação entre nariz e boca	$MNS = \frac{MNSO}{1024}$
MW0	Largura da boca	$MW = \frac{MW0}{1024}$

Apêndice B

Construção das Expressões Faciais no MPEG-4

O padrão MPEG-4 define, para cada FAP, o FP sobre o qual ele atua, a unidade de normalização, a direção do movimento positivo e o valor mínimo e máximo de deslocamento. Dependendo da aplicação e da face a ser animada, utiliza-se algum valor conveniente entre este limite. Neste projeto, as seis expressões faciais de Ekman foram construídas a partir dos parâmetros de baixo nível conforme é apresentado na Tabela B.1. Nem todos os 46 FAPs considerados no projeto estão na tabela, pois são mostrados apenas os parâmetros que foram usados para gerar as seis expressões faciais. Para o caso dos visemas, embora não esteja apresentada no texto, montou-se uma tabela análoga, utilizando apenas FAPs que modificam a região da boca.

Tabela B.1: Valores para os FAPs de baixo nível utilizados para construir as expressões faciais no sistema de animação. Os campos com (*) denotam que o FAP não é utilizado na expressão e valores negativos denotam que o movimento deve ser feito na direção inversa.

FAP	Unidade	Direção	Alegria	Tristeza	Surpresa	Raiva	Nojo	Medo
open_jaw	MNS	<i>y</i>	430	*	490	*	285	290
lower_t_midlip	MNS	<i>y</i>	*	220	-200	200	100	-220
lower_t_lip_lm	MNS	<i>y</i>	*	100	-100	*	60	-100
lower_t_lip_rm	MNS	<i>y</i>	*	100	-100	*	60	-100
lower_t_lip_lm_o	MNS	<i>y</i>	*	*	-100	*	100	-100
lower_t_lip_rm_o	MNS	<i>y</i>	*	*	-100	*	100	-100
raise_b_midlip	MNS	<i>y</i>	-500	-220	-600	80	-170	-270
raise_b_lip_lm	MNS	<i>y</i>	-180	-100	-120	50	-120	-150
raise_b_lip_rm	MNS	<i>y</i>	-180	-100	-120	50	-120	-150
raise_b_midlip_o	MNS	<i>y</i>	-100	*	-50	*	*	*
raise_b_lip_lm_o	MNS	<i>y</i>	-200	*	-200	*	-100	*
raise_b_lip_rm_o	MNS	<i>y</i>	-200	*	-200	*	-100	*
raise_l_cornerlip	MNS	<i>y</i>	70	*	*	-200	*	*
raise_r_cornerlip	MNS	<i>y</i>	70	*	*	-200	*	*
raise_l_cornerlip_o	MNS	<i>y</i>	230	*	*	*	-130	*
raise_r_cornerlip_o	MNS	<i>z</i>	230	*	*	*	-130	*
push_b_lip	MNS	<i>z</i>	*	*	*	200	*	*
depress_chin	MNS	<i>x</i>	*	*	*	250	*	*
stretch_l_cornerlip	MW	<i>x</i>	90	*	-280	*	*	*
stretch_r_cornerlip	MW	<i>x</i>	90	*	-280	*	*	*
stretch_l_cornerlip_o	MW	<i>x</i>	70	*	110	*	*	*
stretch_r_cornerlip_o	MW	<i>x</i>	70	*	110	*	*	*
close_t_l_eyelid	IRISD	<i>y</i>	-50	150	-100	*	250	-100
close_t_r_eyelid	IRISD	<i>y</i>	-50	150	-100	*	250	-100
close_b_l_eyelid	IRISD	<i>y</i>	-50	150	-100	*	250	-150
close_b_r_eyelid	IRISD	<i>y</i>	-50	150	-100	*	250	-150
lift_l_cheek	ENS	<i>y</i>	100	*	*	*	*	*
lift_r_cheek	ENS	<i>y</i>	100	*	*	*	*	*
raise_nose	ENS	<i>y</i>	*	*	*	50	30	*
raise_l_m_eyebrow	ENS	<i>y</i>	90	*	160	-110	-100	70
raise_r_m_eyebrow	ENS	<i>y</i>	90	*	160	-110	-100	70
raise_l_i_eyebrow	ENS	<i>y</i>	*	120	120	-90	*	100
raise_r_i_eyebrow	ENS	<i>y</i>	*	120	120	-90	*	100
raise_l_o_eyebrow	ENS	<i>y</i>	*	-100	60	-70	*	-70
raise_r_o_eyebrow	ENS	<i>y</i>	*	-100	60	-70	*	-70
squeeze_l_eyebrow	ES	<i>x</i>	*	*	*	80	*	60
squeeze_r_eyebrow	ES	<i>x</i>	*	*	*	80	*	60

