



Licença Creative Atribuição-Vedada a
Criação de Obras Derivadas 2.5 Brasil Commons

Fundação Universidade Federal do Rio Grande
Engenharia de Computação

Animador Tridimensional de Faces

Thales Sehn Körting

Rio Grande, 4 de Maio de 2007

Fundação Universidade Federal do Rio Grande
Engenharia de Computação

Animador Tridimensional de Faces

Thales Sehn Körting

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

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

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

Rio Grande, 4 de Maio de 2007

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

Prof. Ms. Alessandro de Lima Bicho

Banca Examinadora:

Prof. Ms. Alessandro de Lima Bicho

DMAT – FURG (Orientador)

Prof^ª. Danúbia Bueno Espíndola

DMAT – FURG

Prof. Dr. Nelson Lopes Duarte Filho

DMAT – FURG

*Poucos dias atrás percebi que gostei dos computadores graças
ao tio Fernando (in memoriam)*

Agradecimentos

A Deus, pela compreensão de meu temporário afastamento e por estar sempre olhando por mim, até mesmo nas intermináveis madrugadas em que este texto foi escrito.

Ao pesquisador Dr. James D. Edge (*University of Sheffield, UK*), pela disponibilidade e paciência em dúvidas cruciais para o bom andamento do trabalho.

Ao professor Alessandro de Lima Bicho, pela ajuda nas decisões de projeto e pelas dicas quentes dadas nas madrugadas precedentes à entrega deste trabalho.

Ao professor Nelson Lopes Duarte Filho, pelas lições, conversas e conselhos que muito me ensinaram durante o andamento do curso.

Aos colegas Felipe Castro da Silva e Rodrigo Mendes Costa, que trilharam comigo os caminhos da Engenharia de Computação, em um ambiente saudável de amizade e companheirismo despretenso.

À Aline, pelo seu carinho e amor, compreensão e companheirismo, em todos os momentos envolvidos nesta longa jornada.

À minha família, por tudo que me ensinou neste quase 1/4 de século de vida...

Às Irmãs do Mosteiro São José, pelo apoio espiritual.

Às pessoas que compreenderam que o texto dos agradecimentos está em ordem alfabética, e que portanto não se sentiram classificadas por qualquer grau de prioridade.

Conteúdo

Lista de Figuras	iii
Lista de Tabelas	v
Lista de Abreviaturas	vi
Resumo	vii
Abstract	viii
1 Introdução	1
1.1 Estrutura do Texto	1
2 Animação	3
2.1 Histórico da Computação Gráfica	3
2.2 Modelagem de Superfícies	5
2.2.1 Métodos Tradicionais	6
2.2.2 Polígonos	6
2.2.3 Patches	7
2.2.4 Superfícies Quádricas	9
2.3 Técnicas de Suavização	9
3 Animação Facial	11
3.1 Trabalhos Correlatos	11
3.2 Aplicações	13
3.2.1 Conferências	13

3.2.2	Agentes Interativos	15
3.3	Regras Básicas da Animação Facial	18
3.4	Emoções	19
3.4.1	Lábios	22
3.4.2	Sobrancelhas	23
3.4.3	Olhos, Dentes e Língua	23
3.4.4	Músculos	24
4	Técnicas de Animação Facial	25
4.1	Modelos Paramétricos	26
4.1.1	<i>Radial Basis Functions</i>	27
4.1.2	Modelo de Waters	36
4.2	Modelos Baseados em Leis da Física	40
4.3	Outras Abordagens	43
4.3.1	<i>Free Form Deformation</i>	43
4.3.2	Interpolação	44
5	Sistema Computacional	45
5.1	Diagramas	45
5.2	Fonte dos Movimentos	46
5.3	Linguagens	49
5.4	Manual	52
6	Considerações Finais	54
6.1	Dificuldades Encontradas	55
6.2	Principais Contribuições	55
6.3	Propostas Futuras	56
6.3.1	Texto-Fala	56
	Bibliografia	58

Lista de Figuras

2.1	Ivan Sutherland(E) e Dave Evans (D), Dezembro de 1969	4
2.2	Tron: filme pioneiro em efeitos gerados por computador	5
2.3	Superfície de Bèzier	7
2.4	Superfície de Hermite	8
2.5	Superfície Catmul-Rom	8
2.6	Curva B-Spline	9
2.7	Subdivisão recursiva de uma malha poligonal [DeRose et al., 1998]	10
2.8	Potencial da técnica de Suavização [DeRose et al., 1998]	10
3.1	Primeira animação 3D de uma face [Parke, 1972]	12
3.2	Primeira animação de face com fala sincronizada	12
3.3	Exemplo de conferência	15
3.4	<i>Woogles</i> do projeto OZ [Loyall, 1997]	17
3.5	MAX [Kopp et al., 2003] – um exemplo de Interface Multimodal	18
3.6	Pontos Característicos do MPEG-4	21
3.7	Unidades de Parâmetros do MPEG-4	22
3.8	Modelos de olho, dentes e língua [Haber et al., 2001]	23
3.9	Estrutura muscular simulada [Kähler et al., 2001]	24
4.1	Algumas Funções de Base existentes	32
4.2	Malha contraída e pontos de controle [Edge and Maddock, 2003]	33
4.3	Pontos de controle em uma malha facial	33
4.4	Regiões de atuação de alguns pontos de controle	34
4.5	Face modificada pelas funções de base radial	35

4.6	Movimento de pontos de controle da Boca aplicados à vizinhança	36
4.7	Modelo baseado em [Waters, 1987]	37
4.8	Exemplo de malha contraída [Waters, 1987]	38
4.9	Músculos utilizados pela proposta de Waters [Noh and Neumann, 1998]	40
4.10	Malhas quadradas contraídas, com ângulos de abertura de $PI/2$ e $PI/6$	41
4.11	Face modificada pela técnica de Waters	42
4.12	Sistema massa-mola aplicado à Animação Facial	42
4.13	Modelo multi-camadas para Animação Facial [Lee et al., 1995]	42
4.14	Interface do Modelo Facial Físico	43
5.1	Diagrama de atividades realizadas pelo sistema	46
5.2	Fluxograma do sistema	47
5.3	Ambiente das filmagens	48
5.4	Exemplo de extração dos pontos de controle da boca	49
5.5	Resultado da animação dos pontos de controle da boca	49
5.6	Tela inicial do sistema	52
5.7	Funcionamento do sistema	53

Lista de Tabelas

3.1	Unidades de Medida	14
3.2	Principais Unidades de Ação [Noh and Neumann, 1998]	20
3.3	Composição das emoções [Noh and Neumann, 1998]	20

Lista de Abreviaturas

API *Application Programming Interface*

AU's *Action Units*

AVI *Audio/Video Interleaved*

CG *Computação Gráfica*

DPC's *Dinâmicas dos Pontos de Controle*

EPC's *Emoções em Pontos de Controle*

FACS *Facial Action Coding System*

FAP's *Facial Animation Parameters*

FAPU's *Facial Animation Parameter Units*

FFD *Free Form Deformation*

FP's *Feature Points*

GUI *Graphical User Interface*

IDE *Integrated Development Environment*

IHC *Interação Humano-Computador*

PPC's *Posições dos Pontos de Controle*

RBF's *Radial Basis Functions*

Resumo

O cenário atual caracteriza-se por ser altamente dinâmico, no qual a comunicação presencial é substituída pela virtual, por motivos de praticidade e redução de custos. Portanto, a pesquisa e o desenvolvimento nesse ramo têm se tornado práticas bastante relevantes. Partindo-se dessa premissa, surge a vontade de se criar um sistema computacional capaz de simular movimentos humanos em uma face virtual, modelada e animada através de técnicas de Computação Gráfica. Essa face será representada em computadores através de uma malha de polígonos, sendo esta definida por meio de um conjunto de diversos pontos (vértices) no espaço 3D, conectados entre si através de arestas. Diversas são as abordagens para realizar a animação facial, sendo classificadas em dois grandes grupos: modelos baseados nas leis da Física e modelos paramétricos. As estratégias mais difundidas são discutidas ao longo deste texto, dando particular atenção à estratégia utilizando Funções de Base Radial, escolhida e implementada neste trabalho. A idéia principal desta estratégia está baseada na inserção de pontos de controle na face virtual onde, através dos movimentos desses pontos, movimentam os demais pontos ao seu redor, em uma região de influência definida com base em heurísticas percebidas na própria anatomia humana.

Abstract

Nowadays the scenario is highly dynamic, where the communication with physical presence is being changed by the virtual, for some reasons like commodity and cost reduction. So, research and development in this area have become important tasks. With this in mind, the will of creating a computer system capable of simulate human movements arises, using a virtual face, through Computer Graphics techniques. This face will be represented in computers by a polygon mesh, being defined through a set of points (vertices) in the 3D space, interconnected by the edges. There are many approaches to facial animation, classified in two groups: the Physics-based and the Parametric models. The most known approaches are discussed in this text, giving special attention to the Radial Basis Functions, that has been chosen and implemented in this work. The main idea in this approach is based upon the insertion of control points in the facial mesh and, through the dynamics of this ones, perform the animation of the other points around this control points, in a influence region defined using heuristics of the human anatomy.

Capítulo 1

Introdução

Uma área fascinante dentro da Informática é a Computação Gráfica (CG), tendo aplicação nas mais diversas áreas, tais como comunicação, medicina, robótica, aviação, cinema, etc. Nestas áreas, a CG é utilizada principalmente em simulações que visam reproduzir situações reais por meio das características mais relevantes nestas aplicações, que são exibidas sem apresentar nenhum impacto verdadeiro, com o intuito de possibilitar uma análise e estudo do ambiente simulado.

Vemos, no cenário atual altamente dinâmico, a comunicação presencial sendo substituída pela virtual, tanto pela praticidade quanto pela redução de custos na maioria dos casos. A pesquisa e o desenvolvimento nesse ramo têm se tornado práticas bastante relevantes. Sendo assim, surge uma proposta inicial de se implementar um sistema capaz de simular uma face humana, reproduzindo os movimentos e expressões da fala. As inúmeras aplicações de tal iniciativa serão apresentadas nos capítulos seguintes, cabendo à introdução apenas a descrição básica das idéias que motivaram o autor no desenvolvimento desse projeto.

1.1 Estrutura do Texto

A estrutura está apresentada da seguinte forma: no Capítulo 2, faz-se uma introdução à Animação realizada pela CG, para a seguir (Capítulo 3), iniciarmos uma discussão detalhada sobre a Animação Facial, objetivo principal deste trabalho. Já com o entendimento sobre a animação facial, parte-se para a descrição das técnicas utilizadas afim de

alcançarmos as metas dessa proposta (Capítulo 4). Em seguida (Capítulo 5), é feita a descrição completa do sistema implementado, englobando características básicas, decisões de projeto, manual de utilização e exemplos de utilização. Por fim, comentários sobre as conclusões e perspectivas futuras são apresentados, também como forma de ratificar o conhecimento adquirido durante a tarefa de implementação desse trabalho.

Capítulo 2

Animação

Optou-se por criar esse capítulo para nele colocar uma visão geral sobre o assunto Animação, dado que o propósito geral desse projeto é um sistema de animação utilizando técnicas de CG. Assim, as seções seguintes tentam traçar um panorama dessa área da computação, visando familiarizar o leitor com o assunto, contemplando desde o histórico até projetos contemporâneos e técnicas para o aumento de realismo.

2.1 Histórico da Computação Gráfica

Atribui-se aos professores David Evans e Ivan Sutherland (Figura 2.1), da Universidade de Utah (Salt Lake City – Utah, EUA), o pioneirismo nas pesquisas de CG. Suas pesquisas e teorias, na década de 60, apontavam para o fato de que computadores poderiam ser utilizados como simuladores. A partir desta motivação, nasceu a idéia de utilizar gráficos e desenhos para facilitar o trabalho que, até então, era feito sem o auxílio do computador. Assim nasceu a Evans & Sutherland (E&S – <http://www.es.com/>), uma empresa cuja principal atividade era a criação de aplicações gráficas para simulação. Além disso, pessoas reconhecidas atualmente em CG foram descobertas na E&S, tais como Jim Clark (fundador da Silicon Graphics), Ed Catmull (fundador da Pixar) e John Warnock (fundador da Adobe) [Straccia, 2002].

Nesse mesmo período, surgem resultados de algumas pesquisas, como o dispositivo de interação denominado *mouse*, criado por Douglas C. Engelbart em 1963. Muito do que conhecemos hoje das teorias de CG (*e.g.*, desenho vetorial, representação 3D em ambiente



Figura 2.1: Ivan Sutherland(E) e Dave Evans (D), Dezembro de 1969

2D, texturas, etc.) teve seu início nesta época. Em 1963, Steve A. Coons apresenta sua teoria de representação de superfícies curvas através de “retalhos” (polígonos) baseados em aproximações polinomiais. Em 1965, Lawrence G. Roberts cria um algoritmo de remoção de partes invisíveis de segmentos de reta e introduz a noção de coordenadas homogêneas na representação geométrica de objetos. Nesse mesmo período, Jack E. Bresenham desenvolve algoritmos eficientes para o desenho de primitivas geométricas 2D. Em 1967, Rod Rougelet cria um simulador interativo de vôo para a NASA [Costa, 2001].

Podemos perceber, nesse primeiro momento, que as universidades americanas têm grande importância na disseminação da CG pelo mundo inteiro, com diversas pesquisas iniciadas e o reconhecimento mundial por seu pioneirismo.

No ano de 1976, temos como outro marco histórico a criação do computador pessoal, o Apple I, seguido do Apple II em 1979 e pelo PC da IBM em 1981. Em seguida, surge o conceito de GUI¹, instituído pela Macintosh, o qual alavancou tanto o uso de computadores em geral quanto em particular o desenvolvimento da CG. Em seguida, surgem as empresas Adobe, seguida da Silicon Graphics, que ganharam mercado devido às potencialidades trazidas com o ramo da CG, especialmente para empresas produtoras de vídeo. Com esse arsenal, no início da década de 80 é lançado o primeiro filme com efeitos especiais gerados por computador: Tron (Figura 2.2), da Disney.

¹ *Graphical User Interface* (GUI) – Interface Gráfica com o Usuário



Figura 2.2: Tron: filme pioneiro em efeitos gerados por computador

2.2 Modelagem de Superfícies²

Na síntese de imagens por computador, uma cena é composta pela disposição de objetos no espaço tridimensional. Em geral, estes objetos são descritos por modelos matemáticos que descrevem superfícies e/ou sólidos.

As principais técnicas para representar superfícies em CG são: polígonos, *patches* e superfícies quádricas. Nas próximas seções estas técnicas serão apresentadas pois, na animação facial, veremos que os modelos de face são definidos através destas superfícies tridimensionais. Como qualquer outra superfície, uma face é composta de uma malha ou nuvem de pontos (vértices) interconectados através de arestas, criando uma malha conexa de polígonos. As diversas estratégias para a geração de superfícies serão apresentadas brevemente, uma vez que a geração da malha poligonal não faz parte da proposta, que leva em consideração um modelo de face previamente definido.

De uma maneira geral, as superfícies são uma generalização das curvas. Uma superfície, da mesma maneira que uma curva, pode ser gerada por famílias de conjuntos de pontos. Podemos ainda interpolar, ajustar ou aproximar superfícies a partir de pontos. Essa forma de geração de objetos por seus contornos é muito importante na modelagem geométrica [Azevedo and Conci, 2003].

²Seção baseada em [Fabbro and Martino, 1998] e [Azevedo and Conci, 2003].

