

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

**UM MODELO DE DRAMATIZAÇÃO BASEADO EM
AGENTES CINEMATOGRAFICOS AUTÔNOMOS
PARA STORYTELLING INTERATIVO**

DISSERTAÇÃO DE MESTRADO

Edirlei Everson Soares de Lima

**Santa Maria, RS, Brasil
2010**

UM MODELO DE DRAMATIZAÇÃO BASEADO EM AGENTES CINEMATOGRAFICOS AUTÔNOMOS PARA STORYTELLING INTERATIVO

por

Edirlei Everson Soares de Lima

Dissertação apresentada ao Curso de Mestrado em Computação do Programa de Pós-Graduação em Informática (PPGI), Área de Concentração em Computação, da Universidade Federal de Santa Maria (UFSM, RS),
como requisito parcial para obtenção do grau de
Mestre em Computação

Orientador: Prof. Dr. Cesar Tadeu Pozzer

Santa Maria, RS, Brasil

2010

**Universidade Federal de Santa Maria
Centro de Tecnologia
Programa de Pós-Graduação em Informática**

A Comissão Examinadora, abaixo assinada,
aprova a Dissertação de Mestrado

**UM MODELO DE DRAMATIZAÇÃO BASEADO
EM AGENTES CINEMATOGRAFICOS AUTÔNOMOS
PARA STORYTELLING INTERATIVO**

**elaborada por
Edirlei Everson Soares de Lima**

como requisito parcial para obtenção do grau de
Mestre em Computação

COMISSÃO EXAMINADORA:

Cesar Tadeu Pozzer, Dr.
(Presidente/Orientador)

Angelo Ernani Maia Ciarlini, Dr. (UNIRIO)

José Antônio Trindade Borges da Costa, Dr. (UFSM)

Santa Maria, 27 de Julho de 2010.

RESUMO

Dissertação de Mestrado
Programa de Pós-Graduação em Informática
Universidade Federal de Santa Maria

UM MODELO DE DRAMATIZAÇÃO BASEADO EM AGENTES CINEMATOGRAFICOS AUTÔNOMOS PARA STORYTELLING INTERATIVO

Autor: Edirlei Everson Soares de Lima

Orientador: Cesar Tadeu Pozzer

Data e local da defesa: Santa Maria, 27 de Julho de 2010.

Sistemas de storytelling interativo são aplicações que buscam contar histórias e permitem que os usuários interajam com estas de modo a alterar os eventos futuros. No que se refere à representação visual das histórias, existem diversas questões relativas à maneira como as cenas devem ser mostradas ao público. Muitas questões que hoje são desafios no desenvolvimento de sistemas de storytelling interativo já foram tratadas antes por vários cineastas. Ao longo dos anos o cinema evoluiu e estabeleceu diversos padrões e princípios que devem ser adotados durante as filmagens de um filme. Entretanto, diferente do cinema, onde cada cena é minuciosamente planejada antes de ser filmada, e caso ocorra algum problema ela é simplesmente filmada novamente, em storytelling interativo não se tem toda esta liberdade; normalmente o enredo da dramatização resulta da interação em tempo real entre o público e os atores virtuais. A dramatização de histórias, cuja sequência de eventos é incerta, depende de um sistema de dramatização robusto, que se adapte de forma adequada a qualquer tipo de cena. Esta dissertação apresenta uma arquitetura de dramatização baseada em agentes autônomos inspirados na cinematografia. Para representar o conhecimento dos agentes utiliza-se uma abordagem baseada em classificadores, mais especificamente, utiliza-se o classificador Support Vector Machine para classificar as melhores tomadas de câmera, clima visual e trilhas sonoras para as cenas.

ABSTRACT

Master's Dissertation
Post-Graduate Program in Informatics
Federal University of Santa Maria

A DRAMATIZATION MODEL BASED ON AUTONOMOUS CINEMATOGRAPHY AGENTS FOR INTERACTIVE STORYTELLING

Author: Edirlei Everson Soares de Lima
Advisor: Cesar Tadeu Pozzer
Santa Maria, July 27, 2010.

Interactive storytelling systems are applications that aim to tell stories and allow the users to interact with the system in order to change the future story events. In respect to the visual representation of the stories, there are several questions related to the manner how the scenes should be shown to the public. Many of these questions that today are faced in the development of interactive storytelling systems have been treated before by several filmmakers. Over the years the cinema has evolved and established various principles and rules to be adopted during the creation of a film. However, unlike movies, where every scene is carefully planned before it was filmed, and if a problem occurs it is simply filmed again, in interactive storytelling this freedom does not exist; usually the storyline of the narrative results from the real-time interaction between the viewers and the virtual actors. The dramatization of stories, whose sequence of events is unknown, depends on a robust dramatization system, which adapts itself appropriately to any type of scene. This dissertation presents a dramatization architecture composed by a set of cinematography-based autonomous agents inspired on the cinematography. To represent the knowledge of the agents is used an approach based on classifiers. More specifically, the architecture uses the Support Vector Machine classifier to classify the best camera shots, visual effects and sound tracks to the scenes.

LISTA DE FIGURAS

Figura 1 - Sistemas de dramatização de histórias.	19
Figura 2 - Arquitetura do Logtell. Fonte: Camanho et al. (2009).	20
Figura 3 - Interface do usuário. Fonte: Camanho et al. (2009).	22
Figura 4 - Estrutura de um filme.	25
Figura 5 - Tipos de Tomadas.	28
Figura 6 - Tipos de ângulos de câmera.	29
Figura 7 - Classificação da altura da câmera.	30
Figura 8 - Diferença entre tomadas usando ângulos com pouco e com bastante senso de profundidade.	30
Figura 9 - Tipos de movimentos de câmera.	32
Figura 10 - Sequência de tomadas que quebram a continuidade visual da cena.	33
Figura 11 – Regras para manter a continuidade durante as filmagens.	34
Figura 12 - Estilos de iluminação.	37
Figura 13 – Método de iluminação three point lighting.	37
Figura 14 - Exemplo de cena utilizando diferentes configurações de iluminação e cores.	39
Figura 15 - Arquitetura de dramatização cinematográfica.	50
Figura 16 - Rede de relações e emoções.	54
Figura 17 - Diagrama de classes dos agentes cinematográficos.	57
Figura 18 - Conexão TPC/IP entre o agente Roteirista e o Client User Interface.	58
Figura 19 - Exemplos de linhas de ação.	65
Figura 20 - Padrão de posicionamento de câmeras para a filmagem de uma cena de diálogo composta por dois personagens.	65
Figura 21 - Padrão de posicionamento de câmeras para a filmagem de uma cena composta por um personagem caminhando em determinada direção.	67
Figura 22 - Padrão de posicionamento de câmeras para a filmagem de uma cena composta por um personagem que inicia um vôo na vertical.	68
Figura 23 - Arquitetura do agente Editor.	74
Figura 24 - Modelo de entrada, processamento e saída de dados das SVMs do agente Editor.	76
Figura 25 - Configuração de câmeras para a filmagem de uma cena de diálogo composta por três personagens.	76

Figura 26 - Possíveis tomadas para a filmagem de uma cena de diálogo composta por três personagens.	77
Figura 27 - Mapa do ambiente 3D da dramatização do cenário de testes do Logtell.....	81
Figura 28 - Conjunto de valores de variação da felicidade/tristeza dos atores envolvidos na cena da proposta de casamento.	83
Figura 29 - Conjunto de valores de variação da coragem/medo dos atores envolvidos na cena da proposta de casamento.	84
Figura 30 - Conjunto de valores de variação da afeição dos atores envolvidos na cena da proposta de casamento.	84
Figura 31 - Resultado visual da cena do pedido de casamento.....	85
Figura 32 - Outras opções de efeitos visuais da cena do pedido de casamento.	85
Figura 33 - Tempo necessário para o treinamento das 9 SVMs do agente Editor utilizando bases de treinamento de tamanho variado.	90
Figura 34 - Tempo necessário para a classificação utilizando as SVMs do agente Editor com bases de treinamento de tamanho variado.	91
Figura 35 - Taxa de reconhecimento das SVMs do agente Editor utilizando bases de treinamento de tamanho variado.....	92
Figura 36 - Tempo necessário para o treinamento da SVM do agente Diretor de Fotografia/Diretor de Música utilizando bases de treinamento de tamanho variado.....	93
Figura 37 - Tempo necessário para a classificação utilizando a SVM do agente Diretor de Fotografia/Diretor de Música com bases de treinamento de tamanho variado.....	94
Figura 38 - Taxa de reconhecimento da SVM do agente Diretor de Fotografia/Diretor de Música utilizando bases de treinamento de tamanho variado.....	95

LISTA DE TABELAS

Tabela 1 - Comandos interpretados pelo agente Roteirista.	60
Tabela 2 - Micro-ações interpretadas pelo agente Diretor.	62
Tabela 3 - Lista de configurações de câmeras implementadas no agente Cameraman.	69
Tabela 4 - Lista de SVMs criadas e treinadas para o agente Editor.	75
Tabela 5 - Perfis de efeitos visuais do agente Diretor de Fotografia.	79
Tabela 6 - Sequência de eventos para a cena do pedido de casamento.	83
Tabela 7 - Perfis de trilhas sonoras do agente Diretor de Música.	87
Tabela 8 - Tempo necessário para o treinamento das 9 SVMs do agente Editor utilizando bases de treinamento de tamanho variado.	89
Tabela 9 - Tempo necessário para a classificação utilizando as SVMs do agente Editor com bases de treinamento de tamanho variado.	90
Tabela 10 - Taxa de reconhecimento das SVMs do agente Editor utilizando bases de treinamento de tamanho variado.	91
Tabela 11 - Tempo necessário para o treinamento da SVM do agente Diretor de Fotografia/Diretor de Música utilizando bases de treinamento de tamanho variado.	93
Tabela 12 - Tempo necessário para a classificação utilizando a SVM do agente Diretor de Fotografia/Diretor de Música com bases de treinamento de tamanho variado.	94
Tabela 13 - Taxa de reconhecimento da SVM do agente Diretor de Fotografia/Diretor de Música utilizando bases de treinamento de tamanho variado.	94
Tabela 14 - Comparativo visual do modelo de dramatização apresentado neste trabalho com duas versões anteriores do sistema.	96

LISTA DE ABREVIATURAS E SIGLAS

DLL	Dynamic-link Library
HTN	Hierarchical Task Network
IPG	Interactive Plot Generator
KNN	K-Nearest Neighbor
SVM	Support Vector Machine
XML	Extensible Markup Language

LISTA DE APÊNDICES

APÊNDICE A - Support Vector Machine	112
--	------------

SUMÁRIO

1 INTRODUÇÃO	12
1.1 Objetivos	13
1.2 Contribuições.....	14
1.3 Metodologia	14
1.4 Estrutura.....	15
2 STORYTELLING INTERATIVO	16
2.1 Logtell	20
2.1.1 Cenário de testes	23
3 CONCEITOS DE CINEMATOGRAFIA	25
3.1 Tomada	26
3.1.1 Tipos de tomadas	26
3.1.2 Ângulo da câmera	27
3.1.3 Altura da câmera.....	29
3.1.4 Ângulo do assunto.....	30
3.2 Movimentos da câmera.....	31
3.3 Continuidade.....	32
3.4 Filmagem de cenas.....	35
3.5 Cortes	35
3.6 Iluminação e cores	36
3.7 Lentes e efeitos	39
3.8 Músicas.....	40
3.9 Equipe de filmagem	41
4 ARQUITETURA DE DRAMATIZAÇÃO CINEMATOGRAFICA	43
4.1 Trabalhos relacionados	43
4.2 Modelo de dramatização cinematográfico proposto	47
4.2.1 Requisitos	47
4.2.2 Arquitetura.....	49
5 IMPLEMENTAÇÃO	52

5.1 Ferramentas e métodos.....	52
5.1.1 Rede de relações e emoções	53
5.2 Agentes cinematográficos	56
5.2.1 Roteirista.....	58
5.2.2 Diretor	62
5.2.4 Cameraman	63
5.2.5 Editor	70
5.2.6 Diretor de fotografia.....	78
5.2.7 Diretor de música	86
6 RESULTADOS	89
6.1 Validação do agente Editor	89
6.2 Validação dos agentes Diretores de Fotografia e Música.....	92
6.3 Validação do modelo de dramatização	95
7 CONCLUSÃO	99
7.1 Considerações gerais.....	99
7.2 Contribuições.....	101
7.3 Publicações.....	102
7.4 Trabalhos futuros	102

1 INTRODUÇÃO

Sistemas de storytelling interativo são aplicações que buscam contar histórias e permitem que os usuários possam interagir com estas de modo a alterar de alguma forma os eventos futuros. São diferentes de jogos, pois não focam em desafios para os usuários e sim no conteúdo dramático apresentado. No que se refere a pesquisas nesta área, distinguem-se claramente três linhas de pesquisa. A primeira é referente à geração das histórias, ou seja, métodos automatizados para gerar enredos coerentes e diversificados. A segunda é referente ao modo como os usuários podem interagir com estas histórias e como esta interação irá interferir no contexto da história. A terceira linha de pesquisa é referente à dramatização das histórias geradas, ou seja, a representação visual dos eventos.

Muitas vezes a etapa de dramatização é considerada a mais simples, pois ao se pensar superficialmente no processo pode se ter a impressão de que tudo se resume a interpretar os eventos gerados e representá-los visualmente em um cenário 2D ou 3D. Entretanto, representar visualmente as histórias de uma forma atraente e envolvente para o público é algo extremamente complexo. Diversas questões surgem, por exemplo:

- Como os atores devem se comportar durante a dramatização de uma ação?
- Como a câmera deve filmar determinada cena?
- Como expressar as emoções das cenas?
- Como garantir a continuidade das cenas?
- E, mais importante de tudo, como apresentar os eventos de forma atraente e envolvente para o público?

Muitas destas questões que hoje são enfrentadas no desenvolvimento de sistemas de dramatização para storytelling interativo já foram tratadas antes por vários cineastas. Ao longo dos anos, o cinema evoluiu e estabeleceu diversos padrões, princípios e conceitos que devem ser adotados no processo de criação e filmagens de filmes. No entanto, aplicar os conceitos e princípios de cinematografia em um programa de computador não é uma tarefa fácil. Diferente do cinema, onde cada cena é minuciosamente planejada antes de ser filmada e, caso ocorra algum problema, ela é simplesmente filmada novamente, em storytelling interativo não se tem toda esta liberdade; normalmente o enredo da dramatização resulta da interação em tempo real entre o público e os atores virtuais. Por este motivo, a simples criação de scripts pré-definidos para a utilização de conceitos e regras cinematográficas pode não ser

o suficiente para a dramatização de histórias de qualidade. O sistema de dramatização deve ser inteligente e autônomo o suficiente para se adaptar em tempo real a qualquer cena que possa vir a ser dramatizada.

1.1 Objetivos

O principal objetivo desta dissertação é propor um modelo de dramatização baseado em conceitos e princípios definidos pela cinematografia. Mais especificamente, propor um modelo que busque enfatizar o conteúdo dramático das cenas e apresente-as de modo atraente e envolvente para os espectadores.

A proposta desta dissertação abrange e enfoca quatro aspectos gerais de um sistema de dramatização:

- Controle de câmeras virtuais;
- Manipulação dos ambientes, iluminação e cores;
- Controle das músicas;
- Direção.

Os três primeiros aspectos estão relacionados com a necessidade de se criar cenas atraentes e envolventes, que expressam e evocam emoções nos espectadores. Já o quarto aspecto está relacionado à maneira como a arquitetura e os seus módulos/agentes são organizados internamente.

Além de seguir os princípios e regras da cinematografia, o modelo proposto neste trabalho também deve ser autônomo e inteligente o suficiente para dramatizar cenas em tempo real, adaptando-se a diferentes situações que podem surgir durante o processo de geração da história e interação com os usuários.

É importante ressaltar que o objetivo desta dissertação não é propor um sistema de dramatização completo, mas sim focar nos aspectos que estão mais relacionados com a cinematografia em si. Mesmo que estes já abranjam grande parte de um sistema de dramatização, este não é o enfoque deste trabalho.

1.2 Contribuições

Esta dissertação visa atingir, de forma resumida, as seguintes contribuições:

- Propor um modelo de dramatização baseado em agentes diretores autônomos que utilizam conhecimentos cinematográficos para a dramatização de histórias com ênfase em conteúdo dramático;
- Propor um método automático para a seleção das melhores tomadas para a filmagem de cenas em tempo real;
- Propor um método automático para a seleção dos melhores efeitos audiovisuais para enfatizar as emoções de cenas em tempo real.

Descrições mais detalhadas das contribuições são apresentadas no Capítulo 7.

1.3 Metodologia

O desenvolvimento do modelo de dramatização proposto nesta dissertação é realizado no contexto do sistema de storytelling interativo Logtell (Pozzer, 2005; Ciarlini *et al.*, 2005). A história utilizada como cenário de testes do Logtell baseia-se em contos de fadas, onde figuram personagens como heróis, vilão e uma princesa, e é representada por um conjunto de eventos modelados através de operações parametrizáveis, como andar, lutar, raptar, dentre outras. Estes eventos são utilizados como entrada para o sistema de dramatização, o qual deve interpretá-los e dramatizá-los de forma adequada no ambiente 3D.

O modelo de dramatização baseia-se em agentes autônomos que utilizam conhecimentos cinematográficos para dramatizar de modo atraente e envolvente os eventos da história. Para representar o conhecimento dos agentes utiliza-se uma abordagem baseada em classificadores, mais especificamente, utiliza-se o classificador Support Vector Machine - SVM (Vapnik, 1995) para classificar as melhores tomadas de câmera, clima visual e trilhas sonoras para as cenas. Para isso, os agentes baseiam-se em dados emocionais dos atores virtuais e informações geométricas do ambiente 3D. Os estados emocionais dos atores são representados através de uma estrutura dinâmica chamada de rede de relações e emoções, a qual é necessária devido a não existência de informações emocionais nas histórias geradas

pelo Logtell. A implementação do sistema de dramatização, assim como todos os agentes que o compõem, foi realizada na *game engine* Unity3D (Unity3D, 2010).

1.4 Estrutura

Esta dissertação está dividida em 7 capítulos. No capítulo 2, apresenta-se uma visão de geral das pesquisas em storytelling interativo sob os aspectos de geração, interação e visualização das histórias. Apresenta-se também o sistema Logtell, no qual o modelo de dramatização proposto neste trabalho será validado.

No capítulo 3, faz-se um levantamento das principais questões relacionadas aos conceitos, princípios e regras definidas pela cinematografia.

O capítulo 4 descreve alguns trabalhos relacionados à utilização de conceitos cinematográficos em ambientes de storytelling interativo e jogos. Em seguida, apresenta-se o novo modelo de dramatização proposto neste trabalho, assim como os requisitos definidos para o desenvolvimento do mesmo.