Referências Bibliográficas

- Amberg et al.(2008)** Brian Amberg, Reinhard Knothe e Thomas Vetter. Expression invariant 3D face recognition with a morphable model. Em *FG*, páginas 1–6. Citado na pág. [5](#)
- Amberg et al.(2009)** Brian Amberg, Pascal Paysan e Thomas Vetter. Weight, sex, and facial expressions: On the manipulation of attributes in generative 3D face models. Em *ISVC09*, páginas I: 875–885. Citado na pág. [2](#)
- Aubrey et al.(2010)** Andrew Aubrey, David Marshall e Paul Rosin. Behaviour transfer between expressive talking heads. Em *Proceedings of the ACM/SSPNET 2nd International Symposium on Facial Analysis and Animation, FAA '10*, páginas 9–9, New York, NY, USA. ACM. Citado na pág. [3](#)
- Balci et al.(2007)** Koray Balci, Elena Not, Massimo Zancanaro e Fabio Pianesi. Xface open source project and SMIL-agent scripting language for creating and animating embodied conversational agents. Em *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*. ACM. Citado na pág. [2](#), [3](#), [62](#)
- Baltrušaitis e Robinson(2010)** Tadas Baltrušaitis e Peter Robinson. Analysis of colour space transforms for person independent AAMs. Em *Proceedings of the ACM/SSPNET 2nd International Symposium on Facial Analysis and Animation, FAA '10*, páginas 21–21, New York, NY, USA. ACM. Citado na pág. [7](#)
- Basso et al.(2006)** Curzio Basso, Pascal Paysan e Thomas Vetter. Registration of expressions data using a 3D morphable model. Em *FGR '06: Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*. IEEE Computer Society. Citado na pág. [4](#)
- Beeler et al.(2010)** Thabo Beeler, Bernd Bickel, Paul Beardsley, Bob Sumner e Markus Gross. High-quality single-shot capture of facial geometry. *ACM Transactions on Graphics*, 29(3). Citado na pág. [2](#)
- Besl e McKay(1992)** Paul J. Besl e Neil D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–258. Citado na pág. [19](#)
- Bickel et al.(2007)** Bernd Bickel, Mario Botsch, Roland Angst, Wojciech Matusik, Miguel Otaduy, Hanspeter Pfister e Markus Gross. Multi-scale capture of facial geometry and motion. Em *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*. ACM. Citado na pág. [3](#)