2.2.1 Métodos Tradicionais

A rotação de uma curva plana em torno de um eixo produz a família mais conhecida de superfícies, definindo as superfícies de **Revolução**. Assim, um segmento de reta girando de 360° , inclinado em relação ao seu eixo de rotação, produzirá um cone. Curvas, ângulos e eixos de rotações diferentes produzem diversas formas de superfícies.

Existem também as superfícies geradas por **Deslocamento**, nas quais translações e deslocamentos genéricos de curvas produzem diversas formas, sendo também conhecidas pela técnica de *sweeping* (varredura).

Por sua vez, as superfícies geradas por **Interpolação Bilinear** são baseadas na expressão da curva ou dos pontos que descrevem seus limites. Após a geração destes limites, por técnicas de interpolação, o interior dessa superfície é gerado, geralmente empregando retas que iniciam em pontos pertencentes a outras retas formadas pelos pontos limites, pré-estabelecidos para a formação da superfície.

2.2.2 Polígonos

Polígono é toda superfície planar com fronteiras definidas por segmentos de reta. Essa forma de representação de superfícies é uma das mais conhecidas no contexto da CG, sendo ideal para superfícies planas, tipicamente encontradas em elementos chamados facetados, tais como pirâmides e paralelepípedos. Para superfícies curvas, como uma esfera, essa representação é aproximada, porém muito utilizada, devido à praticidade e à facilidade de entendimento no seu uso. Mais especificamente, polígonos são superfícies cujo contorno é dado por uma série de vértices ordenados. Cada vértice é conectado com o próximo, sendo que o último vértice é conectado com o primeiro [Fabbro and Martino, 1998].

Na medida em que estamos tentando representar superfícies curvas através de pequenas superfícies retas, iremos perceber um erro nesse tipo de aproximação, o qual se reduz na medida em que mais polígonos (cada vez menores) são utilizados para representar determinada superfície.

2.2.3 Patches

Pode-se utilizar uma abordagem de aproximação polinomial por partes, em uma representação paramétrica com polinômios de terceira ordem (cúbicos) ou com dois parâmetros (bicúbicos), o que traz maior flexibilidade, particularmente às superfícies curvas. À primitiva geométrica associada a esta representação é dado o nome de *patch*, sendo os principais *patches* bicúbicos Bèzier, Hermite, Catmull-Rom e B-Spline.

Bèzier

A superfície é gerada interpolando o primeiro e o último pontos de controle, sendo que mais dois pontos são responsáveis pela direção da curva nos pontos extremos. Esta curva provê grande controle sobre sua forma por parte do usuário. Podemos extrapolar seu uso para um maior número de pontos, como vemos no exemplo da Figura 2.3. Os pontos de controle (em azul) estão conectados como em uma malha poligonal e a superfície de Bèzier é gerada em função de tais pontos.

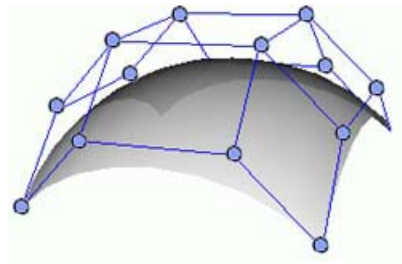


Figura 2.3: Superfície de Bèzier

Hermite

A superfície de Hermite é especificada por 4 pontos de interpolação, 8 vetores tangentes e 4 vetores obtidos por derivações de segunda ordem [Cochran, 2003]. O comprimento destes vetores define o grau de curvatura nos pontos extremos. Podemos ver um exemplo dessa técnica na Figura 2.4, sendo os pontos em azul os 4 pontos de interpolação, conectados a 2 vetores tangentes cada qual.

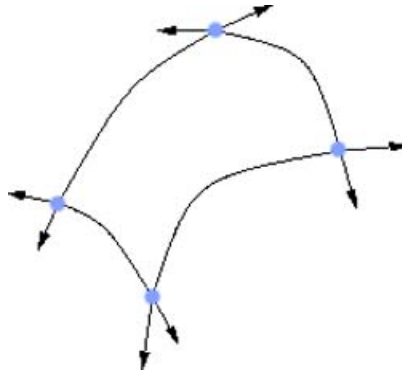


Figura 2.4: Superfície de Hermite

Catmull-Rom

Para a superfície Catmull-Rom, a curva base interpola o segundo e o terceiro pontos de controle. A tangente nestes pontos é paralela à linha que liga o ponto anterior (ao segundo ou terceiro ponto) ao ponto seguinte. Uma característica particular é o fato de que as superfícies geradas passam sempre por todos os pontos de controle (Figura 2.5) [Dunlop, 2002].

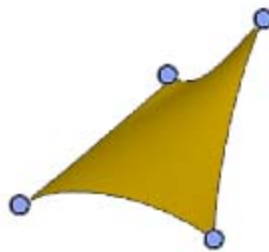


Figura 2.5: Superfície Catmull-Rom

B-Spline

A curva na qual se baseia a superfície B-Spline é mais difícil de ser controlada, uma vez que ela é totalmente aproximada (Figura 2.6). No entanto, são utilizadas pelo fato de oferecerem suavidade e flexibilidade, características difíceis de serem obtidas utilizando modelos poligonais convencionais [Noh and Neumann, 1998].

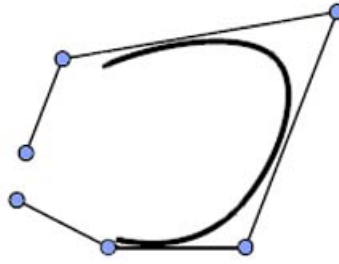


Figura 2.6: Curva B-Spline

2.2.4 Superfícies Quádricas

Utilizada amplamente em ferramentas de modelagem do tipo *Computer Aided Design* (CAD), pelo fato de se adaptarem à representação denominada *Constructive Solid Geometry* (CSG), a qual representa sólidos pela combinação de primitivas e operadores de união, diferença e intersecção. Superfícies Quádricas são construídas por revolução, através da seguinte formulação algébrica:

$$Ax^2 + By^2 + Cz^2 + Dxy + Eyz + Fxz + Gx + Hy + Jz + K = 0 \quad (2.1)$$

sendo A, B, \dots, K constantes, sendo cada superfície quádrlica (*e.g.* cilindro, cone, elipse, esfera, hiperbolóide, parabolóide) dependente dos valores dessas constantes.

2.3 Técnicas de Suavização

O propósito dos modelos que subdividem superfícies é suavizar as pontas (ou vincos), podendo representá-las com intensidades variando de zero (significando suavidade) à infinito (pontas) [DeRose et al., 1998]. Muito se pesquisa nessa área, dando especial enfoque de aplicação para os estúdios de animação, responsáveis pela criação de filmes utilizando apenas computadores, nos quais métodos de suavização se aplicam à modelagem de tecidos para simular roupas e a pele dos próprios personagens, dando particular atenção à face e às mãos.

Para suavizar uma malha poligonal, pode-se optar pela estratégia de se criar regras de subdivisão cujos pesos sejam parametrizados pelo nível de intensidade das pontas

que se desejar alcançar. Outra técnica é a de usar um conjunto de regras para um número arbitrário de passos de subdivisão, seguidos por outro conjunto de regras que são aplicadas ao limite. A suavidade, portanto, depende somente do segundo conjunto de regras, definido *a priori*. Geralmente, a suavização é conseguida através de um processo recursivo, cujas superfícies mais rígidas vão se tornando suavizadas conforme o número de iterações. Um exemplo pode ser visto na Figura 2.7, no qual uma malha de controle é mostrada ao longo de diversos níveis de suavização.

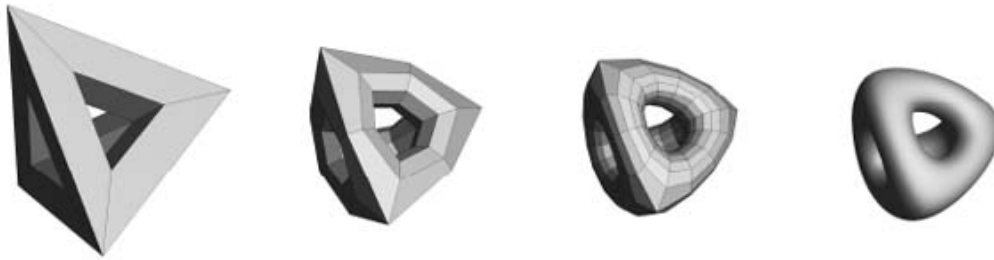


Figura 2.7: Subdivisão recursiva de uma malha poligonal [DeRose et al., 1998]

De uma forma mais prática, podemos exemplificar o uso dessa técnica e de seu potencial baseado em pesos tanto para limitar quanto para definir zonas de influência de suavização. Na Figura 2.8 (imagem retirada do curta metragem *Geri's game* – *Pixar Animation Studios*), vemos a suavização da mão juntamente com vincos bem definidos nas pontas dos dedos e das unhas.



Figura 2.8: Potencial da técnica de Suavização [DeRose et al., 1998]

Capítulo 3

Animação Facial

Um capítulo sobre esse tema foi introduzido com o intuito de mostrar uma perspectiva histórica desse conceito, aliada ao surgimento de aplicações que pudessem utilizar tal abordagem. Devemos deixar claro o fato de que o sistema proposto por esse trabalho não implementa tais sugestões de aplicabilidade, porém poderá vir a servir como parte de um sistema maior, que o utilize como sub-sistema para prover tal funcionalidade.

3.1 Trabalhos Correlatos

Um dos primeiros trabalhos de representação facial por meio de computadores foi realizado no início dos anos 70. A primeira animação tridimensional de uma face foi criada por Parke em 1972 [Parke, 1972] (Figura 3.1). Posteriormente, no início dos anos 80 ocorreu o desenvolvimento do primeiro modelo facial controlado por músculos [Platt and Badler, 1981]. Já o primeiro modelo com fala sincronizada foi apresentado em 1985 no curta metragem “Tony de Peltrie” [Breton et al., 2001] (Figura 3.2).

Em [Kähler et al., 2001], a animação é feita por meio da construção de um modelo facial multi-camadas, onde a geometria do esqueleto e a disposição dos músculos faciais são criados de maneira semi-automática, baseados na malha facial. A animação é alcançada pela simulação de um sistema massa-mola que conecta as três camadas.

Uma outra abordagem [Lee et al., 1995] trata a animação facial particular a cada indivíduo, a partir da criação da malha poligonal específica de cada pessoa, automatizando a desafiadora tarefa de criar modelos faciais com características realistas. Através da

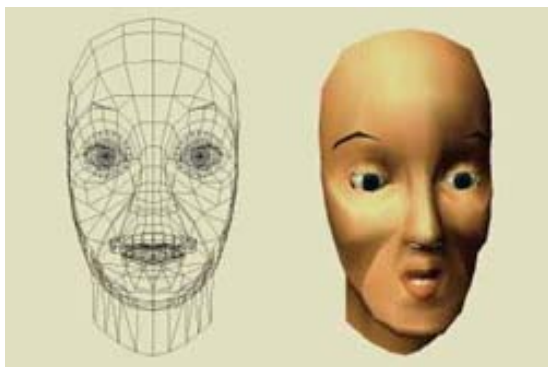


Figura 3.1: Primeira animação 3D de uma face [Parke, 1972]



Figura 3.2: Primeira animação de face com fala sincronizada

aplicação de algoritmos que lidam com dados adquiridos por *scanners* tridimensionais, a face de cada indivíduo é modelada utilizando suas próprias características, e animada por uma simulação de pele dinâmica em conjunto com um modelo muscular em forma de uma camada física inserida logo abaixo da pele, conectada por vértices que distribuem as forças aplicadas.

Já [Edge et al., 2004] realiza a animação facial baseada em fragmentos de fala previamente captados, em uma combinação de deformação, aplicação na malha e mistura entre os estados, gerando estados intermediários. Isso é obtido em um processo dividido em cinco etapas: aquisição dos dados (através de microfones e câmeras – áudio/vídeo), pré-processamento, geração dos fragmentos de dados, síntese do movimento e a animação, através do uso da técnica de *Bézier triangle Induced Deformations* (BID's).

O sistema intitulado *Video Rewrite* [Bregler et al., 1997] realiza uma modificação em vídeos reais contendo personagens falando. O efeito obtido é a modificação da fala presente em uma filmagem real, sincronizando os movimentos labiais com a nova fala. Uma das intenções deste projeto é a utilização em dublagem de filmes, recriando os movimentos labiais dos atores, mantendo sua voz, mas alterando as palavras e o idioma.

A modelagem e a animação de faces humanas é tida atualmente como uma das mais difíceis tarefas na Computação Gráfica [Haber et al., 2001]. A principal meta é dar a ilusão de que as expressões e os movimentos são motivados pelo próprio personagem, ao invés de serem manipulados pelo animador. Talvez o maior desafio seja a síntese de faces artificiais que possam parecer e agir como uma pessoa real. A solução para isso envolverá

não somente Computação Gráfica, mas também outras áreas do conhecimento, tais como Psicologia e Inteligência Artificial (IA).

3.2 Aplicações

Esta seção apresenta algumas das utilidades nas quais um sistema de Animação Facial pode se enquadrar, também com o intuito de justificar a escolha por esta área de pesquisa.

Com o advento de espaços sociais virtuais, onde as pessoas se comunicam face-a-face, a demanda por personagens virtuais com características realistas aumenta [Haber et al., 2001]. Reconstrução de faces específicas também tem aplicações na medicina (*e.g.* cirurgia plástica), entretenimento com a construção de atores virtuais, etc.

3.2.1 Conferências

Podemos invadir um pouco a seara filosófica para comentarmos a respeito do novo paradigma de relacionamento interpessoal instituído pela tecnologia. Além de hoje termos, por exemplo, as telefonia fixa e móvel, com os quais as pessoas podem nos encontrar em qualquer lugar, temos os programas de computador, nos quais as pessoas apenas se comunicam através de mensagens de texto, nos famosos programas de *chat*. Diversas são as tentativas de se expressar emoções por esses programas, algumas representadas por onomatopéias e outras por simbologias que lembram algumas expressões (*e.g.* :), ;), :(, etc.). No entanto, sabemos que diversas vezes o que ocorre são representações de sentimentos através do texto que podem não estar refletindo o atual sentimento dos participantes, como se tais programas servissem até mesmo como máscara para os usuários. Podemos pensar e certamente lembraremos de diversos exemplos nos quais tal situação ocorre.

Assim, uma das propostas é a de se capacitar o computador a transmitir a própria imagem dos comunicantes durante um diálogo (também chamado de conferência). Algumas iniciativas já estão disponíveis aos usuários comuns, através de sistemas que filmam os usuários utilizando câmeras de baixa resolução (conhecidas por *webcams*). No entanto, alguns fatores ainda são limitantes nesse tipo de abordagem, na medida em que a velocidade de transmissão de informações ainda não é satisfatória, conduzindo a uma representação

unidade	significado
s	segundo
f	quadro
i	informação
px	<i>pixel</i>
pt	ponto

Tabela 3.1: Unidades de Medida

não condizente com a movimentação a que estamos acostumados a ver ao vivo.

Diante do que foi explicado acima, se pode estabelecer a primeira área de aplicação deste projeto. Utilizando uma *webcam*, é possível extrair, por técnicas de visão computacional, a dinâmica das principais regiões de uma face, que juntas representam alguma expressão ou representações para os padrões visuais de movimentação articulatória observáveis na face durante a fala (o que é definido por visema) [Martino, 2005]. Conclui-se que uma quantidade muito pequena de pontos (menos de 50) representa um bom conjunto de informações representativos do estado atual de movimento de uma face, e com este se pode representar a dinâmica em um modelo tridimensional. Também se percebe que transmitir esse conjunto de pontos a cada estado da face é muito mais simples do que transmitir uma imagem de uma face, que no caso exemplificado possui um total de $76800px$ (uma *webcam* com resolução de $320px \times 240px$).