O capítulo 5 apresenta detalhes sobre a implementação do sistema de dramatização proposto. São descritas as técnicas e algoritmos empregados para o desenvolvimento dos módulos que compõem o sistema.

No capítulo 6 são apresentados os resultados gerais da validação do sistema de dramatização, assim como alguns testes específicos de desempenho para alguns dos módulos do sistema.

No capítulo 7 são apresentadas as conclusões gerais, contribuições, bem como perspectivas de trabalhos futuros.

2 STORYTELLING INTERATIVO

Storytelling interativo é um meio de entretenimento digital onde autores, público e agentes virtuais participam em uma experiência colaborativa (Scientific American Magazine, 2000). Crawford (2004) define storytelling interativo como uma forma de entretenimento interativo na qual o espectador desempenha o papel do protagonista em um ambiente dramaticamente rico. Difere-se de jogos, pois enquanto que em jogos as histórias são essencialmente utilizadas para criar desafios ao jogador, em storytelling as histórias são criadas para surpreender e divertir os espectadores. A experiência proporcionada ao público por uma história interativa difere-se substancialmente de uma história linear. Uma história linear convencional “segue os seus trilhos” do início ao fim da melhor maneira possível. Uma história interativa oferece aos espectadores um universo de possibilidades dramáticas (Crawford, 2004). O principal objetivo de um sistema de storytelling interativo é entreter os espectadores, desta forma, a qualidade da narrativa em termos de coerência e conteúdo dramático deve ser considerada como uma das principais preocupações. Um dos principais desafios para a implementação de tais sistemas é a conciliação de um bom nível de interatividade com a coerência das histórias.

Os registros das primeiras tentativas de se entender storytelling interativo datam da década de 70. Tale-Spin (Meehan, 1977) foi um dos primeiros programas com objetivo de gerar histórias automaticamente de que se têm relatos. Para a geração das histórias o sistema utilizava-se da simulação do comportamento de personagens. Em 1984, seguindo uma linha semelhante ao Tale-Spin, Lebowitz (1984) apresenta o Universe. O sistema baseia-se em objetivos autorais para a geração das histórias. A partir da década de 90, houve o surgimento de diversos trabalhos em storytelling interativo, dentre os mais conhecidos pode-se citar: Oz Project (Bates *et al.*, 1992), Minstrel (Turner, 1992), Mimesis (Young, 2001), Façade (Mateas 2002; Mateas e Stern, 2003) e o Logtell (Ciarlini *et al.*, 2005; Pozzer, 2005).

Qualquer sistema de storytelling interativo normalmente é composto por três módulos distintos: um responsável pela geração da história; outro específico para tratar a interação dos usuários; e, por último, o módulo responsável pela dramatização, ou seja, a representação visual das histórias.

O módulo de geração de histórias é a base para qualquer sistema de storytelling interativo. A maneira como a história é gerada depende da abordagem utilizada. Atualmente

existem dois modelos comumente utilizados para a geração de histórias: o baseado em personagens (*character-based*) (Cavazza *et al.*, 2002; Young, 2001; Aylett *et al.*, 2006) e o baseado em roteiro (*plot-based*) (Grasbon e Braun, 2001; Paiva, 2001; Spierling *et al.*, 2002; El-Sattar, 2008). Na abordagem baseada em personagens, a história é gerada em tempo real através do comportamento e da interação entre os personagens autônomos. A principal vantagem deste modelo é a possibilidade da intervenção do usuário a qualquer momento. Entretanto, como desvantagem, existe a falta de controle sobre a qualidade das histórias a serem geradas, pois normalmente não há uma maneira de se estimar quais as decisões ou ações serão executadas pelos atores virtuais. Na abordagem baseada em roteiro, a história é normalmente gerada em uma etapa que vem antes da dramatização, podendo ser gerada automaticamente ou com ajuda de um autor. Nesta abordagem os personagens incorporam um comportamento reativo, seguindo regras rígidas definidas pelo roteiro. Com isso tem-se um maior controle sobre a história a ser dramatizada. Existem também algumas abordagens híbridas (Mateas e Stern, 2005; Si *et al.*, 2008; Cai *et al.*, 2007) que procuram combinar os dois modelos. Façade (Mateas e Stern, 2005), por exemplo, procura combinar as vantagens do modelo orientado a personagens com as do modelo orientado a enredo, de forma a minimizar as limitações das duas estratégias.

A geração e planejamento de histórias tem sido alvo de diversas pesquisas nos últimos anos. Trabalhos como (Cavazza *et al.*, 2002; Charles e Cavazza, 2004) focam no modelo *character-based* e utilizam HTNs (*Hierarchical Task Networks*) como estrutura central de planejamento e representação do conhecimento dos personagens. Uma HTN pode ser vista como uma representação implícita de um conjunto de possíveis soluções para uma tarefa. Desta forma, cada personagem está associado a uma HTN que contém todas as possíveis regras para a história. Cada tarefa está associada a pré- e pós-condições. Em outro trabalho, Bates *et al.* (1992) busca criar, em sua abordagem *character-based*, personagens que pareçam realistas, naturais e cujas ações e reações façam sentido. Para isso todos os personagens possuem personalidade, emoções, motivações e relações sociais com os outros personagens. As histórias são geradas através das interações entre estes personagens. Para garantir maior consistência das histórias geradas, Kelso *et al.* (1993) introduziu no sistema um guia para os eventos, onde um grafo direcionado acíclico é usado para guiar os acontecimentos da história. Um dos principais desafios do modelo *character-based* é a geração de histórias coerentes e ao mesmo tempo atraentes.

Em trabalhos utilizando a abordagem *plot-based* não existe tanta preocupação com a geração da história, pois esta normalmente é criada por um autor em uma etapa de pré-processamento, como em Grasbon e Braun (2001), onde o autor define manualmente o modelo global da história. Sua pesquisa questiona a forma de geração da configuração inicial das histórias, visto que os autores não acreditam na geração automática de todos os detalhes das histórias. Mesmo em modelos *plot-based*, muitas vezes é necessário garantir a consistência das histórias geradas a partir da configuração inicial definida pelo autor, como em Ciarlini *et al.* (2005) onde é utilizado um planejador hierárquico não-linear que permite a definição de uma hierarquia de pré-condições, de modo que haja uma priorização na busca de soluções mantendo-se a consistência das histórias geradas. Uma das principais fontes de inspiração para o modelo *plot-based* é o trabalho de Vladimir Propp (Propp, 1968). Propp analisou um conjunto grande de contos de fadas Russos e mostrou que todos podiam ser descritos através de 31 funções típicas, como: partida do herói, resgate, recompensa, punição, etc.

A interação dos espectadores com as histórias dramatizadas também é um grande campo de pesquisas em storytelling. Normalmente, a forma como os espectadores podem interagir com a narrativa está diretamente ligada ao modelo de geração de histórias adotado. Em abordagens *character-based* é possível dar maior liberdade de interação, permitindo que a história seja gerada a partir das escolhas do usuário em relação às ações e reações dos personagens autônomos. Em trabalhos como Cavazza *et al.* (2002), Charles e Cavazza (2004), o usuário é tratado como um espectador e pode realizar interações físicas diretas com os objetos virtuais da cena (adição, remoção ou deslocamento); Além disso, também é possível utilizar mensagens linguísticas para dar sugestões aos personagens. Estas interações têm influência direta na tomada de decisões dos personagens e consequentemente na história gerada. Já em abordagens *plot-based*, as possibilidades de interações são bastante limitadas, permitindo que o usuário possa apenas realizar interferências “sutis” com o objetivo de apenas guiar o andamento da narrativa para os pontos predefinidos no roteiro, como em Grasbon e Braun (2001) e Ciarlini *et al.* (2005) onde o usuário apenas dá sugestões sobre o que deve acontecer nos próximos passos da história.

A etapa final em um sistema de storytelling interativo é a dramatização (exibição ou representação gráfica da narrativa). Nesta fase, os dados numéricos da história são transformados em uma representação visual, onde fatores temporais, espaciais e emocionais são adicionados à narrativa. Em Charles e Cavazza (2004), o espectador assume uma

perspectiva em terceira pessoa e pode visualizar a dramatização sob a perspectiva de qualquer personagem (Figura 1 - a); é permitida ainda a navegação livre pelo cenário 3D. O sistema também incorpora um módulo de câmera que desempenha o papel de diretor virtual e apresenta ao espectador as situações mais relevantes que estão acontecendo (Charles *et al.*, 2002). Mateas e Stern (2005) dramatizam a sua história em um mundo 3D representado por um apartamento; o espectador assume uma perspectiva em primeira pessoa e participa ativamente da narrativa (Figura 1 - b). Ciarlini *et al.* (2005) adota uma abordagem semelhante à de Cavazza *et al.* (2002), com um ambiente 3D onde o espectador assume uma perspectiva em terceira pessoa (Figura 1 - c). Em outra abordagem mais simplificada, Szilas (2008) apresenta as histórias geradas pelo seu sistema de storytelling através de linhas de texto (Figura 1 - d).



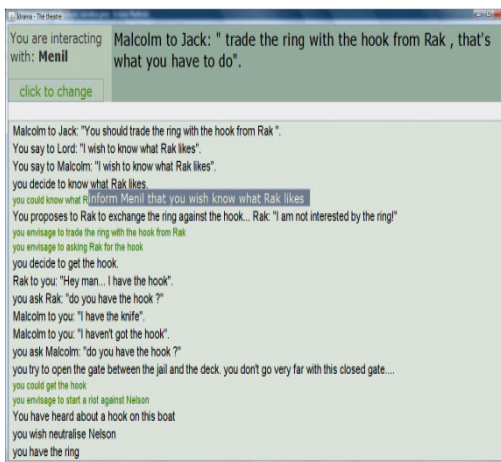
(a) Ambiente 3D de Charles e Cavazza (2004).



(b) Ambiente 3D de Mateas e Stern (2005).



(c) Ambiente 3D de Ciarlini et al. (2005).



(d) Ambiente texto de Szilas (2008).

Figura 1 - Sistemas de dramatização de histórias.

2.1 Logtell

O Logtell (Pozzer, 2005; Ciarlini *et al.*, 2005; Camanho *et al.*, 2009; Doria *et al.*, 2008) é um sistema de storytelling interativo destinado a TV digital interativa. O sistema privilegia a coerência lógica na sua estratégia de geração de narrativas. Baseia-se na abordagem *plot-based*, mas utiliza também algumas características *character-based* definidas através de regras de inferência de objetivos, que fornecem os objetivos a serem alcançados pelos personagens quando determinadas situações são observadas.

O sistema possui uma arquitetura cliente/servidor (Figura 2) (Camanho *et al.*, 2009) que suporta diversos usuários compartilhando e interagindo na mesma ou em diferentes histórias. O lado cliente é responsável pela interação com o usuário e pela dramatização das histórias. O lado servidor é responsável por gerenciar os múltiplos clientes e enredos das histórias.

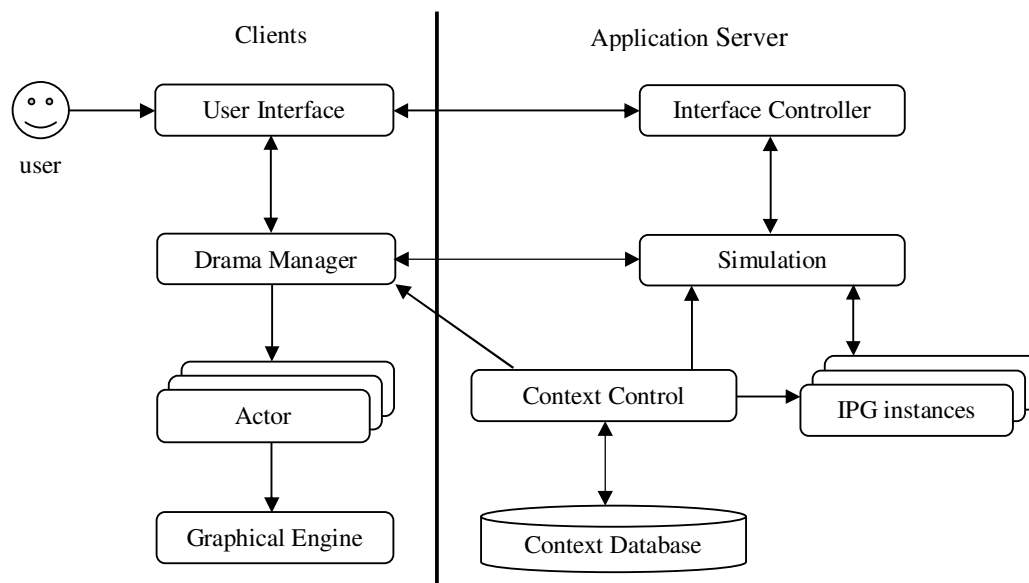


Figura 2 - Arquitetura do Logtell. Fonte: Camanho et al. (2009).

O contexto das histórias é armazenado em um banco de dados de contextos (*Context Database*) e é acessado pelos módulos do sistema durante a geração e dramatização dos enredos. O contexto contém a descrição do gênero segundo o qual as histórias devem ser

geradas, o estado inicial dos personagens e do ambiente no início da história (Camanho *et al.*, 2009). O gênero é descrito basicamente por:

- a. um conjunto de operações básicas parametrizadas, com pré- e pós-condições, que especificam logicamente o repertório pré-determinado de eventos que podem ocorrer;
- b. um conjunto de regras de inferência de objetivos, especificadas formalmente através de lógica temporal, com as situações que levam ao surgimento de objetivos para os personagens ou para a história como um todo; e
- c. uma biblioteca de planos, correspondentes a combinações de operações para o alcance de objetivos específicos.

A geração dos enredos é realizada pelo IPG (*Interactive Plot Generator*) (Ciarlini, 1999; Furtado e Ciarlini, 1999), que consiste em um gerador interativo de histórias implementado em Prolog e que, por meio de planejamento e regras de inferência de objetivos, gera conjuntos parcialmente ordenados de eventos. A criação de um enredo é realizada por um processo de simulação, onde tanto a descrição da situação inicial dos personagens, seus objetivos e as operações que caracterizam o gênero, são levadas em consideração. Uma operação descreve os fatos válidos antes e depois do evento em termos de pré- e pós-condições. Estas funções são baseadas numa classificação de eventos típicos proposto por Propp. Para que uma operação seja executada, é necessário que suas pré-condições sejam satisfeitas, ou no próprio estado inicial, ou então pelas pós-condições de operações que a precedem (Pozzer, 2005). A geração das sequências de eventos (capítulos) passa por múltiplos estágios que alternam entre fases de inferência de objetivos, planejamento e interação com o usuário.

O *Simulation Controller* é responsável por: informar ao *Drama Manager*, no lado do cliente, os próximos eventos a serem dramatizados; receber pedidos de interação e incorporá-los à história, quando coerentes; selecionar sugestões de intervenções viáveis a serem feitas aos usuários; ativar o IPG, a fim de obter os próximos eventos a serem dramatizados; e sincronizar as dramatizações com a geração o enredo (Camanho *et al.*, 2009).

No lado do cliente, o usuário interage com o sistema através da interface do usuário (*User Interface*), que informa as interações desejadas para o *Interface Controller* no lado do servidor. O *Drama Manager* solicita os próximos eventos a serem dramatizados ao *Simulation Controller* e controla as instâncias dos atores para cada personagem no ambiente 3D de dramatização (*Graphical Engine*). No lado do servidor, o *Interface Controller* centraliza as sugestões feitas pelos diversos clientes. Quando vários usuários compartilham a mesma

história, as interações são selecionadas de acordo com o número e clientes que as solicitam. Quando há apenas um cliente, as sugestões são enviadas automaticamente para o *Simulation Controller* (Camanho *et al.*, 2009).

O modo de interação do usuário com a história pode variar de um nível no qual o usuário apenas assiste à história passivamente, a um nível em que o usuário está apto a realizar intervenções fortes e explorar as possibilidades permitidas pelo gênero. As interações podem ser divididas em fortes e fracas. Nas interações fortes, o usuário indica que deseja a ocorrência de determinados eventos ou situações no próximo capítulo. Tais interações são feitas com base em sugestões que são apresentadas automaticamente pelo sistema. Nas interações fracas, o usuário força a história a retornar a um capítulo anterior para que uma alternativa para o capítulo seja automaticamente apresentada ou para assistir novamente ao capítulo, com possibilidade de alterar a continuidade da história a partir desse ponto através de interações fortes. A coerência lógica de uma intervenção solicitada pelo usuário é sempre verificada antes desta ser incorporada ao roteiro. Caso não seja consistente, ela é simplesmente desconsiderada. A interface com o usuário pode ser vista na Figura 3.

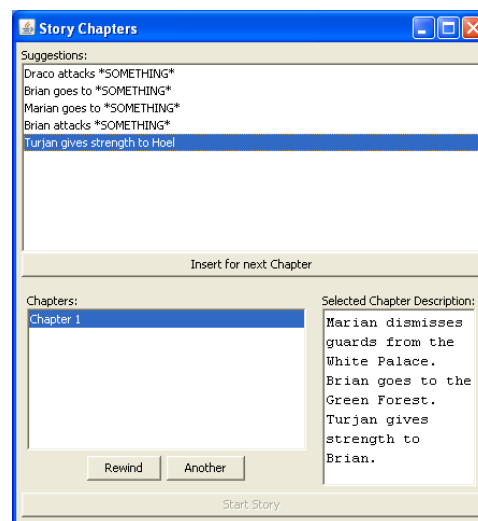


Figura 3 - Interface do usuário. Fonte: Camanho et al. (2009).

O Logtell é apto a gerar e dramatizar histórias simultaneamente, garantindo a diversidade e coerência dos enredos, além de oferecer alternativas de interação em vários níveis demandando pouco esforço do usuário. Além disso, a arquitetura do sistema leva em

consideração questões relativas ao compartilhamento do sistema por grande quantidade de usuários, com histórias individuais ou compartilhadas.

2.1.1 Cenário de testes

O cenário de testes utilizado pelo Logtell, e conseqüentemente na dramatização de testes do modelo proposto nesta dissertação, corresponde a uma pequena subclasse do gênero de contos de fadas. Os eventos das histórias são modelados através de operações parametrizáveis que permitem a geração de uma variedade considerável de roteiros. Estas operações são:

- go(CH, PL): personagem CH vai para o local PL;
- reduce_protection(VIC, PL): a proteção de um local PL (representada por um número de guardas) é reduzida pela possível vítima VIC;
- kidnap(VIL, VIC): o vilão VIL rapta a vítima VIC;
- attack(CH, PL): o personagem CH ataca o local PL (lutando com os guardas);
- fight(CH1, CH2): o personagem CH1 luta contra o personagem CH2;
- kill(CH1, CH2): o personagem CH1 mata o personagem CH2;
- free(HERO, VIC): o personagem HERO liberta o personagem VIC, aumentando o grau de afeição da VIC pelo HERO;
- marry(CH1, CH2): ambos os personagens casam;
- donate(CH1, CH2): o nível de força do personagem CH2 é aumentando por poderes mágicos do personagem CH2;
- bewitch(CH1, CH2): o personagem CH1 enfeitiça o personagem CH2, aumentando a maldade de CH2 e ao mesmo tempo tornando-o mais forte;

O modelo do gênero é completado pelas seguintes regras de interferência:

- Se um personagem interpretar o papel de vítima, ele irá espontaneamente fazer algo que o coloque em uma situação de desproteção;
- Se um personagem interpretando o papel de herói for mais fraco que o vilão, este personagem irá querer ficar mais forte;
- Se o nível de proteção do local onde se encontra uma vítima é reduzido, o vilão irá querer raptá-la;

- Se uma vítima é raptada, o herói irá querer libertá-la;
- Se o nível de afeição entre dois personagens atinge determinado nível, eles irão querer casar;
- Se uma vítima for morta, o herói irá querer vingança.

