

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**AUMENTO DO REALISMO NA DRAMATIZAÇÃO DE STORYTELLING COM A
CRIAÇÃO DE PERSONAGENS QUE DEMONSTREM EMOÇÕES ATRAVÉS DE
EXPRESSÕES FACIAIS**

TRABALHO DE GRADUAÇÃO

SUZANA AMARAL SANGOI

**Santa Maria, Rio Grande do Sul, Brasil
2008**

**AUMENTO DO REALISMO NA DRAMATIZAÇÃO DE STORYTELLING COM A
CRIAÇÃO DE PERSONAGENS QUE DEMONSTREM EMOÇÕES ATRÁVES DE
EXPRESSÕES FACIAIS**

por

Suzana Amaral Sangoi

Monografia apresentada ao Curso de Ciência da Computação,
Área de Computação Gráfica,
da Universidade Federal de Santa Maria (UFSM, RS),
como requisito parcial para a obtenção do grau de
Bacharel em Ciência da Computação

Orientador: Professor Dr. Cesar Tadeu Pozzer

Trabalho de Graduação N. 278

Santa Maria, RS, Brasil

2008

**Universidade Federal de Santa Maria
Centro de Tecnologia
Curso de Ciência da Computação**

A Comissão Examinadora, abaixo assinada,
aprova a Monografia

**AUMENTO DO REALISMO NA DRAMATIZAÇÃO DE STORYTELLING COM A
CRIAÇÃO DE PERSONAGENS QUE DEMONSTREM EMOÇÕES ATRAVÉS DE
EXPRESSÕES FACIAIS**

elaborada por
Suzana Amaral Sangoi

como requisito parcial para a obtenção do grau de
Bacharel em Ciência da Computação

COMISSÃO EXAMINADORA:

Cesar Tadeu Pozzer, Prof. Dr.
(Presidente/Orientador)

Andrea Schwertner Charão, Profa. Dra. (UFSM)

Benhur de Oliveira Stein, Prof. Dr. (UFSM)

Santa Maria, dezembro de 2008

*“Merecem louvor os homens que em si
mesmo encontraram o impulso e subiram
nos seus próprios ombros”
(Séneca)*

Dedico à minha família e a todos que me apoiaram.

AGRADECIMENTOS

Agradeço, primeiramente, aos meus pais e irmã, que sempre me deram apoio e me incentivaram nos momentos de desânimo, para que eu seguisse em frente.

Agradeço às pessoas que, de alguma forma, fizeram parte da minha vida e contribuíram positivamente para a realização desta etapa. Aos meus colegas de curso que sempre estavam dispostos a ajudar uns aos outros, sendo mais que apenas colegas, mas amigos.

Agradeço ao meu orientador, o professor Cesar Tadeu Pozzer, por ter sido sempre compreensivo e paciente e me ajudado nos momentos em que tudo parecia não dar certo.

RESUMO

Monografia de Graduação
Curso de Ciência da Computação
Universidade Federal de Santa Maria

AUMENTO DO REALISMO NA DRAMATIZAÇÃO DE STORYTELLING COM A CRIAÇÃO DE PERSONAGENS QUE DEMONSTREM EMOÇÕES ATRAVÉS DE EXPRESSÕES FACIAIS

AUTORA: SUZANA AMARAL SANGOI

ORIENTADOR: CESAR TADEU POZZER

Local e Data da Defesa: Santa Maria, 17 de dezembro de 2008.

A Computação Gráfica é aplicada em diversas áreas. Uma subarea é a Computação Gráfica em Tempo Real, empregada em aplicações de entretenimento, como jogos de computador e a exibição automática de narrativas, denominada *Storytelling*. Atualmente, o realismo é bastante explorado nessas aplicações, onde faz-se uso de técnicas de representação de personagens que se assemelham cada vez mais a criaturas do mundo real, seja através de sua aparência física ou de seu comportamento. O trabalho descrito aqui faz parte de um projeto que tem como objetivo a visualização gráfica tridimensional de histórias sendo representadas com o desenvolvimento de uma câmera virtual “inteligente” capaz de desempenhar o papel de um diretor que, como em um filme, define diversos parâmetros para ressaltar, da melhor forma possível, os principais fatos que estão acontecendo. Com o intuito de contribuir com o realismo destas aplicações, este presente trabalho propõe uma técnica que enriqueça o conteúdo emotivo das cenas, oferecendo personagens capazes de representar seus sentimentos através de suas expressões faciais, de forma que seu comportamento se assemelhe bastante ao de criaturas do mundo real. Para isso, é descrita uma forma de representação da face do personagem, bem como a abordagem utilizada para animação da mesma, de forma que diferentes emoções possam ser representadas de acordo com a ação que está sendo realizada pelo personagem ou segundo o enredo da história. Assim, espera-se aumentar o realismo da história a ser contada, tornando-a mais interessante e convincente aos olhos do espectador.

Palavras-chave: computação gráfica; *storytelling*; expressões faciais.

ABSTRACT

Monografia de Graduação
Course of Computer Science
Universidade Federal de Santa Maria

IMPROVING THE REALISM OF DRAMATIC STORYTELLING BY CREATING CHARACTERS THAT DEMONSTRATE CHANGING FACIAL EMOTIONS AND EXPRESSIONS

AUTHOR: SUZANA AMARAL SANGOI

ADVISOR: CESAR TADEU POZZER

Place and Date of Defense: Santa Maria, December 17, 2008.

Computer graphics is applied in many areas. The Real-Time Computer Graphics is a sub-area used in entertainment applications like computer games and automatic display of narratives called storytelling. The realism in CG is highly explored within these applications, and realistic expression and natural representation of characters increase every day. This monograph describes part of a project that focuses on the three-dimensional graphic visualization of stories, represented by the development of an "intelligent" virtual camera able to perform the role of a director, that like in movies, determines the parameters to emphasize the story being told. In order to enrich the realism of these applications, this study proposes a technique to improve the emotional content of scenes, by creating characters that effectively represent their feelings by facial expressions, closely mirroring the creatures of the real world. This way, this work presents a representation of the character's face and an animation approach to ensure that different emotions may be represented according to the present action of the character or the plot. Thus, an improvement to the realism of the storyline is expected, making it more interesting and convincing to the eyes of the audience.

Keywords: computer graphics; storytelling; facial expressions.

SUMÁRIO

LISTA DE FIGURAS	10
1 INTRODUÇÃO	11
1.1 OBJETIVOS DO TRABALHO	13
2 FUNDAMENTOS TEÓRICOS	15
2.1 O STORYTELLING INTERATIVO	15
2.2 O LOGTELL	17
2.2.1 <i>Interação com o Usuário</i>	18
2.2.2 <i>Drama Manager</i>	19
2.2.3 <i>Limitações do Sistema</i>	20
2.3 STORYTELLING COM AUXÍLIO DE UM DIRETOR	21
2.4 INSERINDO PERSONAGENS EMOTIVOS.....	23
2.4.1 <i>As Expressões Faciais</i>	24
2.4.2 <i>A Interpolação Linear na Geração de Novas Expressões Faciais</i>	25
2.4.3 <i>Técnicas de Implementação de Expressões Faciais em Personagens</i>	26
3 PROPOSIÇÃO DO TRABALHO	29
3.1 COMUNICAÇÃO DIRETOR-PERSONAGEM.....	29
3.2 MODELAGEM E ANIMAÇÃO DA FACE DO PERSONAGEM.....	30
4 IMPLEMENTAÇÃO DO TRABALHO	32
4.1 CRIAÇÃO DOS ELEMENTOS DA FACE	32
4.2 GERAÇÃO DAS SOBRANCELHAS E DA BOCA DO PERSONAGEM.....	32
4.2.1 <i>O Editor Facial</i>	39
4.2.2 <i>Técnica de Transição entre Expressões Faciais</i>	43
4.2.3 <i>A Interpolação Linear e a Transição de Expressões Faciais</i>	43
5 CONCLUSÃO E TRABALHOS FUTUROS	46
5.1 CONTRIBUIÇÕES ALCANÇADAS	46
5.2 TRABALHOS FUTUROS	47
6 REFERÊNCIAS BIBLIOGRÁFICAS	48
APÊNDICE A – CURVAS PARAMÉTRICAS	51

LISTA DE FIGURAS

FIGURA 1- Arquitetura do LOGTELL.....	17
FIGURA 2- Interface do LOGTELL.....	18
FIGURA 3- <i>Gingy</i> zangado.....	24
FIGURA 4- <i>Gingy</i> feliz.....	24
FIGURA 5- <i>Gingy</i> triste.....	24
FIGURA 6- Resultado do estudo de <i>Ortony</i>	24
FIGURA 7- Detalhes da malha facial.....	28
FIGURA 8- Comunicação entre diretor, câmera e personagem.....	30
FIGURA 9- Cada uma das 6 expressões básicas.....	31
FIGURA 10- Pontos de controle e a curva <i>B-Spline</i> gerada para a boca.....	33
FIGURA 11- Pontos de controle e a curva <i>B-Spline</i> para a boca “feliz”.....	34
FIGURA 12- Vetores diretores dos pontos da curva.....	34
FIGURA 13- Vetores perpendiculares dos pontos da curva.....	34
FIGURA 14- “Anéis” envolvendo cada ponto da curva.....	35
FIGURA 15- Malha da boca.....	35
FIGURA 16- Malha da boca na versão final.....	36
FIGURA 17- Pontos de controle e a curva <i>Bézier</i> gerada para a sobrancelha.....	37
FIGURA 18- Malha da sobrancelha.....	37
FIGURA 19- Malha da sobrancelha com as esferas nas extremidades.....	38
FIGURA 20- Representação final da sobrancelha da face.....	38
FIGURA 21- Expressão “neutra” para a face.....	39
FIGURA 22- Expressão “feliz” para a face.....	39
FIGURA 23- Tela inicial do editor facial.....	41
FIGURA 24- Expressão de raiva gerada no editor facial.....	42
FIGURA 25- TXT com as informações dos ptos de controle da expressão.....	42
FIGURA 26- Expressão inicial de nojo.....	44
FIGURA 27- Expressão final de alegria.....	44
FIGURA 28- Expressão intermediária entre nojo e alegria.....	45
FIGURA 29- Continuidade C0, C1 e C2.....	52
FIGURA 30- Fecho convexo.....	52
FIGURA 31- Representação gráfica das <i>blending functions</i> de grau 3.....	54
FIGURA 32- Curva de <i>Bézier</i> com 4 pontos de controle.....	54
FIGURA 33- Curva <i>B-Spline</i> cúbica.....	55
FIGURA 34- <i>Blending Functions</i> da <i>B-Spline</i>	56

1 INTRODUÇÃO

A definição de Computação Gráfica dada pela ISO (*International Standards Organization*), descreve-a como um conjunto de técnicas e métodos utilizados para converter dados para um dispositivo gráfico, via computador. Se nos basearmos apenas nesta definição, não conseguiremos ter noção da grande variedade de aplicações que a computação gráfica é utilizada.

O primeiro jogo de computador que fez uso da computação gráfica foi criado no início dos anos 60 e batizado de “Spacewar” (MARKOWITZ, 2008). A partir daí, houve uma crescente utilização da computação gráfica, bem como uma grande evolução em termos de ferramentas e dispositivos computacionais, de forma que rapidamente percebeu-se que esta poderia ser uma grandiosa forma de entretenimento.

Com o tempo, os jogos eletrônicos foram se aprimorando para jogos mais complexos e interativos, fazendo com que surgisse um novo segmento altamente lucrativo. Essa crescente demanda por jogos de computador fez com que este segmento crescesse muito mais rápido do que as outras diversas áreas da computação gráfica.

Uma área que possui vários pontos em comum e que acabou crescendo paralelamente à área de jogos é a construção automática de narrativas, também conhecida como *storytelling*. Este é um novo paradigma de entretenimento digital que avança rapidamente com a possibilidade de se produzir histórias interativas com a criação de novas técnicas e ferramentas, e que são visualizadas e guiadas com o auxílio do computador. Esta área possui um campo de aplicação muito vasto, principalmente no campo do entretenimento, podendo ser vista como uma forma de unir histórias e jogos, no sentido de aumentar o conteúdo narrativo de ambos.

Podemos dividir as pesquisas em *storytelling* em três partes distintas, sendo elas: geração, interação e visualização de histórias. Na geração, nos referimos à forma com que a história é gerada, através da estrutura da aplicação que guia aspectos gerais como os personagens, as ações, os objetos e os relacionamentos entre eles. A interação, também chamada de direcionamento, estabelece a interação entre o usuário, o enredo e os personagens da história. Este segmento também é responsável pela gerência das ações dos personagens autônomos, conhecidos como “agentes” (POZZER, 2005), de modo que a história a ser contada tenha

coerência. Já na visualização, é tratada a forma de representação gráfica da história, ou seja, a forma com que as abstrações das estruturas internas dos personagens serão transformadas em ações realistas dentro do espaço gráfico, geralmente tridimensional.