Para deixar clara a questão da diferença do volume de dados, vamos pensar em uma transmissão de um vídeo por um período de 1 minuto ($t = 60s$), a uma taxa de captação de 15 quadros/segundo. Baseado nas unidades da Tabela 3.1, teremos:

Via *webcam*¹:

$$1f = 320px \times 240px \longrightarrow 76800i/f$$

$$60s \times 15f/s = 900f$$

$$900f \times 76800i/f \longrightarrow 69120000i$$

Via captação de pontos:

$$1f = 50pt \longrightarrow 50i/f$$

$$60s \times 15f/s = 900f$$

¹Esse exemplo sugere a captação no seu pior caso, desconsiderando estratégias de codificação de vídeo.

$$900f \times 50i/f \rightarrow 45000i$$

Considerando que uma unidade de informação é tanto um *pixel* quanto um ponto, chega-se, por cada quadro enviado, a uma redução de 99,93% no número de pontos transmitidos a cada segundo. Assim, serão transmitidas apenas as informações sobre a dinâmica dos comunicantes, deixando para o processamento local a movimentação de um personagem 3D, simulando os movimentos captados pelos participantes da conferência.

Além disso, em conversas realizadas “ao vivo”, os participantes desejam conversar olhando nos olhos, característica que não é possibilitada por conferências através de filmagens. Como os participantes estão olhando sempre para o dispositivo de exibição (*e.g.* monitor, telas de projeção, etc.) e não para as câmeras que o estão filmando, os participantes, por conseguinte, poderão se ver, mas não terão uma conversa focalizada. A Figura 3.3 exemplifica tal situação.



Figura 3.3: Exemplo de conferência

3.2.2 Agentes Interativos

Uma área relativamente recente dentro da computação é a chamada Interação Humano-Computador (IHC), na qual se estudam maneiras de aproximar a máquina do ser humano, dotando-a de características simuladas do comportamento humano, através de técnicas de IA. Dito de outra forma, IHC é uma área de pesquisa dedicada a estudar os fenômenos de comunicação entre pessoas e sistemas computacionais [Baranauskas, 2005]. Uma das maneiras de dotar um computador de tal capacidade é através da criação de um Agente

que consiga comunicar-se com o usuário. Na literatura, encontra-se o termo *Embodied Conversational Agent* (ECA) para descrevê-los.

Um ECA é um agente essencialmente comunicativo que imita o ser humano na forma como usa o corpo durante uma conversação, como reage emocionalmente e como mantém um diálogo. Por agente define-se uma entidade que percebe seu ambiente através de sensores e responde através de algum dispositivo de saída, seja visual, sonoro, etc. Há ainda uma subdivisão desses tipos de agentes, nas categorias que estão descritas a seguir [Longhi et al., 2004].

Sistemas de Diálogo (*Chatterbots*)

Tal classe de sistemas tenta imitar a conversa humana, em modo textual. Embora distante do foco de interesse deste trabalho, serve para ilustrar esse ramo da IHC.

Ultimamente, vêm ganhando um certo interesse por parte do setor comercial, na medida em que uma empresa pode utilizá-lo como estratégia de aproximação com os clientes, pois tal sistema está sempre disponível, além de poder manter conversas simultâneas com diversos usuários.

Agentes Autônomos Animados

Podemos representar o significado de agentes autônomos, dentro da IHC, como personagens que integram um mundo dinâmico, podendo ou não interagirem com pessoas. Assim, simulam personalidade, emoção e comportamento social em resposta a estímulos exteriores.

Tais personagens podem ser das mais diversas formas, desde que consigam representar um personagem e de alguma maneira, transmitir informações. Um exemplo dessa abordagem foi apresentada pelo projeto OZ [Loyall, 1997], que traz uma implementação de um mundo simulado, e dentro dele diversos personagens (os agentes autônomos) mutuamente interativos, chamados de *woogles* (Figura 3.4).

Avatares Falantes (*Talking Heads*)

Essa abordagem implementa uma representação facial (2D ou 3D) aliada à capacidade de falar, combinando sons e expressões com movimentos condizentes para representá-los.

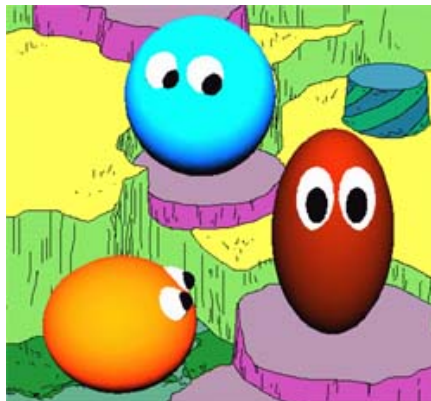


Figura 3.4: *Woogles* do projeto OZ [Loyall, 1997]

Um sistema recente, que está sendo amplamente utilizado por diversos *sites*, visando o aumento da interatividade, é o *talking heads* da Companhia Oddcast (<http://vhost.oddcast.com/>).

A implementação da fala pode contar com um sintetizador Texto-Fala (TF)², um subsistema que deve ser capaz de ler qualquer texto [Körting et al., 2005a], ou também apenas simular os movimentos faciais em um modelo, adicionando sons reais à animação.

Pelo que já foi citado acima, percebemos que a aplicação proposta pelo sistema ao qual esse trabalho se refere pertence a essa classe de agentes interativos, implementado com uma estratégia de modelagem e de animação de uma face em 3D, aliada à voz captada de um usuário real.

Interface Multimodal

O termo multimodal refere-se ao fato da comunicação se dar por diversas maneiras (*e.g.* modos), resultando em uma simulação “natural” da comunicação de um humano, tais como voz, expressão, gestos do corpo e mãos, etc., gerando resultados próximos à realidade, mas atingindo um grau de complexidade alto.

Um exemplo dessa aplicação é desenvolvido pelo grupo de Inteligência Artificial do Laboratório de Realidade Virtual da Universidade de Bielefeld, Alemanha. Conhecido por MAX (*Multimodal Assembly eXpert*), o modelo da Figura 3.5 é simulado em um ambiente virtual no qual um determinado usuário é inserido de modo a interagir com o

²Termo referenciado na bibliografia por *Text-To-Speech* (TTS)



Figura 3.5: MAX [Kopp et al., 2003] – um exemplo de Interface Multimodal

agente.

Além das diversas etapas da CG presentes na implementação de um sistema deste tipo, outras áreas estão inseridas para dar ao agente o comportamento completo, como a IA, a Psicologia, a Física, entre outras.

3.3 Regras Básicas da Animação Facial

Nesta seção, identificaremos uma série de regras que devem ser percebidas para realizar uma animação facial. Algumas delas já foram identificadas há bastante tempo, porém continuam valendo para as implementações contemporâneas.

Os principais aspectos que devem ser considerados em um animador de faces são [Parke and Waters, 1996]:

- quanto mais o modelo facial assemelhar-se a um humano real, maior será nossa percepção crítica sobre o mesmo;
- o potencial de um personagem encontra-se na sua animação e não na sua forma: personagens extremamente convincentes também podem ser obtidos através de simples primitivas geométricas;
- é mais importante preocupar-se com a animação de determinadas características do que outras para se expressar a emoção:

características de alta prioridade são os olhos, a boca, as sobrancelhas e as pálpebras;

certas vezes, um grau de expressão pode ser alcançado apenas pela dilatação ou contração das pupilas [Watt and Watt, 1992];

atributos de menor prioridade são nariz, língua, bochecha, cabelo e orelhas;

- a animação da cabeça em si é algo extremamente importante e não deve ser esquecida, sob pena de a animação parecer muito robotizada;
- animar características faciais simetricamente pode produzir resultados não convincentes – em verdade este é um princípio básico da animação, não somente aplicado à face: um bom exemplo pode ser a animação do piscar dos olhos: alguns instantes devem separar o início da piscada de cada olho, a fim de gerar uma animação mais real.

3.4 Emoções

De um modo geral, as emoções podem ser representadas como, por exemplo, alegria, surpresa, dúvida, medo, raiva, tristeza e assim por diante [Tao and Tan, 2004]. Em [Ekman et al., 2005], temos a descrição de um modelo representativo que serve de notação para as expressões faciais, denominado *Facial Action Coding System* (FACS). Tal notação é dividida em diversas Unidades de Ação – *Action Units* (AU's), as quais servem para descrever o controle das ações da face para expressar as emoções. As principais AU's estão mostradas na Tabela 3.2 e as principais emoções supracitadas estão expressas em forma de AU's na Tabela 3.3.

Considerando aspectos de validade, a relação entre expressões faciais e emoções é baseada apenas em movimentos musculares. Portanto, o desenvolvimento de um método computacional para se medir a emoção de uma expressão pode seguir apenas dois caminhos [Ekman et al., 2005]:

1. o de mostrar claramente que as medidas são baseadas em ações musculares;

AU's	FACS <small>português</small>	FACS <small>inglês</small>
1	Levantar Sobrancelha <small>(interna)</small>	Inner Brow Raiser
2	Levantar Sobrancelha <small>(externa)</small>	Outer Brow Raiser
4	Abaixar Sobrancelha	Brow Lower
5	Levantar Pálpebra Superior	Upper Lid Raiser
6	Levantar a Bochecha	Cheek Raiser
7	Pressionar Pálpebra	Lid Tightener
9	Fransir o Nariz	Nose Wrinkler
10	Levantar a Pálpebra Inferior	Upper Lid Raiser
12	Puxar o Canto do Lábio	Lip Corner Puller
14	Trazer o Lábio para Trás	Dimpler
15	Abaixar o Canto do Lábio	Lip Corner Depressor
16	Abaixar o Lábio Inferior	Lower Lip Depressor
20	Esticar os Lábios	Lip Stretcher
23	Apertar os Lábios	Lip Tightener
26	Abaixar a mandíbula	Jaw Drop

Tabela 3.2: Principais Unidades de Ação [Noh and Neumann, 1998]

Emoção Básica	AU's envolvidas
Alegria	1, 6, 12, 14
Surpresa	1, 2, 5, 15, 16, 20, 26
Dúvida	2, 6, 7, 14
Medo	1, 2, 4, 5, 15, 20, 26
Raiva	2, 4, 7, 9, 10, 20, 26
Tristeza	1, 4, 15, 23

Tabela 3.3: Composição das emoções [Noh and Neumann, 1998]

2. o de conduzir experimentos de validação mais árduos para mostrar que novas medidas são relacionadas com cada emoção em que se tiver interessado.

O estudo do sistema FACS também serve para medir e distinguir todas as diferentes expressões faciais. Todos os músculos faciais foram classificados em 58 AU's, as quais descrevem todas as possibilidades de movimentos. Essa notação é relativamente neutra e se tornou um padrão para a calibração de sistemas de animação facial [Breton et al., 2001].

No entanto, algumas desvantagens da notação FACS devem ser mostradas [Noh and Neumann, 1998]:

- AU's são puramente padrões locais; no entanto a movimentação facial é raramente localizada, e sim, distribuída;
- FACS oferece a descrição da dinâmica espacial, porém não a temporal; isso acaba trazendo problemas na animação co-articulada dos pontos da face;
- mesmo apresentando a descrição espacial, o modelo FACS ainda peca nas deformações mais periféricas do rosto.

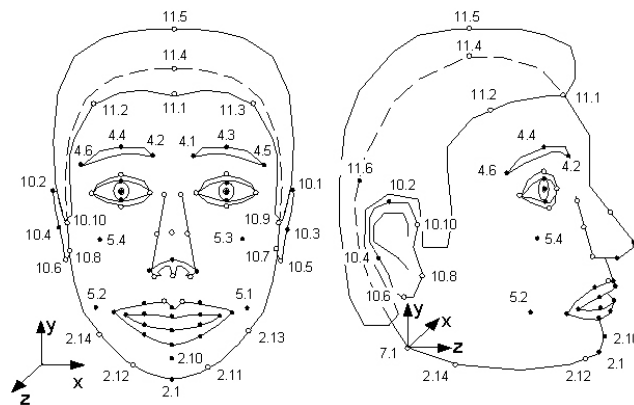


Figura 3.6: Pontos Característicos do MPEG-4

Além do FACS, um outro esforço para a padronização dos modelos paramétricos de face, idealizado originalmente para os propósitos de uma codificação eficiente de imagens em movimento, resultou no desenvolvimento, pelos desenvolvedores do MPEG, do padrão MPEG-4 para *Facial Animation* (FA). Tal padrão define 68 Parâmetros de Animação Facial – *Facial Animation Parameters* (FAP's) e 84 Pontos Característicos – *Feature*

Points (FP's). Esses FP's são pontos chamados *landmarks*, definidos sobre a face humana, tais como *base do queixo*, *canto esquerdo da boca*, etc. Na Figura 3.6 temos a descrição do padrão de forma visual, no qual podemos perceber os diversos FP's presentes na face.

Uma característica do MPEG-4 é que os mesmos valores de FAP's devem produzir um resultado similar em faces diferentes (do contrário o padrão seria inútil). Isto é alcançado através da definição dos FAP's com base nas Unidades de Parâmetros da Animação Facial – *Facial Animation Parameter Units* (FAPU's), definidas pelas distâncias entre determinados FP's do modelo a ser animado. Existem seis FAPU's definidas na especificação do MPEG-4, a saber (Figura 3.7): *diâmetro da íris (IRISD0)*, *distância dos olhos (ES0)*, *distância entre olhos e nariz (ENS0)*, *distância entre nariz e boca (MNS0)*, *largura da boca (MW0)* e *unidade de ângulo* [Beskow, 2003].

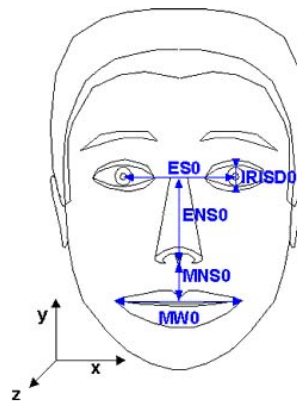


Figura 3.7: Unidades de Parâmetros do MPEG-4

A seguir, a relação das principais características faciais, e as estratégias geralmente adotadas quando da animação de cada uma delas. No Capítulo 5 serão abordadas exclusivamente as estratégias escolhidas para a implementação do sistema proposto por esse trabalho.

3.4.1 Lábios

Para movimentar a boca em qualquer abordagem de animação facial, a distinção entre os lábios superior e o inferior deve ser feita. No entanto, a maioria das malhas poligonais possui lábios sem a adequada separação. Uma idéia poderia ser a de separá-los em duas malhas distintas, mas isso não seria recomendável. Este procedimento destruiria os cantos

dos mesmos, produzindo assim um resultado muito artificial, pois os lábios não estariam conectados quando a boca estiver aberta [Breton et al., 2001].

3.4.2 Sobrancelhas

As sobrancelhas estão especialmente envolvidas com o realismo das expressões faciais sendo, portanto, foco de parte das atenções na modelagem facial. Assim, modelos que a implementem de forma satisfatória obterão resultados certamente mais fidedignos do que outros que não a façam.

3.4.3 Olhos, Dentes e Língua

Importantes para uma animação facial mais realista, são extremamente difíceis de serem modelados, especialmente pela dificuldade encontrada em captar sua completa estrutura e movimentação em imagens reais. Sendo assim, diversos sistemas encontrados na literatura utilizam modelos genéricos para essas características, ajustando poucos parâmetros de maneira a adaptá-los o mais próximo possível dos personagens reais.