A configuração inicial do mundo virtual da dramatização é a seguinte:

- Marian é uma princesa que vive no White Palace (a vítima);
- Brian e Hoel são cavaleiros (os heróis);
- Turjan é um mago que habita o Green Castle (o doador);
- Draco é um dragão cujo covil fica no Red Castle (o vilão);
- A princesa, o dragão e o mago possuem guardas protegendo os seus lares;
- Cada personagem é dotado com certo nível de força;
- Os dois heróis possuem um nível de afeição com princesa, o qual não é recíproco;
- Turjan é neutro e respeita todos os outros personagens.

3 CONCEITOS DE CINEMATOGRAFIA

A cinematografia pode ser definida como a arte de criar filmes (Brown, 2002). O termo foi criado pela indústria do cinema para descrever o processo de criação de imagens em filmes e cobre todos os aspectos do trabalho envolvendo câmeras, incluindo os aspectos criativos envolvidos com a produção de imagens esteticamente agradáveis e os aspectos técnicos sobre o uso de câmeras, luzes e outros equipamentos (Newman, 2008). Mais especificamente, a cinematografia descreve um conjunto de orientações e princípios para o uso eficaz da iluminação, trilhas sonoras, posicionamento de personagens e câmeras. A aplicação bem sucedida dos conceitos de cinematografia resulta na representação coerente e atraente de uma narrativa (Kneafsey, 2006). Além de ajudar a enriquecer a dramatização, a cinematografia também fornece meios para a geração de respostas emocionais nos espectadores e pode ser usada para comunicar informações complementares sobre os eventos da narrativa.

Um filme poder ser considerado uma sequência linear de imagens, entretanto para cinematografia, um filme possui uma estrutura básica (Figura 4). No nível mais alto encontram-se as sequências de cenas. Cada cena é composta por várias tomadas. Uma tomada consiste em uma visão contínua de uma cena filmada por uma câmera sem interrupções. A transição de uma tomada para o outra é conhecido como corte (Mascelli, 1965).

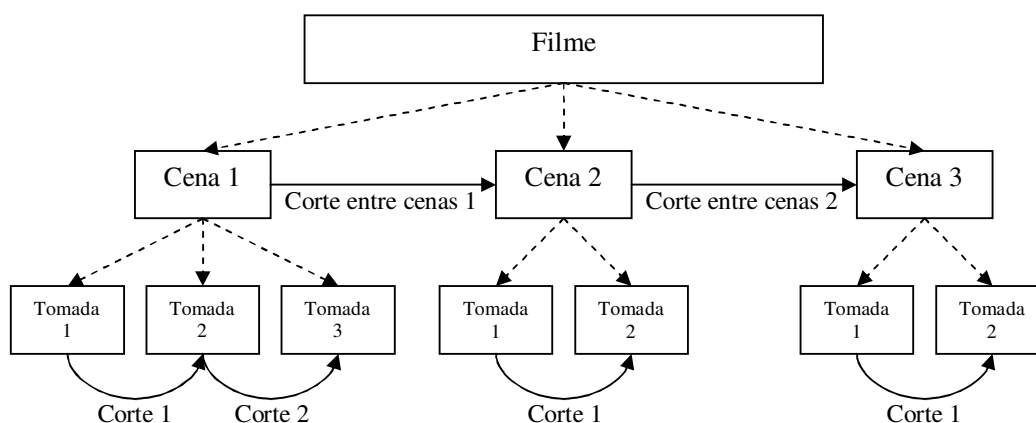


Figura 4 - Estrutura de um filme.

3.1 Tomada

Thompson e Bowen (2009) definem uma tomada como a menor unidade de um filme, a gravação de uma ação de um determinado ponto de vista por uma câmera. A combinação de várias tomadas produz uma cena. Diversos fatores devem ser considerados para a criação de uma tomada. Os mais importantes são:

- Tipo de Tomada: Define o quanto da cena ou do assunto estará visível;
- Ângulo da Câmera: Determina o ponto de vista do espectador em relação à cena;
- Altura da Câmera: Determina a altura da câmera e consequentemente a relação psicológica do espectador com os objetos e personagens da cena;
- Ângulo do Assunto: Define a profundidade em relação aos objetos e personagens da cena;

3.1.1 Tipos de tomadas

As tomadas são utilizadas para enquadrar os objetos, pessoas, ações ou eventos em um filme. Embora cada tomada seja realizada de uma maneira diferente, existe uma lista de tipos de tomadas comumente utilizadas. Segundo Thompson e Bowen (2009), os tipos de tomadas podem ser classificados em (Figura 5):

- *Extreme Long Shot*: Tipo de tomada onde a cena é filmada a uma longa distância abrangendo um grande campo de visão. O resultado é uma tomada que mostra todo o ambiente onde a cena está acontecendo. Tradicionalmente é utilizado em cenas externas;
- *Very Long Shot*: Tipo de tomada que procura mostrar o ambiente da cena, porém dando mais ênfase aos personagens, sendo possível distingui-los claramente. Pode ser utilizado em ambientes internos e externos;
- *Long Shot*: Tipo de tomada que procura enfatizar os personagens no ambiente, mostrando-os de corpo completo; expressões faciais podem ser levemente percebidas. Este tipo de tomada dá menos ênfase ao ambiente, mas este ainda continua perceptível na cena;

- *Medium Long Shot*: Tipo de tomada que enquadra o personagem na cena a partir dos joelhos. Detalhes das roupas e expressões faciais são visíveis. Parte do ambiente continua ainda visível;
- *Medium Shot*: Tipo de tomada que enquadra o personagem na cena a partir da cintura. O personagem é a figura mais proeminente da cena; detalhes como a direção onde o personagem está olhando podem ser percebidos;
- *Medium Close-Up*: Tipo de tomada que enquadra o personagem na cena a partir do cotovelo. As características faciais são bem visíveis, assim como a direção do olhar e expressões emocionais. É um dos tipos de tomada mais utilizados, pois fornece grande quantidade de informações visuais sobre o personagem quando ele está falando, ouvindo ou realizando ações que não envolvam o seu corpo;
- *Close-Up*: Tipo de tomada que enquadra o personagem na cena a partir dos ombros. Mostra detalhes nos olhos e permite a percepção de emoções sutis transmitidas através do olhar, boca e músculos faciais do personagem. Os espectadores ficam totalmente focados na face do personagem, especialmente olhos e boca;
- *Big Close-Up*: Tipo de tomada que enquadra a face do personagem na cena. Detalhes emocionais são extremamente perceptíveis;
- *Extreme Close-Up*: Tipo de tomada utilizado para enquadrar detalhes específicos de um personagem, como os olhos, boca, mão, etc.

3.1.2 Ângulo da câmera

O ângulo da câmera tem forte influência na forma como uma cena será percebida pelos espectadores. Mascelli (1965) descreve três tipos de ângulos de câmera (Figura 6):

- *Objetivo (Objective)*: Ângulo de câmera impessoal. Os espectadores vêem a cena pelos olhos de um observador invisível. Este tipo de ângulo é conhecido como o ponto de vista do espectador, uma vez que ele não apresenta a cena do ponto de vista de nenhum dos personagens presentes na cena. Os personagens nunca olham diretamente para a câmera;



Extreme Long Shot



Very Long Shot



Long Shot



Medium Long Shot



Medium Shot



Medium Close-Up



Close-Up



Big Close-Up



Extreme Close-Up

Figura 5 - Tipos de Tomadas.

- **Subjetivo (*Subjective*):** Ângulo de câmera pessoal. Os espectadores participam das cenas como um participante ativo ou através dos olhos de outro personagem da narrativa. Os personagens sempre olham diretamente para a câmera estabelecendo um contato olho a olho com os espectadores;
- **Ponto de Vista (*Point-of-View*):** Ângulo de câmera objetivo, mas que combina aspectos de um ângulo subjetivo. A câmera é normalmente posicionada ao lado de um personagem, assim os espectadores têm a visão sobre o ponto de vista do personagem, mas sem serem colocados no papel do personagem.



Figura 6 - Tipos de ângulos de câmera.

3.1.3 Altura da câmera

A altura da câmera em relação aos personagens de uma cena pode alterar o conteúdo psicológico e dramático da narrativa, de modo a influenciar a relação entre o espectador e os personagens. Mascelli (1965) classifica a altura da câmera em três tipos (Figura 7):

- **Eye-Level:** A altura de câmera é baseada na altura dos olhos dos personagens. Em ângulos de câmera objetivos a câmera é posicionada na altura relativa aos olhos do personagem que está sendo filmado. Em ângulos subjetivos, a câmera deve ser posicionada na altura dos olhos da pessoa que está observando a cena, normalmente deve-se considerar a altura de uma pessoa média;
- **High Angle:** A câmera é posicionada acima dos personagens que estão sendo filmados de modo que ela precise estar direcionada para baixo para filmar a cena. Pode ser utilizado para que os espectadores se sintam superiores aos personagens na cena;
- **Low Angle:** A câmera é posicionada abaixo dos personagens que estão sendo filmados de modo que ela precise estar direcionada para cima para filmar a cena. Pode ser utilizado para que os espectadores se sintam inferiores, demonstrando a dominância e autoridade aos personagens na cena.



Figura 7 - Classificação da altura da câmera.

3.1.4 Ângulo do assunto

Outro ponto importante no processo de criação de uma tomada é o ângulo do assunto (objeto ou personagem alvo na tomada). É importante manter o senso de profundidade e de tridimensionalidade da cena, para isso deve-se filmar a cena em ângulos que permitam a visualização de mais lados do assunto (Mascelli, 1965).

Como pode ser observado na Figura 8, a falta de profundidade da imagem (a) impossibilita a visualização de muitos detalhes do personagem, que são claros na imagem (b). Além disso, também é perceptível a maior vivacidade presente na imagem (a) em comparação com a imagem (b).

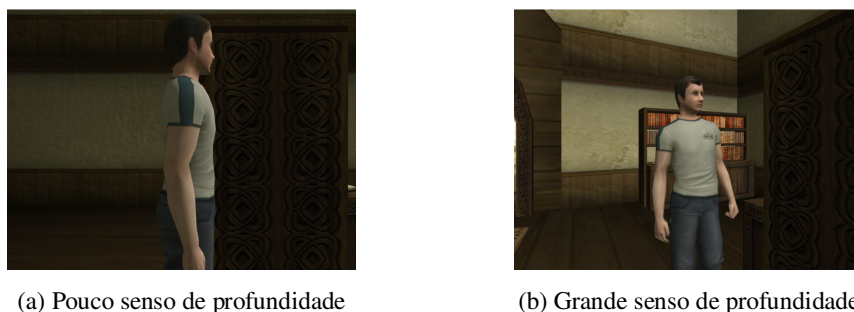


Figura 8 - Diferença entre tomadas usando ângulos com pouco e com bastante senso de profundidade.

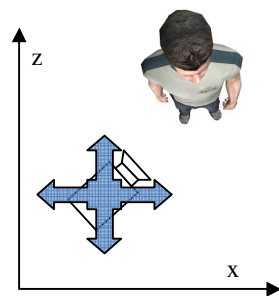
3.2 Movimentos da câmera

A movimentação das câmeras é um aspecto extremamente importante a ser levado em consideração durante o planejamento das filmagens de qualquer filme (Brown, 2002). Segundo Brown (2002), existem dois motivos que podem requerer a movimentação da câmera durante as filmagens: o primeiro deles é relativo a ações que envolvam personagens ou objetos móveis. Neste caso, os movimentos da câmera são utilizados para manter o assunto sempre enquadrado, mesmo quando este estiver se movendo. A outra motivação é relativa ao roteiro. Algumas vezes é necessário apresentar aos espectadores novos objetos ou personagens importantes para a história e ao mesmo tempo relacionar estes com o espaço da cena. Neste caso, os movimentos de câmera são utilizados para fazer esta relação, de modo que os espectadores possam entender claramente a introdução do novo elemento na cena.

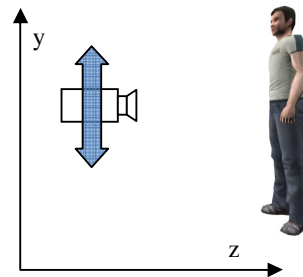
Movimentos de câmera também podem ser utilizados para definir o clima e expressar emoções nas cenas. Algumas características, como o estilo, trajetória e velocidade contribuem de formas diferentes no clima da narrativa. Por exemplo, movimentos lentos passam uma sensação de tranquilidade, enquanto movimentos rápidos podem expressar raiva ou fúria (Brown, 2002).

De acordo com Brown (2002), a cinematografia define diferentes tipos de movimentos que pode ser utilizados pelas câmeras, entre eles, os mais comuns são (Figura 9):

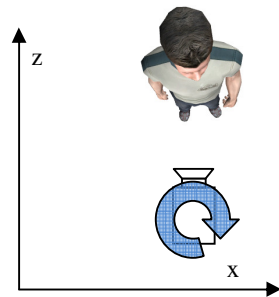
- *Dolly*: Consiste em qualquer movimento realizado pela câmera nos planos X ou Z. O seu nome é derivado do nome da estrutura com rodas em que a câmera é posicionada para realizar este tipo de movimento (Figura 9 - a);
- *Crane*: Movimento realizando pela câmera no plano Y. Seu nome é derivado dos guindastes utilizados nos estúdios de filmagens para realizar este tipo de movimento (Figura 9 - b);
- *Pan*: Consiste no movimento de rotação da câmera no eixo Y. É um movimento muito comum utilizado para mostrar os elementos que estão ao redor e compõem o ambiente das cenas (Figura 9 - c);
- *Tilt*: Consiste no movimento de rotação da câmera no eixo X. É utilizado para mostrar os elementos que estão acima ou abaixo do campo de visão normal (Figura 9 - d).



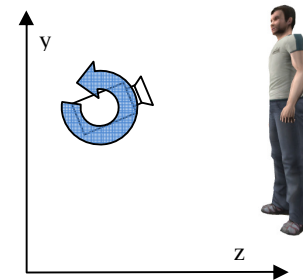
(a) Movimento de câmera Dolly.



(b) Movimento de câmera Crane.



(c) Movimento de câmera Pan.



(d) - Movimento de câmera Tilt.

Figura 9 - Tipos de movimentos de câmera.

3.3 Continuidade

Para a cinematografia, a continuidade refere-se à consistência da movimentação e posicionamento de personagens e objetos nas cenas durante a execução de cortes entre tomadas (Mascelli, 1965). Assegurar a continuidade evita que os espectadores possam ficar confusos pela mudança de ângulo da câmera durante a troca de tomadas.

Supondo uma cena de um diálogo envolvendo dois personagens P1 e P2. Considerando que a cena é iniciada com a primeira tomada do personagem P1 sendo realizada em um ângulo de aproximadamente 45° , com a câmera posicionada no lado esquerdo de P1 (Figura 10 - a). Caso a segunda tomada de P2 fosse realizada com a mesma configuração da anterior, porém do lado direito de P2 e consequentemente do lado direito de P1 (Figura 10 - B). Uma tomada assim estaria quebrando a continuidade da cena e pareceria confusa para o público. À primeira vista pareceria que os dois personagens trocaram de lugar, porém, em seguida, poderia se perceber que o cenário de fundo está diferente, mas, como ainda não houve uma ligação espacial dos personagens com o cenário, os espectadores poderiam pensar

que se trata de outro local. Identificar que a câmera está em um determinado local do cenário é algo extremamente complexo de ser realizado pelos espectadores, principalmente quando estes não conhecem totalmente o ambiente onde a cena está acontecendo.



(a) Primeira Tomada



(b) Segunda Tomada

Figura 10 - Sequência de tomadas que quebram a continuidade visual da cena.

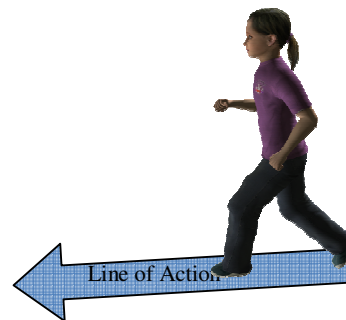
Para manter a continuidade das cenas, a cinematografia define algumas regras para serem seguidas durante as filmagens. Thompson e Bowen (2009) descrevem as seguintes regras básicas:

- *Screen Direction*: As ações devem manter a mesma direção durante o corte entre tomadas em uma mesma cena. Por exemplo, se em uma tomada um ator sair correndo pelo lado direito da tela, na próxima tomada ele deve entrar na tela pelo lado esquerdo. É importante mencionar que esta regra não é totalmente necessária durante cortes entre cenas; se a próxima cena estiver ocorrendo em um local diferente, esta regra pode ser quebrada sem maiores consequências para a continuidade;
- *Line of Action*: Para toda e qualquer cena deve-se estabelecer uma linha de ação. A linha de ação consiste em uma linha imaginária conectando e direcionando o foco de atenção da cena. Normalmente, a linha de ação se dá pela direção na qual o personagem principal da cena está olhando (Figura 11 - b). Os espectadores costumam observar estas linhas de ação e conectar os personagens com o seu foco de atenção, estabelecendo assim uma relação espacial nas cenas (Figura 11 - c). Além disso, a linha de ação também é utilizada para a aplicação de outras regras para manter a continuidade;
- *180 Degree Rule*: Todas as tomadas de uma cena devem ser feitas de apenas um dos lados da linha de ação. Após a primeira tomada em um dos lados, a câmera não deve realizar nenhuma nova tomada do outro lado (Figura 11 - c). Isso evita que o espectador possa ficar confuso devido à posição e orientação dos personagens e objetos na cena (Figura 10);

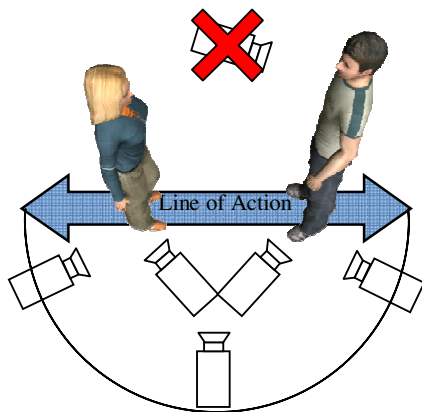
- *30 Degree Rule*: Considerando um arco de 180 graus formado em um dos lados da linha de ação, ao se iniciar uma nova tomada dentro deste arco deve-se assegurar que a câmera esteja no mínimo a 30 graus da posição onde foi feita a última tomada (Figura 11 – d). Deste modo as duas tomadas podem ser consideradas diferentes o suficiente para evitar que o filme pareça segmentado para os espectadores. O mesmo deve ser aplicado em relação à aplicação de zoom nas tomadas; este deve sempre manter no mínimo 20% de diferença para a tomada anterior;
- *Reciprocating Imagery*: As configurações de uma tomada para enquadrar um personagem devem ser exatamente as mesmas para enquadrar outros personagens na mesma cena. Mudanças na altura da câmera, tipo de lente, foco e iluminação podem comprometer a continuidade da cena.