Uma técnica que se mostra bastante interessante quando se deseja aumentar o realismo das histórias representadas é a implementação de uma câmera inteligente, que possui conhecimentos de um diretor cinematográfico, para o aumento do poder de dramatização de *storytelling*. Neste contexto, a câmera virtual tem o papel de capturar e orquestrar as cenas sendo apresentadas, que podem ser compostas por um ou vários personagens. Além das tarefas que são desempenhadas por câmeras virtuais em jogos, neste caso, a câmera tem um papel de maior responsabilidade. A câmera deve desempenhar o papel de um diretor que, como em um filme, deve definir diversos parâmetros para ressaltar, da melhor forma possível, os principais fatos que estão ocorrendo na história. Ela deve ser autônoma o suficiente para selecionar o conteúdo a exibir, se posicionar, se locomover e fazer transições entre tomadas sucessivas, bem como reposicionar personagens no cenário para realizar tomadas mais realistas. Assim, a câmera assume dois papéis: o de um agente inteligente que toma decisões pertinentes e o de um elemento gráfico que se utiliza de técnicas cinematográficas para melhor capturar cada cena. Para isso, é necessário desenvolver recursos para dar à câmera a autonomia de definição do início das ações, a duração das ações, posicionamento e movimentação, levando-se em conta cenários abertos e fechados, bem como a topologia do terreno (POZZER, 2005). Além disso, é interessante adicionar à câmera efeitos especiais que caracterizam o conteúdo emotivo de cada cena. Para ressaltar esse tipo de conteúdo, é importante, também, incorporar aos personagens tridimensionais recursos de expressões faciais, de modo que suas emoções sejam expressas nas cenas que se decorrem, tornando a animação como um todo muito mais realista e interessante aos olhos do espectador. Para isso, existem diversas técnicas de representação e animação da face de personagens emotivos, que utilizam desde abordagens simples até técnicas mais complexas, dependendo do objetivo da aplicação que implementa este tipo de recurso.

1.1 Objetivos do Trabalho

O trabalho proposto nesta monografia se encaixa no contexto de *storytelling*, pois está inserido em um projeto maior, desenvolvido no Laboratório de Computação Aplicada (LaCA), sob orientação do professor Dr. Cesar Tadeu Pozzer. Este projeto foca-se quase que exclusivamente na visualização gráfica tridimensional de histórias sendo representadas, através do desenvolvimento de uma câmera “inteligente”, que por possuir auxílio de um “diretor virtual”, passa a se comportar de forma semelhante às câmeras em filmes, realizando transições de tomadas que valorizem a história a ser contada. Ele pode ser interpretado como uma tentativa de inserir uma maior flexibilidade e um maior realismo às histórias contadas pela técnica de *storytelling* já conhecida. No âmbito do realismo, percebeu-se que um fator que contribuiria de forma bastante relevante seria a representação de personagens que pudessem expressar as suas emoções através da face, ou seja, alterando suas expressões faciais ao mesmo tempo que executam suas ações, de acordo com o seu estado de espírito ao realizá-las. Com isso, o trabalho desenvolvido e descrito nesta monografia propõe uma técnica que possa criar personagens capazes de enriquecer o conteúdo emotivo das cenas com a representação de seus sentimentos através de suas expressões faciais.

Este trabalho de graduação foi organizado da seguinte forma: O Capítulo 2 apresenta os Fundamentos Teóricos necessários para o desenvolvimento da proposta, onde é feita uma introdução à abordagem de *Storytelling* Interativo. Segue-se com a descrição do sistema LOGTELL, onde são introduzidos conceitos da Interação do Usuário com o sistema, seguindo com a descrição do módulo *Drama Manager*. Na seqüência, são mostradas algumas Limitações do Sistema e a descrição da idéia de *Storytelling* com Auxílio de um Diretor, seguida de alguns conceitos sobre a Inserção de Personagens Emotivos em aplicações gráficas. As Expressões Faciais são tratadas na seqüência, assim como as Técnicas de Implementação de Expressões Faciais em Personagens. No Capítulo 3, tem-se a Proposição do Trabalho, onde é descrita a proposta de Comunicação entre Diretor e Personagem. A implementação do Trabalho é mostrada no Capítulo 4, onde tem-se a descrição do processo de Criação dos Elementos da Face e a Geração das Sobrancelhas e da Boca do Personagem. Segue-se com a introdução ao Editor Facial, utilizado como ferramenta de auxílio ao trabalho desenvolvido, seguida da

Técnica de Transição entre Expressões Faciais utilizada e, fechando a seção, tem-se a descrição sobre a Transição entre as Expressões Faciais. No Capítulo 5 encontram-se a Conclusão e os Trabalhos Futuros. Para finalizar, o Capítulo 6 traz as Referências Bibliográficas, seguido pelo Apêndice A, que contém uma revisão de conceitos básicos sobre curvas paramétricas.

2 FUNDAMENTOS TEÓRICOS

2.1 O Storytelling Interativo

A forma com que as pessoas interagem com os computadores atualmente vem sofrendo uma revolução, tanto em termos de conteúdo como de paradigmas de interface. Na narração de histórias, a necessidade de se prover recursos que permitam ir além da narrativa tradicional, composta por simples ramificações, é bastante evidente, uma vez que pode-se observar que conteúdos lineares, como em livros, não se mostram mais suficientes para atender à nova demanda de aplicações (MATEAS; STERN, 2002). Histórias não lineares, também chamadas de “narrativas participativas” ou “ficção interativa”, são pré-requisitos que o conteúdo deve permitir, para que o mesmo possa ser chamado de interativo (GLASSNER, 2004).

O novo paradigma de entretenimento digital, conhecido como *storytelling*, está avançando rapidamente através da criação de novas técnicas e ferramentas computacionais que fazem com que histórias interativas possam ser geradas, visualizadas e guiadas com o uso do computador.

Qualquer sistema de *storytelling* opera sobre algum tipo de dado que possa representar a história com a qual o usuário interage, podendo consistir, por exemplo, de regras, personagens, tipos de cenas e eventos. Segundo alguns estudiosos, máquinas não possuem a capacidade de dinamicamente criar histórias com detalhes convincentes, uma vez que esta é uma tarefa que envolve um ciclo iterativo de especificações e testes, de modo que se assegure que o conteúdo resultante seja tanto coerente como interativo.

Um fator que tem uma grande influência sobre a forma como a interação do usuário ocorre é a estrutura interna de representação da história. O nível de interação pode ser tratado sob diferentes abordagens, entre elas a abordagem orientada a enredos (*plot-based*) e a abordagem orientada a personagens (*character-based*). Na primeira, tanto a interação como a criação do enredo são realizados sobre operações em alto-nível, ou seja, o usuário não possui permissão para alterar qualquer atributo a qualquer momento (GRASBON; BRAUN, 2001). Nesta abordagem, tem-se um controle maior sobre a história que está sendo representada, de forma que seja evitada a fuga do contexto inicial que o autor

definiu. O autor gera previamente a estrutura geral da história e já pode definir um início, meio e fim. Apenas interferências “sutis” podem ser realizadas pelo usuário, as quais têm o objetivo de guiar o andamento para chegar aos pontos predefinidos. Geralmente utiliza-se grafos para representar esse tipo de estrutura, onde cada nó contém um evento da história, uma informação sobre o personagem ou uma posição discreta no ambiente. Esses nós são conectados de forma que diferentes caminhos podem ser percorridos, definindo, assim, o formato da apresentação da narrativa.

Já na abordagem orientada a personagens (CAVAZZA; CHARLES; MEAD, 2001), (CAVAZZA; CHARLES; MEAD, 2002), (MATEAS; STERN, 2000), (YOUNG, 2001), a interação com o usuário é dada em mais baixo-nível, tornando possível a alteração de atributos que influenciam no desenvolvimento do enredo. É o comportamento e a interação dos personagens autônomos, também chamados de “agentes”, que formam a base da geração do enredo. Pode ser uma abordagem um tanto perigosa, uma vez que as tecnologias existentes podem não ser capazes de garantir coerência nos roteiros criados ou histórias com muito significado (GRASBON; BRAUN, 2001). Outras abordagens que permitem que o usuário tenha um controle muito detalhado também podem resultar no mesmo problema.

A abordagem orientada a enredo trata a técnica de *storytelling* como narração, diferente da abordagem orientada a personagens, onde a mesma é tratada como simulação. Na narração, a história é parcialmente definida com algumas possibilidades de interação, desde que limitadas ao escopo do enredo, tendo-se à disposição um conjunto de parâmetros reais, geralmente em alto nível, significando a possibilidade de processar dados já conhecidos que foram projetados com o intuito de serem ajustados (SPIERLING; BRAUN; GRASBON, 2002). Na abordagem orientada a personagens, a história é gerada à medida em que ocorre a interação dos personagens entre si e com o mundo. Apesar de permitir um nível muito mais fino de interatividade, pode torná-la um tanto complexa e trabalhosa.

Há também as abordagens híbridas (CRAWFORD, 1999), que fazem uso de verbos como componentes básicos das ações, onde estas são definidas pelo autor, uma vez que também acredita-se na impossibilidade da máquina gerar histórias coerentes automaticamente. Esta abordagem parece muito adequada, visto que nela combinam-se eventos dinâmicos com ações previamente definidas.

2.2 O LOGTELL

O LOGTELL é um sistema de *storytelling* desenvolvido por Cesar Tadeu Pozzer, como tese de Doutorado defendida no ano de 2005, na PUC-Rio (CIARLINI; POZZER; FURTADO; FEIJÓ, 2005), e que serve como suporte lógico para a geração de histórias. O sistema possui módulos de suporte à geração, interação com o usuário e visualização das histórias geradas, conforme pode ser visto na figura 1 (CIARLINI; POZZER, 2005), que mostra a estrutura completa da aplicação.

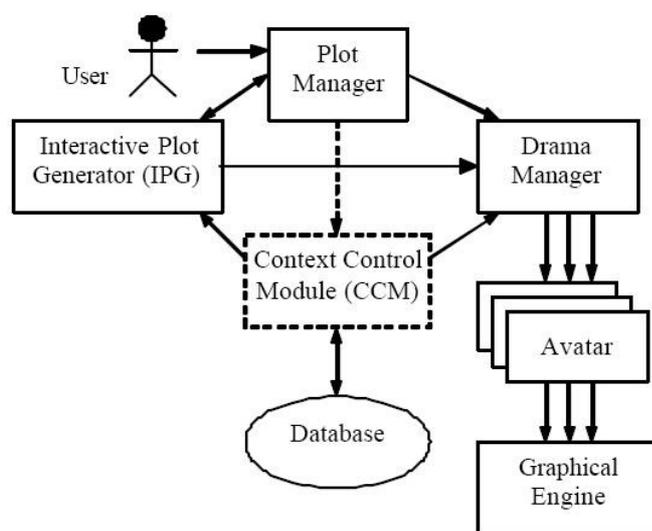


Figura 1: Arquitetura do LOGTELL

A geração de fatos (pequenas ações que, quando unidas, formam uma determinada seqüência de eventos a serem mostrados) é realizada pelo IPG (*Interactive Plot Generator*). Por meio da interface do *Plot Manager*, o usuário pode interagir com o processo de criação da história, escolhendo uma determinada ordem para os fatos, pois o IPG gera apenas uma ordem parcial para os mesmos, solicitando, então, que o *Plot Manager* ative o módulo de dramatização (visualização), conhecido por *Drama Manager*.

O módulo IPG realiza simulações de acordo com o contexto especificado pelo usuário. A ordem dos fatos, informada pelo usuário através do *Plot Manager*, é levada em consideração quando esse processo de simulação é feito. O *Drama Manager* é responsável por controlar a dramatização dos fatos gerados, fazendo o controle dos personagens em um ambiente tridimensional. Durante a dramatização, este módulo consulta o IPG para manter uma certa coerência entre as

representações lógicas e gráficas dos fatos.

O processo para a criação de histórias no LOGTELL possui os seguintes passos:

- configuração inicial, introduzindo os personagens e suas situações iniciais, assim como a descrição dos cenários e outras informações necessárias para a geração de histórias;
- conjunto de regras lógicas para determinar metas a serem seguidas por cada personagem e para guiar certas situações que acontecem no decorrer dos eventos;
- uma biblioteca de operações típicas da escolha do gênero da história.

No LOGTELL, o gênero da história a ser contada está inserido em um contexto de conto de fadas, cujo elenco principal de personagens são dois heróis, uma princesa e um vilão (dragão), que tem como objetivo tentar raptar a princesa. O papel dos heróis é libertar a vítima para com ela se casar. Os personagens dispõem de um elenco de ações que inclui deslocamentos, ataques, raptos, libertação e casamento. Para a captura das ações a serem visualizadas, a câmera desenvolvida identifica a ação que estão ocorrendo atualmente e desloca-se para local do acontecimento, de forma a mostrar o que está se passando naquele instante de tempo.

A configuração inicial da história é especificada por um estado da base de dados a ser modificada a partir da execução das operações. Alguns exemplos de fatos armazenados na base de dados em um contexto de contos de fadas, usando uma notação em Prolog, são:

- `personagem(vilão, dragão): "O dragão é um vilão";`
- `vilão_estado(dragão, força, 0.5): "O nível de força inicial do dragão é 0.5";`
- `estado_herói(cavaleiro, princesa, 0.2): "O nível de afeição inicial entre o cavaleiro e a princesa é 0.2".`

2.2.1 Interação com o Usuário

Em termos gerais, o usuário tem controle direto apenas com o módulo *Plot Manager*, o qual se comunica com o IPG para executar a geração de eventos e impor coerência nos fatos e com o módulo *Drama Manager* para controlar a visualização gráfica 3D dos fatos. O *Plot Manager* possui uma interface gráfica em

Java que permite ao usuário participar da escolha dos eventos que serão visualizados e decidir a seqüência final dos mesmos, conforme ilustrado na figura 2 (CIARLINI; POZZER; FURTADO; FEIJÓ, 2005). Cada evento é representado por uma caixa retangular que assume uma cor específica de acordo com o seu atual estado.