Alterações nos olhos incluem apenas a redução ou aumento do raio do globo ocular. Nos dentes faz-se geralmente modificações na angulação, visando encaixá-los internamente no modelo. Modelos de língua são mais empregados quando a simulação contempla representações da fala, pois sabemos que esta tem um papel bastante importante na produção dos sons em conjunto com os dentes (Figura 3.8).

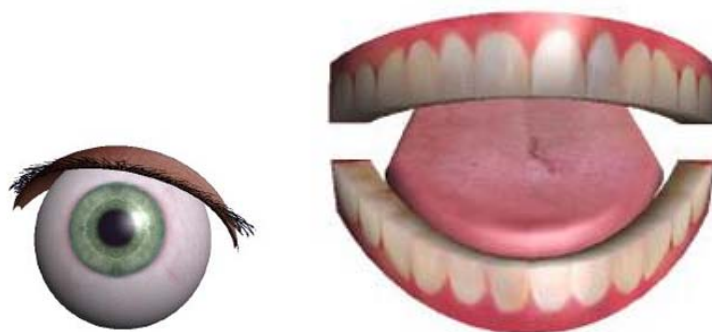


Figura 3.8: Modelos de olho, dentes e língua [Haber et al., 2001]

3.4.4 Músculos

Nas abordagens que os implementam, cada um dos músculos geralmente é construído como um conjunto de fibras individuais ao redor de segmentos lineares, formadores de polígonos de controle (Figura 3.9). Um elipsóide é alinhado a cada segmento, visando prover o formato e o volume de cada fibra muscular [Haber et al., 2001]. Assim, a deformação da malha da pele é causada pela simulação da contração dos músculos faciais.



Figura 3.9: Estrutura muscular simulada [Kähler et al., 2001]

Capítulo 4

Técnicas de Animação Facial

Destina-se um capítulo às técnicas existentes de animação facial com a finalidade de explicar a pesquisa realizada, apresentando decisões de projeto e resultados de testes com diversas técnicas pesquisadas que não entraram na implementação final desse trabalho, para que se possa mostrar mais claramente os motivos que levaram o autor a optar por determinadas técnicas em relação a outras.

Na Seção 4.1 são expostos os dois principais modelos que utilizam a abordagem Paramétrica (*Radial Basis Functions* (RBF's) e o modelo de Waters). Dá-se particular atenção à Seção 4.1.1 (Funções de Base Radial), abordagem escolhida para a implementação do sistema proposto por esse trabalho. A seguir, na Seção 4.2, um outro modelo bastante citado pela bibliografia é exposto, de maneira a situar o leitor com as técnicas tradicionais na área da animação facial. Por fim, na Seção 4.3, há uma descrição um pouco mais superficial de estratégias alternativas de animação facial, não somente aplicadas a faces tridimensionais, como também a animação e adaptação de seqüências de vídeos em duas dimensões.

Já foi mencionado, em capítulos anteriores, de que maneira as faces tridimensionais são representadas e modeladas através de computadores, tal como qualquer outra superfície mais simples. Nesse capítulo, as malhas poligonais de faces utilizadas serão apresentadas, cabendo um pequeno comentário de sua origem. Existem diversos sistemas para modelagem tridimensional, dentre eles 3D Studio (<http://www.autodesk.com/>), Maya (<http://www.alias.com/>), etc. No entanto, esses sistemas não se aplicam especificamente a faces, mas sim a superfícies poligonais em geral. Um sistema específico para a

modelagem de faces bastante conhecido é o *FaceGen Modeller*, da *Singular Inversions Inc.* (<http://www.facegen.com/>). Com ele é possível a criação de malhas poligonais que se assemelhem a faces de usuários, através da extração de características a partir de imagens do mesmo e do estabelecimento de pontos de controle. Tal sistema possui a opção de exportar a malha gerada, em um formato conhecido, o qual pode ser interpretado por outros programas, permitindo a utilização das malhas por ele criadas. As figuras apresentadas neste e nos próximos capítulos contém malhas que foram geradas por este sistema, sendo deformadas utilizando as técnicas a seguir apresentadas.

4.1 Modelos Paramétricos

A superfície facial é tipicamente descrita como uma malha de polígonos, geralmente em 3D. Durante a animação, essa superfície é deformada através do movimento dos vértices dessa malha, mantendo sempre sua topologia (*i.e.* estrutura e conexão dos pontos) constante. O movimento dos vértices é então governado por um conjunto de parâmetros de controle [Beskow, 2003]. Essas transformações espaciais podem ser entendidas como funções aplicadas sobre cada vértice \vec{x} , classificadas com base no seu escopo (região de influência), podendo ser tanto global quanto local. Uma transformação é dita global se $f(\vec{x})$ é dependente de todos os pontos de controle, com base em suas distâncias ao ponto \vec{x} . Do contrário, se $f(\vec{x})$ não depende de todos esses pontos, mas apenas dos que estão a uma certa distância de \vec{x} , a transformação é dita local [Wirth, 2000].

A chamada “parametrização de faces” permite um controle específico sobre diversas configurações faciais, sendo utilizada para afetar apenas determinadas regiões da face. A combinação desses parâmetros proporcionam uma gama considerável de expressões faciais com um custo computacional relativamente baixo [Noh and Neumann, 1998].

No entanto, devemos mostrar também que a animação paramétrica apresenta algumas desvantagens. Uma limitação é que a escolha dos parâmetros dependerá sempre da topologia da malha de polígonos da face, ou seja, uma parametrização genérica não é possível; resultando em outra limitação, que é a da realização de ajustes manuais para os valores dos parâmetros, a cada nova malha que se desejar animar. Existem outras estratégias paramétricas descritas na literatura. Porém, duas delas serão apresentadas

com maior nível de detalhamento.

4.1.1 *Radial Basis Functions*

O uso de funções de base radial, RBF's, proporciona deformações suavizadas com um comportamento facilmente controlável. Da sua definição, temos que RBF's são funções suaves e contínuas que realizam interpolações entre dados fornecidos, produzindo resultados contínuos. Também a geometria dos pontos de controle não necessita ter qualquer restrição, implicando em uma distribuição dos mesmos de forma esparsa e irregular. Além disso, o comportamento facilmente controlável pode ser empregado para se chegar aos requisitos desejados em uma animação [Wirth, 2000].

De um modo geral, o funcionamento das RBF's aplicado à animação facial é dado da seguinte maneira: pontos de controle são definidos na superfície facial em um estado inicial, com a face relaxada. Para se obter a animação de alguma característica (*e.g.* movimento do canto esquerdo da boca), é atribuído um ponto de controle (muitas vezes chamados *landmark*) para essa região, e se movimenta tal ponto para uma região alvo específica. A seguir, são feitos os cálculos das posições de todos os pontos que estiverem na vizinhança desse *landmark*, através da aplicação das fórmulas a serem apresentadas nesta seção. Assim, temos o estado inicial ou relaxado, e o final, contendo a movimentação do(s) ponto(s) de controle e da sua região de influência. Com esses dois estados (ou quadros), produz-se a animação através de uma técnica denominada *morphing*¹.

O mapeamento realizado entre as posições inicial e final dos pontos de controle pode ser descrito em termos de um vetor de coeficientes [yong Noh et al., 2000]. Esses coeficientes devem ser encontrados a cada movimento que se deseja realizar sobre a face. Os demais nós na região de influência desses coeficientes têm sua posição modificada com base nesses valores calculados.

A seguir serão descritas as equações que regem o funcionamento das RBF's, através de exemplos práticos e de resultados na própria animação facial.

¹Algoritmos que lidam com a transição de duas imagens digitais, pela busca de um alinhamento geométrico e pela interpolação das cores afim de produzir uma transição suavizada e realista.

Realizando Deformações²

RBF's proporcionam funções de mapeamento para cada ponto de controle em uma imagem, e pontos correspondentes em outra imagem, além de proporcionar a interpolação desse mapeamento em situações intermediárias. Uma transformação espacial em d dimensões, denotada $T(x, y)$, é composta de k funções de mapeamento ($k = 1, 2, \dots, d$) tal que

$$T(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_d(\vec{x})] \quad (4.1)$$

$f_1(\vec{x})$ representa a função que mapeia a primeira dimensão, $f_2(\vec{x})$ a segunda dimensão, e assim por diante. Nesta abordagem, cada função pode ser decomposta em dois componentes, um local e outro global, aplicados a cada dimensão, permitindo o surgimento de uma família de transformações a serem definidas na região de influência dos pontos de controle que se deseja. Dados n pares de pontos de controle (origem e destino), cada uma das k funções de mapeamento seguem a forma geral assim definida:

$$f_k(\vec{x}) = P_{mk}(\vec{x}) + \sum_{i=1}^n A_{ik}g(r_i) \quad (4.2)$$

nas quais $P_{mk}(\vec{x})$ (equação 4.3) é um polinômio com grau $m = 1$, podendo ou não estar presente no equacionamento, conforme o grau de influência global que se desejar obter. O uso de polinômios de maior grau ($m > 1$) pode causar um comportamento irregular [Wirth, 2000].

$$P_{1k}(\vec{x}) = a_{0k} + a_{1k}x + a_{2k}y + a_{3k}z \quad (4.3)$$

O último componente $g(r_i)$ é o somatório de diversas funções bases, cada qual com seu peso de influência (A_{ik}) sobre o resultado, onde r_i é calculada através da distância Euclidiana entre o ponto em questão \vec{x} , e o ponto atual do somatório \vec{x}_i , da seguinte maneira:

$$r_i = \|\vec{x} - \vec{x}_i\| \quad (4.4)$$

Portanto, a função de base radial $f_k(\vec{x})$ é uma combinação linear entre uma função radialmente simétrica $g(r_i)$ e um polinômio de baixo grau $P_{mk}(\vec{x})$.

Uma questão que deve surgir é quanto ao custo computacional envolvido com a realização de tantos cálculos numéricos que devem ser realizados para a simples transição de

²Seção baseada em [Wirth, 2000]

um ponto de controle. Porém, a abordagem que será mostrada representa esses cálculos, e de fato realiza-os sob a forma matricial, que além de visualmente ser mais fácil para o entendimento, possui diversos métodos numéricos já implementados para a resolução de tais cálculos, que na maioria das vezes envolve a resolução de sistemas lineares. Na Seção 5.3 estão mostradas as linguagens de programação envolvidas no desenvolvimento desse sistema, juntamente com bibliotecas específicas para o trabalho com matrizes e resolução de sistemas lineares.

Para o cálculo da transformação RBF em 3D são utilizados $n + 4$ coeficientes em cada dimensão, onde $\vec{x} = (x, y, z)$ [Wirth, 2000]. Os coeficientes da função $f_k(\vec{x})$ são restritos pelas seguintes condições de interpolação:

$$f_1(\vec{x}_j) = u_j, \quad f_2(\vec{x}_j) = v_j, \quad e \quad f_3(\vec{x}_j) = w_j \quad \text{onde } j = 1, \dots, n \quad (4.5)$$

juntamente com as definições de compatibilidade, que estabelecem:

$$\sum_{i=1}^n A_{ik} = \sum_{i=1}^n A_{ik}x_i = \sum_{i=1}^n A_{ik}y_i = \sum_{i=1}^n A_{ik}z_i = 0 \quad (4.6)$$

Como foi dito anteriormente, pode ser que o polinômio não esteja presente, assim seus coeficientes não precisarão ser calculados, tendo então um cálculo de n coeficientes. Utilizando ou não o polinômio $P_{mk}(\vec{x})$ todos os coeficientes são encontrados através da resolução do seguinte sistema linear:

$$W = L^{-1}Y \quad (4.7)$$

onde

$$L = \begin{bmatrix} G & P \\ P^T & 0 \end{bmatrix} \quad (4.8)$$

com 0 sendo uma matriz de tamanho 4×4 , e

$$Y^T = \begin{bmatrix} u_1 & u_2 & \dots & u_n & 0 & 0 & 0 \\ v_1 & v_2 & \dots & v_n & 0 & 0 & 0 \\ w_1 & w_2 & \dots & w_n & 0 & 0 & 0 \end{bmatrix} \quad (4.9)$$

a matriz formada pelos destinos dos pontos de controle u_1, \dots, u_n , v_1, \dots, v_n e w_1, \dots, w_n .

Os elementos componentes da matriz L estão nas matrizes

$$G = \begin{bmatrix} g(r_{11}) & g(r_{12}) & \dots & g(r_{1n}) \\ g(r_{21}) & g(r_{22}) & \dots & g(r_{2n}) \\ \vdots & \vdots & \ddots & \vdots \\ g(r_{n1}) & g(r_{n2}) & \dots & g(r_{nn}) \end{bmatrix} \quad (4.10)$$

e

$$P^T = \begin{bmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ y_1 & y_2 & \dots & y_n \\ z_1 & z_2 & \dots & z_n \end{bmatrix} \quad (4.11)$$

Porém, algumas vezes ocorre que a matriz G apresenta uma característica peculiar, resultado de uma alta entropia dos dados. O que interfere na resolução do sistema por matrizes, pela geração de uma matriz singular. Quando isso ocorre, pode-se utilizar como solução a adição de um “pequeno” parâmetro λ na matriz G , tão logo esta for calculada:

$$G = G + \lambda I$$

Esse termo geralmente é definido empiricamente, não alterando os resultados, apenas eliminando o problema da impossibilidade de se resolver o sistema para o cálculo dos coeficientes [yong Noh et al., 2000].

A matriz de coeficientes descoberta pela solução do sistema linear exposto na equação 4.7 é estruturada da seguinte forma:

$$W^T = \begin{bmatrix} A_{11} & A_{21} & \dots & A_{n1} & a_{01} & a_{11} & a_{21} & a_{31} \\ A_{12} & A_{22} & \dots & A_{n2} & a_{02} & a_{12} & a_{22} & a_{32} \\ A_{13} & A_{23} & \dots & A_{n3} & a_{03} & a_{13} & a_{23} & a_{33} \end{bmatrix} \quad (4.12)$$

sendo os últimos coeficientes a_{0k}, a_{1k}, a_{2k} e a_{3k} ($k = 1, 2, 3$) aqueles utilizados pelo polinômio de primeiro grau já comentado anteriormente. Observe que algumas matrizes estão no formato transposto apenas para melhorar a visualização; não modificando, portanto, o significado de seus valores.

Podemos também notar que o (i, j) -ésimo elemento de G , $g(r_{ij})$, tem o raio calculado por:

$$r_{ij} = \|\vec{x}_i - \vec{x}_j\| \quad i.e. \quad r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \quad (4.13)$$

Intuitivamente, $g(r_{ij})$ mede o efeito do j -ésimo ponto de controle na transformação do i -ésimo ponto. Ainda falta definirmos a função $g(r)$, que é chamada de função de base, o que adicionado ao termo radial (r), significa que dado um ponto de controle, os cálculos são feitos com base na distância (raio) desse ponto até os demais da sua vizinhança. Essa função, $g(r)$, aplicada a cada distância calculada, pode ser de diversos tipos, conforme o próprio comportamento que desejamos atribuir à deformação a ser aplicada à malha de pontos da face. A seguir, a apresentação de algumas das diversas funções de base mais utilizadas.

Funções de Base

Como já dito, a estratégia das RBF's se utiliza de uma função, chamada “de base”, a qual recebe por entrada a distância r entre os pontos de controle da malha. Algumas delas crescem a medida em que r cresce; outras, decrescem. A escolha da função mais apropriada é realizada por fatores como a dimensão do problema, as condições de interpolação e as propriedades desejadas para a própria interpolação. As funções aqui descritas podem ser aplicadas a quaisquer dimensões; mas vale salientar que algumas funções são específicas para determinadas dimensões, como algumas para 2D, por exemplo, mas estão fora do escopo deste trabalho e por isso não serão citadas.