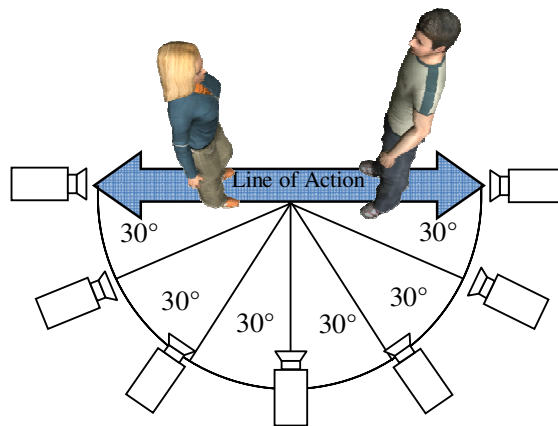
(a) Linha de ação para uma cena de diálogo entre dois personagens



(b) Linha de ação para uma cena de um personagem caminhando em um direção.



(c) Regra dos 180 graus.



(d) Regra dos 30 graus.

Figura 11 – Regras para manter a continuidade durante as filmagens.

3.4 Filmagem de cenas

Existem diversas maneiras de se filmar uma cena. Brown (2002) descreve os três métodos mais comuns:

- *Master Scene Method*: Consiste na filmagem da cena inteira com apenas uma tomada (*master shot*), juntamente com um conjunto de outras tomadas realizadas por outras câmeras da mesma cena (*coverage*). Este método busca fornecer ao editor material suficiente para realizar a edição consistente da cena, de modo que o *master shot* possa ser usado de base para garantir que nenhum momento da cena possa ser perdido. Este é o método mais comum de filmagem utilizado na maioria dos filmes atuais;
- *Plán Scene*: Método semelhante ao *master scene method*, com a diferença de que somente o *master shot* é filmado; toda a cena é apresentada através de uma única tomada, onde a câmera simplesmente efetua rotações e outros movimentos para manter a ação da cena enquadrada. Era um método bastante utilizado nos filmes até os anos 60;
- *Triple Take* ou *Overlapping Method*: Consistem em um método onde a parte final das ações realizadas em uma tomada é repetida no início da próxima. Deste modo o editor tem à sua disposição algumas partes duplicadas da cena para garantir a continuidade das ações. Entretanto, este método somente pode ser aplicado em cenas onde as ações são controláveis, ou seja, ações que podem ser repetidas sem dificuldade;

3.5 Cortes

Um corte consiste na mudança instantânea entre diferentes posicionamentos e ângulos de câmera. Brown (2002) classifica os cortes em 5 tipos:

- *Content Cut*: Corte feito com objetivo de adicionar novas informações na cena ou para avançar a história. Na sua forma mais simples, este tipo de corte é o utilizado para filmar uma cena de diálogo, onde, normalmente, são feitos diversos cortes deste tipo para realizar diversas tomadas entre os personagens;

- *Action Cut*: Corte utilizado quando uma ação começa em uma tomada e termina em outra. Por exemplo, quando na primeira tomada um personagem aparece abrindo uma porta e na próxima ele aparecer saindo pelo outro lado da porta;
- *POV Cut*: Corte efetuado quando um personagem olha para algo que não esteja visível na primeira tomada o que motiva a segunda tomada. Por exemplo, quando um personagem olha para cima e na próxima tomada aparece uma torre com um relógio;
- *Match Cut*: Corte que procura combinar duas tomadas associando-as com algum objeto similar. Serve como uma forma de transição entre as duas tomadas;
- *Conceptual Cut*: Corte semelhante ao *match cut*, entretanto o *conceptual cut* utiliza-se, além da similaridade dos objetos, alguma metáfora entre eles. Por exemplo, no filme “2001: Uma Odisseia no Espaço” existe uma cena onde um macaco descobre que é possível utilizar um osso como ferramenta; ao lançá-lo para cima, ocorre um *conceptual cut* onde é feita uma tomada de uma estação espacial que combina exatamente com a forma do osso.

3.6 Iluminação e cores

Segundo Mascelli (1965), a iluminação é um importante elemento da cinematografia. A correta manipulação da luz é capaz de melhorar as cenas em diferentes aspectos, desde a estética até o clima da narrativa.

Existem dois estilos de iluminação comumente utilizados em filmagens: o primeiro, conhecido como *high-key lighting* (Figura 12 - a), é normalmente utilizado em cenários cotidianos. Este tipo de iluminação expõe detalhes dos objetos provendo grande quantidade de informação visual sobre a cena. Entretanto, em alguns casos, é necessário criar um clima de suspense ou medo nas cenas, para isso utiliza-se o segundo estilo de luz conhecido como *low-key lighting* (Figura 12 - b). Este estilo gera sombras na cena e grande contraste nos objetos, sendo possível esconder informações visuais de modo a criar um clima medo e suspense (Abrams *et al.*, 2001).

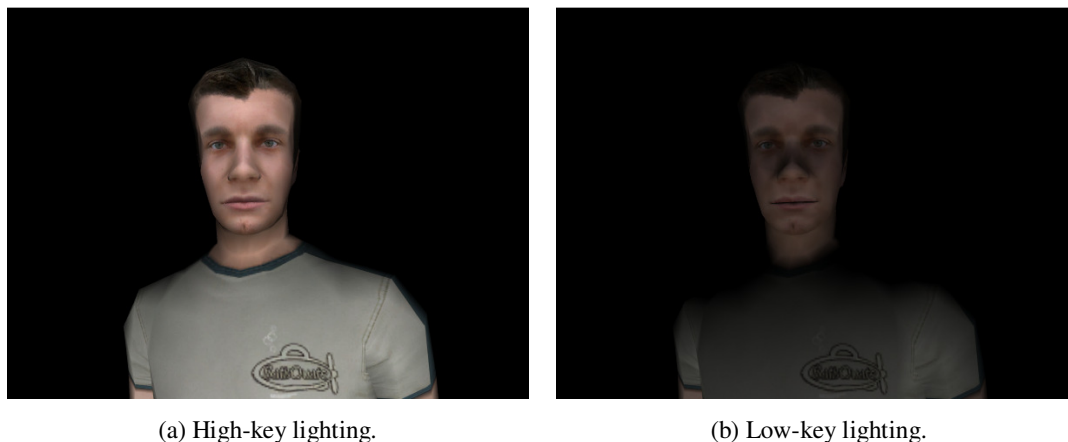


Figura 12 - Estilos de iluminação.

Para a criação de uma iluminação de qualidade no estilo *high-key lighting*, normalmente é necessária a utilização de pelo menos três fontes de luz na cena. Este método é conhecido como *three point lighting* (Figura 13); nele, cada luz tem uma responsabilidade particular para controlar a aparência da cena. A *key light* é a principal fonte de luz da cena, define a direção, intensidade, cor, entre outros aspectos. A *fill light* permite um controle adicional sobre o tom e o sombreamento dos objetos. Por ultimo a *back light* é utilizada para enfatizar a silhueta dos objetos e destacar os elementos mais importantes da cena (Millerson, 1991). Uma variação deste modelo, conhecida como *four point lighting*, considera uma quarta fonte de luz, a *background light* é utilizada para criar um senso de profundidade entre o assunto da cena e os objetos de fundo (Abrams *et al.*, 2001).



Figura 13 – Método de iluminação three point lighting.

Segundo O'Brien e Sibley (1995), a iluminação é um elemento importante para se expressar as emoções das cenas. Cenas bem iluminadas aumentam a sensação de bem-estar, enquanto cenas escuras e com sombras aumentam o impacto de emoções como medo e mau agouro. O humor é outro ingrediente vital provido pela luz. Uma iluminação brilhante, como um dia ensolarado, tende a passar aos espectadores sentimentos alegres e descontraídos. Se a luz é dura, como em um dia nebuloso, o humor passado aos espectadores tende a ser pesado e hostil. Uma iluminação fraca, como uma névoa, tende a passar sensações nostálgicas e melancólicas. A escuridão da noite, ou de uma tempestade se aproximando, passa aos espectadores sentimentos de preocupação e seriedade (O'Brien e Sibley, 1995).

A cor é um importante aspecto provido pela luz. As cores afetam psicologicamente a percepção das cenas, podem alterar clima o das cenas, assim como transmitir emoções. Segundo LoBrutto (2002), a cor pode comunicar tempo e local, definir personagens, estabelecer a emoção, humor e o clima das cenas. As cores devem ser utilizadas para transcrever visualmente as camadas de sentimentos que estão definidas no roteiro. A cor pode ser muito subjetiva, mas certas tonalidade e conjuntos de cores podem ser usados para comunicar mensagens para os espectadores. As cores quentes tendem a representar ternura e humanidade, assim como sexualidade, raiva e paixão. Cores frias representam a perda, emoções frias e sentimentos distantes. Uma paleta de cores monocromática ou uma gama limitada de cores podem ser usadas para expressar monotonia, emoções mascaradas ou uma sensação de simplicidade (LoBrutto, 2002).

Um exemplo de como a iluminação e cores influenciam a maneira como as cenas são percebidas e interpretadas pelos espectadores pode ser observado na Figura 14. Neste exemplo, a mesma cena pode ser observada através de quatro configurações diferentes de iluminação e cores.



(a) Luz brilhante na cor branca e com alta intensidade; Neblina distante.



(b) Luz suave na cor cinza e com intensidade média; Neblina a uma distancia média.



(c) Luz escura na cor cinza escuro e com baixa intensidade; Neblina densa.



(d) Luz suave na cor vermelha e com intensidade média; Neblina a uma distancia média.

Figura 14 - Exemplo de cena utilizando diferentes configurações de iluminação e cores.

3.7 Lentes e efeitos

Efeitos visuais criados com o uso de lentes especiais, técnicas de filmagem ou pós-processamento também podem ser usados para melhorar a estética das cenas, assim como expressar a época em que a cena está acontecendo e provocar determinadas emoções nos espectadores (Brown, 2002). Dentre os efeitos mais comuns utilizados no cinema, podemos citar:

- *Depth of Field*: Técnica que procura ajustar o plano de foco da câmera para que determinados objetos estejam mais nítidos que outros. Esta técnica pode ser usada para

focar a atenção dos espectadores para um determinado assunto na cena, aumentar a estética e também trazer à tona as emoções do assunto em foco;

- *Film grain*: Técnica que faz com que as cenas gravadas no filme possuam falhas nas imagens, como se fossem gravações antigas. Este efeito é bastante utilizado em alguns filmes para mostrar cenas que aconteceram há muito tempo. Em ambientes escuros e sombrios também pode ser utilizada para aumentar a sensação de medo;
- *Gray Scale*: Técnica que procura deixar as cenas em tons de cinza. Assim como o *film grain* (que pode ser usado em conjunto em alguns casos) é utilizado para representar cenas que aconteceram muito tempo atrás;
- *Vignette*: Técnica que cria bordas mais escuras nas imagens gravadas. Este efeito, quando utilizando em cenas internas e escuras, tem um efeito psicológico que passa aos espectadores uma sensação claustrofóbica e de medo;
- *Misty Border*: Técnica que adiciona bordas nebulosas ou levemente borradas na tela. É usada em cenas de flashback como um método para os espectadores reconhecerem que aquela cena está acontecendo em outra época.

3.8 Músicas

A música é uma poderosa ferramenta para expressar emoções. Em um filme a música pode mudar a sensação causada por uma cena, enfatizar emoções, assim como reforçar o entendimento que os espectadores terão das cenas. A música é parte integrante de um filme, uma vez que ajuda a conectar o conteúdo emocional com os acontecimentos visuais (Davis, 2000).

Existem várias características que são sugeridas como fontes de influência nas emoções causadas pela música. Por exemplo, segundo Gabrielsson e Lindstrom (2001), trilhas sonoras em escala maior, ritmo rápido causam felicidade, enquanto que trilhas sonoras em escala menor e ritmo lento causam tristeza; ritmos rápidos juntamente com dissonância causam medo. Um aspecto importante das características das músicas é que elas são muito dependentes do contexto musical. Por exemplo, uma música em escala maior pode não ser necessariamente triste se ela possuir ritmo rápido. É importante levar todos os fatores em consideração. Existem inúmeros estudos que analisaram os elementos da composição musical

no que se refere às emoções expressadas. Gabrielsson e Lindstrom (2001) resumem um grande número desses estudos.

A música pode evocar emoções e sentimentos tão bem porque trabalha abaixo do nível consciente das pessoas (Miller, 1988). A música pode acrescentar conteúdo emocional a um filme sem ocupar mais tempo de dramatização. Pode colocar o espectador em um mundo diferente, um país diferente, ou em um tempo diferente. Pode criar um mundo obscuro e misterioso acrescentando tensão às cenas de modo a reforçar a seriedade de situações (Larsen, 2005).

A música pode produzir diferentes tipos de emoções em diferentes espectadores e também pode ter respostas emocionais diferentes na mesma pessoa em momentos diferentes. A experiência de vida e o bom senso são responsáveis por essas variações nas reações emocionais (Miller, 1988). Da mesma maneira como uma música pode melhorar uma cena, uma música incorreta pode arruinar completamente a cena, anular, ou até mesmo mudar as emoções que estão sendo expressadas pelos atores.

3.9 Equipe de filmagem

As filmagens e a produção de um filme envolvem uma grande equipe de profissionais cineastas. Cada um possui determinadas responsabilidades durante o processo de criação de filme. Salles (2008) descreve os seguintes profissionais principais envolvidos neste processo:

- **Diretor:** No cinema, o diretor é responsável pelo filme como um todo. Deve conhecer perfeitamente todos os detalhes do roteiro e ter previamente uma imagem feita de cada cena. Além disso, o diretor deve conhecer detalhadamente cada função técnica da criação de um filme, saber o que pode e o que não pode ser feito. Deve ter uma cultura literária, musical e dramática elevada, pois tudo servirá como referência em sua criação. Deve ajudar a escolher a melhor trilha sonora, melhores tomadas e a melhor forma de extrair a dramaticidade desejada de seus atores;
- **Diretor de Fotografia:** O diretor de fotografia é responsável por todos os aspectos visuais de um filme. Tem a função de definir a iluminação, escolher os equipamentos de filmagem, indicar a exposição e o uso de filtros.

- **Roteirista:** No cinema o roteirista é responsável por escrever a história do filme ou, em alguns casos, adaptar uma obra literária. É o criador do texto roteiro com alguma trama especializada para o cinema de fatos reais ou ficcionados;
- **Cameraman:** O cameraman é o técnico de cinema que opera as câmeras de vídeo. É responsável pelos enquadramentos e tomadas que compõem as cenas, sejam tomadas fixas ou em movimento. O cameraman deve estar apto a operar a câmera utilizando tripé ou equipamentos especiais utilizados para filmagens em movimento. Deve conhecer os tipos de lentes, filtros, acessórios de câmera, controlar a qualidade do foco ou o movimento de zoom, assim como a qualidade de execução de movimentos. O cameraman recebe ordens do diretor de fotografia.
- **Diretor de Música:** É responsável pela criação de todas as músicas a serem colocadas na trilha sonora do filme, acompanhando em background as imagens montadas de um filme, visando auxiliar a obtenção do "clima dramático" da cena desejado pelo diretor. Seu trabalho pode ser feito isolado em um estúdio com seus músicos e depois ajustado na montagem ou a partir das imagens visionadas previamente.
- **Editor:** É o técnico de cinema e vídeo que reúne, ordena e dá ritmo às imagens filmadas e sons captados no set de filmagem. Procura dar um sentido narrativo à história proposta pelo roteiro literário do filme.

4 ARQUITETURA DE DRAMATIZAÇÃO CINEMATOGRAFICA

Neste capítulo são apresentadas algumas abordagens utilizadas em trabalhos anteriores que buscam, de alguma forma, utilizar alguns dos princípios de cinematografia em ambientes interativos. Em seguida, será apresentada a proposta do novo modelo de dramatização baseado em regras princípios e cinematográficos, bem como a estrutura e os requisitos adotados.

4.1 Trabalhos relacionados

A manipulação de câmeras em ambientes tridimensionais tem sido alvo de diversas pesquisas nos últimos anos. Dentre as linhas de pesquisa mais comuns é possível distinguir duas abordagens de maior enfoque:

- Satisfação de restrições (*Constraint satisfaction*): Abordagem que busca utilizar restrições para definir o posicionamento e comportamento das câmeras;
- Baseados em *idioms* (*Idiom-based*): Abordagem que utiliza uma estrutura de dados para armazenar a forma como a câmera deve filmar cada tipo de cena. Este tipo de estrutura é conhecida como *idiom*. Consiste em um modo simplificado de se codificar conhecimento cinematográfico.

Um dos primeiros trabalhos a utilizar satisfação de restrições no controle de câmeras foi proposto por Drucker e Zeltzer (1995). O trabalho propõe um modelo que encapsula todas as tarefas relacionadas com câmeras em módulos. O sistema expressa esses módulos utilizando restrições visuais, como, por exemplo, qual tamanho cada personagem deve ter na tela e em que ângulo eles devem estar posicionados. Isso evita que o usuário precise lidar com comandos de baixo nível para controlar a câmera. Por exemplo, para se gerar um close-up é possível especificar como entrada para o sistema que a cena deve exibir a face de determinado personagem e que ela deve ocupar 90% da tela. A partir desta entrada o sistema de restrições retorna a posição e ângulo de rotação para a câmera virtual. Porém, o sistema não oferece uma forma automatizada para se lidar com excessos de restrições, o que pode ocorrer frequentemente em ambientes complexos e dinâmicos.

Em uma abordagem baseada em *idioms*, He *et al.* (1996) utiliza a estrutura para descrever o posicionamento e comportamento das câmeras para determinados tipos de cena. As regras cinematográficas são codificadas como máquinas de estados finitos hierárquicas e controlam o posicionamento e transições entre tomadas automaticamente. O sistema também utiliza uma biblioteca de módulos de câmera que contém informações sobre a posição dos atores na tela e o tipo de tomada. O sistema é apto a ser utilizado com sucesso em cenas simples envolvendo dois ou três atores. Entretanto, os autores mencionam que o sistema pode falhar em cenas complexas devido a oclusões inesperadas. A outra limitação que merece destaque é que o sistema não consegue lidar com cenas que não foram previstas durante a sua modelagem.