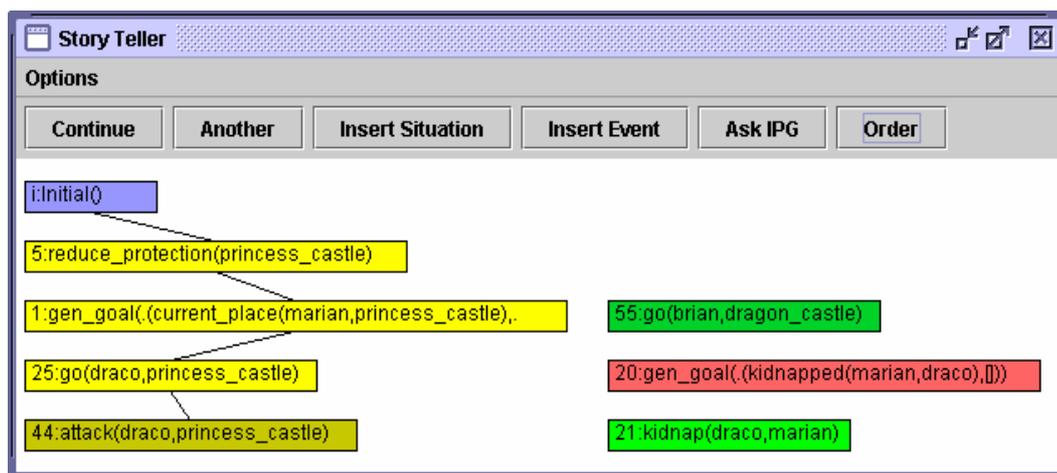


Figura 2: Interface do LOGTELL

O usuário não tem controle direto da cena e nem dos personagens presentes nela. A intervenção do usuário é sempre indireta, uma vez que cada intervenção do usuário deve ser validada pelo IPG antes de ser incorporada no sistema.

A geração de eventos e a dramatização são dois processos separados, contrastando com a abordagem de *storytelling* puramente baseada em personagem, onde a interação do usuário afeta a estrutura da história em tempo de execução. Na seção a seguir é descrito em mais detalhes o módulo de dramatização, conhecido como *Drama Manager*.

2.2.2 Drama Manager

O módulo de dramatização, conhecido como Drama Manager, é uma *engine* de suporte gráfico aos eventos gerados conforme descrito na seção anterior. É implementado em C++ e usa a API gráfica OpenGL para suporte à renderização em tempo-real dos elementos tridimensionais que compõem a história. Os personagens envolvidos nos eventos são referenciados por atores na dramatização. Cada ator é implementado como um agente reativo, associado a um avatar tridimensional.

Este módulo de visualização é responsável por renderizar, a cada quadro, a cena e os respectivos personagens e suas ações, como resultado de interações em tempo real com a cena e, ocasionalmente, com outros atores. Para fazer isso, é seguida a ordem na seqüência de eventos gerada no estágio anterior da simulação. O *Drama Manager* é o módulo que sincroniza e temporiza as ações dos personagens, assim como a representação gráfica dos mesmos.

O trabalho desempenhado pelo *Drama Manager* não está limitado a designar as ações que um determinado personagem deve desempenhar. Ele traduz operações simbólicas em animações gráficas tridimensionais e garante o sincronismo e coerência lógica entre o mundo a ser representado e a sua representação gráfica.

2.2.3 Limitações do Sistema

Algumas limitações podem ser constadas no desenvolvimento do sistema LOGTELL. A primeira delas está relacionada com a forma com que a visualização da história é apresentada, através da abordagem utilizada para a câmera da aplicação. Neste sistema, a câmera tem a função de descobrir, em tempo real, a ação a exibir e se configurar em função dela. Desta forma, algumas tomadas iniciam em cortes bruscos, pois a câmera dá início à captura da nova cena quando ela já está ocorrendo, dificultando, assim, o entendimento pelo usuário do novo evento. Uma solução proposta para este problema está descrita na seção a seguir.

Outra questão é a existência de personagens apáticos, que não possuem nenhuma expressão facial para demonstrar as suas emoções, o que compromete a sensação de realismo que a animação gerada deveria transmitir. Assim, o trabalho descrito nesta monografia contribui exatamente neste ponto, introduzindo personagens que possuam a capacidade de transmitir as emoções que estão sentindo ao realizar uma determinada ação, através das expressões faciais. O simples fato de inserir essas expressões nos personagens de uma animação tem o poder de aumentar consideravelmente a carga emocional das cenas apresentadas, tornando-as muito mais envolventes e convincentes aos olhos do espectador.

2.3 Storytelling com Auxílio de um Diretor

A câmera, em um ambiente de *storytelling*, é tão importante quanto os mecanismos de geração de histórias. Não é muito conveniente desenvolver um poderoso ambiente de geração de histórias, que permita ao mesmo tempo a criação de histórias interativas e “interessantes”, se o processo de visualização gráfica do conteúdo gerado for deficiente. A proposta de *storytelling* com auxílio de um diretor atribui à câmera características visualizadas em câmeras de filmes, através de tomadas específicas e técnicas que valorizam o conteúdo da história a ser mostrada. Desta forma, usa-se o termo câmera “inteligente”, uma vez que esta possui o auxílio de um diretor virtual responsável por coordenar, temporizar e sincronizar a realização dos eventos, interpretando o roteiro e definindo como cada cena deverá ser filmada, informando à câmera o tipo de tomada que ela deverá realizar em cada momento. Questões como a duração das cenas, posicionamento de atores, posição e transições de câmeras devem estar sob responsabilidade do diretor. Desta forma, o diretor virtual deve contemplar os seguintes quesitos:

1. Definir o tempo de início das ações: este é um fator extremamente importante para uma melhor compreensão da história a ser apresentada. Com o poder do diretor de controlar o início das cenas, é possível melhor contextualizar o cenário e os personagens que farão parte da tomada. Depois de contextualizar o cenário, pode-se, então, dar início à dramatização da cena propriamente dita.

2. Definir o tempo de duração de cada ação: neste caso, o diretor deve incorporar recursos para que sejam feitos cortes de tomadas muito longas, como no caso de cenas que representem a ida de um personagem a um determinado local distante. Para fazer isso, o diretor deve ser capaz de contextualizar o início da cena, fazer o corte e finalmente estimar a posição do personagem próximo ao seu destino, de modo que seja possível reposicioná-lo e voltar a gravar.

3. Posicionamento: Além de sincronizar as cenas, o diretor deve saber posicionar a câmera no cenário, segundo técnicas existentes de cinematografia (HE; COHEN; SALESIN, 1996), para que seja possível melhor capturar a essência de cada cena. Em alguns casos, o diretor deve posicionar os personagens de modo a melhor contextualizar a cena (tomada) a ser gravada. Para isso, ele deve avaliar o cenário, a posição e o número de personagens envolvidos na cena.

4. Movimentação: Diversas cenas podem envolver personagens se locomovendo. Neste caso, a câmera deve saber se locomover para acompanhar a cena. Além disso, as cenas estáticas também podem se utilizar desse recurso para uma melhor representação, podendo a câmera fazer um “vão” panorâmico ao redor da ação. Assim, pode-se ver a cena sob vários ângulos de visão, o que também facilita a contextualização do cenário onde a cena está sendo realizada.

5. Linguagem de especificação: a grande maioria de sistemas tidos como inteligentes incorporam conhecimentos inseridos pelo programador. Na direção de um filme, por exemplo, o diretor faz uso de vários conhecimentos e processos cognitivos para escolher a melhor tomada a ser feita. Não espera-se que o diretor virtual seja tão inteligente a ponto de se igualar a um diretor real. Dessa forma, propõe-se que devam ser passadas dicas, por meio de uma linguagem de especificação, onde o usuário do sistema ou o criador da história possa informar parâmetros que auxiliem o diretor no seu trabalho. Esses parâmetros podem ser globais, como “cena de pânico” ou “cena de romance”, ou de mais baixo nível, como “abraçe um personagem” ou “converse com um personagem por 5 segundos”.

Para que o diretor possa desempenhar essas metas propostas, ele deve agir como um personagem inteligente, capaz de tomar decisões próprias. Assim, ele deve incorporar algoritmos de técnicas de Inteligência Artificial comumente aplicadas em jogos de computador no controle de personagens autônomos, também conhecidos como NPCs (*Non Player Characters*)¹.

De posse de um diretor com recursos inteligentes, um componente a ser adicionado, que visa o enriquecimento da dramatização das cenas, é a agregação de características emotivas à câmera. Como cada evento tem um traço emotivo característico, como alegria, medo e fúria, o diretor pode fazer uso desta informação para atribuir à câmera um comportamento apropriado. Dessa forma, o estado emocional dos agentes da cena tem influência no estado emocional da câmera, que passa a realizar tomadas que refletem o estado emocional dos personagens. A emoção pode afetar tanto as tomadas de câmeras como os movimentos que esta realiza para realçar o conteúdo da cena.

¹ NPCs representam oponentes que são controlados pelo computador

2.4 Inserindo Personagens Emotivos

Em aplicações de entretenimento, como jogos de computador e *storytelling*, é de fundamental importância que os personagens possam se comportar como criaturas do mundo real, uma vez que este ramo da computação gráfica vem se desenvolvendo rapidamente, assim como a exigência por parte dos usuários desse tipo de aplicação. O realismo é cada vez mais explorado nestes, fazendo-se uso da evolução tanto em termos de inteligência de *software* como de *hardware*. Além da agregação de características emotivas à câmera, conforme citado na seção anterior, a representação de personagens que expressem suas emoções através da face se torna um quesito extremamente importante na busca pelo realismo em aplicações de entretenimento.

Personagens sem expressão emocional não nos trazem a ilusão de “estarem vivos”. Esta ilusão, quando alcançada através de técnicas de implementação de agentes emotivos, resulta em uma maior “simpatia” do usuário com a animação em questão. É fundamental para o realismo de uma aplicação de entretenimento gráfico que os seus personagens se assemelhem ao máximo possível à criaturas do mundo real, as quais estamos acostumados a nos relacionar. Isso traz um maior envolvimento do usuário com os personagens, criando um laço afetivo que torna a animação muito mais interessante e envolvente. Animações com personagens que não possuem expressões faciais ou até mesmo sem uma face definida tendem a tornarem-se desinteressantes muito mais rapidamente.

Em se tratando do vínculo afetivo criado pelo usuário com o personagem da animação, estudos realizados apontaram que o ser humano possui maior simpatia por personagens que não necessariamente se assemelham fisicamente à criaturas do mundo real, mas que o fazem através da demonstração de diferentes emoções, seja através da face, de movimentos corporais ou de ambos. Já personagens que possuem uma semelhança física muito grande com seres reais, mas que não possuem expressões emocionais, comumente não alcançam esse tipo de vínculo afetivo. Um exemplo que pode ser citado é o caso do personagem *Gingy*, do filme *Shrek*, que possui uma forma bem simples de biscoito, mas que demonstra características bem próximas às do ser humano através de suas expressões emotivas muito bem representadas, conforme pode ser visto nas figuras 3 a 5 (FILMONIC, 2008).

Figura 3: *Gingy zangado*Figura 4: *Gingy feliz*Figura 5: *Gingy triste*

2.4.1 As Expressões Faciais

Uma questão importante para a representação de expressões faciais em personagens virtuais é a definição de expressões básicas. Mas será que é possível definir uma quantidade finita de expressões, das quais todas as outras podem ser geradas? Estudos realizados na área de computação afetiva concluíram que a resposta à esta pergunta é positiva. Mas esta é uma questão muito controversa, pois apesar de diferentes estudiosos chegarem à esta conclusão, as descobertas divergem em relação ao número de expressões básicas existentes (ORTONY, 1990). Não há um acordo quanto ao conjunto e número de expressões básicas, nem quanto aos critérios que as definem, conforme pode ser visto na figura 6.

	EMOÇÕES BÁSICAS
Arnold	Raiva, aversão, coragem, desânimo, desejo, desespero, medo, ódio, esperança, amor, tristeza
Ekman, Friesen e Ellsworth	Raiva, desgosto, medo, surpresa, tristeza
Frijda	Desejo, felicidade, interesse, surpresa, admiração, tristeza
Gray	Raiva, terror, ansiedade, alegria
James	Medo, desgosto, amor, raiva
Mowrer	Dor, prazer
Panksepp	Expectativa, medo, raiva, pânico
Watson	Medo, amor, raiva
Weiner and Graham	Felicidade, tristeza

Figura 6: Resultado do estudo de Ortony

Uma vez definidas as expressões básicas, expressões intermediárias podem ser geradas através delas, utilizando-se alguma técnica para isto, onde, talvez, a mais conhecida e utilizada seja a de interpolação linear de expressões básicas (PARKE; WATERS, 1996). Nesta, expressões intermediárias são geradas através da interpolação das posições dos vértices que compõem as expressões-chave. Obviamente, esta é uma técnica bastante simples que pode não ser satisfatória para determinadas aplicações, porém, devido ao fato de ser uma técnica comumente utilizada, o processo realizado na geração de novas expressões faciais fazendo-se uso da interpolação linear será descrito em mais detalhes na seção a seguir.