O comportamento de algumas funções está mostrado na Figura 4.1, e suas descrições e definições estão mostradas a seguir:

Multiquadrática: Abreviada MQ, tem a forma geral

$$g(r) = (r^2 + \delta)^\mu \quad (4.14)$$

com $\delta > 0$ e $\mu > 0$. Essa função incorpora um parâmetro de localidade, δ , o qual é utilizado para controlar seu comportamento, dando menos peso para a distância entre os pontos de controle e mais para os da vizinhança próxima.

Multiquadrática Inversa: Assume a mesma forma da MQ, porém com a condição de $\mu < 0$. Seu comportamento torna-se similar ao da função Gaussiana, com o parâmetro σ se aproximando de zero para a resposta se tornar mais localizada, e aumentando de valor para aumentar a região de influência dos pontos de controle.

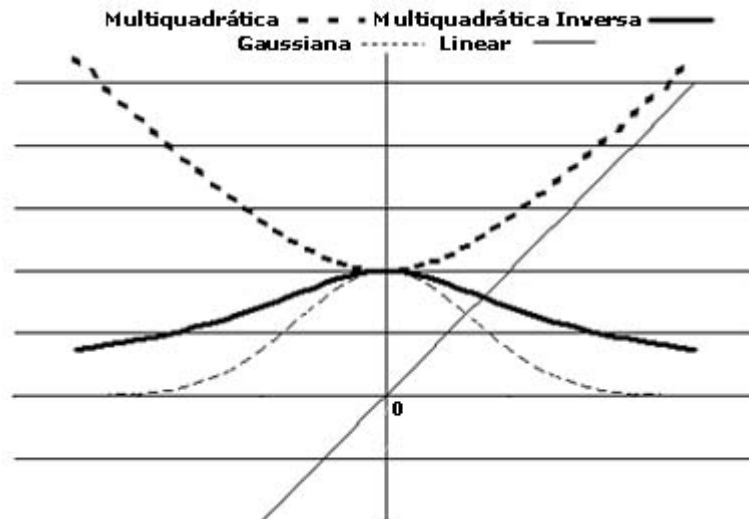


Figura 4.1: Algumas Funções de Base existentes

Gaussiana: Essa função é dada por

$$g(r) = e^{-r^2/\sigma} \quad (4.15)$$

e à medida em que o valor de σ cresce, a contribuição de pontos mais distantes aumenta, produzindo uma aproximação suavizada e mais global. Se σ diminui, as contribuições da vizinhança próxima aumentam, produzindo aproximações mais localizadas e, portanto, mais bruscas.

Linear: Tal função não apresenta aplicabilidade direta para a animação facial pelo fato de não contemplar as características de suavização e influência nas proximidades dos pontos de controle, o que produziria resultados não satisfatórios.

Exemplos

De modo a entendermos de maneira mais clara a funcionalidade das RBF's, podemos mostrar a Figura 4.2, em que se expõe uma malha de polígonos com diversos pontos de controle em duas posições: uma inicial, relaxada, e outra final, com os deslocamentos realizados pelos pontos de controle e aplicados à vizinhança.

Passando para o domínio facial, podemos mostrar a aplicação das RBF's em malhas de polígonos representando faces, e para isso devemos ter os pontos de controle

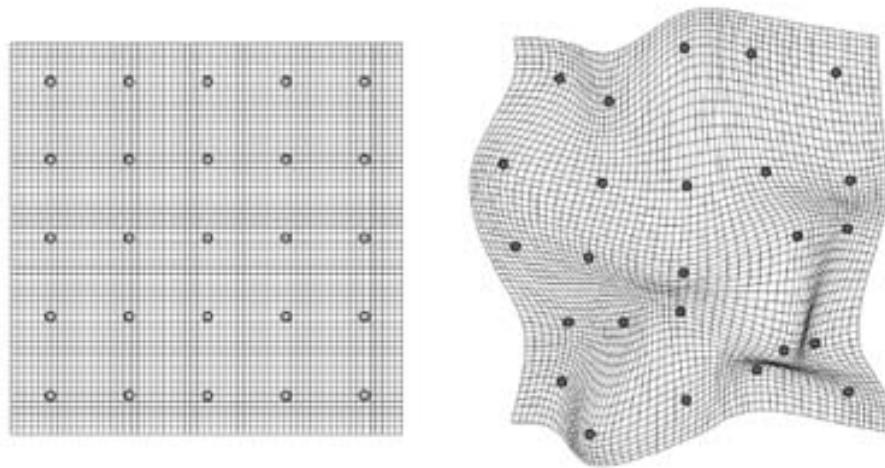


Figura 4.2: Malha contraída e pontos de controle [Edge and Maddock, 2003]

pré-estabelecidos para, ao movê-los, aplicar os resultados sobre toda a vizinhança. Na Figura 4.3 apresentamos uma malha poligonal de uma face, com os pontos de controle já definidos. Observe que a localização desses pontos é baseada no padrão MPEG-4, embora a abordagem atual não implemente a totalidade de pontos desse padrão.

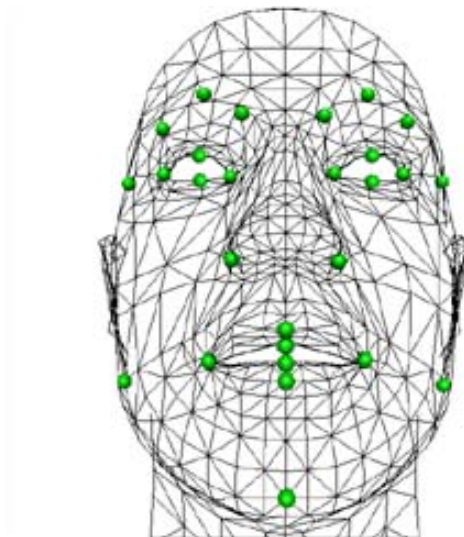


Figura 4.3: Pontos de controle em uma malha facial

Com os pontos de controle já estabelecidos, nos resta saber qual a região de interesse para cada um deles, em particular, tendo em vista que alguns pontos não podem influen-

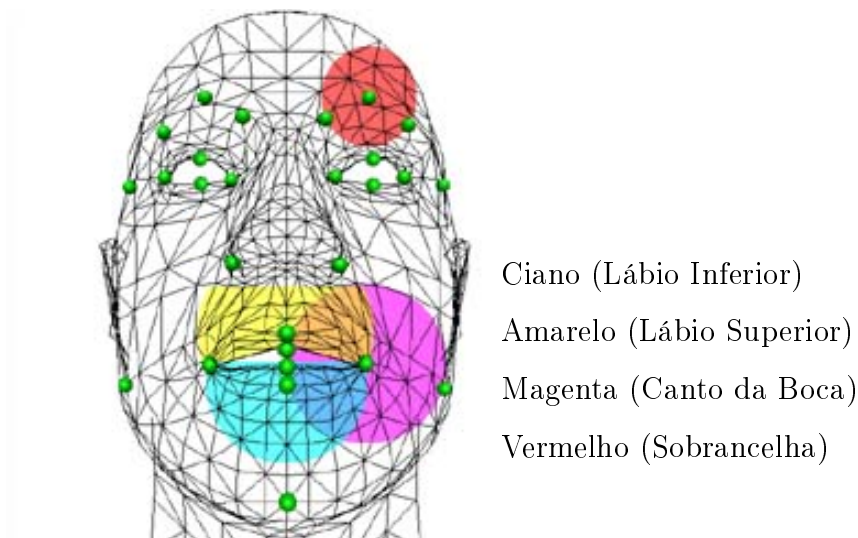


Figura 4.4: Regiões de atuação de alguns pontos de controle

ciar toda a região circunscrita em um determinado raio. Para compreendermos melhor, basta lembrarmos que a dinâmica de um ponto estabelecido sobre o lábio superior não pode influenciar quaisquer pontos presentes no lábio inferior, mesmo que estejam mais próximos do que pontos presentes na parte superior do lábio. As principais regiões de influência estão mostradas na Figura 4.4

De posse desses pontos de controle, e das regiões de atuação de cada um deles, podemos aplicar então a técnica de RBF's para realizar a deformação, resultando em uma nova expressão facial. Em um primeiro exemplo (Figura 4.5) aplica-se o deslocamento dos dois cantos da boca para cima, da mesma forma que no exemplo da Figura 4.11, animado pelo modelo de Waters.

No segundo exemplo, apresentado na Figura 4.6, temos a aplicação das RBF's para simular o movimento da boca, apenas com o deslocamento de dois pontos de controle, conforme mostrado na imagem. Podemos observar um pequeno orifício presente nos dois cantos; embora não prejudique consideravelmente a apresentação do resultado final, deve-se esclarecer que permaneceu aberto devido à utilização de pontos de controle apenas nas regiões periféricas da boca. A intenção disso é a de reduzir o cálculo para cada quadro, tendo em vista que o custo computacional deve sempre ser levado em consideração.

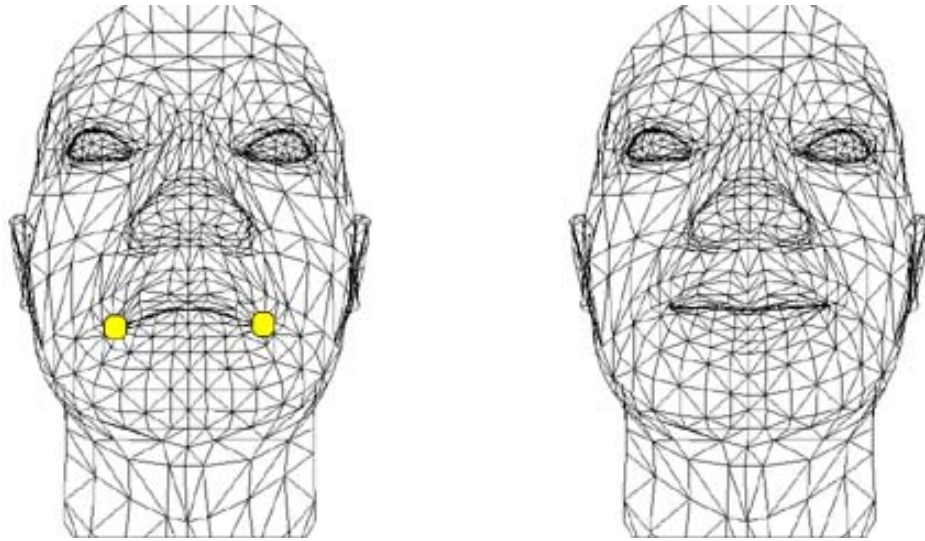


Figura 4.5: Face modificada pelas funções de base radial

Outras Aplicações

Podemos citar também algumas outras áreas nas quais o uso de RBF's tem resultados satisfatórios, como no processo de *morphing*, utilizado em 2D e 3D, tanto para a animação quanto para a geração de novas formas geométricas. Além disso, outra área de aplicação é a interpolação entre camadas planas em volumes tridimensionais, com grande utilidade para a área médica. Também na medicina, aplicações com RBF's incluem planejamento de cirurgias através da reconstrução de superfícies ausentes, sendo uma alternativa bastante utilizada em soluções para o problema de interpolação de superfícies incompletas derivadas de imagens médicas tridimensionais [Wirth, 2000].

Limitações

Devem-se expor algumas limitações do método, também como forma de apresentar uma avaliação geral da técnica que foi adotada para ser utilizada na implementação do sistema exposto neste trabalho.

No caso de funções de base RBF's como a MQ, a seleção do parâmetro de localidade (δ) é fundamental para se conseguir bons resultados, o que pode algumas vezes ser um trabalho demorado, pois os pontos são escolhidos, os cálculos são realizados, e devem-se observar manualmente os resultados para sua validação.

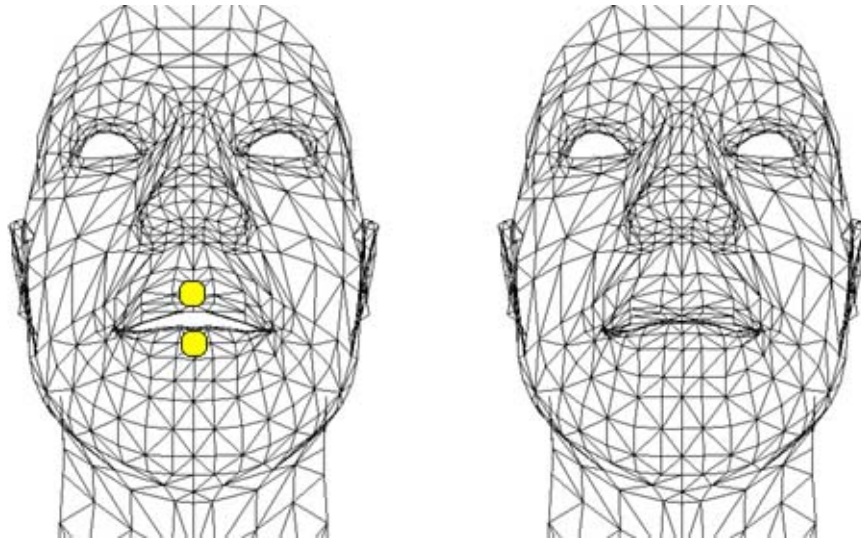


Figura 4.6: Movimento de pontos de controle da Boca aplicados à vizinhança

A seleção de pontos de controle apropriados também é crítica para o sucesso do funcionamento das RBF's. Uma das maiores desvantagens do uso de RBF's é o aumento considerável na complexidade computacional quando se aumenta a região de influência de um determinado ponto de controle. Assim, já se antecipa o comentário a respeito do sistema implementado, que no estado atual, não está capacitado, devido a esta limitação, a realizar a animação em tempo real, apenas após o cálculo das equações a cada movimento que se deseja realizar sobre um ou mais pontos específicos da face.

A solução adotada na maioria dos casos é a aplicação de funções com limites mais localizados, para reduzir a gama de pontos processados e, conseqüentemente, o processamento. Os limites aplicados à malha facial, particularizados a cada região, estão ressaltados na Figura 4.2.

4.1.2 Modelo de Waters

Também conhecido por (*Muscle Based Animation*), foi primeiramente implementado por Waters, em 1987 [Waters, 1987]. Esse modelo se baseia na construção de músculos virtuais, interagindo com a malha representativa da face, sendo ainda foco das pesquisas que visam incrementá-lo na busca de animações mais fidedignas. Nessa estratégia, cada músculo é representado como um vetor (presente em algum ponto da face) que, ao ser

aplicado, desloca na sua direção os demais vértices das regiões vizinhas, de acordo com sua zona de influência, definida por uma série de parâmetros.

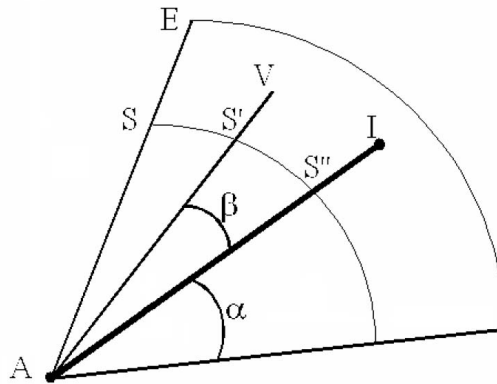


Figura 4.7: Modelo baseado em [Waters, 1987]

Podemos perceber, de acordo com a Figura 4.7, os seguintes pontos:

- um ponto de controle A que pode ser entendido como a raiz do músculo – de um ponto de vista biológico, como se estivesse conectado ao osso;
- um ponto de inserção I (ponto de controle também) na pele;

O músculo então pode ser considerado como um vetor \overrightarrow{AI} . Quando é simulada a sua contração, os pontos de controle não se movem e o vetor \overrightarrow{AI} mantém seu comprimento. Esse vetor age como uma “atração magnética” sobre os vértices ao redor de sua zona de influência. Tal modelo é complementado com os seguintes parâmetros:

- um ângulo de abertura α , que estabelece um limite angular, de modo que um vértice \overrightarrow{V} , com ângulo β , esteja na zona de influência do músculo;
- duas distâncias radiais, S e E , as quais determinam áreas de influência. Nessa implementação, S e E são proporções do vetor \overrightarrow{AI} .