Amerson e Kime (2000) descreve um modelo similar ao utilizado por He *et al.* (1996), porém utilizando uma árvore para codificar os *idioms* em diversos níveis. As cenas são expressas em sequências de restrições, onde cada restrição possui um peso indicando a sua importância relativa para a tomada. O sistema adota uma arquitetura formada por três módulos: tradutor, diretor e cinematógrafo. O tradutor utiliza como entrada o mundo virtual e transforma estas informações em dados que podem ser entendidos pelo Diretor, que utiliza essas informações para realizar uma pesquisa de profundidade para selecionar o melhor *idiom* na árvore e envia esta informação para o cinematógrafo, que, por fim, utiliza os parâmetros do *idiom* para posicionar e manipular a câmera no mundo virtual.

Adotando uma abordagem diferenciada das apresentadas anteriormente, Tomlinson *et al.* (2000) apresenta um sistema onde a câmera, assim como todos os outros personagens no ambiente, são agentes autônomos. Sensores captam informações sobre as emoções e as motivações dos personagens. Estas informações são então combinadas com as emoções e motivações da própria câmera para determinar o comportamento que a câmera terá durante a filmagem das cenas. Por exemplo, uma câmera feliz realizará cortes mais frequentes e adotará movimentos oscilantes, enquanto que uma câmera triste adotará movimentos longos e violentos. O sistema também utiliza as informações emocionais para controlar a iluminação das cenas.

A importância de existir um bom equilíbrio entre a satisfação de restrições e a coerência nos frames para a geração de movimentos e transições suaves na câmera são discutidos por Halper *et al.* (2001). Os autores apresentam um método que, quando ocorrem mudanças entre tomadas, assegura que a câmera não salte para a nova localização, mas se mova suavemente para a nova posição.

Em um trabalho focado especificamente para storytelling interativo, Charles *et al.* (2002) propõem um módulo de câmera que desempenha um papel de diretor virtual, apresentando ao espectador as situações mais relevantes a cada momento. Para isso, todos os eventos possuem um peso/importância para a narrativa, o sistema utiliza esses pesos para criar uma classificação heurística dos eventos ativos e escolher o mais significativo. Em seguida o sistema seleciona o *idiom* mais adequado para exibir a cena.

Friedman e Feldman (2002) afirmam que a utilização de *idioms* é uma abordagem simplificada para formalizar os princípios cinematográficos. Por este motivo, os autores propõem uma abordagem mais refinada de regras, onde procuram utilizar em conjunto as melhores características das abordagens baseadas em *idioms* e satisfação de restrições. O sistema, chamado de “Mario”, gera automaticamente filmes de animação 3D a partir de um roteiro utilizando um sistema multicamadas de regras cinematográficas. Entretanto o sistema não trata problemas de oclusões e é destinado para cenas compostas por apenas um ator.

El-Nasr (2004) propõe uma arquitetura de dramatização de histórias onde o autor busca representar através de um conjunto de regras, o comportamento da câmera, luz e atores virtuais. O comportamento da câmera é definido através de *idioms*, baseando-se em informações geométricas das cenas. O controle da iluminação das cenas é feito através de um conjunto de regras e funções que buscam estabelecer a melhor posição, foco visual, clima do ambiente, visibilidade dos personagens e aumentar o senso de realismo das cenas (El-Nasr, 2003).

Hawkins (2005) propõe uma possível arquitetura de dramatização onde ele procura representar através de módulos, o papel de alguns elementos da indústria cinematográfica. O autor divide a arquitetura em 3 módulos distintos: o Diretor, cujo objetivo é sugerir *idioms* para as filmagens; o Editor que escolhe as tomadas que serão filmadas; e, por último, o Cinematógrafo, que faz a manipulação direta das câmeras. Um dos problemas deste sistema é a utilização de apenas informações geométricas para a geração das tomadas, o que pode ser insuficiente para a geração de filmagens com ênfase no conteúdo dramático.

Bourne e Sattar (2005) propõem um sistema baseado em conjuntos de restrições representados através de perfis de câmera. Os autores apresentam um método que utiliza algoritmos genéticos para evoluir os perfis e assim encontrar o mais adequado.

Christie e Normand (2005) argumentam que uma tomada pode ter diversas soluções visuais, ou seja, uma mesma tomada pode ser composta de diversas maneiras. Para demonstrar isso, foi proposto um sistema que gera um conjunto de possíveis tomadas para as

cenar. Entretanto, o sistema apenas apresenta as possíveis soluções; algum processamento adicional seria necessário para selecionar a melhor solução. Além disso, atualmente ainda seria inviável a sua aplicação em tempo real devido ao alto custo computacional do modelo proposto.

Em uma abordagem baseada em satisfação de restrições, Hermann (2005) propõe um módulo cinegrafista para o posicionamento automático da câmera em ambientes 3D. No sistema, os requisitos de posicionamento são expressos em um conjunto de restrições que devem ser atendidas pelo modelo de câmera. Para isso, a câmera usa um modelo físico baseado em um sistema de partículas regido pelo método de Verlet (Verlet, 1967), empregando o método de relaxação para a convergência do sistema a fim de atender às restrições impostas.

Cozic (2007) apresenta uma abordagem de modelo cinematográfico voltada para jogos que procura utilizar conceitos de cinematografia, mas mantendo ao mesmo tempo a jogabilidade como principal objetivo. O sistema busca detectar quando o jogador está próximo a um objeto ou personagem importante do jogo para então incluí-lo suavemente no enquadramento da câmera. Entretanto, o autor não menciona como o sistema se comportaria em cenas complexas compostas por diversos personagens. Além disso, por priorizar a jogabilidade, são poucos os conceitos de cinematografia que podem ser aplicados no modelo. Dando continuidade ao trabalho apresentado por Cozic (2007), Schwartz (2009) propõem um método que utiliza lógica fuzzy para usar a câmera como uma ferramenta de auxílio aos jogadores durante a execução de tarefas e resolução de mistérios.

Burelli e Jhala (2009) propõem um sistema para o controle de câmeras que emprega *artificial potential fields* para movimentação da câmera e composição de frames. As colisões da câmera com o ambiente são evitadas utilizando forças repulsivas, e a composição de frames é obtida através da transformação das constantes da tomada em forças que afetam a câmera e o ponto para o qual ela vai estar direcionada (*look-at-point*). Porém o sistema não utiliza outros princípios cinematográficos, além de apresentar problemas de mínimos locais em ambientes complexos.

4.2 Modelo de dramatização cinematográfico proposto

O modelo de dramatização proposto neste trabalho visa utilizar conceitos e princípios definidos pela cinematografia com objetivo de enfatizar o conteúdo dramático das cenas dramatizadas, apresentando-as de modo atraente e envolvente para os espectadores.

4.2.1 Requisitos

Com base nos princípios e regras definidas pela cinematografia e nos trabalhos relacionados, foram identificados alguns requisitos necessários para um sistema de dramatização de qualidade. Desta forma, foram estabelecidos os requisitos necessários para o desenvolvimento da arquitetura proposta neste trabalho. Estes requisitos são descritos a seguir:

1. **Flexibilidade:** Um dos principais desafios na criação de um sistema de dramatização é torná-lo flexível, adaptável e funcional independente do ambiente ou história a ser dramatizada. Um modelo de dramatização completo deve ser autônomo e adaptável a qualquer contexto;
2. **Expressar e enfatizar as emoções das cenas:** Um dos principais objetivos da cinematografia é enfatizar e expressar adequadamente a emoção das cenas para os espectadores. Um sistema de dramatização com ênfase no conteúdo dramático deve atuar de modo a ser guiado pelas emoções e sentimentos dos personagens. A escolha das tomadas, assim como a iluminação, música e outros aspectos precisam sempre levar em consideração o estado emocional e as relações existentes entre os personagens envolvidos nas cenas;
3. **Manter a continuidade:** Manter a continuidade é algo primordial para qualquer filme. Cenas que quebram a continuidade costumam deixar os espectadores confusos, com a sensação de que algo está errado ou de que algo foi perdido. Na representação virtual de uma história é função do sistema de dramatização assegurar a continuidade das cenas. As mesmas regras utilizadas pela cinematografia podem ser aplicadas em ambientes virtuais; mais especificamente um sistema de dramatização deve:

- a. Manter a mesma direção das ações durante cortes entre tomadas em uma mesma cena;
 - b. Em todas as cenas deve ser estabelecida uma linha de ação conectando e direcionando o foco de atenção da cena;
 - c. Todas as tomadas realizadas para uma cena devem ser feitas de apenas um dos lados da linha de ação;
 - d. Ao efetuar um corte para uma nova tomada o deve-se assegurar que a câmera esteja no mínimo a 30 graus da posição onde foi feita a última tomada;
 - e. As configurações de altura da câmera, iluminação e efeitos visuais devem ser mantidas inalteradas entre cortes realizados na mesma cena.
4. **Realizar tomadas consistentes e adequadas:** Uma tomada é a menor unidade de um filme; tomadas inadequadas podem comprometer completamente a narrativa. Um sistema de dramatização deve assegurar que as tomadas realizadas sejam adequadas e consistentes com as cenas dramatizadas. Para isso, o sistema deve ser capaz de:
- a. Sempre enquadrar adequadamente o assunto;
 - b. Evitar oclusões;
 - c. Utilizar uma configuração de câmera adequada para cada cena;
 - d. Utilizar tomadas que garantam a continuidade.
5. **Realizar cortes entre tomada e entre cenas:** Filmagens contínuas e sem cortes de câmera eram comuns somente em filmes antigos. Isso indica que ao longo dos anos a indústria cinematográfica adaptou-se a um modo mais adequado de filmagens. Um modelo de dramatização deve assumir que devem existir várias tomadas em uma cena e, conseqüentemente, cortes entre elas, além dos cortes que ocorrem entre as cenas;
6. **Utilizar adequadamente a iluminação:** A iluminação é parte essencial de qualquer ambiente e deve ser utilizada adequadamente tanto para melhorar o aspecto visual como para expressar as emoções das cenas. O sistema de dramatização deve ser capaz de manipular corretamente a iluminação do ambiente de modo a enfatizar as emoções e deixar as cenas visualmente agradáveis. Ao mesmo tempo o sistema deve garantir que a continuidade não seja quebrada devido à utilização de diferentes tipos de iluminação durante a dramatização;
7. **Utilizar adequadamente as músicas:** A música é uma poderosa ferramenta para expressar as emoções das cenas. Diferentes trilhas sonoras podem mudar a maneira como as cenas são percebidas pelos espectadores. Um sistema de dramatização deve

ser capaz de utilizar adequadamente as músicas, de modo que estas sejam coerentes com as cenas que estão sendo dramatizada, e possam enfatizar corretamente as emoções das cenas.

4.2.2 Arquitetura

A arquitetura proposta neste trabalho é baseada no modelo apresentado por Hawkins (2005), onde o autor procura dividir o sistema em módulos baseando-se no modelo adotado na criação de um filme. Entretanto, este trabalho se diferencia em vários aspectos:

- Agentes autônomos: A proposta inicial de Hawkins era de que o modelo deveria ser autônomo. Entretanto, em momento algum o autor menciona como os módulos apresentados poderiam agir de forma autônoma. Na arquitetura proposta neste trabalho os módulos são modelados e desenvolvidos na forma de agentes autônomos;
- Informações emocionais: Hawkins utiliza no seu modelo somente informações geométricas para a geração das tomadas. No entanto isso pode ser insuficiente para a geração de filmagens com ênfase no conteúdo dramático. Neste trabalho o modelo será guiado pelas emoções das cenas. O principal objetivo dos agentes é enfatizar as emoções e apresentar o conteúdo dramático de modo coerente, atraente e evolvente para os espectadores;
- Storytelling: O modelo apresentado por Hawkins é destinado a jogos e não a sistemas de storytelling. A arquitetura proposta neste trabalho se destina a ambientes de storytelling, diferente de jogos onde a jogabilidade é considerada uma das maiores preocupações. Em storytelling a qualidade das histórias em termos de coerência e conteúdo dramático deve ser considerada como a maior preocupação;
- Diretor de fotografia e diretor de música: A arquitetura proposta neste trabalho acrescenta dois novos agentes cinematográficos na estrutura proposta por Hawkins. O diretor de fotografia é responsável pela manipulação da luz e a criação de efeitos especiais, enquanto que o diretor de música é responsável por adicionar trilhas sonoras adequadas às cenas;
- Implementação: Segundo Cozic (2007), o modelo proposto por Hawkins não foi completamente implementado. A arquitetura proposta neste trabalho foi totalmente

implementada, testada e validada, tanto em questões de desempenho em tempo real, como o resultado visual e emocional transmitido pela dramatização.

A arquitetura de dramatização proposta neste trabalho e a interligação desta com o sistema de geração de histórias do Logtell é apresentada na Figura 15.

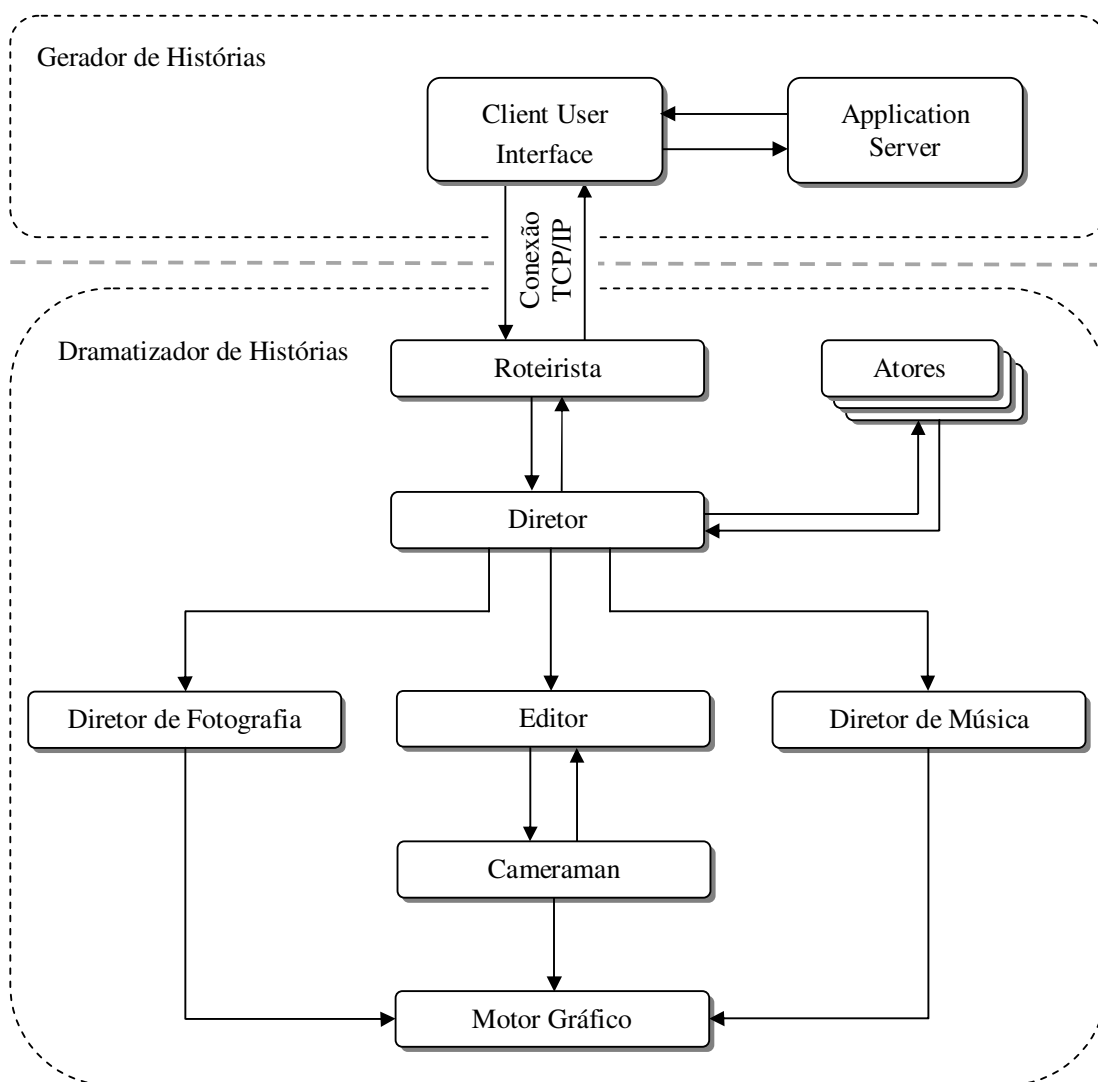


Figura 15 - Arquitetura de dramatização cinematográfica.

Na arquitetura de dramatização proposta, o módulo Roteirista é responsável por fazer a comunicação entre o sistema de dramatização e o módulo gerador de histórias. É também

responsável por expandir os eventos da história criando micro-ações detalhadas sobre os eventos de alto nível. O Diretor é responsável por interpretar os eventos da história recebidos do Roteirista, enviar os comandos necessários para os atores, em seguida comunicar o Diretor de Fotografia, Diretor de Música e Editor sobre os eventos que serão dramatizados. O Editor é responsável por escolher qual a melhor tomada para filmar a cena em um determinado momento. O Cameraman tem o objetivo de criar um conjunto de possíveis tomadas para filmar determinada cena e em seguida sugerir estas possíveis soluções para o Editor selecionar a melhor opção. O Diretor de Fotografia é responsável por determinar o aspecto visual da dramatização, manipular a iluminação e o ambiente para criar o clima emocional da cena. O Diretor de Música tem o objetivo de criar o clima emocional das cenas, trabalhando com as trilhas sonoras utilizadas durante a dramatização.

Esta arquitetura será incorporada ao sistema de storytelling Logtell (descrito no capítulo 2), substituindo o antigo módulo de dramatização. A arquitetura deverá funcionar de forma compatível com os demais módulos do Logtell, sendo compatível com os mecanismos de geração de enredos e interação com os usuários.

5 IMPLEMENTAÇÃO

Neste capítulo, são abordadas as principais questões relativas à implementação da arquitetura de dramatização apresentada no capítulo 4.

5.1 Ferramentas e métodos

A primeira versão do módulo de dramatização do Logtell (Pozzer, 2005) era implementada em C++ utilizando a API OpenGL. Apesar de funcional, alguns problemas podem ser apontados:

- **Complexidade:** Devido à não utilização de uma *game engine*, todos os módulos necessários para a dramatização 3D (física, animações, *loaders* de modelos 3D, entre outros) tiveram que ser implementados, o que resultou em inúmeras linhas de código e tornou complexo o processo de adição de novos recursos ou melhoria dos existentes;
- **Baixa qualidade gráfica:** O formato dos modelos 3D utilizados na implementação (MD2 e 3DS) já podem ser considerados ultrapassados. Além disso, a não utilização de *shaders* resulta em uma qualidade gráfica que hoje pode ser considerada rudimentar;
- **Falta de conceitos de cinematografia:** Poucos, ou praticamente nenhum, conceito de cinematografia foram utilizados na implementação do sistema de dramatização.