Algumas regras para a realização de uma animação facial já foram identificadas há bastante tempo, porém, continuam válidas para implementações contemporâneas. Alguns dos principais aspectos que devem ser considerados em um animador de faces são citados a seguir (KÖRTING; SILVA; COSTA; BOTELHO; BICHO, 2006).

- Quanto mais o modelo facial se assemelhar ao de um ser 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 de outras para expressar emoção; características de alta prioridade são os olhos, boca, sobrancelhas e pálpebras.

2.4.2 A Interpolação Linear na Geração de Novas Expressões Faciais

Em matemática, denomina-se Interpolação Linear o método de interpolação que se utiliza de uma função linear (polinômio de primeiro grau) $p(x)$ para representar, por aproximação, uma suposta função $f(x)$ que, originalmente, representaria as imagens de um intervalo descontínuo contido no domínio de $f(x)$.

A interpolação linear gera o segmento de reta entre dois pontos ordenados em seqüência. O princípio básico do equacionamento é desenvolver a expressão:

$$a(1 - t) + b(t), \text{ para } 0 \leq t \leq 1$$

onde a e b são os pontos inicial e final do intervalo descontínuo que se deseja obter a interpolação.

A interpolação linear leva cada ponto por transformações mantidas entre os valores 0 e 1. Dependendo-se do valor atribuído ao parâmetro t , tem-se diferentes quantidades de pontos gerados entre a e b .

Conforme já foi citado anteriormente, essa técnica é bastante utilizada na geração de expressões intermediárias, tendo-se definidas as expressões inicial e final e aplicando o método a cada ponto que define cada elemento da face. Desta forma, definindo-se um valor para o parâmetro de interpolação t , tem-se como resultado uma determinada expressão facial, representada pela combinação da expressão inicial e final utilizadas.

2.4.3 Técnicas de Implementação de Expressões Faciais em Personagens

Além da técnica de definição de expressões faciais básicas descritas na seção 2.4.1, atualmente, diversas técnicas são utilizadas para a representação de expressões, bem como mecanismos para a animação facial. O desenvolvimento de um modelo facial envolve determinar sua descrição geométrica e sua capacidade de animação. Adicionalmente, está envolvida nesta etapa a representação de atributos adicionais para a face, tais como superfícies coloridas e a utilização de texturas. Diversas técnicas de representação facial são conhecidas, entre elas as de representação volumétrica, de superfícies e a técnica de geração de novas faces a partir de faces existentes (RODRIGUES; FEIJÓ; VELHO, 2005).

Na representação volumétrica, a abordagem utilizada é a de modelar faces através da utilização de técnicas de representação volumétrica. Isso inclui *Constructive Solid Geometric (CSG)*, *arrays* de elementos de volume (*voxels*) e elementos de volume agregados, tais como *octrees* (PARKE; WATERS, 1996). Normalmente, faces realistas não são representadas dessa forma. Com isso, *CSG* não é uma técnica de base geométrica popular para faces. Por outro lado, é possível imaginar um personagem tridimensional em estilo *cartoon* modelado a partir dessa técnica.

Superfícies primitivas e estruturas são, atualmente, a base geométrica preferida para modelos faciais (PARKE; WATERS, 1996), (ZHANG; LIU; GUO; SHUM, 2003). As estruturas de superfície utilizadas permitem formatos de superfície

e mudanças nos formatos quando necessário para as várias conformidades e expressões faciais. A técnica de representação de superfícies inclui superfícies implícitas, superfícies paramétricas e superfícies poligonais. Superfícies paramétricas incluem *Bézier* bivariável, *Catmull Rom*, *Beta-Spline*, *B-Spline*, *B-Spline* hierárquica e superfícies *NURBS*. Superfícies poligonais incluem malhas regulares poligonais e redes de polígonos arbitrárias. Na criação de novas faces a partir de faces já existentes (PARKE; WATERS, 1996), há várias abordagens conhecidas. Essas técnicas incluem interpolação entre faces existentes, deformação à faces já existentes e transformação de uma face canônica em faces de indivíduos específicos.

Ainda no contexto de expressões faciais, é importante pensarmos em como realizar a transição entre duas expressões faciais de forma satisfatória, o que é essencial se estamos preocupados com o realismo das animações. Existem, assim, ao menos quatro categorias fundamentais para a animação facial. Essas categorias são: interpolação de poses-chave, parametrização geométrica, parametrização *data-driven* e simulação biomecânica (CÔRREA; MAGALHÃES; MARTINO, 2005).

A interpolação de poses-chave baseia-se na especificação de um conjunto de modelos geométricos apresentados em diferentes poses faciais. As poses intermediárias são calculadas através da interpolação das posições dos vértices que compõem as poses-chave.

Na parametrização geométrica, oferece-se um conjunto de mecanismos geométricos para a realização da animação facial. Uma série de transformações geométricas como rotação, translação e mudança de escala são empregadas para a produção do movimento facial (PARKE, 1974), (KALRA; MANGINI; MAGNENAT-THALMANN, 1992).

Já a parametrização *data-driven* utiliza como parâmetros dados obtidos com equipamentos especiais de mensuração de ações humanas. Encontra-se freqüentemente na literatura a parametrização *data-driven* combinadas a outras estratégias como, por exemplo, com a parametrização geométrica (DE MARTINO; MAGALHÃES, 2004), (DE MARTINO; MAGALHÃES; VIOLARO, 2006), ou com a simulação biomecânica (ESSA; BASU; DARRELL; PENTLAND, 1996).

Com o objetivo de produzir animações realistas, a estratégia de simulação biomecânica tem como princípio básico a representação de propriedades físicas da face, conforme a figura 7 (LEE; TERZOPOULOS; WATERS, 1995). Nesta estratégia,

modela-se os músculos faciais como parâmetros de manipulação, aptos a desencadear a movimentação da face virtual. O tecido facial é usualmente representado por uma malha de sistemas massa-mola, que procura simular as propriedades elásticas da pele humana (LEE; TERZOPOULOS; WATERS, 1995), (LUCERO; MUNHALL, 1999).

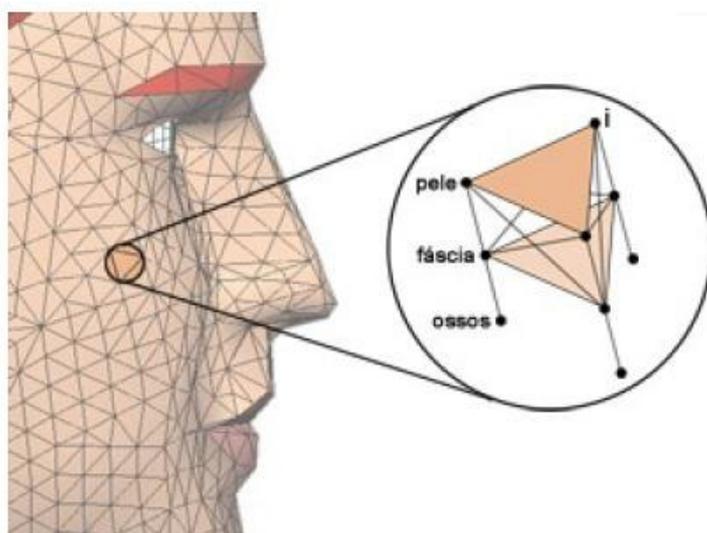


Figura 7: Detalhes da malha facial

A escolha entre uma técnica ou outra está diretamente ligada à forma com que a face está sendo representada na aplicação. Esta forma depende do objetivo da aplicação, por exemplo, se é desejável criar um personagem em estilo *cartoon* ou um personagem dispendo de um grau elevado de semelhança ao ser humano, o que vai resultar na escolha da melhor abordagem de representação de faces a ser utilizada.

3 PROPOSIÇÃO DO TRABALHO

3.1 Comunicação Diretor-Personagem

Dentro do contexto de *storytelling* interativo com o auxílio de um diretor, a comunicação do personagem com o diretor é um fator muito importante que define a dramatização da cena. O diretor deve se comunicar com o ator (personagem) através de uma linguagem específica, enviando-lhe parâmetros a respeito de sua posição no espaço tridimensional, sobre a ação que ele deve desempenhar em determinado momento ou qual expressão facial emotiva ele deve demonstrar e em que espaço de tempo isso deve acontecer. O diretor também pode solicitar ao ator que este o envie posições nas quais a câmera deve se posicionar para realizar tomadas específicas, como um *close* no rosto do personagem ou uma tomada *over the shoulder* (acima dos ombros), posições estas muito utilizadas na cinematografia. Neste caso, o ator retorna a posição e a direção em que a câmera deve ser posicionar para que tal tomada seja realizada. Isto se faz necessário, uma vez que o diretor não possui informações sobre a posição em que se encontra a cabeça do personagem ou localizações específicas do corpo deste, mas sim possui conhecimento apenas de um ponto ao qual a entidade personagem como um todo está localizada no espaço tridimensional.

Dentro do projeto de *storytelling* interativo com auxílio de um diretor, um trabalho que foi desenvolvido em paralelo ao descrito nesta monografia (RAMOS, 2008) é o de criação de um personagem que possa desempenhar qualquer ação (movimento corporal) solicitada pela câmera, mesmo que esta não tenha sido definida na criação do personagem. Não cabe ao escopo desta monografia descrevê-lo em detalhes, porém, é importante citá-lo por estar intimamente relacionado à proposta que está sendo descrita aqui, uma vez que ambos definem o personagem que irá interagir com a câmera (seja através da definição de seus movimentos corporais ou de suas expressões faciais).

É importante salientar que o diretor possui apenas o conhecimento do personagem como um todo, independente deste ser formado por corpo e face, que desempenham papéis diversificados. Por exemplo, ao solicitar que o personagem deve correr estando muito apavorado, o diretor envia esses parâmetros (correr,

apavorado) ao personagem, sendo que este deve separar estas informações e enviá-las para o corpo (ação = correr) e para a face (expressão facial = apavorado). Desta forma, temos o personagem da animação estruturado da seguinte maneira: uma entidade corpo, (formado por tronco, membros e cabeça) e a entidade face (formada pelos elementos que compõem a face – olhos, sobrancelhas e boca – e que definem as expressões faciais do personagem). A figura 8 mostra como é feita a comunicação entre o diretor, a câmera e o personagem (formado por corpo e face).

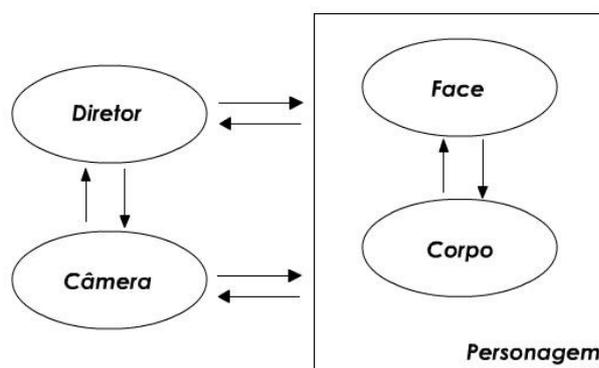


Figura 8: Comunicação entre diretor, câmera e personagem

Uma vez que a face do personagem é implementada de forma independente da cabeça, onde será inserida, esta deverá lhe informar a posição e direção que a face deverá possuir a cada momento da animação, para que ela possa acompanhar o movimento da cabeça na ação em que o personagem estiver executando.

3.2 Modelagem e Animação da Face do Personagem

A seguir, será descrita a abordagem e as ferramentas utilizadas para representação das diferentes expressões faciais que o personagem deverá possuir, juntamente com a técnica de animação facial utilizada. Para o desenvolvimento deste trabalho, utilizou-se o *Bloodshed Dev-C++*, um ambiente de desenvolvimento de aplicações para as linguagens C/C++ para *Windows*. A programação foi feita através da linguagem de programação C++, fazendo-se uso da biblioteca *OpenGL* (*Open Graphics Library*), para geração dos elementos gráficos da aplicação. A API *OpenGL* possui centenas de funções que podem ser usadas para o desenvolvimento e visualização de objetos gráficos tridimensionais complexos através de simples primitivas nela definidas.

Conforme citado anteriormente, existem diferentes conclusões por parte de estudiosos no que se refere às expressões básicas que definem as principais emoções. Neste trabalho, foi utilizado como base a teoria que definem 6 (seis) como sendo as expressões básicas, sendo elas: raiva, nojo, medo, alegria, tristeza e surpresa (PARKE; WATERS, 1996). Além disso, acrescentou-se a esse conjunto de expressões a face neutra, ou seja, aquela expressa pelo personagem quando este não possui definição alguma sobre o estado emocional que deverá estar apresentando. Essa expressão é definida como padrão na geração da face e permanece ativa até que seja informado à face a nova expressão que a mesma deve passar a ter. Na figura 9 (RODRIGUES; FEIJÓ; VELHO, 2005), tem-se cada uma das seis expressões básicas a serem utilizadas neste trabalho representadas em uma face humana, para efeito de visualização.