- Bickel et al.(2008)** Bernd Bickel, Manuel Lang, Mario Botsch, Miguel A. Otaduy e Markus Gross. Pose-space animation and transfer of facial details. Em *Proc. of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, páginas 57–66. Citado na pág. [2](#), [5](#), [7](#)
- Bickel et al.(2009)** Bernd Bickel, Moritz Bächer, Miguel A. Otaduy, Wojciech Matusik, Hanspeter Pfister e Markus Gross. Capture and modeling of non-linear heterogeneous soft tissue. *ACM Transactions on Graphics*, 28:89:1–89:9. Citado na pág. [2](#)
- Blanz e Vetter(1999)** Volker Blanz e Thomas Vetter. A morphable model for the synthesis of 3D faces. Em *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co. Citado na pág. [1](#), [2](#), [4](#), [51](#)
- Blanz et al.(2003)** Volker Blanz, Curzio Basso, Thomas Vetter e Tomaso Poggio. Reanimating faces in images and video. Em *Proceedings of EUROGRAPHICS*. Citado na pág. [5](#)
- Blanz et al.(2007)** Volker Blanz, Kristina Scherbaum e Hans-Peter. Seidel. Fitting a morphable model to 3D scans of faces. Em *IEEE International Conference on Computer Vision*, volume 2007. IEEE Computer Society. Citado na pág. [4](#)
- Bradley et al.(2010)** Derek Bradley, Wolfgang Heidrich, Tiberiu Popa e Alla Sheffer. High resolution passive facial performance capture. Em *ACM SIGGRAPH 2010 papers*, SIGGRAPH '10, páginas 41:1–41:10, New York, NY, USA. ACM. Citado na pág. [2](#), [3](#)
- cai Yin et al.(2005)** Bao cai Yin, Cheng zhang Wang, Qin Shi e Yan feng Sun. MPEG-4 compatible 3D facial animation based on morphable model. Em *Proceedings of 2005 International Conference on Machine Learning and Cybernetics*. IEEE Computer Society. Citado na pág. [2](#), [3](#), [62](#)
- Çeliktutan et al.(2008)** Oya Çeliktutan, Hatice Çinar Akakin e Bülent Sankur. Multi-attribute robust facial feature localization. Em *FG*, páginas 1–6. IEEE. Citado na pág. [7](#)
- Chai et al.(2003)** Jin-xiang Chai, Jing Xiao e Jessica Hodgins. Vision-based control of 3D facial animation. Em *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*. Eurographics Association. Citado na pág. [3](#)
- Choe e Ko(2005)** Byoungwon Choe e Hyeong-Seok Ko. Analysis and synthesis of facial expressions with hand-generated muscle actuation basis. Em *SIGGRAPH '05: ACM SIGGRAPH 2005 Courses*. ACM. Citado na pág. [3](#)
- Colombo et al.(2006)** Alessandro Colombo, Claudio Cusano e Raimondo Schettini. 3D face detection using curvature analysis. *Pattern Recogn.*, 39:444–455. ISSN 0031-3203. Citado na pág. [23](#), [29](#)
- Cosker et al.(2010)** Darren Cosker, Eva Krumerhuber e Adrian Hilton. A FACS validated 3D human facial model. Em *Proceedings of the ACM/SSPNET 2nd International*