Para o desenvolvimento do novo sistema de dramatização proposto neste trabalho optou-se pela utilização de uma *game engine*. Deste modo, o problema da complexidade da versão antiga foi reduzido, pois com uma *game engine* é possível concentrar o desenvolvimento na lógica do sistema e menos em detalhes como a simulação física, modelos 3D, animações, etc. Uma *game engine* tem como objetivo abstrair e facilitar o processo de desenvolvimento de jogos, além de proporcionar um código eficiente e otimizado. A *game engine* escolhida para o desenvolvimento da arquitetura proposta neste trabalho foi a Unity3D (Unity3D, 2010).

A Unity3D é uma *game engine* que possui um ambiente integrado de desenvolvimento que permite a criação de cenários 3D de maneira rápida e simplificada. Proporciona a criação

de jogos com ótima qualidade gráfica utilizando um *pipeline* otimizado para o DirectX ou OpenGL. Permite a criação de *shaders* utilizando a linguagem CG, além de possuir uma completa biblioteca de *shaders* pré-definidos. A simulação física é realizada internamente utilizando a *engine* física Ageia PhysX e permite a simulação realista da física em ambientes 3D. Para a programação, a Unity3D permite a utilização de 3 linguagens script: Javascript, C# e Boo (Unity3D, 2010).

Dentre as linguagens de programação disponíveis na Unity3D optou-se pela utilização do C#, que é uma linguagem de programação orientada a objetos desenvolvida pela Microsoft como parte da plataforma .NET. A sua sintaxe foi baseada no C++ com muitas influências da linguagem Java. A escolha do C# como linguagem para o desenvolvimento deste trabalho foi feita devido à característica orientada a objetos, que não está diretamente presente nas outras linguagens utilizadas na Unity3D.

A Unity3D atende a todas as necessidades para o desenvolvimento da arquitetura proposta neste trabalho, além disso, proporciona uma ótima qualidade gráfica e suporta modelos 3D de ótima qualidade, suprindo assim o problema da baixa qualidade gráfica da versão anterior do sistema de dramatização.

A arquitetura de dramatização foi implementada de modo a substituir o antigo módulo de dramatização do Logtell. Buscou-se manter o mesmo padrão utilizado na versão anterior de modo a evitar alterações nas camadas superiores de interação e de geração da história. A única mudança relevante foi a substituição do método de comunicação entre os módulos. Na versão anterior a comunicação era feita através de uma DLL e de chamadas nativas entre as classes. Isso é necessário devido à utilização de linguagens de programação distintas nas implementações dos módulos. Na nova arquitetura, este método de comunicação foi substituído por uma conexão TCP/IP local para a troca de mensagens entre os módulos. Este método foi adotado devido à sua simplicidade e maior compatibilidade com a Unity3D.

5.1.1 Rede de relações e emoções

As emoções e relações dos personagens envolvidos em uma dramatização são a base para a utilização da maioria dos conceitos de cinematografia. Entretanto, o Logtell não dispõe deste tipo de informação, o que inviabilizaria completamente a dramatização de histórias com

ênfase no conteúdo dramático. Por este motivo, o sistema de dramatização proposto incorpora uma rede de relações e emoções. Na estrutura cada ator é representado através de um nó e a relação entre os atores através de conexões entre os nós (Figura 16).

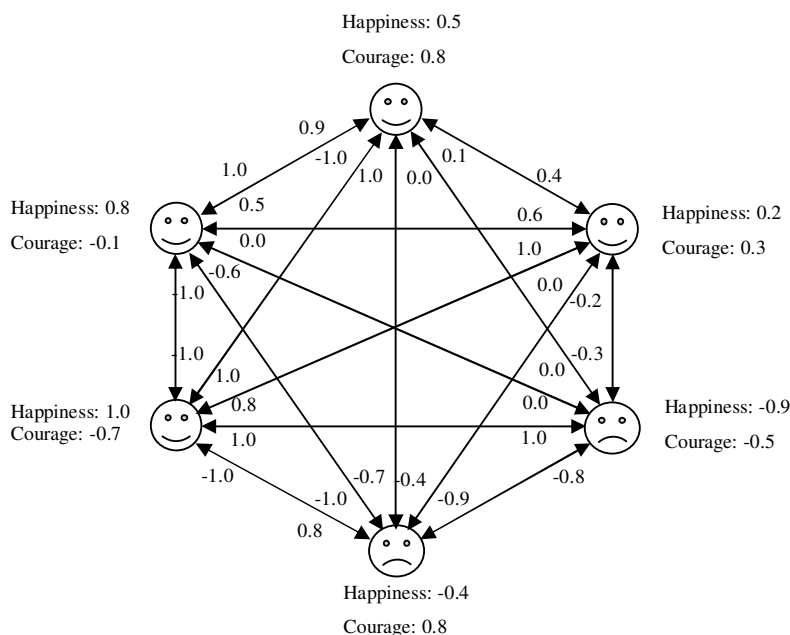


Figura 16 - Rede de relações e emoções.

O estado emocional de um ator i no tempo t é definido pelos níveis de intensidade de suas emoções básicas. Este nível é representado numericamente no intervalo de $[-1, +1]$. Por exemplo, o nível de felicidade do um ator i no tempo t pode ser descrito como $e_i^1(t) = happiness_i(t)$. Valores de $happiness_i(t)$ próximos de $+1$ representam um estado emocional de felicidade, enquanto valores próximos de -1 representam um estado emocional de tristeza. A utilização de níveis de intensidade de emoções permite representar uma variação entre duas emoções básicas com estados emocionais opostos.

Os níveis de intensidade das relações entre os atores também são representadas numericamente no intervalo de $[-1, +1]$. Supondo dois atores i e j em um determinado tempo t . A relação do ator i com o ator j , do ponto de vista de i , em um tempo t , pode ser formalizada pela função $affection_{i,j}(t)$. Quanto mais próximo de -1 for o valor da relação, menos o ator i

gosta do ator j , quanto mais próximo de +1 for o valor, mais o ator i gosta do ator j . As relações são diretas e não são necessariamente simétricas:

$$\exists i, j \text{ relation}_{i,j}(t) \neq \text{relation}_{j,i}(t)$$

A rede possui uma topologia fixa envolvendo todos os atores da história. A inicialização da rede é feita através de um arquivo XML (*Extensible Markup Language*), onde se define o estado emocional e a relação inicial de todos os personagens da narrativa. A criação deste arquivo deve ser incluída no processo de autoria da história.

A partir da definição dos estados emocionais e relações iniciais dos atores, a rede de relações e emoções pode ser utilizada de forma dinâmica durante a dramatização, sendo atualizada em tempo real sempre que algum evento importante ocorrer com os atores. A função de atualização pode ser descrita como:

$$\varphi(\text{action}, \text{executor}, \text{victim})$$

onde:

- *Action* é a ação do evento;
- *Executor* é o vetor de atores que executaram a ação;
- *Victim* é o vetor de atores que sofreram a ação;

A função de atualização φ usa uma função de avaliação para definir o impacto emocional causado pelas ações. Cada ação tem uma maneira diferente de afetar a emoção e a relação dos atores que executaram e sofreram a ação. Por exemplo, para atualizar as emoções e relações dos atores envolvidos uma ação *Kill* (onde um ator mata um ou mais atores), a função de avaliação irá classificar esta ação em três tipos diferentes baseado no estado atual dos atores:

- **Kill Intencional:** ocorre quando o executor da ação tem uma relação de ódio com a vítima ($\text{affection}_{e,v}(t) < -0.3$), neste caso o ator executor da ação irá incrementar a sua emoção $e_i^1(t) = \text{happiness}_i(t)$ em +0.3 e $e_i^2(t) = \text{courage}_i(t)$ em +0.2 após executar a ação, pois era algo que o executor desejava. Deste modo ele ficou mais feliz ao conseguir matar alguém que ele odiava e consequentemente a sua coragem também será incrementada.
- **Kill indiferente:** ocorre quando o ator tem uma relação de indiferença com a vítima ($\text{affection}_{e,v}(t) > -0.3$ e $\text{affection}_{e,v}(t) < +0.3$), neste caso, se o ator executor da ação possuir uma índole má ($\text{alignment}_i(t) < 0$) ele irá incrementar a sua emoção $e_i^1(t) = \text{happiness}_i(t)$ em +0.1 e $e_i^2(t) = \text{courage}_i(t)$ em +0.2 após executar a ação. Já

se o ator executor da ação possuir uma índole boa ($alignment_i(t) > 0$) ele irá decrementar a sua emoção $e_i^1(t) = happiness_i(t)$ em -0.1 e $e_i^2(t) = courage_i(t)$ em -0.2 após executar a ação.

- **Kill não-intencional:** ocorre quando o executor da ação tem uma relação de amor ou simplesmente gosta da vítima ($affection_{e,v}(t) > +0.3$), neste caso o ator executor da ação irá decrementar a sua emoção $e_i^1(t) = happiness_i(t)$ em -1.0 e $e_i^2(t) = courage_i(t)$ em -1.0 após executar a ação, isso por que não era algo que o executor desejava. Deste modo ele ficou mais triste por ter matado alguém que ele gostava e sua coragem diminuiu por ele ter feito algo que ele não desejava.

A função de atualização da rede é desencadeada pelos eventos da dramatização. Se nenhum evento ocorrer, a emoção e as relações dos atores não irão evoluir. No caso contrário, a função de atualização é usada para atualizar as emoções e as relações na rede.

Na implementação utilizada neste trabalho optou-se pela utilização de duas emoções básicas (felicidade/tristeza e coragem/medo), além da índole de cada personagem. A escolha de somente estas emoções se justifica devido a elas serem consideradas as emoções mais básicas. Praticamente todos os outros estados emocionais podem ser considerados derivações destes. Além disso, atualmente, a história gerada pelo gerador de enredos pode ser considerada simples e não requer estados emocionais complexos para os personagens.

5.2 Agentes cinematográficos

A arquitetura de dramatização proposta neste trabalho é composta por 6 agentes autônomos. O diagrama de classes e a estrutura hierárquica resumida dos agentes cinematográficos que compõem o sistema de dramatização são mostrados na Figura 17. Nas sessões seguintes serão apresentados detalhes sobre a implementação de cada um dos agentes.

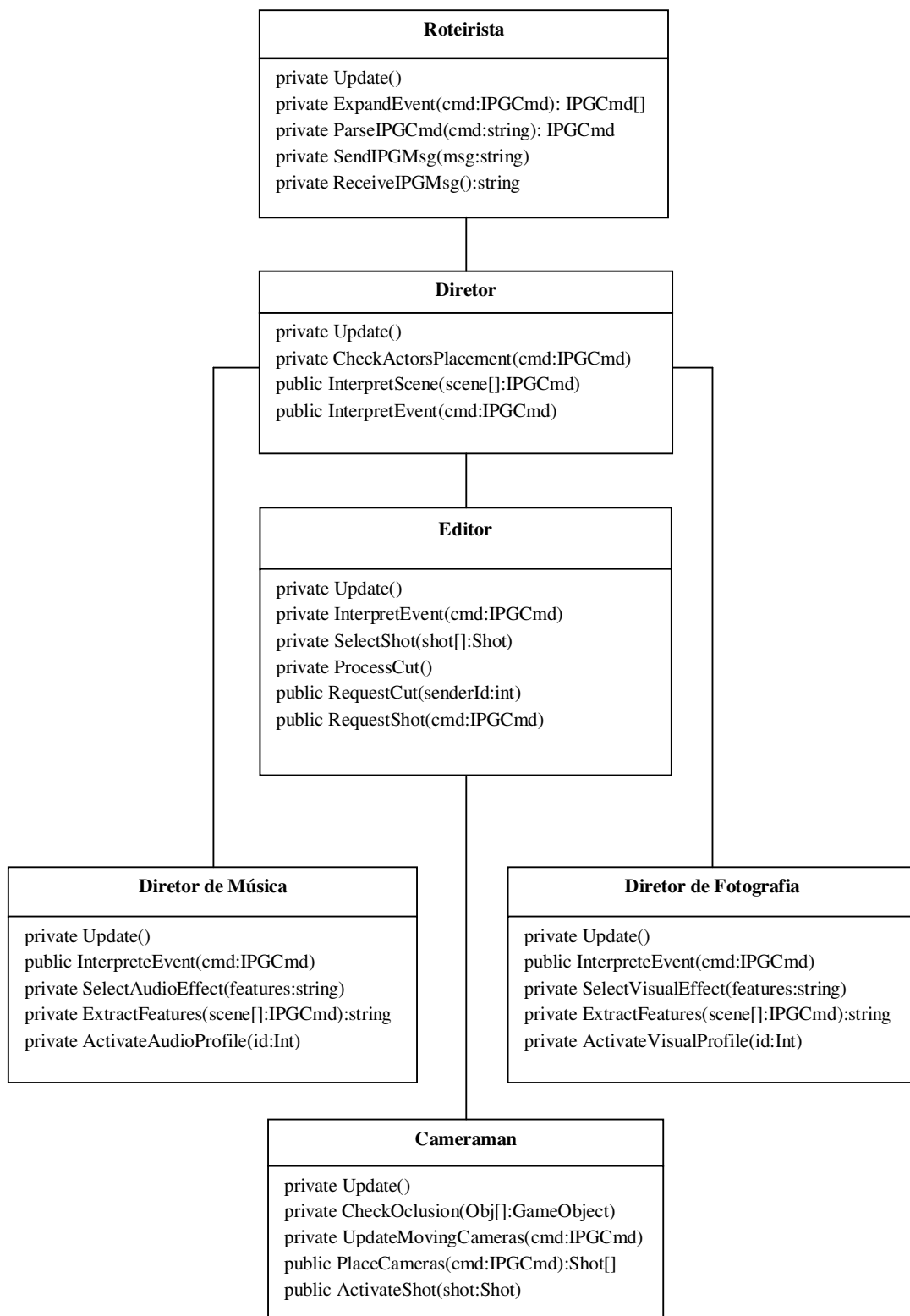


Figura 17 - Diagrama de classes dos agentes cinematográficos.

5.2.1 Roteirista

Como mencionado anteriormente, no Logtell a camada de geração da história é implementada em Prolog e o módulo de gerenciamento e interação com os usuários é implementado em Java. Para que o sistema de dramatização funcione de forma compatível com o Logtell é necessário que este se comunique com os demais módulos para o recebimento dos eventos a serem dramatizados.

Na arquitetura proposta, o agente Roteirista é responsável por fazer esta integração entre o sistema de dramatização e as camadas superiores de gerenciamento e interação com os usuários. A comunicação entre as duas camadas é feita utilizando uma conexão TCP/IP local. Adota-se um modelo cliente/servidor onde o módulo de dramatização atua como servidor e o *Client User Interface* (módulo do Logtell responsável pelo gerenciamento local da história e interação com os usuários) como cliente (Figura 18). Para isso, o agente Roteirista ao ser inicializado instancia um *socket* na porta 3000 e aguarda por conexões. No momento que uma nova história é solicitada pelo usuário, o *Client User Interface* se conecta ao endereço IP 127.0.0.1 (localhost) na porta 3000, pressupõe-se que o agente Roteirista já estará aguardando por conexões. Depois de conectado o *Client User Interface* pode enviar os comandos da história (no formato de strings) para serem dramatizados.

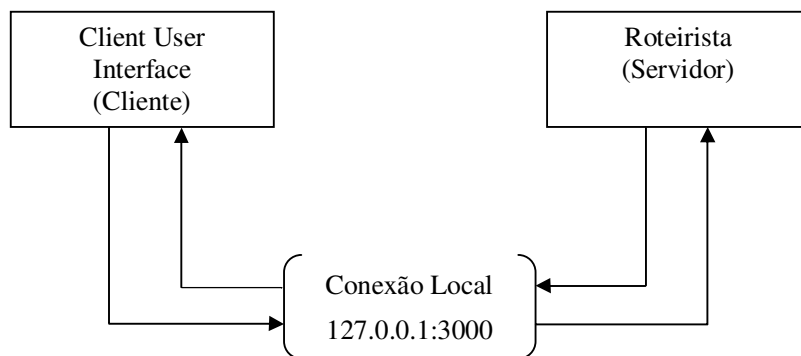


Figura 18 - Conexão TPC/IP entre o agente Roteirista e o Client User Interface.

As regras do protocolo utilizado para a comunicação entre os módulos consistem em:

- O módulo de dramatização (servidor) deve sempre aguardar por conexões de clientes na porta 3000;
- O módulo que gerencia os eventos da história (cliente) deve, sempre que desejar iniciar uma dramatização, se conectar ao endereço IP da máquina local (127.0.0.1) na porta 3000;
- As mensagens enviadas pelo gerenciador dos eventos da história devem ser formadas por strings no seguinte formato: {ID|Command|Parameter1|Parameter2|Subtitle}, onde:
 - **ID**: É um código sequencial e único identificando a mensagem;
 - **Command**: String referente ao comando/evento que deve ocorrer na dramatização. Este comando deve fazer parte do conjunto de comandos reconhecido pelo roteirista {reduce_protection, go, attack, kidnap, donate, bewitch, fight, kill, free, marry} para que a ação seja executada na dramatização;
 - **Parameter1**: Primeiro parâmetro utilizado pelo comando/evento. Cada comando possui uma sintaxe e parâmetros diferentes;
 - **Parameter2**: Segundo parâmetro utilizado pelo comando/evento;
 - **Subtitle**: String descrevendo a ação. Esta string será mostrada durante a execução da ação;
- Para solicitar a dramatização de uma ação, o gerenciador dos eventos da história (cliente) deve enviar a ação utilizando o formato descrito acima. Exemplo: {22|kidnap|Draco|Marian|Draco kidnaps Marian|};
- Após o envio de um comando para ser dramatizado, o gerenciador dos eventos da história (cliente) deve aguardar até que uma mensagem de confirmação de que a ação solicitada terminou de ser dramatizada. A mensagem de confirmação irá conter somente o ID do evento que foi dramatizado;
- Após o recebimento da confirmação do término de uma ação, o gerenciador dos eventos da história (cliente) pode solicitar a dramatização de novos eventos;
- Para solicitar que um mesmo evento seja dramatizado novamente, o gerenciador dos eventos da história (cliente) deve enviar novamente o mesmo comando assegurando que o ID do comando seja o mesmo utilizado da primeira vez;
- Para reiniciar o processo de dramatização e restaurar todos os atributos do sistema de dramatização, o gerenciador dos eventos da história (cliente) deve enviar uma

mensagem com o comando *restart*, com o ID -1 e os demais parâmetros como “null”.

Exemplo: {-1|restart|null|null|null|}.

Os possíveis comandos interpretados pelo agente Roteirista são descritos na Tabela 1.