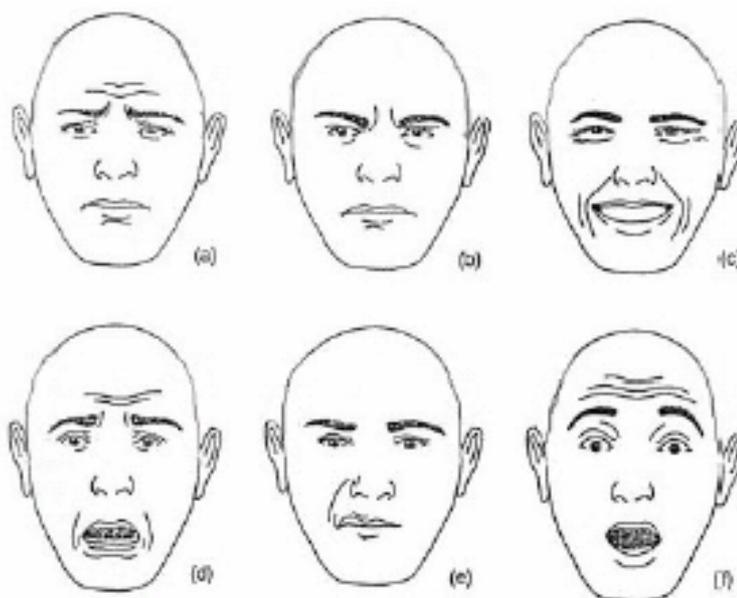


Figura 9: Cada uma das 6 expressões básicas (tristeza, raiva, alegria, medo, nojo e surpresa)

É importante salientar que o objetivo deste trabalho não é criar expressões faciais e animá-las em personagens com uma representação física realista. Desta forma, as faces geradas possuem uma representação física bastante simples, como é o caso do personagem *Gingy*, já citado na seção 2.3. Assim, cada expressão é formada por sobrancelhas, olhos e boca. A diferença entre uma expressão e outra se dá apenas por alterações nas formas das sobrancelhas e da boca, elementos chave para a identificação de uma determinada expressão facial.

4 IMPLEMENTAÇÃO DO TRABALHO

4.1 Criação dos Elementos da Face

Foram utilizadas diferentes abordagens para representação de cada um dos elementos da face (sobrancelhas, olhos e boca). Uma vez que a distinção entre uma expressão e outra se dá através da forma das sobrancelhas e da boca, estes elementos devem possuir uma estrutura onde seja possível fazer alterações nas suas formas, conforme a expressão facial que se deseja gerar. Possibilitar essas deformações de uma maneira que não seja muito complexa é uma questão que também deve ser considerada. Uma abordagem que se mostrou apropriada levando-se em conta essas considerações é a representação das sobrancelhas e da boca utilizando-se curvas paramétricas para guiar a forma física destes dois elementos, uma vez que elas podem ser modificadas facilmente gerando diferentes formas, de acordo com a necessidade. Algumas considerações sobre a geração de curvas para compreensão das abordagens utilizadas podem ser encontradas no apêndice A.

4.2 Geração das Sobrancelhas e da Boca do Personagem

As sobrancelhas são um dos elementos mais expressivos no que diz respeito ao reconhecimento de uma expressão facial. Elas podem estar mais arqueadas, em uma expressão de surpresa, ou mais suaves, quando se está sorrindo. A boca também possui um papel muito importante neste contexto. Dependendo da expressão contida na face, ela pode estar com uma certa curvatura para baixo, demonstrando tristeza, ou aberta, em uma expressão de pavor. Sem a análise desses dois elementos, certamente o reconhecimento de expressões faciais se torna uma tarefa difícil.

Conforme já citado anteriormente, as formas de representação desses elementos chave devem possibilitar deformações em suas estruturas que gerem as diferentes formas que eles devem possuir para cada uma das expressões básicas utilizadas. Assim, utilizou-se curvas paramétricas para gerar a forma física destes dois elementos, uma vez que é simples modificá-las, gerando diferentes formas, de acordo com a necessidade.

Devido às características das curvas *B-Spline*, esta se mostrou mais vantajosa em relação à curva de *Bézier* para gerar a forma da boca do personagem. Isso devido ao fato de que as curvas *B-Spline* de grau 3 são bastante apropriadas para a geração de curvas fechadas, por não necessitarem de técnicas desenvolvidas para garantir a continuidade desejada no ponto de junção do primeiro e último segmento, como seria o caso da curva de *Bézier*.

Assim, para a geração da boca do personagem, utilizou-se *B-Splines* de grau 3, juntamente com 12 pontos de controle. O último segmento gerado interliga-se com o primeiro. Desta forma, temos uma curva fechada que, dependendo da posição dos pontos de controle, obtém-se diferentes formas para a boca a ser gerada para uma determinada expressão. A figura 10 mostra a boca do personagem representada pelos pontos de controle e a curva gerada através deles.

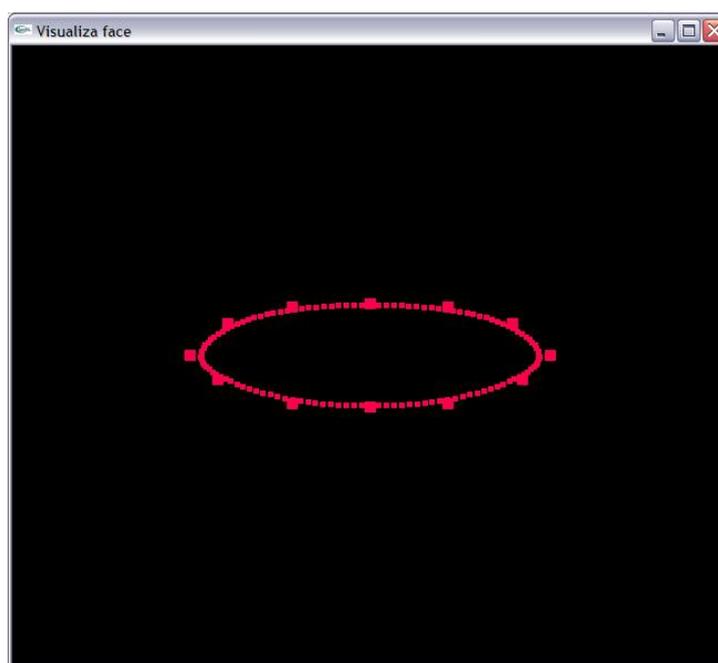


Figura 10: Pontos de controle e a curva *B-Spline* gerada para a boca

Nota-se que os pontos de controle (pontos maiores) estão sendo tocados pela curva. Isto se dá ao fato de que, em determinadas regiões, estes pontos são praticamente colineares (região superior e inferior da curva fechada). Alterando-se os pontos de controle, tem-se diferentes curvas geradas. Na figura 11 tem-se uma representação da boca para a expressão “feliz”.

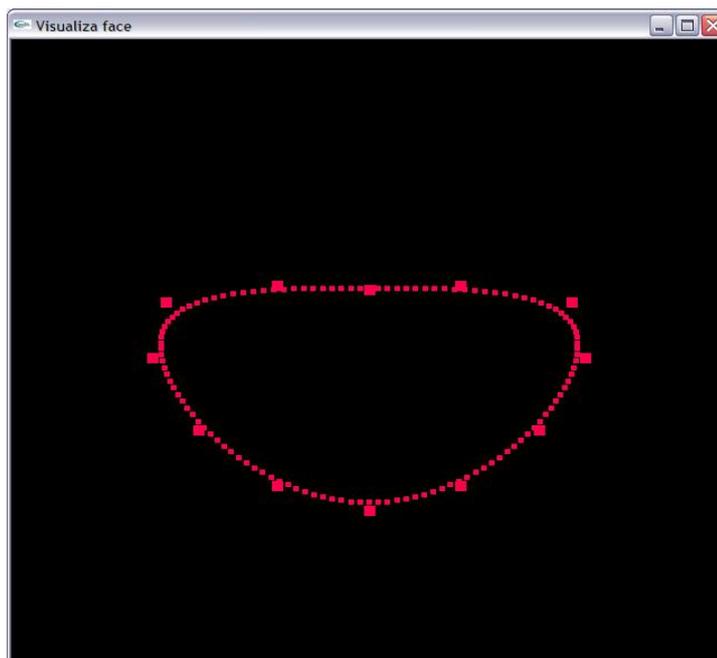


Figura 11. Pontos de controle e a curva *B-Spline* gerada para a boca “feliz”

Uma vez definido o formato base da boca, cria-se uma malha que envolve a curva *B-Spline* gerada. Para isso, é necessário que para cada ponto da curva gerada (pontos menores), seja calculado um vetor perpendicular a ele (vetor normal). Para o cálculo desse vetor normal, utilizou-se as coordenadas dos pontos que definem a curva, criando-se vetores diretores para cada um deles, conforme a figura 12.

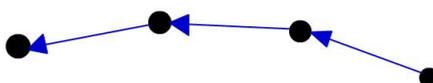


Figura 12: Vetores diretores dos pontos da curva

Uma vez tendo-se os vetores diretores de cada ponto da curva, calculou-se os vetores perpendiculares a eles, mostrados na figura 13.

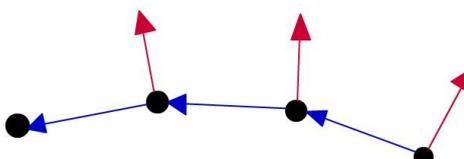


Figura 13: Vetores perpendiculares dos pontos da curva

Fazendo-se uso de uma função *glRotate* da API gráfica utilizada, onde, dado um eixo arbitrário, realiza a rotação de um vetor ao redor deste, teve-se como resultado “anéis” que envolvem cada um dos pontos que definem a boca do personagem. A figura 14 mostra o processo realizado, também conhecido como *sweep* rotacional, utilizando-se o vetor diretor de cada ponto da curva como eixo de rotação e o vetor perpendicular do ponto correspondente como vetor a ser rotacionado. Considera-se o vetor diretor do ponto (eixo de rotação) como se estivesse saindo do plano do papel.

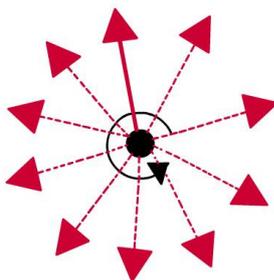


Figura 14: Sweep rotacional em cada ponto da curva

Através desses anéis, gera-se a malha da boca, interligando os pontos de cada um dos anéis com os dos seus pontos vizinhos. A figura 15 mostra a malha resultante da junção dos pontos dos anéis de toda a curva gerada.

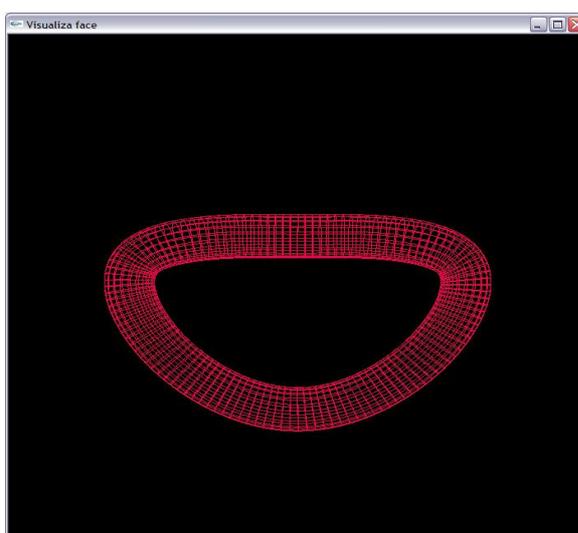


Figura 15: Malha da boca

Uma vez gerada a malha e aplicando-se recursos de iluminação nela, tem-se como resultado a representação final da boca do personagem, conforme a figura 16.

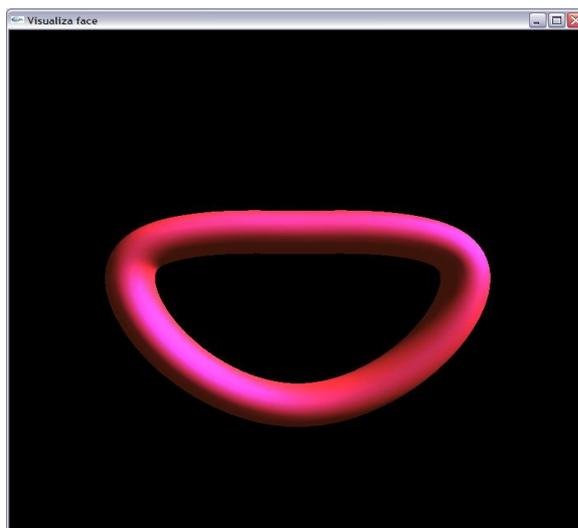


Figura 16: Malha da boca na versão final

Para a geração das sobrancelhas, a técnica que se mostrou mais apropriada foi modelá-las através de curvas de *Bézier*. Neste caso, não é necessário obter uma curva fechada, como no caso da boca. Logo o problema da continuidade no ponto de junção do primeiro e último segmento não precisa ser considerado. Outra vantagem é a não necessidade de se definir mais pontos de controle do que o número de segmentos gerados, o que acontece com as curvas *B-Spline*. Fazendo-se uso de um único segmento de curva gerado através de 4 pontos de controle (*Bézier* de grau 3), tem-se a possibilidade de gerar diversas curvas, de forma que a sua utilização para representação dos diferentes formatos das sobrancelhas se mostrou de forma suficientemente satisfatória.

A figura 17 mostra os pontos de controle para uma das sobrancelhas do personagem, bem como a curva Bézier de grau 3 gerada a partir deles.