O equacionamento é aplicado a cada vértice V que estiver contido na zona de influência estabelecida por um determinado músculo, gerando um vetor deslocamento $\delta\overrightarrow{V}$ a ser somado na posição do referido vértice. A direção deste deslocamento deve ser oposta à

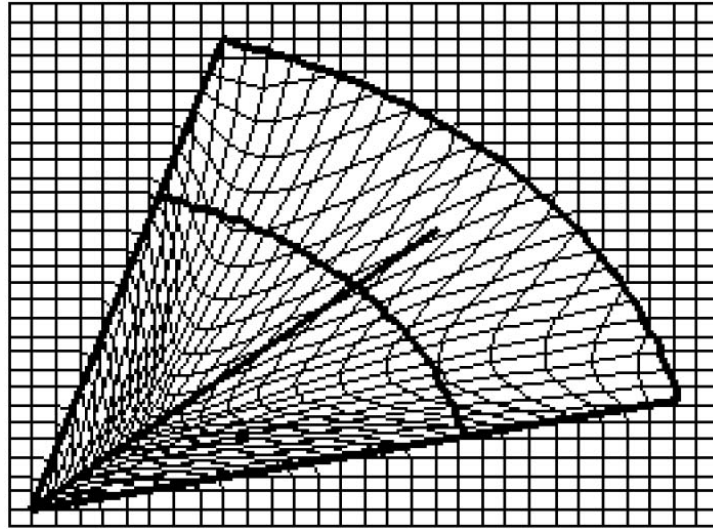


Figura 4.8: Exemplo de malha contraída [Waters, 1987]

do vetor \overrightarrow{AV} , o que conseguimos perceber pela própria descrição do método; e o módulo é delimitado pela constante C , pré-definida, da seguinte maneira:

$$\delta \overrightarrow{V} = -C \cdot \delta \cdot \overrightarrow{AV} \quad (4.16)$$

$$\delta = \begin{cases} \delta A \cdot \delta R, & \text{se } |AV| > |AS| \\ \delta A & \text{demais casos} \end{cases} \quad (4.17)$$

onde:

$$\delta A = \frac{\cos \beta - \cos \alpha}{1 - \cos \alpha} \quad (4.18)$$

$$\delta R = \cos \left(\frac{|\overrightarrow{S'V}| \pi}{|\overrightarrow{SE}| 2} \right) \quad (4.19)$$

Podemos ver um exemplo de contração muscular aplicada a uma malha de pontos genérica, na Figura 4.8. Observemos as zonas de influência determinadas pelos parâmetros radiais (S e E): a área mais externa ao efeito da contração (setor limitado pelos raios S e E) é atenuada com o intuito de suavizar o produto final, o que torna a implementação mais fiel.

A animação de sistemas musculares é interessante por diversas razões. Como cada expressão facial pode ser reduzida a um grupo de contrações musculares, elas geralmente requerem menos parâmetros do que outros modelos [Breton et al., 2001]. A elasticidade da pele é geralmente modelada pela própria contração do músculo e esse é um dos maiores ganhos computacionais trazidos por essa estratégia; algoritmos de elasticidade geralmente requerem a resolução de sistemas de equações diferenciais.

Uma limitação a ser estudada no modelo de contração muscular se dá no momento em que se deseja associar ações musculares. Isso não pode ser feito com a simples adição de vetores deslocamento, pois os resultados podem acabar puxando determinados vértices além dos seus limites físicos. Waters propôs uma estratégia de se atribuir a cada vértice um Grau de Liberdade – *Degree of Freedom* (DOF). Esse DOF seria uma porção do espaço que limita os deslocamentos de cada vértice fora dos quais resultariam em efeitos desagradáveis [Waters, 1987].

Uma outra tentativa para solucionar o problema gerado pela associação de músculos é a de agrupá-los de acordo com suas funções e orientações [Breton et al., 2001]: músculos frontais esquerdos e direitos e músculos labiais superiores e inferiores. Assim, quando um movimento de mais de um músculo dos grupos acima for solicitado, somente o de maior módulo será efetuado, preservando assim os limites da elasticidade facial. A proposta inicial de Waters continha uma pequena quantidade de músculos (Figura 4.9), com os quais já se conseguiu atingir objetivos bastante relevantes para a área de animação facial.

Exemplos

Como a técnica baseada no modelo de Waters foi implementada, mostram-se agora alguns resultados em uma malha regular quadrada (Figura 4.10) e em malhas de faces (Figura 4.11).

Já foi dito que esse modelo é baseado na inserção de alguns pontos de controle, sendo assim, exemplificamos a colocação dos dois pontos *A* e *I* na malha quadrada e a utilização de duas diferentes angulações, visando mostrar a adaptação da técnica para vários tipos de casos. As duas angulações aplicadas sobre a malha quadrada estão mostradas na Figura 4.10.

Ao ver esses exemplos, podemos extrapolar para o propósito desse projeto: a face

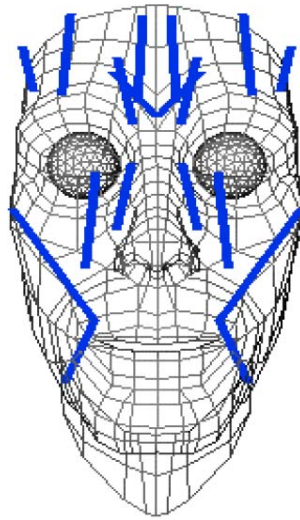


Figura 4.9: Músculos utilizados pela proposta de Waters [Noh and Neumann, 1998]

humana. Para simular a expressão de um sorriso, basicamente realizamos deslocamentos para cima dos cantos da boca em ambos os lados. Pensando no sistema de Waters, poderemos inserir músculos nas regiões indicadas e realizar a “contração” desses músculos, baseados nos parâmetros estabelecidos. Tanto os pontos de controle quanto o resultado da expressão são mostrados na Figura 4.11.

4.2 Modelos Baseados em Leis da Física

Tal técnica, embora não utilizada na implementação do sistema proposto, merece também uma descrição mais detalhada, pelo fato de outras implementações existentes utilizarem-na em sua integridade ou, em alguns casos, de parte de sua teoria, caracterizando os modelos híbridos encontrados na literatura.

Os modelos faciais baseados em leis da Física tipicamente utilizam um sistema massa-mola (*mass-spring*) ou rede de elementos finitos para simular as propriedades da pele. Uma implementação derivada dessa categoria (Figura 4.12) nos mostra um modelo contendo três camadas (tecido cutâneo, adiposo e músculos), todos conectados através de molas [Kähler et al., 2001].

Uma abordagem semelhante é a multi-camadas, simulando as três camadas básicas da face (*i.e.* ossos, músculos e pele) e a interface entre elas, como mostrado na Figura

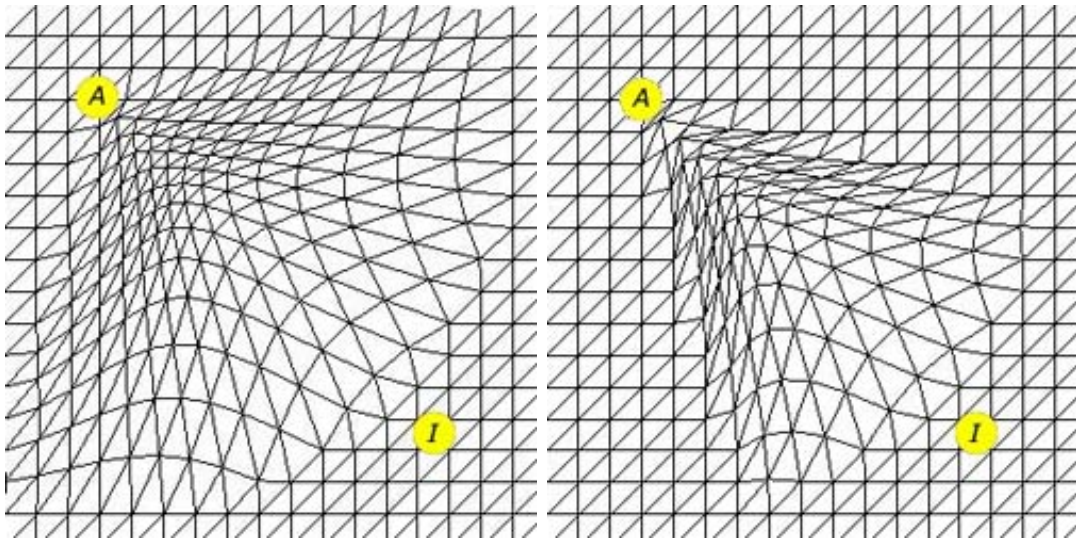


Figura 4.10: Malhas quadradas contraídas, com ângulos de abertura de $PI/2$ e $PI/6$

4.13. Nessa estratégia há que se levar em conta a impossibilidade da sobreposição de camadas, na medida em que a ordem dos elementos (*i.e.* camadas) deverá ser preservada na geração das expressões e movimentos. Assim, quando uma malha de pontos da pele é modelada como uma simples membrana, ela pode penetrar as camadas musculares e ósseas quando esticada em demasia [Kähler et al., 2001]. Sendo assim, métodos para a preservação volumétrica e limítrofe das camadas devem sempre ser levados em consideração.

A teoria que rege as técnicas baseadas em leis da Física integra a manipulação física e geométrica do modelo facial (Figura 4.14) de maneira bastante coerente [Qin and Terzopoulos, 1996]. O processo de modelagem e animação também é facilitado pelo uso de energias, forças e heurísticas de restrições. O algoritmo pode aplicar forças variantes no tempo para modificar a face interativamente.

A complexidade computacional e o conhecimento necessário da anatomia do corpo humano, fundamentais para a simulação baseada em leis da Física, são razões para que essa abordagem não fosse utilizada na implementação deste projeto.

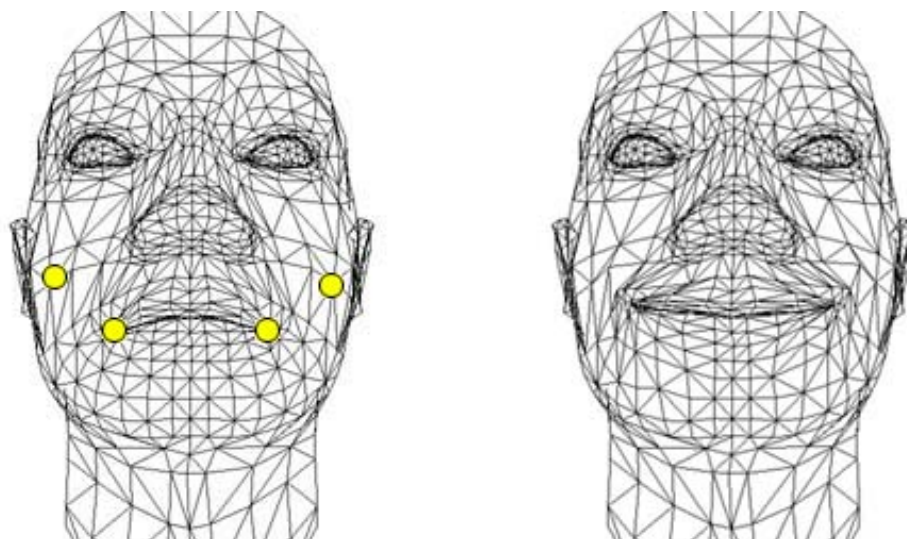


Figura 4.11: Face modificada pela técnica de Waters

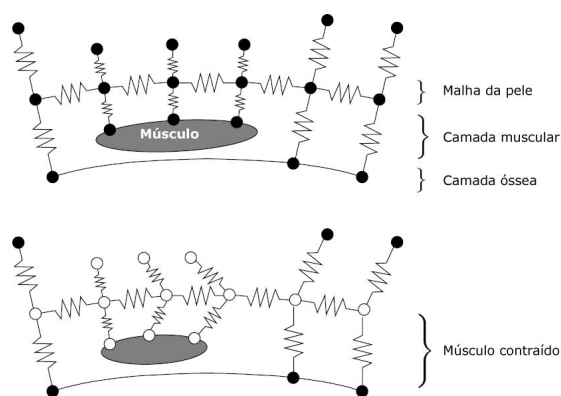


Figura 4.12: Sistema massa-mola aplicado à Animação Facial

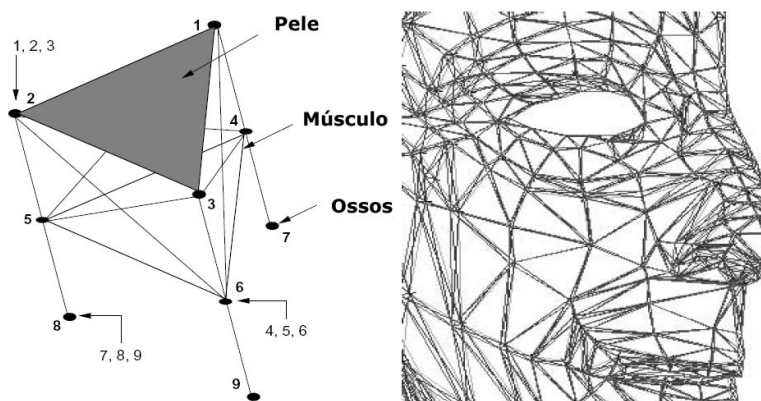


Figura 4.13: Modelo multi-camadas para Animação Facial [Lee et al., 1995]



Figura 4.14: Interface do Modelo Facial Físico

4.3 Outras Abordagens

Existem diversas abordagens já propostas na bibliografia para a animação facial. Nesse momento, algumas das mais utilizadas serão brevemente discutidas, para que se conheça sua existência, possibilitando buscar maiores referências no caso de querer implementá-las. No entanto, essas técnicas não foram as escolhidas para a implementação desse projeto, não merecendo tanto destaque, portanto, quanto as demais já citadas.

Deve-se lembrar também que classificações exatas de abordagens para a animação facial são complicadas, pela falta de limites claros entre os métodos, além do fato de que estratégias mais recentes geralmente integram diversos métodos na tentativa de produzir melhores resultados.

4.3.1 *Free Form Deformation*

A abordagem *Free Form Deformation* (FFD) realiza uma deformação volumétrica de objetos através da manipulação de pontos de controle pré-estabelecidos, posicionados em um cubo exterior ao objeto que se deseja deformar [Sederberg and Parry, 1986]. Conceitualmente, um objeto flexível é inserido nesse cubo, contendo uma grade tridimensional de

linhas e pontos que, quando modificadas, aplicam o resultado sobre o elemento contido em seu interior.

Quando aplicado a faces, um paralelepípedo de controle é definido sobre a região de interesse. As deformações na pele correspondem às ações de esticar, achatar, expandir e comprimir o volume pertencente ao controle, através do deslocamento dos pontos de controle e modificação dos pesos associados com cada um deles [Noh and Neumann, 1998].