Tabela 1 - Comandos interpretados pelo agente Roteirista.

Comando	Descrição	Exemplo
go(CH, PL)	O personagem CH vai para o local PL.	1 go Marian White_Palace Marian goes to the White Palace
reduce_protection(CH, PL)	A proteção do local PL é reduzida pelo personagem CH.	1 reduce_protection Marian White_Palace Marian dismisses guards from the White Palace
kidnap(CH1, CH2)	O personagem CH1 rapta o personagem CH2.	1 kidnap Draco Marian Draco kidnaps Marian
attack(CH, PL)	O personagem CH ataca o local PL.	1 attack Brian Red_Castle Brian attacks the Red Castle
fight(CH1, CH2)	O personagem CH1 luta contra o personagem CH2.	1 fight Brian Draco Brian fights against Draco
kill(CH1, CH2)	O personagem CH1 mata o personagem CH2.	1 kill Brian Draco Brian kills Draco
free(CH1, CH2)	O personagem CH1 liberta o personagem CH2.	1 free Brian Marian Brian frees Marian
marry(CH1, CH2)	O personagem CH1 casa com o personagem CH2.	1 marry Brian Marian Brian and Marian get married
donate(CH1, CH2)	O personagem CH1 doa poderes para o personagem CH2.	1 donate Turjan Brian Turjan gives strength to Brian
bewitch (CH1, CH2)	O personagem CH1 enfeitiça o personagem CH2.	1 bewitch Turjan Brian Turjan bewitches Brian

Além de ser responsável por fazer a comunicação entre o módulo de dramatização e o de geração de histórias, o Roteirista é também responsável por expandir os eventos da história criando novas ações com base nos eventos de alto nível. Isso é necessário, pois o planejador utilizado para a geração das histórias (IPG) trabalha somente com eventos de alto nível, acrescentar eventos de mais baixo nível poderia tornar o processo de planejamento inviável. Entretanto, dramatizar somente os eventos de alto nível resultaria em uma história muito simplificada e inviabilizaria a utilização da maioria dos conceitos cinematográficos. Por este motivo, o agente Roteirista tem também o objetivo de incrementar os eventos da história, acrescentando ações mais detalhadas, diálogos e até mesmo outros personagens.

No modelo de expansão de histórias adotado pelo Roteirista, cada ação de alto nível possui um conjunto de micro-ações que são adicionadas à história sempre que o evento de alto

nível for enviado para a dramatização. Por exemplo, para o evento de alto nível “reduce_protection(Marian, White_Palace)” (onde o personagem Marian deve reduzir a proteção do local White_Palace), o agente Roteirista irá expandir este evento com o seguinte conjunto de micro-ações:

- CreateActor(White_PalaceGuarda, White_Palace, Guarda1);
- CreateActor(White_PalaceGuarda, White_Palace, Guarda2);
- Talk(Marian, [Guarda1, Guarda2], “Guards, get out! I want to be alone...”);
- Talk(Guarda1, [Marian, Guarda2], “Yes my lady!”);
- LookToSmooth([Guarda1, Guarda2], White_PalaceAway);
- Go([Guarda1, Guarda2], White_PalaceAway);
- Talk(Marian, [], “Finally alone...”);
- DestroyActor([Guarda1, Guarda2]);

A descrição de cada uma das micro-ações utilizadas para expandir os eventos de alto nível é apresentada na Tabela 2. Apesar de ser uma abordagem simplificada de expansão das histórias, pode ser considerado um método funcional. Mesmo se tratando de um conjunto pré-definido de micro-ações para cada evento, a diversidade das histórias não é afetada, pois o próprio gerador de histórias vai garantir que novas sequências de eventos de alto nível sejam geradas. Além disso, todas as micro-ações são disponibilizadas para as camadas superiores de geração de histórias. Desta maneira as micro-ações podem ser utilizadas pelo próprio planejador quando este suportar o planejamento de histórias mais detalhadas.

Após expandir um evento de alto nível, o Roteirista cria uma lista das ações que deverão ser dramatizadas para aquele evento. Fazendo uma analogia com a cinematografia, esta lista de ações pode ser considerada uma “cena”. Esta cena é composta por diversas ações que vão ocorrer em um determinado local e deverão ser filmadas por diversas tomadas. Por último, o Roteirista envia a cena criada para o agente Diretor.

A função do agente Roteirista pode ser comparada com a de um roteirista de filmes. Ambos são responsáveis por escrever o roteiro/história da narrativa. É claro que para o agente o ato de “criar” uma história envolve todo o processo de planejamento e interação com os usuários, mas isso é transparente para o sistema de dramatização.

5.2.2 Diretor

Na criação de um filme, o diretor criativamente traduz as palavras escritas no roteiro em imagens. Ele visualiza o roteiro e dá aos conceitos abstratos uma forma concreta. O mesmo é feito pelo agente Diretor no sistema de dramatização. Basicamente, o Diretor deve interpretar as ações descritas nas cenas geradas pelo Roteirista de modo a coordenar a dramatização destas ações, seja passando tarefas para os atores ou então executando intervenções diretas no ambiente.

O agente Diretor é constituído de um conjunto de instruções específicas para cada tipo de ação que pode ser dramatizada. Por exemplo, ao interpretar uma ação *Kill* (onde um personagem deve matar outro), o Diretor irá enviar para o ator executor da ação um comando informando que ele deve executar a animação de ataque e para o ator vítima da ação um comando informando que ele deve executar a animação de morte. Os possíveis comandos interpretados pelo Diretor são descritos na Tabela 2. É importante ressaltar que os parâmetros identificados com “[n]” após o seu nome são vetores. Estes permitem a utilização de um conjunto de objetos como parâmetro, sendo possível assim que vários personagens executem uma mesma ação ao mesmo tempo.

Tabela 2 - Micro-ações interpretadas pelo agente Diretor.

Comando	Descrição
CreateActor(CH, PL, NM)	Cria um novo personagem CH no local PL com o nome N.
Talk(CH1, CH2[n], MSG)	O personagem CH1 diz a mensagem MSG para o personagem CH2.
NarratorSay(MSG)	O narrador da história diz a mensagem MSG.
LookTo (CH[n], PL)	O personagem CH olha para o local ou personagem PL.
LookToSmooth (CH[n], PL)	O personagem CH olha lentamente para o local PL.
Go(CH[n], PL)	O personagem CH vai para o local PL.
DestroyActor(CH[n])	Destrói o personagem CH.
Fight(CH1[n], CH2)	O personagem CH1 luta contra o personagem CH2.
Die(CH[n])	O personagem CH morre.
Hit(CH1[n], CH2)	O personagem CH1 bate no personagem CH2.
Faint(CH[n])	O personagem CH desmaia.
Teleport(CH[n], PL)	O personagem CH é teletransportado para o local PL.
Take(CH1, CH2)	O personagem CH1 segura o personagem CH2.
Magic(CH1[n], CH2)	O personagem CH1 executa uma magia no personagem CH2.
LookAround(CH[])	O personagem CH olha para os lados.
Kiss(CH1, CH2)	O personagem CH1 beija o personagem CH2.

Em algumas micro-ações o agente Diretor também é responsável por assegurar que o posicionamento dos atores está adequado para a cena. Por exemplo, em uma micro-ação Talk de uma cena de diálogo envolvendo dois atores, o Diretor irá, antes de enviar o comando Talk para os atores, verificar se a distância entre os dois atores não é muito grande ou muito pequena. Caso a distancia esteja fora dos limites que permitam um diálogo, o agente Diretor reposiciona os atores de forma a permitir que o diálogo seja realizado de forma consistente.

Além de interpretar os eventos da dramatização e garantir que o posicionamento dos atores seja consistente, o Diretor também é responsável por enviar uma mensagem para o agente Editor, Diretor de Fotografia e Diretor de Música informando sobre as ações e cenas que serão dramatizadas, para que estes possam executar as suas tarefas.

Após a dramatização de uma cena, o Diretor é responsável por solicitar ao Roteirista uma nova cena para ser dramatizada.

5.2.3 Cameraman

O agente Cameraman, assim como no cinema, é responsável pelo posicionamento e a manipulação direta das câmeras. A principal tarefa do Cameraman é, para uma dada cena, propor as possíveis tomadas para filmá-la.

A primeira etapa do processo de definição das possíveis tomadas para uma cena é a criação de uma linha de ação. Como detalhado no capítulo 2, esta linha de ação consiste em uma linha imaginária que conecta e direciona o foco de atenção da cena. Esta linha é de grande importância para a aplicação de algumas regras para se manter a continuidade na cena.

Para se definir uma linha de ação no ambiente 3D, utiliza-se a equação de um plano:

$$Ax + By + Cz + D = 0$$

Por exemplo, dada uma cena composta por um ator em uma posição p andando em uma direção d (Figura 19 - a), e considerando o vetor up $u = (0, 1, 0)$, o vetor normal da linha de ação pode ser definido como:

$$n = \frac{u \times d}{\|u \times d\|}$$

A partir de n e o ponto p é possível determinar D :

$$D = -n \cdot p$$

Aplicando estes valores à equação do plano, é possível verificar que um ponto (x, y, z) pertence ao plano se:

$$n_x x + n_y y + n_z z + D = 0 \quad (1)$$

Da mesma maneira, é possível determinar a qual dos lados da linha da ação um dado ponto pertence. Para isso, basta analisar o resultado da equação (1), caso $n_x x + n_y y + n_z z + D > 0$ significa que o ponto pertence a um dos lados do plano, caso $n_x x + n_y y + n_z z + D < 0$ o ponto pertence ao outro lado do plano.

O mesmo pode ser feito para cenas contendo mais atores. Supondo uma cena composta por dois atores (Figura 19 - b), a mesma metodologia adotada para o exemplo anterior pode ser aplicada, entretanto, em vez do vetor direção d ser determinado pela direção para onde o ator estava andando, deve-se considerar o vetor normalizado entre os dois atores:

$$d = \frac{p_1 - p_2}{\|p_1 - p_2\|}$$

O mesmo deve ser feito para cenas compostas por 3 ou mais atores, entretanto deve-se considerar os dois atores de maior importância na cena, para então a linha de ação ser estabelecida entre eles.

É importante mencionar que a linha de ação não é estabelecida somente entre atores; qualquer objeto importante para a cena pode ser utilizado para definir a linha de ação.

Após a definição de uma linha de ação é possível posicionar as câmeras na cena. A escolha de apenas um dos lados da linha de ação irá garantir que a regra dos 180 graus não seja quebrada. Como foi detalhado no capítulo 2, manter as tomadas de uma mesma cena em apenas um dos lados da linha de ação é uma regra importante para se manter a continuidade entre tomadas.

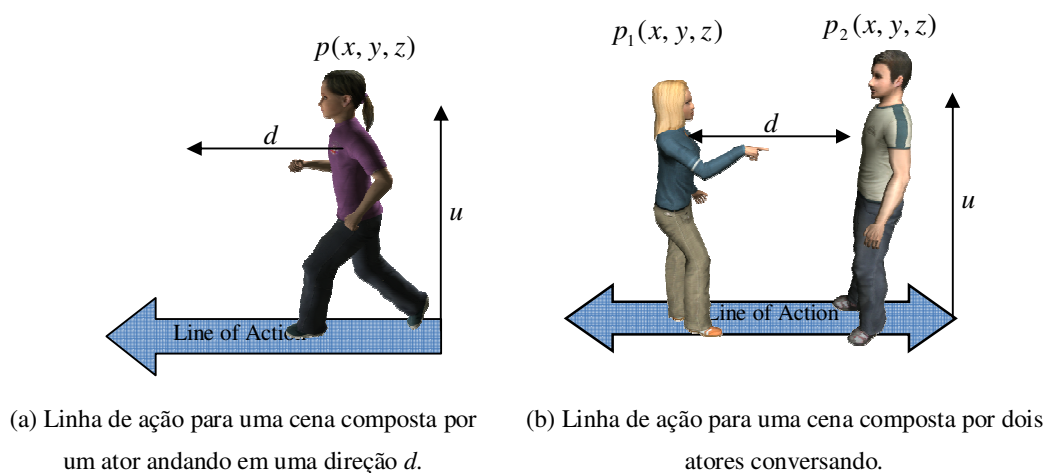


Figura 19 - Exemplos de linhas de ação.

Para o posicionamento das câmeras nas cenas, adota-se uma abordagem baseada nos padrões de posicionamento de câmeras propostos pela cinematografia. Arijon (1976) e Katz (1991) descrevem diversos destes padrões. Para cada tipo de cena, existem alguns modelos de posicionamento de câmeras comumente adotados. Por exemplo, para cenas de diálogos é comum a utilização de uma formação conhecida como *triangle system*. Nesta formação, as câmeras são posicionadas na cena formando um triângulo entre os atores. Um exemplo deste padrão pode ser visto na Figura 20.

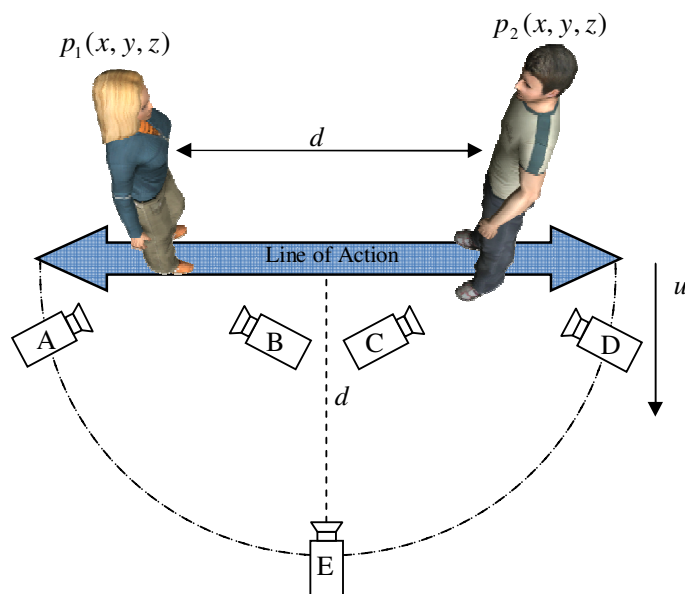


Figura 20 - Padrão de posicionamento de câmeras para a filmagem de uma cena de diálogo composta por dois personagens.

Para se determinar a posição exata das câmeras existe um cálculo específico para cada uma delas, por exemplo, para se determinar a posição da câmera E, primeiramente, calcula-se o ponto central da cena. Considerando os atores $p_1(x, y, z)$ e $p_2(x, y, z)$ o ponto central da cena c é dado por:

$$c = (p_1 + p_2)/2$$

Em seguida calcula-se a distância d entre os atores:

$$d = \|p_1 - p_2\|$$

Considerado o vetor perpendicular dos atores $u = (0, 0, 1)$, a posição da câmera E é dada por:

$$g = c + (u \times d)$$

Para alterar o lado da linha de ação em que a câmera está posicionada basta alterar o sinal do vetor perpendicular u para $-u$.

A mesma metodologia de posicionamento pode ser aplicada para cenas com 3 atores. A única diferença é que a escolha do lado da linha de ação em que as câmeras serão posicionadas deve ser o lado oposto onde o terceiro ator se encontra. Assume-se que a linha de ação será traçada entre os dois atores de maior importância para a cena. Considerando que o terceiro ator é o de menor importância, dificilmente uma tomada mais detalhada deste ator será necessária.

Em cenas mais complexas (envolvendo quatro ou mais atores), os atores são divididos em grupos. Para cada um dos grupos traça-se uma linha de ação e os padrões de posicionamento de câmeras para dois ou três atores são utilizados. Também é incluída uma câmera que possa fazer uma tomada de todos os grupos.

A mesma abordagem é utilizada em cenas de ação. Entretanto, em algumas destas cenas a câmera também incorpora movimentos. Como, por exemplo, em uma cena onde um ator caminha para uma determinada direção (Figura 21). Para este tipo de cena, é comum a utilização de duas câmeras. A tomada realizada com a câmera A é utilizada para mostrar de onde o ator está vindo, enquanto a câmera B enfatiza para onde o ator está indo. Para este

exemplo, as câmeras executam o movimento conhecido na cinematografia como *dolly* para acompanhar o ator enquanto este executa a caminhada.

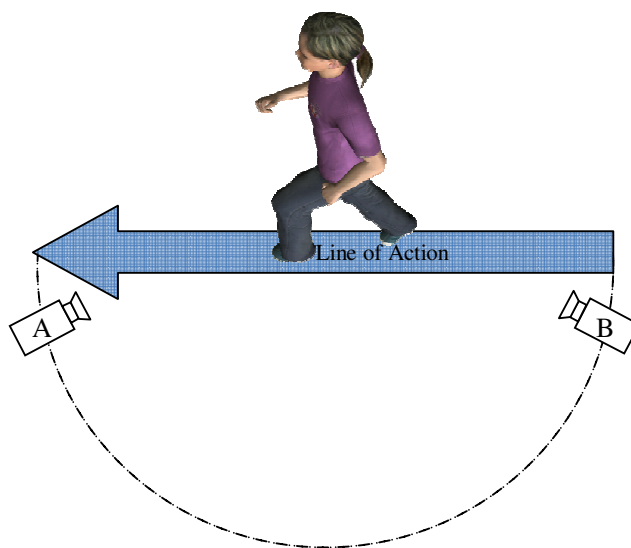


Figura 21 - Padrão de posicionamento de câmeras para a filmagem de uma cena composta por um personagem caminhando em determinada direção.

Em cenas que possuem uma configuração onde as câmeras acompanham os movimentos dos atores (ou qualquer outro objeto) é necessário recalcular as posições das câmeras a cada frame. Entretanto, em caminhadas longas os atores normalmente mudam de direção várias vezes e, teoricamente, a linha de ação também estaria mudando. Porém, se o posicionamento das câmeras fosse recalculado a cada frame, mesmo quando os personagens estivessem mudando a sua direção, a câmera faria o mesmo movimento de rotação executado pelo ator e a sensação que se teria é de que a câmera estaria de certa forma presa ao ator. Para solucionar este problema, em cenas de ação o cálculo de uma nova linha de ação só é feito quando ocorre um corte entre tomadas, desta maneira a câmera acompanha os personagens, mas respeitando a linha de ação que foi estabelecida no início da tomada atual.

Em outros tipos de cenas de ação a câmera permanece parada e executa movimentos de *Pan* e *Tilt* para acompanhar a movimentação dos atores da cena. Como, por exemplo, uma cena onde um dragão inicia um vôo na vertical (Figura 22). Para este tipo de cena, é comum a utilização de somente uma tomada para filmar a cena. Neste caso, a câmera deve acompanhar os movimentos do dragão executando um movimento conhecido na cinematografia como *Tilt*.