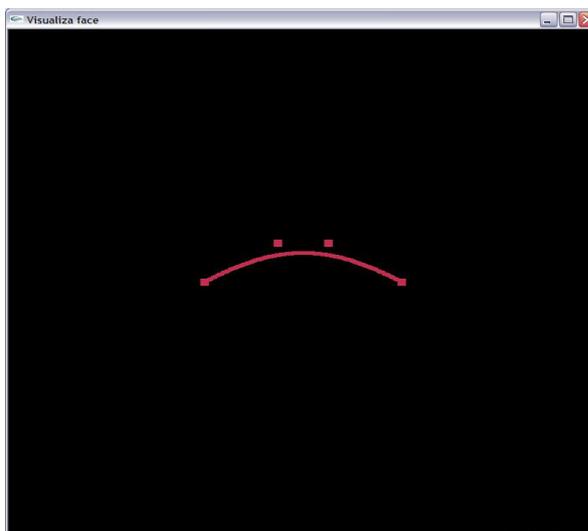


Figura 17: Pontos de controle e a curva *Bézier* gerada para a sobrancelha

Nas curvas de *Bézier* os pontos das extremidades são tocados pela curva, diferentemente dos demais pontos que são apenas aproximados (ver apêndice A). Para a geração da malha da sobrancelha, utilizou-se a mesma técnica descrita para a boca do personagem, rotacionando-se os vetores normais a cada ponto da curva, gerando, assim, os “anéis” de pontos a serem interligados, formando a malha. A figura 18 mostra a malha gerada para a sobrancelha.

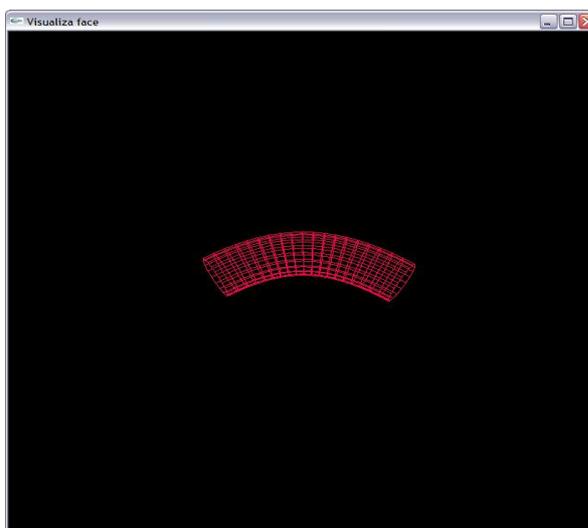


Figura 18: Malha da sobrancelha

Para gerar as extremidades arredondadas da sobrancelha, utilizou-se uma das funções pré-definidas da *glut* (*OpenGL Utility Toolkit*), para desenho de uma esfera. Na figura 19, temos a imagem da malha da sobrancelha juntamente com as esferas para arredondamento das extremidades.

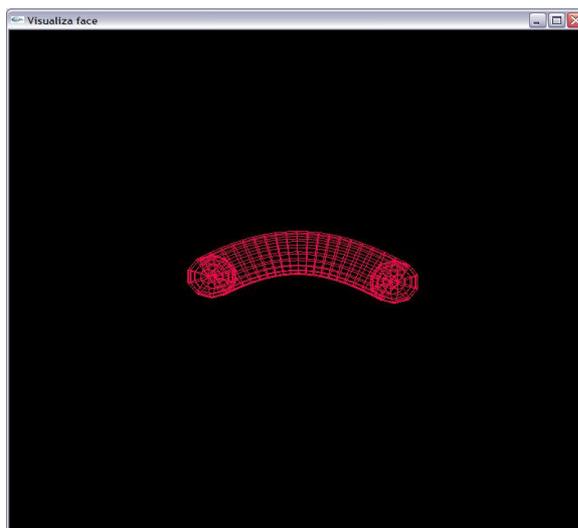


Figura 19: Malha da sobrancelha com as esferas nas extremidades

Aplicando os recursos para iluminação, como foi realizado para a boca, tem-se a representação final da sobrancelha da face, conforme a figura 20.

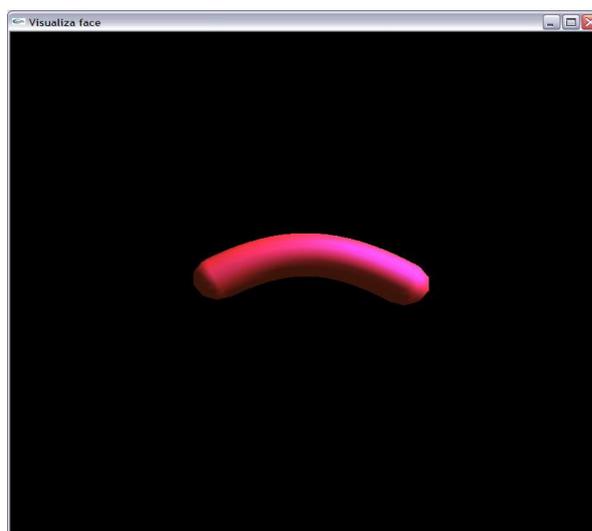


Figura 20: Representação final da sobrancelha da face

Conforme citado anteriormente, os olhos da face não sofrem alterações na sua forma, permanecendo estáticos durante a transição das expressões. Assim, para sua representação na face do personagem, utilizou-se a mesma função de geração de esferas contida na biblioteca *OpenGL*. A figura 21 mostra a junção de todos elementos que compõem a face (sobrancelhas, olhos e boca). Nela, é demonstrada a expressão “neutra” gerada para a face. Na figura 22, tem-se a

expressão “feliz”, através da mudança dos pontos de controle da boca e das sobrancelhas.

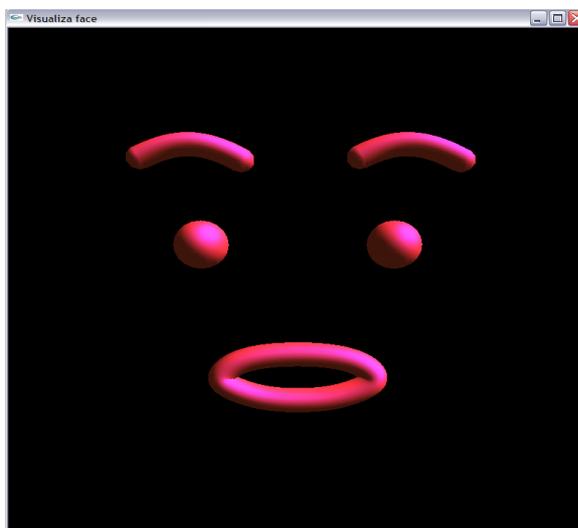


Figura 21: Expressão “neutra” para a face

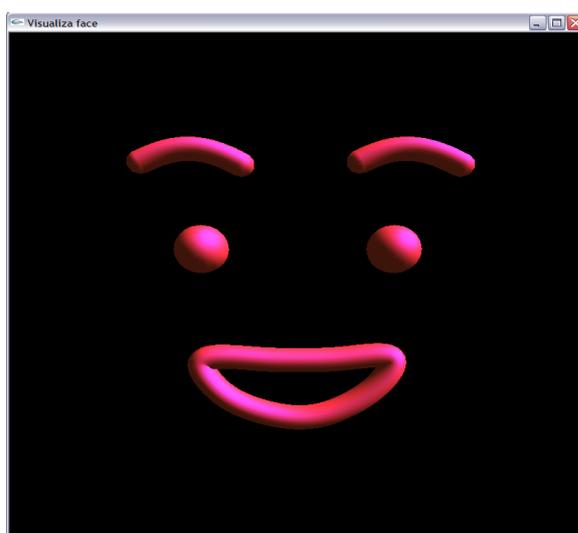


Figura 22: Expressão “feliz” para a face

4.2.1 O Editor Facial

A seção anterior abordou a forma com que os elementos da face (em especial aqueles que definem a distinção entre as seis expressões básicas mais a neutra) estão representados no sistema, através do detalhamento da geração de cada elemento, etapa por etapa. Uma vez tendo-se a posição dos pontos de controle da boca e das sobrancelhas, basta gerar a curva correspondente (*Bézier* ou *B-Spline*)

e, por cima dela, criar a malha, preenchê-la e iluminá-la para gerar a face que se deseja mostrar. Desta forma, a definição das posições de cada ponto de controle se mostra como tendo o mais importante papel na geração das expressões faciais do personagem.

Uma forma de se gerar esses pontos de controle que definem as formas das curvas que geram cada expressão é utilizar um programa de modelagem gráfica tridimensional onde se possa criar pontos aleatórios e ter acesso à posição deles dado um eixo de coordenadas conhecido. Um exemplo de *software* que pode ser citado nesta categoria é o *3DMax Studio*. Neste, é fornecido um eixo de coordenadas tridimensional (x, y e z), onde o usuário pode usar diferentes ferramentas para gerar diversas formas no espaço tridimensional, inclusive curvas paramétricas. Após a geração dessas formas, pode-se ter acesso a cada ponto que se queira obter a posição no espaço simplesmente posicionando o *mouse* sobre o ponto desejado.

Inicialmente, a idéia para a obtenção da posição dos pontos de controle das curvas da boca e sobrancelhas da face era a de se utilizar o programa descrito anteriormente para a captura dessas informações, através do processo descrito acima. Esse processo não se mostra muito prático, uma vez que se torna necessário anotar cada posição de cada ponto de controle e transcrevê-las dentro das funções para a geração das expressões faciais. Uma alternativa é a utilização de exportadores específicos para salvar as informações sobre os pontos modelados. Esses arquivos podem ser posteriormente utilizados para o carregamento dos pontos referentes a uma dada expressão para utilização nas funções geradoras das expressões faciais. Um exemplo de exportador para o *3D Max Studio* é um *plugin* instalado automaticamente com o OGRE (*Open Source Graphics Engine*), uma *engine* de auxílio na geração de aplicativos gráficos como jogos e animações (OGRE, 2008).

Devido à necessidade de uma técnica específica e prática para a geração dos pontos de controle necessários para a definição de cada expressão, decidiu-se pela implementação de um editor facial próprio, ao invés de se utilizar programas e *plugins* já existentes. Este editor possui interatividade em tempo real na geração das expressões, ou seja, o usuário tem acesso à face neutra onde os elementos boca e sobrancelhas são representados apenas por pontos de controle e a curva correspondente à eles (*Bézier*, no primeiro caso e *B-Splines* no segundo) e pode

alterar a posição desses pontos com o *mouse* obtendo, no mesmo instante, o formato da curva gerada por essa modificação. Ao se executar o editor facial, a face neutra é então visualizada. Com o auxílio do mouse altera-se a posição dos pontos de controle desejados e, após se chegar ao resultado desejado, guarda-se a posição de cada ponto de controle na sua atual posição, representando uma expressão facial específica. O salvamento desses pontos se dá através do teclado, pressionando a tecla *s*. Ao pressionar a tecla *r*, tem-se a opção de restaurar os pontos alterados para as suas posições iniciais (expressão neutra) e começar todo o processo novamente.

A figura 23 mostra a tela inicial ao se executar o editor facial descrito.

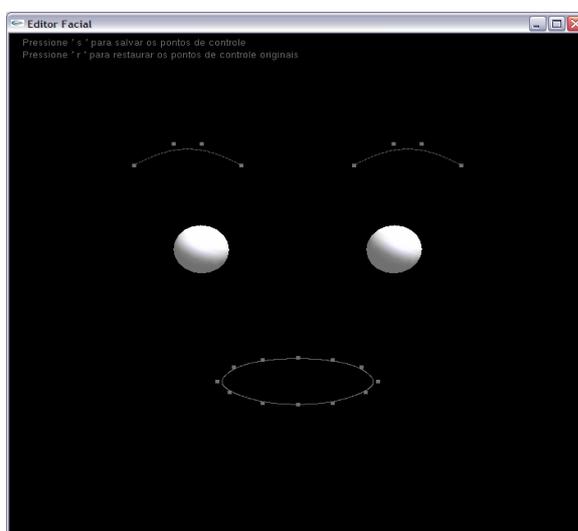


Figura 23: Tela inicial do editor facial

Para a geração da expressão “raiva”, altera-se os pontos de controle da forma desejada, resultando na curva conforme a figura 24.