- Symposium on Facial Analysis and Animation*, FAA '10, páginas 12–12, New York, NY, USA. ACM. Citado na pág. 62
- da Fontoura Costa e Cesar Jr.(2009)** Luciano da Fontoura Costa e Roberto Marcondes Cesar Jr. *Shape Classification and Analysis: Theory and Practice*. CRC Press, Inc., Boca Raton, FL, USA. Citado na pág. 17
- Dibeklioglu et al.(2008)** Hamdi Dibeklioglu, Albert Ali Salah e Lale Akarun. 3D facial landmarking under expression, pose, and occlusion variations. páginas 1–6. Citado na pág. 7
- Dutreve et al.(2008)** Ludovic Dutreuve, Alexandre Meyer e Saïda Bouakaz. Feature points based facial animation retargeting. Em *VRST '08: Proceedings of the 2008 ACM symposium on Virtual reality software and technology*. ACM. Citado na pág. 3
- Ekman e Friesen(1978)** P. Ekman e W. V. Friesen. The facial action coding system: A technique for the measurement of facial movement. Em *Consulting Psychologists*, Palo Alto, California. Citado na pág. 12
- Fratarcangeli e Schaerf(2005)** Marco Fratarcangeli e Marco Schaerf. Fast facial motion cloning in MPEG-4. Em *4th International Symposium on Image and Signal Processing and Analysis (ISPA 2005)*. Citado na pág. 62
- Garchery e Magnenat-Thalmann(2001)** S. Garchery e N. Magnenat-Thalmann. Designing MPEG-4 facial animation tables for web applications. Em *Multimedia Modeling 2001*, páginas 39–59. Citado na pág. 62
- Golovinskiy et al.(2006)** Aleksey Golovinskiy, Wojciech Matusik, Hanspeter Pfister, Szymon Rusinkiewicz e Thomas Funkhouser. A statistical model for synthesis of detailed facial geometry. *ACM Transactions on Graphics*, 25(3):1025–1034. Citado na pág. 2, 42
- Guskov et al.(2000)** Igor Guskov, Kiril Vidimče, Wim Sweldens e Peter Schröder. Normal meshes. Em Sheila Hoffmeyer, editor, *Proceedings of the Computer Graphics Conference 2000 (SIGGRAPH-00)*, páginas 95–102, New York. ACM Press. Citado na pág. 42
- Hashimoto e Cesar(2009)** Marcelo Hashimoto e Roberto Cesar. Object detection by keygraph classification. Em Andrea Torsello, Francisco Escolano e Luc Brun, editors, *Graph-Based Representations in Pattern Recognition*, volume 5534 of *Lecture Notes in Computer Science*, páginas 223–232. Springer Berlin / Heidelberg. Citado na pág. 73
- Hu et al.(2009)** Yongli Hu, Mingquan Zhou e Zhongke Wu. A dense point-to-point alignment method for realistic 3D face morphing and animation. *Int. J. Comput. Games Technol.*, 2009:1–9. ISSN 1687-7047. Citado na pág. 6, 7
- Jimenez et al.(2010)** Jorge Jimenez, Timothy Scully, Nuno Barbosa, Craig Donner, Xenxo Alvarez, Teresa Vieira, Paul Matts, Verónica Orvalho, Diego Gutierrez e Tim Weyrich. A practical appearance model for dynamic facial color. *ACM Transactions on Graphics*, 29:141:1–141:10. Citado na pág. 2

- Joshi et al.(2003)** Pushkar Joshi, Wen C. Tien, Mathieu Desbrun e Frédéric Pighin. Learning controls for blend shape based realistic facial animation. Em *SCA '03: Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*. Eurographics Association. Citado na pág. 2
- Ju e Lee(2008)** Eunjung Ju e Jehee Lee. Expressive facial gestures from motion capture data. *Comput. Graph. Forum*, 27(2):381–388. Citado na pág. 3
- Katsamanis et al.(2008)** A. Katsamanis, G. Papandreou e P. Maragos. Audiovisual-to-articulatory speech inversion using active appearance models for the face and hidden markov models for the dynamics. Em *Proc. IEEE Int'l Conference on Acoustics, Speech, and Signal Processing (ICASSP-2008), Las Vegas, Nevada, U.S.A., Mar.-Apr. 2008*. Citado na pág. 7
- Kim e Ko(2007)** Ig-Jae Kim e Hyeong-Seok Ko. Intuitive quasi-eigen faces. Em *GRAPHITE '07: Proceedings of the 5th international conference on Computer graphics and interactive techniques in Australia and Southeast Asia*. ACM. Citado na pág. 2
- Kim et al.(2010)** Jeong-Sik Kim, Young-Ju Cho, Gyoung-Ah Lee e Myoung-Hee Kim. A realistic 3D facial deformation constrained with facial asymmetry of healthy subjects. Em *ACM SIGGRAPH ASIA 2010 Posters, SA '10*, páginas 4:1–4:1, New York, NY, USA. ACM. Citado na pág. 62
- Krumhuber et al.(2010)** Eva Krumhuber, Lucas Tamarit, Klaus R. Scherer e Etienne Roesch. FACSGen 2.0: facial expression animation based on FACS. Em *Proceedings of the ACM/SSPNET 2nd International Symposium on Facial Analysis and Animation, FAA '10*, páginas 7–7, New York, NY, USA. ACM. Citado na pág. 47
- Lau et al.(2007)** Manfred Lau, Jinxiang Chai, Ying-Qing Xu e Heung-Yeung Shum. Face poser: interactive modeling of 3D facial expressions using model priors. Em *Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation, SCA '07*, páginas 161–170, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association. Citado na pág. 3
- Liu et al.(2006)** Junfa Liu, Yiqiang Chen e Wen Gao. Mapping learning in eigenspace for harmonious caricature generation. Em *MULTIMEDIA '06: Proceedings of the 14th annual ACM international conference on Multimedia*, páginas 683–686, New York, NY, USA. ACM. Citado na pág. 7
- Loop(1987)** Charles Loop. Smooth subdivision surfaces based on triangles. Dissertação de Mestrado, University of Utah, Department of Mathematics. Citado na pág. 42
- Lu e Jain(2006)** Xiaoguang Lu e Anil K. Jain. Deformation modeling for robust 3D face matching. Em *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2, CVPR '06*, páginas 1377–1383, Washington, DC, USA. IEEE Computer Society. Citado na pág. 7
- Ma et al.(2009)** Xiaohan Ma, Binh Huy Le e Zhigang Deng. Style learning and transferring for facial animation editing. Em *SCA '09: Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, páginas 123–132, New York, NY, USA. ACM. Citado na pág. 2