4.3.2 Interpolação

Um dos métodos mais diretos e provavelmente o mais comum, a interpolação é bastante aplicada à animação facial em 3D. O princípio básico é ter um conjunto de formas faciais pré-definidas, geralmente correspondendo aos visemas e/ou a determinadas expressões faciais. Para cada uma dessas formas, a posição dos vértices é guardada, mantendo sempre a topologia constante. A animação é feita pela estratégia de *key-framing* (quadro a quadro), sendo os quadros intermediários (*i.e.* que contém as novas posições dos vértices) determinados por técnicas de interpolação [Beskow, 2003].

Capítulo 5

Sistema Computacional

Cabe a este capítulo uma descrição detalhada sobre o sistema desenvolvido durante a execução deste projeto. Ao longo das próximas seções, serão vistas algumas características básicas que regeram as decisões de projeto tomadas para sua implementação, tais como linguagens de programação e bibliotecas utilizadas, além de um breve manual de utilização do sistema, para que o leitor possa entender os diversos detalhes que regem o funcionamento do sistema como um todo.

5.1 Diagramas

Alguns diagramas serão expostos como forma de melhor expor as funcionalidades realizadas pelo sistema. Primeiramente, a figura 5.1 apresenta uma visão geral das atividades necessárias para interagir com o sistema e os impactos causados sobre cada componente. Cabe ressaltar algumas definições já presentes nesta figura, em relação aos termos utilizados. O controle da animação presente no diagrama diz respeito às funcionalidades presentes em um *player* de mídia, com opções de “Iniciar”, “Pausar” e “Continuar” a animação, além da alteração da velocidade da mesma, com o intuito de se perceber certos detalhes em uma exposição mais lenta ou aumentar a velocidade para uma visão mais superficial do todo.

Algumas decisões de projeto apontaram para a criação de arquivos contendo diversos parâmetros que podem ser alterados pelo usuário como, por exemplo, as Posições dos Pontos de Controle (PPC's), Emoções em Pontos de Controle (EPC's) e as Dinâmicas

dos Pontos de Controle (DPC's), que agem sobre a malha poligonal presente no arquivo que contém a definição de um objeto (por isso a extensão OBJ). Os PPC's representam a configuração da posição tridimensional dos pontos de controle sobre uma determinada face. Os EPC's contém algumas configurações, no mesmo padrão dos DPC's (comentados a seguir), necessárias para a movimentação dos pontos de controle de modo a animar a face com algum tipo pré-definido, em uma abordagem que mistura a construção de emoções pelo sistema FACS, aplicadas a alguns pontos de controle presentes na abordagem MPEG-4. Por fim, os DPC's são os que possuem a dinâmica dos pontos de controle, em cada quadro, para o sistema realizar as deformações na face padrão para esses novos estados, e através das técnicas de *morphing* animar de maneira suavizada.

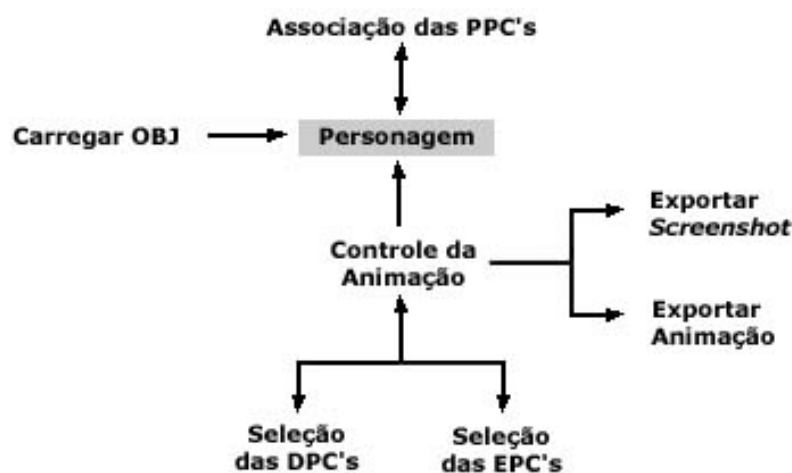


Figura 5.1: Diagrama de atividades realizadas pelo sistema

Na Figura 5.2 é apresentado o fluxograma que rege o funcionamento do sistema. Sobre esse diagrama, deve-se ressaltar a ocorrência da decisão intitulada “Botão OK”, o qual libera o programa para instanciar os objetos relacionados às regiões de influência de cada um dos pontos de controle, conforme especificado no Capítulo 4.

5.2 Fonte dos Movimentos

Para os experimentos realizados e exibidos como exemplo, configurou-se um ambiente propício para a captação dos movimentos de um personagem real através de filmagens

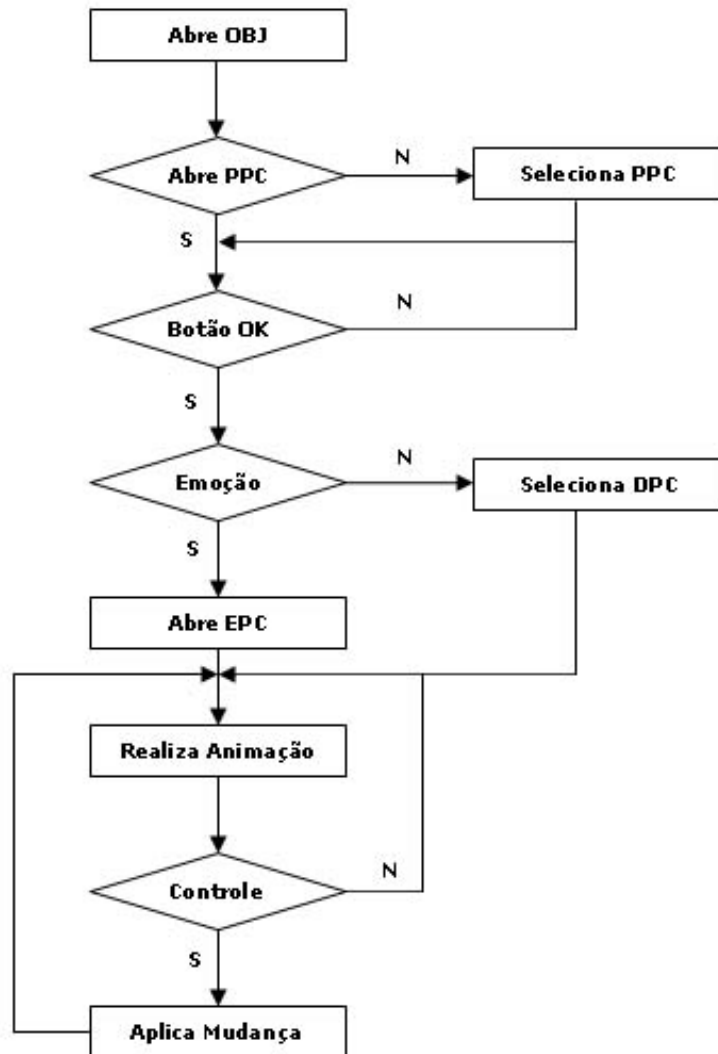


Figura 5.2: Fluxograma do sistema

e a extração de suas características a serem animadas (*e.g.* olhos, movimentos labiais, etc.). A Figura 5.3 mostra esse ambiente, montado em um fundo de cor padrão (azul, para a aplicação da técnica de cromaqui) para a extração das características e de sua dinâmica, realizada pelo trabalho de [da Silva and Costa, 2005]. A adoção de um fundo simples reduz significativamente a complexidade da etapa de localização da face.

Na medida em que a idéia é a utilização de dispositivos de baixo custo, foram utilizadas câmeras simples (*webcams* D-LINK DSBC-110) para a captação das imagens, armazenadas em arquivos do formato *Audio/Video Interleaved* (AVI). A resolução adotada foi a de 352 x 288 com um *framerate* (taxa de *frames*) variável em função das características da câmera,



Figura 5.3: Ambiente das filmagens

influenciadas pela iluminação.

Como já dito, o sistema é baseado em arquivos de configuração, contendo as posições dos pontos de controle, além das dinâmicas a serem aplicadas a cada ponto em cada unidade de tempo. Essa dinâmica é exposta em arquivos no formato DPC's, no seguinte padrão:

```

2                # Total de quadros

1.0  2.0  0.0   #
1.1  0.0  0.5   # Primeiro Quadro
...                #
0.0 -3.0  3.0   #

2.0  2.0  0.0   #
0.0 -1.0  3.0   # Segundo Quadro
...                #
0.0 -3.0  3.0   #

```

no qual a primeira linha indica o total de estados (ou quadros) a serem simulados pelo personagem, e as linhas posteriores representam a movimentação de todos os n pontos de controle em cada um dos 2 quadros para o exemplo em questão. A movimentação de cada ponto de controle é expressa com base no estado inicial dos pontos de controle e

seu deslocamento relativo a essa posição em cada quadro. Um exemplo da extração dos pontos de controle, restritos à boca para fins de ilustração, é mostrado na Figura 5.4, e o resultado da animação com base na extração desses pontos de controle está na Figura 5.5



Figura 5.4: Exemplo de extração dos pontos de controle da boca

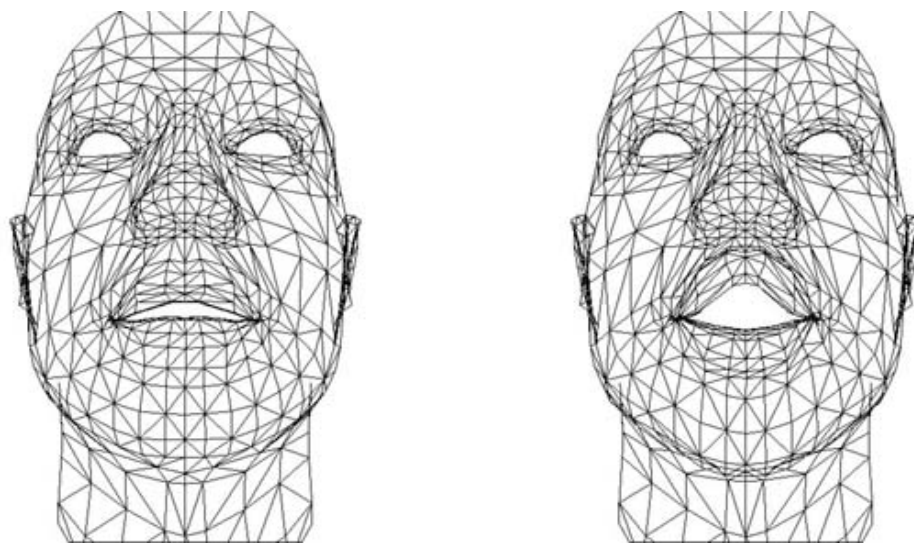


Figura 5.5: Resultado da animação dos pontos de controle da boca

5.3 Linguagens

JAVA [(SDN), 2005] é uma linguagem de programação relativamente nova, de alto nível, com sintaxe similar à C++. Apresenta diversos aspectos que a tornam atraente aos programadores das mais diversas áreas de pesquisa. Podemos citar alguns:

Orientação a objetos

Observando códigos implementados, podemos perceber as grandes vantagens da abordagem orientada a objetos, permitindo o uso de todas as potencialidades desse paradigma de programação para a construção de um código simplificado e reutilizável.

Exceções

JAVA possui a abstração de máquina virtual, que faz uma verificação em tempo de execução quanto aos acessos de memória, abertura de arquivos e uma série de outros eventos, dando a possibilidade ao programador de tratá-las em particular, da maneira mais conveniente.

Independência de plataforma

Talvez a característica decisiva para a escolha dessa linguagem seja essa abordagem multiplataforma, tendo em vista que novos Sistemas Operacionais estão ganhando destaque, especialmente os sistemas de Software Livre, além de futuramente, em uma proposta mais ousada, vir portar uma versão simplificada do sistema para arquiteturas móveis, como telefones celulares e *handhelds*.

Ampla documentação e fácil acesso

No próprio *site* da empresa mantenedora da linguagem (<http://java.sun.com/>), encontra-se uma infinidade de documentação que contempla grande parte das APIs¹ disponíveis para a utilização dos programadores, além de ferramentas de fórum e listas de discussão que englobam muitos problemas, dúvidas e soluções propostas por programadores ao redor do mundo (<http://forum.java.sun.com/>).

Bibliotecas gráficas para animação

Dá-se particular atenção à API chamada JAVA3D, um conjunto de funções específicas para a animação em computação gráfica, também com ampla documentação, através de

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

tutoriais, fóruns e referências às API's.

Garbage Collector

Os programadores não necessitam preocupar-se com o gerenciamento de memória. JAVA, utiliza-se de um conceito de *garbage collection* (coleta de lixo), cuja função é a de varrer a memória de tempos em tempos, liberando automaticamente os blocos que não estão sendo utilizados.

Geração de aplicações baseadas na *web*

JAVA possui um diferencial com relação a outras linguagens de programação no que diz respeito a sua amplitude de utilização; com ela podem ser implementados sistemas para a Internet, executados sob a forma de *applets*.

Livre para ser utilizada

JAVA está disponível para *download* sem qualquer obrigação de cadastramento ou registros, sendo também disponibilizadas IDE's² gratuitamente.

A linguagem JAVA possui bibliotecas diversas, que servem como apoio para o programador. Dentre elas podemos citar JAVA3D (como citado acima), que é uma interface criada para o desenvolvimento de aplicações gráficas tridimensionais em JAVA, executada no topo de bibliotecas gráficas de mais baixo nível, tais como OpenGL e Direct3D [Bicho et al., 2002].

Assim, através da percepção das diversas características encontradas na linguagem JAVA, favoráveis ao desenvolvimento de aplicações deste tipo, optou-se por utilizá-la na criação de um sistema que atenda, na medida do possível, às necessidades encontradas pela comunidade de usuários.

²*Integrated Development Environment* (IDE) – ambiente de programação do qual o programador se utiliza para implementar um programa de computador

5.4 Manual

Nesta seção é apresentado o sistema computacional desenvolvido para realizar a animação facial, baseado nas técnicas explicadas em capítulos anteriores. Sua descrição será feita através de algumas imagens de seu funcionamento (*screenshots*), seguindo o fluxo de execução mencionado pelos diagramas presentes na Seção 5.1.

Na Figura 5.6 é apresentada a primeira tela do sistema.

Como já mencionado, primeiramente deve-se fazer a importação da malha de polígonos contida em arquivos do formato OBJ, para a seguir proceder com a adaptação dos pontos da face à malha que for importada.



Figura 5.6: Tela inicial do sistema

Após a abertura da malha de polígonos, o usuário deve realizar a associação entre os pontos de controle da imagem 2D presente no canto inferior esquerdo da tela com o modelo 3D, e ao final, pressionando o “botão OK”. Após esse momento, pode-se utilizar uma “Emoção” padrão, presente no menu lateral, ou importar uma dinâmica de movimento, com arquivos DPC’s.