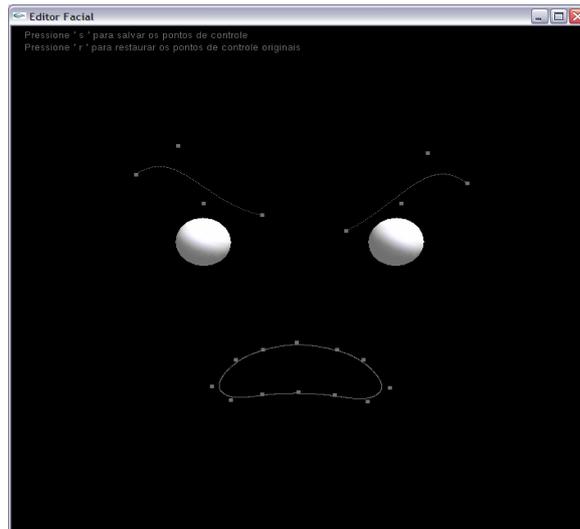


Figura 24: Expressão de raiva gerada no editor facial

Após o salvamento dos pontos de controle, tem-se um arquivo TXT com a informação necessária para geração da expressão, conforme a figura 25.

```
//Sobrancelha Esquerda//
ptos_controle1[0].set(56.0, -30.0, 0.0);
ptos_controle1[1].set(32.0, 14.0, 0.0);
ptos_controle1[2].set(-12.0, 10.0, 0.0);
ptos_controle1[3].set(-39.0, 0.0, 0.0);
//Sobrancelha Direita//
ptos_controle2[0].set(39.0, 0.0, 0.0);
ptos_controle2[1].set(10.0, 39.0, 0.0);
ptos_controle2[2].set(-10.0, 18.0, 0.0);
ptos_controle2[3].set(-71.0, -82.0, 0.0);
//Boca//
ptos_controle[0].set(85.0, -21.0, 0.0);
ptos_controle[1].set(31.0, 8.0, 0.0);
ptos_controle[2].set(17.0, 12.0, 0.0);
ptos_controle[3].set(-3.0, 32.0, 0.0);
ptos_controle[4].set(-17.0, 12.0, 0.0);
ptos_controle[5].set(-55.0, 37.0, 0.0);
ptos_controle[6].set(-59.0, 3.0, 0.0);
ptos_controle[7].set(-33.0, -6.0, 0.0);
ptos_controle[8].set(-17.0, -12.0, 0.0);
ptos_controle[9].set(0.0, -13.0, 0.0);
ptos_controle[10].set(17.0, -12.0, 0.0);
ptos_controle[11].set(33.0, -6.0, 0.0);
```

Figura 25: Informações dos pontos de controle de uma dada expressão

Uma vez tendo-se o arquivo com essas informações, basta copiar o conteúdo e transportá-lo para a aplicação que usa esses pontos para a geração das faces do personagem. Esta forma de obtenção das informações do arquivo certamente poderia ser melhorada através da importação automática dos pontos de controle pela aplicação que os utiliza, sendo citada a seguir como um dos trabalhos futuros a ser desenvolvido.

4.2.2 A Interpolação Linear e a Transição de Expressões Faciais

Para a animação facial entre expressões, fez-se uso da técnica de interpolação linear, onde, para um dado valor t do parâmetro de interpolação dentro do intervalo $[0, 1]$, gera-se uma face “intermediária”, cujos pontos que definem os seus elementos de face modificáveis são representados pela interpolação dos pontos da face inicial com a final, em t . Assim, para diversos valores de t haverá diversas faces intermediárias geradas entre a face que representa a expressão atual do personagem e a face correspondente à expressão final que se deseja obter, cada uma delas, em seqüência, representando uma aproximação da face inicial com a face final.

A seguir, é mostrado um exemplo da interpolação entre duas expressões básicas, mostrando-se a face que representa a expressão inicial, a expressão final e a face intermediária obtida para o valor $t = 0,5$ do parâmetro de interpolação. Para este valor, tem-se a face que representa exatamente a expressão na metade do tempo em que dura a animação facial. Na figura 26, temos a representação para a expressão de nojo, definida como a expressão inicial do personagem e na figura 27, tem-se a expressão final, de alegria.



Figura 26: Expressão inicial de nojo



Figura 27: Expressão final de alegria

Na seqüência, a figura 28 mostra a expressão intermediária gerada para o valor de $t = 0,5$.



Figura 28: Expressão intermediária entre nojo e alegria

Desta forma, definindo-se quaisquer expressões inicial e final, obtém-se a animação facial de transição entre expressões faciais, variando o parâmetro de interpolação dentro do intervalo $[0, 1]$.

5 CONCLUSÃO E TRABALHOS FUTUROS

5.1 Contribuições Alcançadas

Através do desenvolvimento do atual projeto, obteve-se como resultado uma técnica de representação de expressões faciais em personagens simplicados, que podem ser usados em aplicações de *storytelling*, atribuindo à esses personagens um comportamento facial semelhante ao do ser humano, resultando em animações mais realistas e envolventes ao usuário final.

A técnica utilizada possibilitou a definição da face de forma tridimensional, onde os elementos que a definem (sobrancelhas, olhos e boca) possuem uma representação que pode ser visualizada com a câmera posicionada em qualquer orientação no espaço 3D. Assim, a geração de uma malha a partir das curvas que definem as sobrancelhas e a boca do personagem possibilitou uma representação física para a face do mesmo que atendessem aos objetivos desejados na realização deste trabalho.

As curvas de *Bézier* se mostraram bastante apropriadas para a geração das sobrancelhas, visto que elas são definidas por curvas abertas, e o fato da curva tocar os pontos de controle das extremidades apresenta vantagens, pois acaba por facilitar a manipulação da mesma. Na geração da boca, seriam necessárias técnicas que garantissem uma continuidade C^2 (apêndice A) para que, na junção dos pontos extremos, não houvesse a formação de “pontas”. Por isso, as curvas *B-Spline* foram utilizadas neste caso, por possuírem a característica de sempre garantirem uma continuidade C^2 ao se utilizar uma curva de grau 3. A quantidade de pontos de controle escolhidos para a geração das curvas se mostrou apropriado para se gerar as diversas formas desejadas na definição das diferentes expressões. Com 4 pontos de controle para a sobrancelha, foi possível gerar a forma para as 6 expressões básicas sem que um polinômio de grau muito alto tivesse que ser utilizado. Para a geração da boca, 12 pontos de controle possibilitaram a definição das mais variadas formas para as diversas expressões.

5.2 Trabalhos Futuros

Como trabalho futuro, pretende-se melhorar a técnica de transições faciais oferecendo não somente seis expressões faciais básicas e a neutra, mas também expressões geradas através da combinação destas, gerando uma gama maior de possibilidade de expressões representadas. Estas combinações podem ser feitas com duas ou mais expressões, por exemplo, gerando-se uma expressão de “tristeza + assustado” ou “raiva + nojo + medo”. Desta forma, ter-se-ia uma riqueza maior na representação da animação facial do personagem.

Outra melhoria a ser feita, já citada na seção 3.1.3, se refere ao melhoramento da técnica de obtenção dos pontos de controle gerados pelo editor facial. Pretende-se implementar uma técnica de exportação dos pontos em um arquivo em formato XML, utilizando *tags* para representar cada um dos elementos da face (sobrancelhas e boca), contendo a lista de vértices representando cada ponto de controle. Desta forma, a aplicação importaria os pontos de controle do documento gerando de maneira automática a expressão representada por eles.

A utilização de expressões faciais para diálogos entre personagens também é um assunto a ser tratado em pesquisas futuras. Possibilitar a interação de personagens através da fala, gerando, para cada sílaba, a forma correta a ser representada pela boca, bem como outros aspectos que englobam o tema, aplicados à *storytelling*, trariam resultados bastante interessantes em termos realistas.

6 REFERÊNCIAS BIBLIOGRÁFICAS

POZZER, C. T.. **Um sistema para geração, interação e visualização tridimensional de histórias para tv interativa**. Tese de Doutorado, Departamento de Informática, PUC-Rio, 2005.

MATEAS, M.; STERN, A.. **Architecture, authorial idioms and early observations os the interactive drama façade**. Technical report, School of Computer Science, Carnegie Mellon University, Pittsburg, PA, December 2002.

GLASSNER, A.. **Interactive Storytelling: Techniques for 21st Century Fiction**. AK Peters LTD, 2004.

GRASBON, D.; BRAUN.. **A morphological approach to interactive storytelling**. In: Fleischmann, M., Strauss, W., editors, Proceedings: CAST01, Living in The Mixies Realities. Special Issue of Netzspannung.org/journal, The Magazine for Media Production and Inter-Media Research, p. 337 – 340, Sankt Augustin, Germany, 2001.

MATEAS, M.; STERN, A.. **Façade: An experiment in building a fully-realized interactive drama**. In: Game Developers Conference, Game Design Track, San Jose, CA, March 2003.

CAVAZZA, M.; CHARLES, F.; MEAD, S.. **Character-based interactive storytelling**. IEEE Intelligent System, special issue on AI in Interactive Entertainment, 17(4):17-24, July 2002.

MATEAS, M.; STERN A.. **Towards integrating plot and character for interactive drama**. In: Dautenhahn, K., editor, Socially Intelligent Agents: The Human in the Loop, AAI Fall Symposium, Technical Report, p. 113-118, Menlo Park, CA, 2000. AAI Press.

YOUNG, R.. **An overview of the mimesis architecture: Integrating narrative control into a gaming environment**. In: AAI Spring Symposium on Artificial Intelligent and Interactive Entertainment, AAI Technical Report, p. 78-81, Stanford, CA, March 2001. AAI Press.

CAVAZZA, M.; CHARLES, F.; MEAD S.. **Interacting with virtual agents in interactive storytelling**. In: ACM Joint Conference on Autonomous Agens and Multi-Agent Systems, AAAMAS'02, p.318-325, 2002.

CAVAZZA, M.; CHARLES, F.; MEAD, S.. **Ai-based animation for interactive storytelling**. In: Proceedings of Computer Animation, p. 113-120, Seoul, Korea, 2001. IEEE Computer Society Press.

CHARLES, F.; CAVAZZA, M.; MEAD, S.. **Character-driven story generation in interactive storytelling**. Technical report, VSMM, Berkeley, 2001.

CAVAZZA, M.; CHARLES, F.; MEAD S.. **Emergent situations in interactive storytelling**. In: SAC '02: Proceedings of the 2002 ACM Symposium on Applied Computing, p. 1080-1085, Madrid, Spain, 2002. ACM Press.

SPIERLING, U.; BRAUN, N.; IURGEL, I.; GRASBON, D.. **Setting the scene: playing digital director in interactive stroytelling and creation**. Computers & Graphics, 26:31-44, 2002.

CRAWFORD, C.. **Assumptions underlying the erasmatron interactive storytelling engine**. In: Mateas, M.; Sengers, P., editors, Proceedings of the AAAI Fall Symposium: Narrative Intelligence, Technical Report, P. 112-114, Menlo Park, CA, 1999. AAAI Press.

HE, L.; COHEN, M. F.; SALESIN, D.H.. **The virtual conematographer: a paradigm for automatic real-time camera control and directing**. In: Proceedings of the 23RD Annual Conference on Computer Graphics and Interactive Techniques. ACM Siggraph, volumen 30, p. 217-224, August 1996.

ORTONY, A., &TURNER, T.T. (1990). **What's basic about basic-emotions?** Psychological Review, 97, 315-331.

KÖRTING, T., SILVA, F., COSTA R., BOTELHO, S., BICHO, A.. **Um estudo sobre a animação tridimensional de faces**. Engenharia da Computação – Fundação Universidade Federal do Rio Grande, Rio Grande, RS, DCA-FEEC-Unicamp, Campinas, SP, 2006.

RODRIGUES, P.; FEIJÓ, B.; VELHO, L.. **Modelos Avançados de animação facial – o estado da arte**. Laboratório VISGRAPH, Instituto de Matemática Pura e Aplicada, Pontífica Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ, 2005.

PARKE, F. I.; WATERS, K.. **Computer Facial Animation**. AK Peters, ISBN 1-56881-014-8,1996.

ZHANG, Q.; LIU, Z.; GUO, B.; SHUM, H.. **Geometry-Driven Photorealistic Facial Expression Synthesis**. Proceedings of the 2003 ACM Siggraph/Eurographics Symposium on Computer Animation, San Diego, California, ISBN/ISSN: 1727-5288, 1-58113-659-5, p.117-189.

CÔRREA, R.; MAGALHÃES, L.; DE MARTINO, J.. **Sistema de animação facial biomecânica**. Faculdade de Engenharia Elétrica, Depto. De Engenharia de Computação e Automação Industrial, Universidade de Campinas (UNICAMP), Campinas, SP, 2005.

PARKE, F. I.. **A parametric model for human faces**. PhD THESIS, Universidade de Utah, Salt Lake City, Utah, UTEC-CSc-75-047, 1974.

KALRA, P.; MANGINI, A.; MAGNENAT-THALMANN, N.; THALMANN, D.. **Simulation of facial muscle actions based rational free form deformations**. Proceedings of Eurographics 92, p. 59-69, 1992.

DE MARTINO, J.; MAGALHÃES, L.. **Um conjunto de visemas para a cabeça falante de português do Brasil**. III Congresso Ibero Americano IBERDISCAP 2004 – Tecnologia de Apoio a la Discapacidade, 2004.

DE MARTINO, J.; MAGALHÃES, L.; VIOLARO, F.. **Facial Animaion Based on Context-Dependent Visemes**. Computers & Graphics, v.30, n.6, 2006.

ESSA, I.; BASU, S.; DARRELL, T.; PENTLAND, A.. **Modeling, tracking and interactive animation of faces and heads using input from video**. CA'96 Proceedings of The Computer Animation. Washington, DC, USA: IEEE Computer Society, p.68-79, 1996.

LEE, Y.; TERZOPOULOS, D.; WATERS, K.. **Realistic modeling for facial animation**. SIGGRAPH'95 Proceedings of the 22ND Annual Conference on Computer Graphics and Interactive Techniques, 1995, p.55-62.

LUCERO, J.; MUNHALL, K.. **A model of facial biomechanics for speech production**. The Journal of the Acoustical Society of America, v.106, n.5, p. 2834-2842, 1999.

WIKIPEDIA. Desenvolvido pela Wikimedia Foundation. **Bézier curve**. Apresenta conteúdo enciclopédico. 2007. Disponível em: <http://en.wikipedia.org/wiki/B%C3%A9zier_curve>. Acesso em: novembro de 2008.

MARKOWITZ, M.. **Spacewar**. 2000. Disponível em <<http://www3.sympatico.ca/maury/games/space/spacewar.html>>. Acesso em novembro de 2008.