- Mäkäräinen e Takala(2009)** Meeri Mäkäräinen e Tapio Takala. An approach for creating and blending synthetic facial expressions of emotion. Em *IVA '09: Proceedings of the 9th International Conference on Intelligent Virtual Agents*, páginas 243–249, Berlin, Heidelberg. Springer-Verlag. Citado na pág. 3
- Mattheyses e Verhelst(2010)** Wesley Mattheyses e Werner Verhelst. Photorealistic 2D audiovisual text-to-speech synthesis using active appearance models. Em *Proceedings of the ACM/SSPNET 2nd International Symposium on Facial Analysis and Animation*, FAA '10, páginas 15–15, New York, NY, USA. ACM. Citado na pág. 7
- Mattos e Cesar-Jr.(2009)** Andréa B. Mattos e Roberto M. Cesar-Jr. 3D facial animation based on structural registration. Em *Workshops of Sibgrapi 2009 - Posters*. Citado na pág. 71
- Mattos et al.(2010)** Andréa B. Mattos, Jesús P. Mena-Chalco, Roberto M. Cesar-Jr. e Luiz Velho. 3D linear facial animation based on real data. Em *Sibgrapi 2010 (XXIII Brazilian Symposium on Computer Graphics and Image Processing)*. Citado na pág. 71
- Mena-Chalco(2010)** Jesús P. Mena-Chalco. *Reconstrução de faces 3D através de espaços de componentes principais*. Tese de Doutorado, IME-USP. Citado na pág. 49
- Mena-Chalco et al.(2008a)** Jesús P. Mena-Chalco, Roberto M. Cesar-Jr. e Luiz Velho. Banco de dados de faces 3D: IMPA-FACE3D. Relatório técnico, Instituto de Matemática Pura e Aplicada - IMPA - VISGRAF Laboratory, Rio de Janeiro, RJ, Brazil. TR 01. Citado na pág. 11, 73
- Mena-Chalco et al.(2008b)** Jesús P. Mena-Chalco, Ives Macêdo, Luiz Velho e Roberto M. Cesar-Jr. PCA-based 3D face photography. Em *Proceedings of the Brazilian Symposium on Computer Graphics and Image Processing, 21 (SIBGRAPI)*, páginas 313–320. IEEE Computer Society. Citado na pág. 20
- Mena-Chalco et al.(2009)** Jesús P. Mena-Chalco, Ives Macêdo, Luiz Velho e Roberto M. Cesar-Jr. 3D face computational photography using PCA spaces. *The Visual Computer*, 25(10):899–909. Citado na pág. 1, 49, 54
- Moreira et al.(2010)** Juliano L. Moreira, Adriana Braun e Soraia R. Musse. Eyes and eyebrows detection for performance driven animation. Em *Sibgrapi 2010 (XXIII Brazilian Symposium on Computer Graphics and Image Processing)*. Citado na pág. 3
- Noh e Neumann(2001)** Jun-yong Noh e Ulrich Neumann. Expression cloning. Em *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM. Citado na pág. 3, 5, 7
- Noma(2010)** Alexandre Noma. *Duas abordagens para casamento de padrões de pontos usando relações espaciais e casamento entre grafos*. Tese de Doutorado, IME-USP. Citado na pág. 22
- Noma et al.(2008)** Alexandre Noma, Alvaro Pardo e Roberto M. Cesar Jr. Structural matching of 2D electrophoresis gels using graph models. Em *SIBGRAPI '08: Proceedings of the 2008 XXI Brazilian Symposium on Computer Graphics and Image Processing*, páginas 71–78, Washington, DC, USA. IEEE Computer Society. Citado na pág. 18, 73