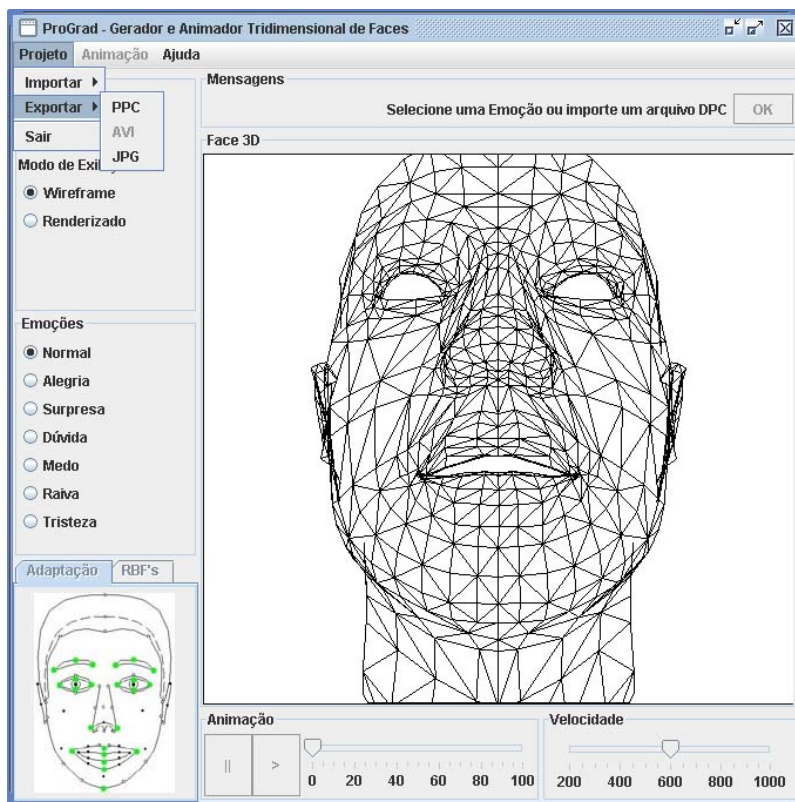


Figura 5.7: Funcionamento do sistema

Além do supracitado, percebemos no sistema algumas funcionalidades que servem para configurar o ambiente e permitir um controle maior sobre a animação por parte do usuário. Tais funções incluem a exibição ou não dos pontos de controle, seleção do modo de exibição (apenas malha de polígonos – *wireframe*, ou renderizado), configuração dos parâmetros das RBF's (o tipo de função utilizada, e o parâmetro principal envolvido no cálculo) e também funções para exportar alguns resultados, tanto no formato JPG (imagem estática) quanto no formato AVI (animação), além do próprio arquivo contendo PPC's, para facilitar o usuário ao reutilizar a malha poligonal posteriormente.

Capítulo 6

Considerações Finais

A animação facial possui uma ampla gama de aplicações abrangendo, por exemplo, cinema, jogos eletrônicos e educação. Além disso, constitui-se em uma alternativa promissora para a compressão de dados em ambientes de vídeo conferência [Martino, 2005]. Essa linha de pesquisa não é trivial dentro da Computação Gráfica. Além de englobar diversas áreas do conhecimento humano (por exemplo, inteligência artificial, matemática, psicologia e anatomia humana), o estudo dessa área impulsiona o desenvolvimento científico e tecnológico, tanto nas áreas computacionais quanto naquelas das quais o seu estudo depende [Körting et al., 2005b].

Um sistema de animação facial no espaço 3D foi proposto. Diversas técnicas para a realização desse tipo de animação foram expostas, dando particular atenção às paramétricas, que através da escolha de pontos de controle e suas respectivas dinâmicas (seus parâmetros), realizam deformações sobre uma malha poligonal que representa uma face a ser animada. Dentre as estratégias paramétricas estudadas, a saber, RBF's e Waters, a escolhida para a implementação do sistema foi a RBF's, pela suas características favoráveis, especialmente a facilidade quando da manipulação de pontos de controle e aplicação nas vizinhanças pré-determinadas. O nome "Funções de Base Radial" deriva do fato de que uma classe particular de funções podem ser aplicadas a um somatório de elementos multiplicados por pesos individuais, elementos esses gerados pela distância euclidiana entre os pontos de controle e cada ponto influenciado por esse mesmo ponto de controle. Também foi exposta a principal limitação das RBF's, o fato de que a cada alteração de um ponto de controle, os pesos (ou coeficientes) devem ser recalculados o que, com o poder

de processamento atual, torna essa aplicação ainda incapaz de ser executada em tempo real.

6.1 Dificuldades Encontradas

Deve-se sempre pensar nas dificuldades como maneiras que nos são apresentadas e que, ao superá-las, nos fazem crescer mais do que se as tais dificuldades não tivessem surgido. É com base nesse pensamento que se devem expor alguns problemas encontrados durante a implementação desse projeto.

A maioria das ferramentas e técnicas mais práticas, envolvidas na construção deste sistema, não foram vistas (e realmente nem deveriam por ser muito específicas) em disciplinas disponíveis no curso de Engenharia de Computação, tornando a atividade da construção do projeto de graduação como um todo uma grande experiência de pesquisa. As desvantagens e problemas encontrados nas técnicas escolhidas estão apresentados diretamente nas seções que as definem.

Outro fator que dificultou, mas felizmente não impediu, a implementação foi o fato do contato em âmbito acadêmico com a disciplina de CG, o qual aconteceu pela primeira vez somente no segundo semestre do ano letivo corrente, época em que toda a revisão bibliográfica e o primeiro contato com termos técnicos já havia sido feitos.

Por fim, deve-se salientar que a ocorrência dessas dificuldades não impediu a construção e o término do sistema proposto para o projeto de graduação, mas apenas estimulou o aluno à prática da pesquisa científica e serviu para o próprio amadurecimento pessoal adquirido ao longo dessa experiência acadêmica, que resume de certa forma, tudo o que foi visto durante o curso de Engenharia de Computação da FURG.

6.2 Principais Contribuições

O sistema proposto por este projeto de graduação não tem a intenção de apresentar uma nova estratégia para a animação facial e tampouco mostrar um programa pronto para ser utilizado como base para aplicações comerciais. Cabe salientar que sua proposta é a de mostrar o funcionamento de uma técnica de animação facial, implementada, comparando-

a com outras também utilizadas para a sua validação.

Com essa experiência adquirida, espera-se disponibilizar as contribuições desta pesquisa para futuros acadêmicos no âmbito interno da FURG, construir artigos científicos para expor os resultados à comunidade em geral e, acima de tudo, aumentar a quantidade de material bibliográfico sobre a área de animação facial na língua portuguesa, o que ainda é bastante escasso.

Além dessas contribuições, também se deseja disseminar a imensa gama de aplicações para as quais esse projeto pode se enquadrar, para que mais pessoas se interessem e possam dar continuidade a esse trabalho, que apresenta características pioneiras dentre os demais projetos de pesquisa e graduação já realizados dentro do curso de Engenharia de Computação da FURG.

6.3 Propostas Futuras

Como já mencionado, a proposta que originou a implementação deste trabalho era mais audaciosa, a de animar uma face aliando esses movimentos à fala humana, com base em movimentos reais detectados previamente. No entanto, sistemas com esse número de funcionalidades apresentam uma grande carga de implementações e pesquisas, que fugiriam ao escopo de um trabalho de graduação, tanto que são apenas teses de doutorado encontradas na bibliografia que mostram a implementação de um sistema completo [Beskow, 2003], [Martino, 2005].

6.3.1 Texto-Fala

Basicamente, um conversor TF é composto de dois módulos principais: o Processador de Linguagem Natural e o Processador de Sinais Digitais. O primeiro é o responsável pela produção da transcrição fonética do texto a ser lido, juntamente com a entonação e o ritmo mais adequados, através da geração de uma informação simbólica. Já o segundo componente é o responsável pela transformação da informação gerada pelo módulo anterior em fala [Körting et al., 2005a].

Tendo esse sistema já implementado, denominado *Free Text-to-Speech System* (FTTSS), disponibilizado para livre utilização, cópia e alteração no endereço [http:](http://)

`//fttss.sourceforge.net/`, espera-se futuramente construir uma base de conhecimento para aliar a voz aos movimentos faciais, construindo o que já foi definido por visema.

Muitas implementações de animação facial ainda se utilizam de fala não sintetizada, mas gravada previamente, e aplicada à animação para produzir o resultado final (voz + imagem). Espera-se também realizar testes com essa abordagem, tanto para o treinamento e construção da base dos visemas, quanto para a validação desse e do método de fala utilizando o próprio sistema TF.

Bibliografia

- [Azevedo and Conci, 2003] Azevedo, E. and Conci, A. (2003). *Computação Gráfica – Teoria e Prática*. Elsevier, Rio de Janeiro, RJ, Brasil.
- [Baranauskas, 2005] Baranauskas, C. (2005). *IHC Brasil*. Disponível em <http://www.serg.inf.puc-rio.br/ihc/>. Acessado em 13/12/2005.
- [Beskow, 2003] Beskow, J. (2003). *Talking Heads*. PhD thesis, Stockholm.
- [Bicho et al., 2002] Bicho, A. L., da Silveira Jr, L. G., da Cruz, A. J. A., and Raposo, A. B. (2002). Programação Gráfica 3D com OpenGL, Open Inventor e JAVA 3D. In *Revista Eletrônica de Iniciação Científica (REIC)*.
- [Bregler et al., 1997] Bregler, C., Covell, M., and Slaney, M. (1997). Video Rewrite: driving visual speech with audio. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 353–360, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.
- [Breton et al., 2001] Breton, G., Bouville, C., and Pelé, D. (2001). FaceEngine a 3D facial animation engine for real time applications. In *Web3D '01: Proceedings of the sixth international conference on 3D Web technology*, pages 15–22, New York, NY, USA. ACM Press.
- [Cochran, 2003] Cochran, W. O. (2003). *Hermite Surfaces*. Disponível em <http://ezekiel.vancouver.wsu.edu/~cs442/>. Acessado em 13/12/2005.
- [Costa, 2001] Costa, A. C. (2001). *História da Computação Gráfica*. Disponível em <http://www.dei.isep.ipp.pt/cg/>. Acessado em 09/12/2005.

- [da Silva and Costa, 2005] da Silva, F. C. and Costa, R. M. (2005). *Um Sistema Computacional para Extração de Características Faciais*. Projeto de Engenharia de Computação (FURG). Rio Grande, RS, Brasil.
- [DeRose et al., 1998] DeRose, T., Kass, M., and Truong, T. (1998). Subdivision surfaces in character animation. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 85–94, New York, NY, USA. ACM Press.
- [Dunlop, 2002] Dunlop, R. (2002). *Introduction to Catmull-Rom Splines*. Disponível em <http://www.mvps.org/directx/articles/catmull/>. Acessado em 13/12/2005.
- [Edge et al., 2004] Edge, J. D., Lorenzo, M. A. S., and Maddock, S. (2004). Animating speech from motion fragments. Technical report, Department of Computer Science Research Memorandum CS - 04 - 02, University of Sheffield.
- [Edge and Maddock, 2003] Edge, J. D. and Maddock, S. (2003). Image-based Talking Heads using Radial Basis Functions. In *Proceedings of IEEE Theory and Practice of Computer Graphics*, pages 74–80.
- [Ekman et al., 2005] Ekman, P., Friesen, W. V., and Hager, J. C. (2005). *Face and Gesture FAQ for Computer Scientists*. Disponível em <http://face-and-emotion.com/>, acessado em 28/06/2005.
- [Fabbro and Martino, 1998] Fabbro, P. A. D. and Martino, J. M. D. (1998). *Imagens Demonstrativas de Técnicas de Síntese de Imagens*. Disponível em <http://www.dca.fee.unicamp.br/~martino/>. Acessado em 10/12/2005.
- [Haber et al., 2001] Haber, J., Kähler, K., Albrecht, I., Yamauchi, H., and Seidel, H.-P. (2001). Face to face: From Real Humans to Realistic Facial Animation. In *In Proc. Israel-Korea Binational Conference on Geometrical Modeling and Computer Graphics*, pages 73–82.
- [Kähler et al., 2001] Kähler, K., Haber, J., and Seidel, H.-P. (2001). Geometry-based muscle modeling for facial animation. In *GRIN'01: No description on Graphics interface 2001*, pages 37–46, Toronto, Canada. Canadian Information Processing Society.

- [Kopp et al., 2003] Kopp, S., Jung, B., Lebmann, N., and Wachsmuth, I. (2003). Max – A Multimodal Assistant in Virtual Reality Construction. pages 11–17.
- [Körting et al., 2005a] Körting, T. S., da Silva, F. C., and Costa, R. M. (2005a). Um Sistema Texto-Fala Livre. In *WSL'05: Workshop sobre Software Livre*.
- [Körting et al., 2005b] Körting, T. S., da Silva, F. C., Costa, R. M., de Lima Bicho, A., and da Costa Botelho, S. S. (2005b). Um estudo sobre a animação tridimensional de faces. In *SIBGRAPI '05: Workshop de Iniciação Científica*.
- [Lee et al., 1995] Lee, Y., Terzopoulos, D., and Waters, K. (1995). Realistic Modeling for Facial Animation. pages 55–62.
- [Longhi et al., 2004] Longhi, M. T., Nedel, L. P., Vicari, R. M., and Axt, M. (2004). Especificação e Interpretação de Gestos Faciais em um Agente Inteligente e Comunicativo. In *SBC Symposium on Virtual Reality*, pages 243–254, São Paulo, SP, Brasil.
- [Loyall, 1997] Loyall, A. B. (1997). *Believable Agents: Building Interactive Personalities*. PhD thesis, School of Computer Science, Computer Science Department, Carnegie Mellon University, Pittsburg, USA.
- [Martino, 2005] Martino, J. M. D. (2005). *Animação Facial Sincronizada com a Fala: Visemas Dependentes do Contexto Fonético para o Português do Brasil*. PhD thesis, Universidade Estadual de Campinas, Campinas, SP, Brasil.
- [Noh and Neumann, 1998] Noh, J. and Neumann, U. (1998). A survey of Facial Modeling and Animation Techniques.
- [Parke, 1972] Parke, F. I. (1972). Computer generated animation of faces. In *ACM'72: Proceedings of the ACM annual conference*, pages 451–457, New York, NY, USA. ACM Press.
- [Parke and Waters, 1996] Parke, F. I. and Waters, K. (1996). *Computer facial animation*. A. K. Peters, Ltd., Natick, MA, USA.

- [Platt and Badler, 1981] Platt, S. M. and Badler, N. I. (1981). Animating facial expressions. In *SIGGRAPH '81: Proceedings of the 8th annual conference on Computer graphics and interactive techniques*, pages 245–252, New York, NY, USA. ACM Press.
- [Qin and Terzopoulos, 1996] Qin, H. and Terzopoulos, D. (1996). D-NURBS: A Physics-Based Framework for Geometric Design. In *IEEE Transactions on visualization and Computer Graphics*, volume 2, pages 85–96.
- [(SDN), 2005] (SDN), S. D. N. (2005). *JAVA Technology*. Disponível em <http://java.sun.com/>.
- [Sederberg and Parry, 1986] Sederberg, T. W. and Parry, S. R. (1986). Free-form deformation of solid geometry models. In *Computer Graphics (SIGGRAPH 86)*, pages 151–160.
- [Straccia, 2002] Straccia, A. (2002). *Introdução à História da Computação Gráfica*. Disponível em <http://www.xuti.net/>. Acessado em 10/12/2005.
- [Tao and Tan, 2004] Tao, J. and Tan, T. (2004). Emotional Chinese talking head system. In *ICMI '04: Proceedings of the 6th international conference on Multimodal interfaces*, pages 273–280, New York, NY, USA. ACM Press.
- [Waters, 1987] Waters, K. (1987). A muscle model for animation three-dimensional facial expression. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 17–24, New York, NY, USA. ACM Press.
- [Watt and Watt, 1992] Watt, A. and Watt, M. (1992). *Advanced animation and rendering techniques : theory and practice*. Addison-Wesley.
- [Wirth, 2000] Wirth, M. A. (2000). *A Nonrigid approach to Medial Image Registration: Matching Images of the Breast*. PhD thesis, RMIT University, Melbourne, Victoria. Australia.
- [yong Noh et al., 2000] yong Noh, J., Fidaleo, D., and Neumann, U. (2000). Animated deformations with radial basis functions. In *VRST '00: Proceedings of the ACM symposium on Virtual reality software and technology*, pages 166–174, New York, NY. USA. ACM Press.