OGRE.. **OGRE 3D: Open Source graphics engine**. 2000. Disponível em <http://www.ogre3d.org/index.php?option=com_content&task=blogcategory&id=0&Itemid=98>. Acesso em novembro de 2008.

CIARLINI, A.; POZZER, C. T.; FURTADO, A.; FEIJÓ, B.. **A logic-based tool for interactive generation ad dramatization of stories**. Proceedings of the ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (ACE2005), Valencia, p. 133-140, 2005.

ROGERS, D.; ADAMS, j.. **Mathematical Elements for Computer Graphics**, 2nd Edition. Mc Graw Hill, 1990.

FILMONIC.. **Molesting your Film Shaft**. 2008. Disponível em <<http://filmonic.com/shrek-4-titled>>. Acesso em novembro de 2008.

RAMOS, H.. **Implementação de um Personagem com Esqueleto para Dramatização de Storytelling com Auxílio à Câmera e ao Diretor**. Monografia de Graduação, Departamento de Eletrônica e Computação, UFSM, 2008.

APÊNDICE A – Curvas Paramétricas

Curvas paramétricas geralmente são representadas em sua extensão por segmentos menores de curvas, representados por polinômios de baixo grau. Isto se dá pelo fato de que curvas mais complexas podem não ser facilmente descritas por expressões analíticas em toda sua extensão, sendo necessário dividi-la em segmentos menores e representar esses segmentos pelos seus respectivos polinômios. Desta forma, para gerar a curva completa, unem-se as extremidades dos segmentos gerados separadamente obtendo-se, assim, a curva completa.

Geralmente são utilizados segmentos de grau 3, devido ao fato de que quanto maior for o grau das funções, mais complexos são os cálculos a serem executados, influenciando diretamente no desempenho de aplicações que os utilizam. Além disso, outro problema que pode vir a surgir é a instabilidade numérica, onde o polinômio não é capaz de representar os valores corretos para qualquer ponto da curva. O fato de polinômios de grau 3 não serem planares, ou seja, representarem o espaço tridimensional também influi diretamente nesta decisão.

Cada segmento de curva é definido por um conjunto de pontos discretos, chamados pontos de controle (*control points*), e por funções básicas (uma para cada ponto) que combinam a influência dos pontos na geração da curva. Essas funções são conhecidas como *blending functions*. No caso de segmentos de curva de grau 3, são utilizados 4 pontos de controle e 4 funções básicas, que geram a curva a partir dos pontos de controle. Para que a união das curvas seja feita de forma adequada, muitas vezes é necessário que a curva resultante da união possua curvatura contínua, ou seja, que a união entre um segmento e outro não gere “pontas” (mudanças bruscas de direção) em sua extensão. Para isso, algumas restrições de continuidade devem ser respeitadas quando se implementa esse tipo de abordagem. Existem três tipos de continuidade paramétrica, conforme listado abaixo:

- C^0 – duas curvas se encontram;
- C^1 – duas curvas se encontram e ambas devem ter tangentes comuns ao ponto de junção;
- C^2 – duas curvas se encontram, possuem tangentes comuns no ponto de junção e ambas possuem a mesma curvatura.

As 3 continuidades podem ser visualizadas na figura 29 (ROGER; ADAMS, 1990):



Figura 29: a) Continuidade C^0 , b) Continuidade C^1 e c) Continuidade C^2

Outro conceito que deve ser mencionado quando se estuda curvas é o de fecho convexo (*convex hull*). O fecho convexo é um polígono que engloba todos os pontos de controle da curva. Fisicamente, pode ser visualizado como um elástico que toca os pontos de controle mais externos, fazendo com que um ponto de controle esteja no interior ou no perímetro deste polígono, conforme a figura 30.

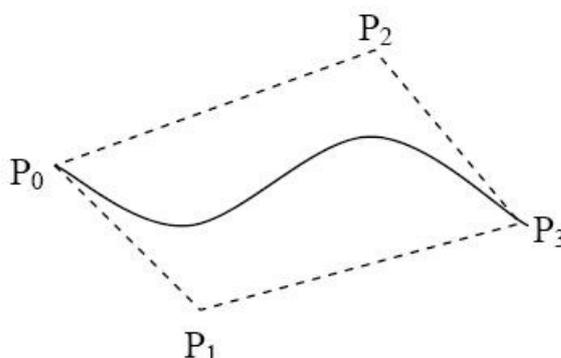


Figura 30: Fecho convexo

Curvas de Bézier

Desenvolvida em 1969 por *Pierre Bézier*, a curva de *Bézier* baseia seu cálculo no Binômio de *Newton* para a resolução de seus coeficientes e é resolvida através da seguinte equação:

$$P(t) = \sum_{i=0}^n B_i J_{n,i}(t), \quad 0 \leq t \leq 1$$

onde

$$J_{n,i}(t) = \binom{n}{i} t^i (1-t)^{n-i}$$

Na equação, B_i representa cada um dos $n+1$ pontos de controle e $J_{n,i}(t)$ são as *blending functions*, descritas pelos polinômios de *Bernstein*.

O índice t é um valor de parametrização para percorrer a curva e pode ser qualquer valor entre zero e um e n é o grau do binômio, tal que utiliza-se $n+1$ pontos de controle para cada curva que se deseja gerar. Por exemplo, para a resolução de $(t+(1-t))^2$ usa-se 3 pontos de controle e obtém-se curvas quadráticas. Já com o uso do binômio $(t+(1-t))^3$, utiliza-se 4 pontos de controle e obtém-se curvas cúbicas. Os pontos de controle B_i podem ser escolhidos aleatoriamente e devem ser multiplicados cada um por uma das parcelas do binômio resolvido.

As funções $J_{n,i}$ devem satisfazer algumas condições, sendo elas:

$$J_{n,i}(t) > 0, \quad 0 \leq t \leq 1$$

$$\sum_{i=0}^n J_{n,i}(t) = 1, \quad 0 \leq t \leq 1$$

Essa propriedade garante que a curva gerada esteja contida dentro de fecho convexo (*convex hull*) definido pelos pontos de controle. Para $n = 3$ (curva de grau 3), tem-se as seguintes funções básicas (*blending functions*):

$$(1-t)^3 \quad (\text{valor máximo em } t = 0)$$

$$3t(1-t)^2 \quad (\text{valor máximo em } t = 1/3)$$

$$3t^2(1-t) \quad (\text{valor máximo em } t = 2/3)$$

$$t^3 \quad (\text{valor máximo em } t = 1)$$

A seguir, tem-se a figura 31 que representa graficamente cada uma das 4 funções citadas:

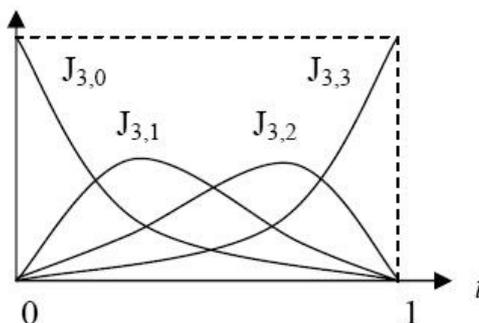


Figura 31: Representação gráfica das *Blending Functions* de grau 3

Através da análise da figura acima, verifica-se que ao ser alterado um ponto de controle da curva, toda curva sofre alteração, pois os valores das funções básicas não são nulos dentro do intervalo $[0, 1]$ de variação do parâmetro de interpolação t . Além disso, para $t = 0$ e $t = 1$ o valor das funções básicas correspondentes a esses pontos são iguais a 1, ou seja, a curva toca o primeiro e último ponto de controle. Para os demais, a curva é apenas aproximada (figura 32).

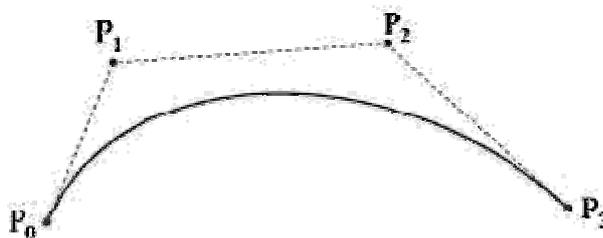


Figura 32: Curva de *Bézier* com 4 pontos de controle

Curvas *B-Spline*

As curvas *B-Spline* possuem algumas vantagens em relação à *Bézier* nos seguintes aspectos:

- O grau dos polinômios pode ser definido independentemente do número de pontos de controle (com algumas limitações);
- Permitem controle local na forma da curva, ou seja, ao se alterar um determinado ponto de controle, somente a região próxima a ele é alterada, e não a curva toda.

Além disso, as curvas *B-Spline* não tocam nenhum dos pontos de controle, ou seja, todos os pontos são aproximados pela curva. Isso facilita a geração de curvas

fechadas (como círculos e elipses), onde, utilizando curvas de grau 3, não se tem a preocupação com a continuidade da curva no ponto de junção, como acontece com as curvas de *Bézier*, uma vez que nesta os pontos de controle das extremidades são tocados pela curva.

Uma curva *B-Spline* pode ser gerada para um número qualquer de pontos de controle e grau de polinômio. O grau do polinômio independe do número de pontos de controle, seguindo a regra: C^{d-2} , onde $d-1$ é o grau da função e C a continuidade da curva. Para um polinômio de grau 3, tem-se continuidade C^2 entre um segmento fechadas (como círculos e elipses), onde, utilizando curvas de grau 3, não se tem a preocupação com a continuidade da curva no ponto de junção, como acontece com as curvas de *Bézier*, uma vez que nesta os pontos de controle das extremidades são tocados pela curva.

Uma curva *B-Spline* pode ser gerada para um número qualquer de pontos de controle e grau de polinômio. O grau do polinômio independe do número de pontos de controle, seguindo a regra: C^{d-2} , onde $d-1$ é o grau da função e C a continuidade da curva. Para um polinômio de grau 3, tem-se continuidade C^2 entre um segmento e outro (grau = 3 = 4-1 $\rightarrow C^{4-2} = C^2$). O número de funções básicas também corresponde ao número de pontos de controle utilizados para cada segmento da *B-Spline*. A figura 33 (ROGER; ADAMS, 1990) mostra uma *B-Spline* formada por segmentos de grau 3 (4 pontos de controle para cada um).

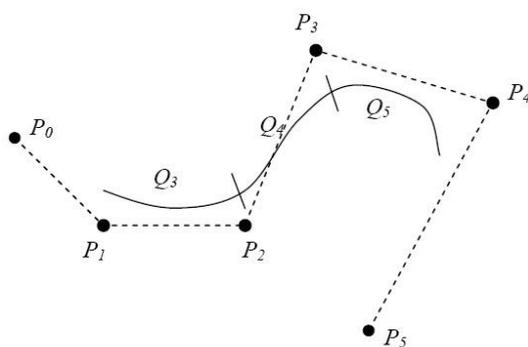


Figura 33: Curva *B-Spline* cúbica

O segmento Q_3 é definido pelos pontos P_0 a P_3 , Q_4 por P_1 a P_4 e Q_4 por P_2 a P_5 . Uma vez que é gerado um pequeno segmento de curva dado os 4 pontos de controle, para que não haja descontinuidade na curva resultante é necessário utilizar

os dois últimos pontos de controle do segmento anterior como sendo os dois primeiros do segmento subsequente.

O cálculo da curva para um dado parâmetro de interpolação t é dado pela equação abaixo:

$$Q_i(t) = \sum_{k=0}^3 P_{i-3+k} B_{i-3+k}(t)$$

O valor em cada segmento de curva varia entre 0 e 1. Desta forma, a curva *B-Spline* é definida como uma série de $m-2$ segmentos de curvas, onde existem $m+1$ pontos de controle. Cada segmento de curva é definido por 4 pontos de controle, ou seja, existem 3 vezes mais pontos de controle e 3 vezes mais funções básicas que segmentos de curva. As funções básicas para a geração de *B-Splines* de grau 3 são mostradas a seguir:

$$\begin{aligned} B_i &= 1/6t^3 \\ B_{i-1} &= 1/6(-3t^3 + 3t^2 + 3t + 1) \\ B_{i-2} &= 1/6(3t^3 - 6t^2 + 4) \\ B_{i-3} &= 1/6(1-t)^3 \end{aligned}$$

Cada função base é não nula para quatro intervalos sucessivos em t , ou seja, a *B-Spline* é não nula no intervalo $t_i, t_{i+1}, \dots, t_{i+2}$. Na prática, cada função básica é composta por 4 segmentos, sendo que cada uma delas é dada como uma versão transladada da função anterior, conforme a figura 34.

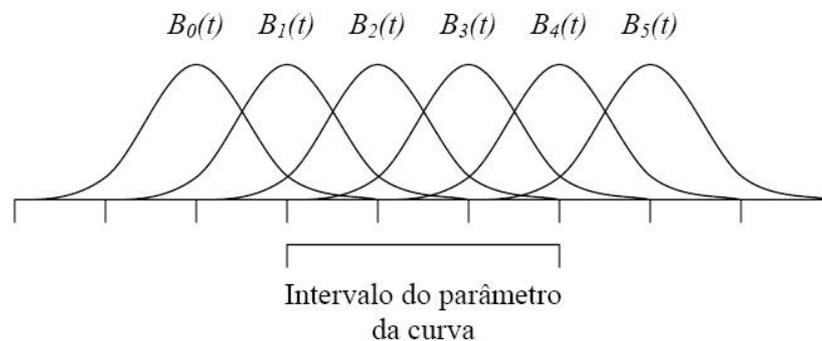


Figura 34: Blending Functions da B-Spline