- Obaid et al.(2009)** Mohammad Obaid, Daniel Lond, Ramakrishnan Mukundan e Mark Billingham. Facial caricature generation using a quadratic deformation model. Em *ACE '09: Proceedings of the International Conference on Advances in Computer Entertainment Technology*, páginas 285–288, New York, NY, USA. ACM. Citado na pág. 7
- Paysan et al.(2009)** Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani e Thomas Vetter. A 3D face model for pose and illumination invariant face recognition. Em *AVSS '09: Proceedings of the 2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, páginas 296–301, Washington, DC, USA. IEEE Computer Society. Citado na pág. 5
- Pelachaud(2009)** Catherine Pelachaud. Modelling multimodal expression of emotion in a virtual agent. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1535):3539–3548. Citado na pág. 47
- Pighin et al.(2006)** Frederic Pighin, Jamie Hecker, Dani Lischinski, Richard Szeliski e David H. Salesin. Synthesizing realistic facial expressions from photographs. Em *SIGGRAPH '06: ACM SIGGRAPH 2006 Courses*. ACM. Citado na pág. 1, 2
- Queiroz et al.(2010)** Rossana B. Queiroz, Marcelo Cohen, Juliano Moreira, Adriana Braun, Julio C. Jacques Junior e Soraia R. Musse. Generating facial ground truth with synthetic faces. Em *Sibgrapi 2010 (XXIII Brazilian Symposium on Computer Graphics and Image Processing)*. Citado na pág. 3, 62
- Rodrigues(2007)** Paula Salgado Lucena Rodrigues. *Um Sistema de Geração de Expressões Faciais Dinâmicas em Animações Faciais 3D com Processamento de Fala*. Tese de Doutorado, PUC-Rio. Citado na pág. 47, 50
- Sattar et al.(2009)** Abdul Sattar, Nicolas Stoiber, Renaud Seguier e Gaspard Breton. Gamer's facial cloning for online interactive games. *International Journal of Computer Games Technology*, 2009. Citado na pág. 7
- Savran et al.(2008)** Arman Savran, Nese Alyuz, Hamdi Dibeklioglu, Oya Celiktutan, Berk Gokberk, Bulent Sankur e Lale Akarun. Bosphorus database for 3D face analysis. Em *Proceedings of the First COST 2101 Workshop on Biometrics and Identity Management (BIOD)*, Denmark. Citado na pág. 73
- Scherbaum et al.(2007)** Kristina Scherbaum, Martin Sunkel, Hans-Peter Seidel e Volker Blanz. Prediction of individual non-linear aging trajectories of faces. Em *The European Association for Computer Graphics, 28th Annual Conference, EUROGRAPHICS 2007*, volume 26 of *Computer Graphics Forum*, páginas 285–294, Prague, Czech Republic. Blackwell. Citado na pág. 2
- Shi e Tomasi(1994)** Jianbo Shi e Carlo Tomasi. Good features to track. Em *1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94)*, páginas 593 – 600. Citado na pág. 20
- Sifakis et al.(2005)** Eftychios Sifakis, Igor Neverov e Ronald Fedkiw. Automatic determination of facial muscle activations from sparse motion capture marker data. Em *SIGGRAPH '05: ACM SIGGRAPH 2005 Papers*. ACM. Citado na pág. 3

- Sifakis et al.(2006)** Eftychios Sifakis, Andrew Selle, Avram Robinson-Mosher e Ronald Fedkiw. Simulating speech with a physics-based facial muscle model. Em *SCA '06: Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer animation*. Eurographics Association. Citado na pág. 3
- Song et al.(2007)** Mingli Song, Zhao Dong, Christian Theobalt, Huiqiong Wang, Zicheng Liu e Hans peter Seidel. A generic framework for efficient 2d and 3d facial expression analogy. Em *IEEE Transactions on Multimedia*, volume 9. Citado na pág. 3
- Tang e Huang(2008)** Hao Tang e Thomas S. Huang. MPEG4 performance-driven avatar via robust facial motion tracking. Em *ICIP*, páginas 249–252. Citado na pág. 3, 62
- Theobald et al.(2007)** Barry-John Theobald, Iain A. Matthews, Jeffrey F. Cohn e Steven M. Boker. Real-time expression cloning using appearance models. Em *ICMI '07: Proceedings of the 9th international conference on Multimodal interfaces*, páginas 134–139, New York, NY, USA. ACM. Citado na pág. 3, 7
- Toma et al.(2009)** Arshed Toma, Alexei Ivanovich Zhurov, Rebecca Playle, Egle Ong e Stephen Richmond. Reproducibility of facial soft tissue landmarks on 3D laser-scanned facial images. *Orthod Craniofac Res*, 12(1):33–42. Citado na pág. 6
- Vlasic et al.(2006)** Daniel Vlasic, Matthew Brand, Hanspeter Pfister e Jovan Popovic. Face transfer with multilinear models. Em *SIGGRAPH '06: ACM SIGGRAPH 2006 Courses*. ACM. Citado na pág. 3, 5, 7
- Weise et al.(2009)** Thibaut Weise, Hao Li, Luc Van Gool e Mark Pauly. Face/off: live facial puppetry. Em *SCA '09: Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM. Citado na pág. 3
- Whitmarsh et al.(2006)** Tristan Whitmarsh, Remco C. Veltkamp, Michela Spagnuolo, Simone Marini e Frank Ter Haar. Landmark detection on 3D face scans by facial model registration, 2006. Citado na pág. 7
- Wilson et al.(2010)** Cyrus A. Wilson, Abhijeet Ghosh, Pieter Peers, Jen-Yuan Chiang, Jay Busch e Paul Debevec. Temporal upsampling of performance geometry using photometric alignment. *ACM Transactions on Graphics*, 29:17:1–17:11. Citado na pág. 2
- Wu et al.(2002)** Zhilin Wu, Petar S. Aleksic e Aggelos K. Katsaggelos. Lip tracking for MPEG-4 facial animation. Em *ICMI '02: Proceedings of the 4th IEEE International Conference on Multimodal Interfaces*. IEEE Computer Society. Citado na pág. 3
- Yano e Harada(2009)** Ken Yano e Koichi Harada. A facial expression parameterization by elastic surface model. *International Journal of Computer Games Technology*, 2009. Citado na pág. 3
- Yin et al.(2006)** Lijun Yin, Xiaozhou Wei, Yi Sun, Jun Wang e Matthew J. Rosato. A 3D facial expression database for facial behavior research. Em *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition, FGR '06*, páginas 211–216, Washington, DC, USA. IEEE Computer Society. Citado na pág. 73

Histograma de referências bibliográficas