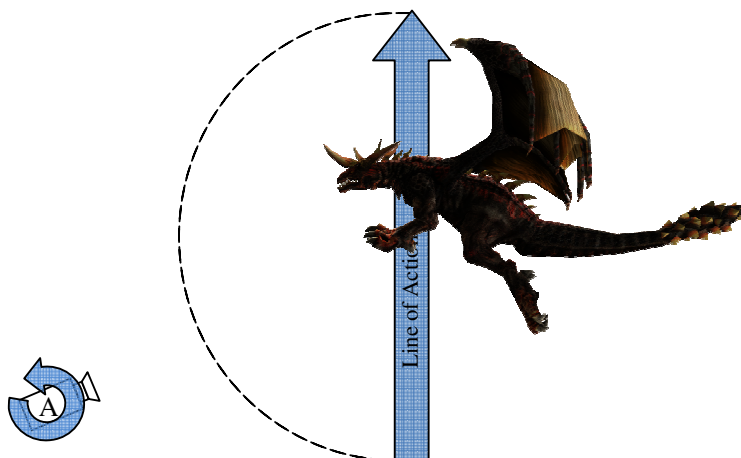


Figura 22 - Padrão de posicionamento de câmeras para a filmagem de uma cena composta por um personagem que inicia um voo na vertical.

Esta metodologia de padrões de posicionamento de câmeras é utilizada para todas as cenas; cada tipo de cena possui uma configuração específica. Para alguns tipos de cenas as câmeras são estáticas, para outras as câmeras acompanham os atores, e em outras elas permanecem paradas, mas mantêm o foco nos atores que estiverem se movendo. O agente Cameraman incorpora este conjunto de configurações de câmeras para os possíveis tipos de cenas que podem ocorrer na dramatização. Estes padrões de câmeras foram baseados nas configurações apresentadas por Arijon (1976) e Katz (1991). A lista completa de configurações de câmeras implementadas no agente Cameraman é mostrada na Tabela 3.

É importante ressaltar que as posições das câmeras propostas pelo agente Cameraman são apenas “possíveis” tomadas que poderão ser utilizadas para filmar as cenas. A escolha de qual destas tomadas é a melhor para filmar a cena em um determinado momento é tarefa do agente Editor. Quando uma tomada é escolhida, o agente Cameraman utiliza o posicionamento sugerido para posicionar a câmera escolhida e então filmar a cena.

Tabela 3 - Lista de configurações de câmeras implementadas no agente Cameraman.

Câmera	Descrição
Talk(CH)	Configuração de câmeras para cenas de diálogos envolvendo um personagem CH.
Talk(CH1, CH2)	Configuração de câmeras para cenas de diálogos entre dois personagens, CH1 e CH2.
Talk(CH1, CH2, CH3)	Configuração de câmeras para cenas de diálogos envolvendo três personagens, CH1, CH2 e CH3.
Go(CH)	Configuração de câmeras para cenas caminhadas envolvendo um personagem CH.
Go(CH1, CH2)	Configuração de câmeras para cenas caminhadas envolvendo dois personagens, CH1 e CH2.
Fight(CH1, CH2)	Configuração de câmeras para cenas de luta entre o personagem CH1 e o personagem CH2.
Kiss(CH1, CH2)	Configuração de câmeras para cenas de beijo entre o personagem CH1 e o personagem CH2.
Die(CH)	Configuração de câmeras para cenas de morte de um personagem CH.
Look(CH, OB)	Configuração de câmeras para cenas onde um personagem CH olha para um objeto/personagens/local OB.
StartFly(CH)	Configuração de câmeras para cenas de início de vôo envolvendo um personagem CH.
Fly(CH)	Configuração de câmeras para cenas de vôo envolvendo um personagem CH.
EndFly(CH)	Configuração de câmeras para cenas de fim de vôo envolvendo um personagem CH.

Além de calcular o posicionamento das câmeras, o agente Cameraman também é responsável por garantir a consistência das tomadas, de modo que os atores estejam devidamente enquadrados pelas câmeras e não estejam oclusos por outros objetos. O enquadramento correto das cenas é garantido pelo próprio cálculo do posicionamento das câmeras que leva em considerações as variáveis geométricas das cenas e dos atores. Para

verificar a existência de oclusões nas tomadas, o agente Cameraman utiliza o algoritmo de *ray casting* para traçar raios na direção dos atores ou objetos de foco da cena; caso o raio não atinja o seu objeto alvo, significa que outro objeto está bloqueando a visibilidade deste.

A verificação de oclusões nas tomadas é realizada em dois momentos: primeiramente quando uma das tomadas sugeridas pelo Cameraman é selecionada pelo agente Editor, antes de ativar a tomada, o Cameraman realiza a verificação de oclusões. Caso algo esteja obstruindo a visibilidade da câmera, o Cameraman seleciona outra tomada que se aproxime mais da escolha feita pelo Editor e que continue mantendo o foco no mesmo ator da tomada selecionada. Caso a próxima tomada também esteja obstruída, verifica-se a próxima. Caso todas as tomadas estejam obstruídas, o Cameraman reposiciona todas as câmeras do lado oposto da linha de ação e seleciona a tomada mais próxima da que foi selecionada pelo Editor. Em último caso, se mesmo do outro lado da linha de ação todas as tomadas continuarem obstruídas, o Cameraman realiza uma tomada de cima dos atores da cena. O outro momento em que o agente verifica oclusões é durante a filmagem de cenas onde a câmera incorpora algum movimento. Neste caso, sempre que a câmera se move, é realizada a verificação de oclusões; caso a visibilidade da câmera seja bloqueada por algum objeto, o Cameraman força um corte selecionando a tomada que mais se aproxime da tomada atual. Adota-se a mesma abordagem do primeiro caso de verificação de oclusões, até que uma tomada que não tenha nenhuma oclusão seja selecionada, ou até que a última opção de filmagem da cena de cima seja adotada.

5.2.4 Editor

No processo de criação de um filme, o editor é responsável por unir, cortar e editar as tomadas feitas durante as gravações para montar a versão final do filme. No cinema o editor trabalha sobre o filme depois de ele ter sido filmado. No modelo de dramatização proposto, o agente Editor deve desempenhar um papel semelhante, entretanto tudo deve ser feito em tempo real. Os cortes entre as tomadas devem ser feitos durante as filmagens, pois não existe um processo de pós-produção; as cenas devem ser exibidas ao público ao mesmo tempo em que elas são dramatizadas.

O principal objetivo do agente Editor é escolher a melhor tomada para filmar uma cena em um determinado momento. Porém, diferente do agente Cameraman, que pode seguir um conjunto de padrões para posicionar as câmeras, a escolha da “melhor tomada” para filmar uma cena feita pelo Editor é algo bem subjetivo. Não existe na literatura nenhum padrão para ser seguido. Por este motivo, é proposto neste trabalho tratar este problema de escolha da “melhor tomada” como um problema de classificação estatística.

A classificação estatística é uma metodologia supervisionada de aprendizagem de máquina que busca classificar dados com base na proximidade entre características de dados anteriores. A utilização de uma abordagem supervisionada de aprendizado é bastante eficaz neste caso, pois é possível deixar a cargo do próprio classificador encontrar um padrão para a seleção da melhor tomada, com base em exemplos de escolhas feitas por um editor real em uma etapa de treinamento.

Existem diversos algoritmos de classificação, dentre os mais comuns pode-se citar: redes neurais artificiais, árvores de decisão, redes bayesianas, k-vizinhos mais próximos (KNN), support vector machines (SVM), entre outros. Dentre os classificadores mais comuns optou-se pela utilização de support vector machines. A escolha do SVM como método de classificação para este trabalho deu-se devido a diversas vantagens, apontadas por vários autores, que o SVM possui em relação aos demais algoritmos.

Segundo Byvatov *et al.* (2003), uma das principais vantagens do SVMs em relação a redes neurais artificial é a não influência de todo o conjunto de dados de treinamento na função de classificação; esta depende apenas dos vetores de suporte. Outra vantagem está relacionada à utilização de funções de kernel para a criação da função de classificação. Isto permite à SVM lidar com grandes conjuntos de características e dados. Segundo Gunn (1998), SVM possui uma melhor generalização que redes neurais artificiais e garante soluções globais e locais ótimas similares às obtidas através de redes neurais artificiais. Hmeidia (2008), aponta também vantagens do SVM em relação ao algoritmo de classificação KNN em questões de desempenho e precisão na classificação dos dados. Nos últimos anos o uso de SVM tem se demonstrado eficaz em diversas aplicações do mundo real, como na detecção de microcalcificações em imagens médicas (El-Naqa *et al.*, 2002), para a categorização hierárquica de documentos (Cai e Hofmann, 2004), para categorização de spam (Drucker *et al.* 1999), entre outros.

Para a utilização do classificador SVMs para representar o conhecimento do agente Editor, primeiramente é necessário definir a estrutura da SVM, as características e classes

utilizadas. A escolha da tomada mais adequada para filmar uma cena normalmente está ligada diretamente com a tentativa de se enfatizar a emoção dos atores. Desta maneira a escolha das características deve estar de certo modo ligada a estas emoções. A rede de emoções e relações apresentada na sessão 5.1.1 pode fornecer estas informações para o agente Editor. Entretanto uma cena pode ser composta por uma quantidade diferente de atores, e conseqüentemente por uma quantidade diferente de emoções ligadas a cada um dos atores. Além disso, o número de possíveis tomadas para filmar uma cena varia conforme o tipo de cena. Isso torna impossível a definição de uma única SVM para atender a todas as cenas, pois a estrutura de uma SVM deve possuir um número fixo de características e classes (possíveis resultados). Para resolver este problema adotou-se uma estrutura composta por um conjunto de SVMs, cada uma delas treinada para classificar as melhores tomadas em um determinado tipo de cena.

As características utilizadas pelas SVMs do agente Editor são:

- Nível de intensidade emocional de felicidade/tristeza dos atores envolvidos na cena
 $e_i^1(t) = happiness_i(t)$;
- Nível de intensidade emocional de coragem/medo dos atores envolvidos na cena
 $e_i^2(t) = courage_i(t)$;
- O principal ator atuante da cena (representado através de um valor numérico único associado a cada um dos atores da cena).

As classes são as possíveis tomadas propostas pelo agente Cameraman para filmar uma cena baseando-se no tipo de cena atual. O número de características assim como o número de classes irá variar para cada uma das SVMs. Por exemplo, uma cena de diálogo entre dois atores irá conter 5 características e 5 classes. As 5 características são: 2 níveis de intensidade de felicidade (um para cada ator), 2 níveis de intensidade de coragem (um para cada ator) e 1 característica representando o ator atuante. O número de classes são as 5 possíveis tomadas para filmar a cena. Um exemplo destas tomadas pode ser visto na Figura 30 apresentada anteriormente.

Com a definição da estrutura das SVMs, o próximo passo é o treinamento. O objetivo desta etapa é dar à SVM exemplos de situações reais e demonstrar qual seria a resposta mais adequada para aquela situação. Baseado nestes exemplos, a SVM irá criar hiperplanos separando as características dos exemplos de modo a definir um padrão que classifique as classes.

Para efetuar o treinamento das SVMs foram feitas diversas simulações de cenas e durante as simulações, foi feita a escolha da melhor tomada para filmar as cenas a cada

momento. Para realizar estas escolhas, foi levado em consideração o contexto emocional dos atores e as ações que estavam acontecendo nas cenas. Para cada seleção de tomada, as características atuais dos atores que são utilizadas pela SVM foram armazenadas no formato de um vetor em um arquivo juntamente com um rótulo representando a tomada selecionada. Cada uma destas seleções representa um exemplo de treinamento. O formato utilizado para armazenar o conjunto de treinamento pode ser descrito como:

$$\{\text{ClassLabel } 1:\text{Feature } 2:\text{Feature } 3:\text{Feature} \dots N:\text{Feature}\}$$

onde:

- **ClassLabel:** Representa um rótulo que será atribuído para a classe de modo a identificá-la. Todos os exemplos que pertencerem à mesma classe devem ter o mesmo rótulo;
- **Feature:** Representa o valor numérico da característica. O número que antecede a característica representa um identificador único e incremental da característica (1, 2, 3... N).

Considerando uma cena de diálogo entre dois personagens, um exemplo de treinamento pode ser representado como:

$$\{2 \ 1:0.2 \ 2:0.1 \ 3:0.7 \ 4:0.1 \ 5:2\}$$

onde, o valor 2 inicial descreve a classe do exemplo, no caso o número 2 é usado para identificar a câmera 2. A primeira característica 1:0.2 representa o nível de felicidade do primeiro ator, 2:0.1 a coragem do primeiro ator, 3:0.7 a felicidade do segundo ator e assim por diante. Uma base de treinamento é formada por um conjunto de exemplos de treinamento.

O processo de treinamento deve ser executado somente uma vez. Após o treinamento as SVMs se tornam aptas a classificar as melhores tomadas em situações futuras. Entretanto, é importante ressaltar que o treinamento deve ser feito de modo correto; caso tomadas inadequadas sejam selecionadas durante a etapa de treinamento é provável que a classificação de tomadas futuras siga um padrão incorreto.

Após o treinamento de todas as SVMs (uma para cada tipo de cena) o agente Editor estará apto a agir como um editor real e selecionar em tempo real as melhores tomadas para

qualquer tipo de cena durante a dramatização. A arquitetura do agente Editor é apresentada na Figura 23. A lista de todas SVMs criadas e treinadas para o agente Editor pode ser vista na Tabela 4.

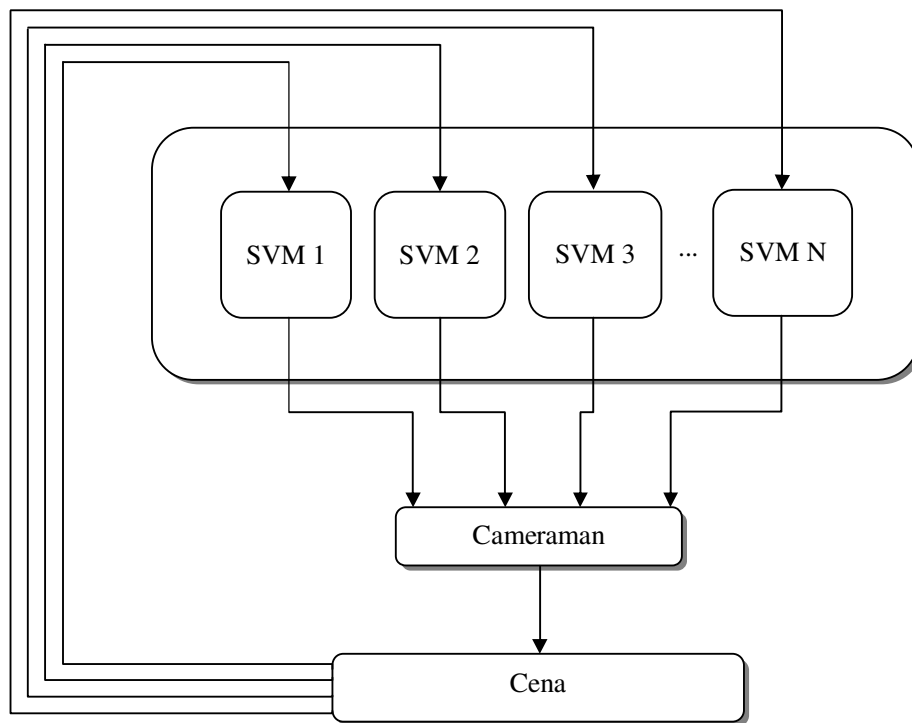


Figura 23 - Arquitetura do agente Editor.

Durante o processo de dramatização, o agente Editor deve selecionar, a cada momento, a melhor tomada para filmar a cena atual. Para isso, no início de qualquer cena o agente Editor recebe do Diretor uma mensagem informando sobre o tipo de cena que será dramatizada. Com base no tipo de cena, o Editor seleciona e ativa a SVM correspondente. Ao mesmo tempo envia uma mensagem para o agente Cameraman informando sobre o tipo de cena que deverá ser filmada, de modo que o Cameraman possa calcular o posicionamento de todas as possíveis tomadas para filmar a cena.

Tabela 4 - Lista de SVMs criadas e treinadas para o agente Editor.

SVM	Descrição
Talk(CH)	SVM treinada para classificar as melhores tomadas para cenas de diálogos envolvendo um personagem CH.
Talk(CH1, CH2)	SVM treinada para classificar as melhores tomadas para cenas de diálogos entre dois personagens, CH1 e CH2.
Talk(CH1, CH2, CH3)	SVM treinada para classificar as melhores tomadas para cenas de diálogos envolvendo três personagens, CH1, CH2 e CH3.
Go(CH)	SVM treinada para classificar as melhores tomadas para cenas de caminhadas envolvendo um personagem CH.
Go(CH1, CH2)	SVM treinada para classificar as melhores tomadas para cenas de caminhadas envolvendo dois personagens, CH1 e CH2.
Fight(CH1, CH2)	SVM treinada para classificar as melhores tomadas para cenas de luta entre o personagem CH1 e o personagem CH2.
Fly(CH)	SVM treinada para classificar as melhores tomadas para cenas de vôo envolvendo um personagem CH.

A seleção de novas tomadas para uma cena é desencadeada pela ocorrência de eventos durante a dramatização. Sempre que um novo evento ocorre, o agente Diretor informa sobre a sua ocorrência para o Editor. Ao receber esta notificação de que um novo evento ocorrerá o agente Editor extrai da rede de emoções e dos atores as características utilizadas pela SVM que foi ativada no início da cena. Em seguida, estas características são aplicadas a SVM de modo que elas possam ser classificadas dentre uma das tomadas disponíveis para o tipo de cena atual. As características servem de entrada para a SVM, a saída é um rótulo identificando a tomada mais adequada para filmar a cena (Figura 24).

Após a classificação das características dentre as possíveis tomadas para filmar a cena, o agente Editor envia uma mensagem para o Cameraman informando sobre a sua escolha de uma nova tomada, identificando a tomada escolhida. Em seguida, o Cameraman interpreta a mensagem recebida e ativa a câmera correspondente.

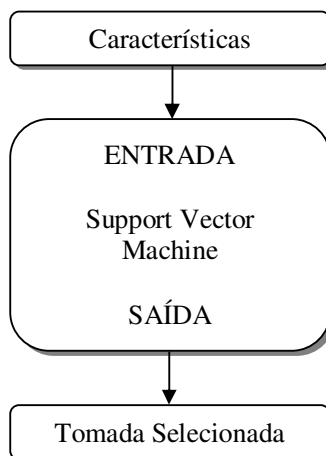


Figura 24 - Modelo de entrada, processamento e saída de dados das SVMs do agente Editor.

Considerando uma cena de um diálogo entre três personagens (Figura 25). O primeiro passo para a escolha da tomada mais adequada é a extração das características. Para isso, o agente Editor acessa a rede de relações e emoções para extrair o valor de felicidade e coragem dos atores envolvidos na cena. O ator atuante é extraído do próprio evento que desencadeou a seleção da tomada. Como resultado, tem-se o seguinte conjunto de características: {1:-0.2 2:-0.5 3:0.3 4:-0.1 5:0.7 6:0.8 7:3}

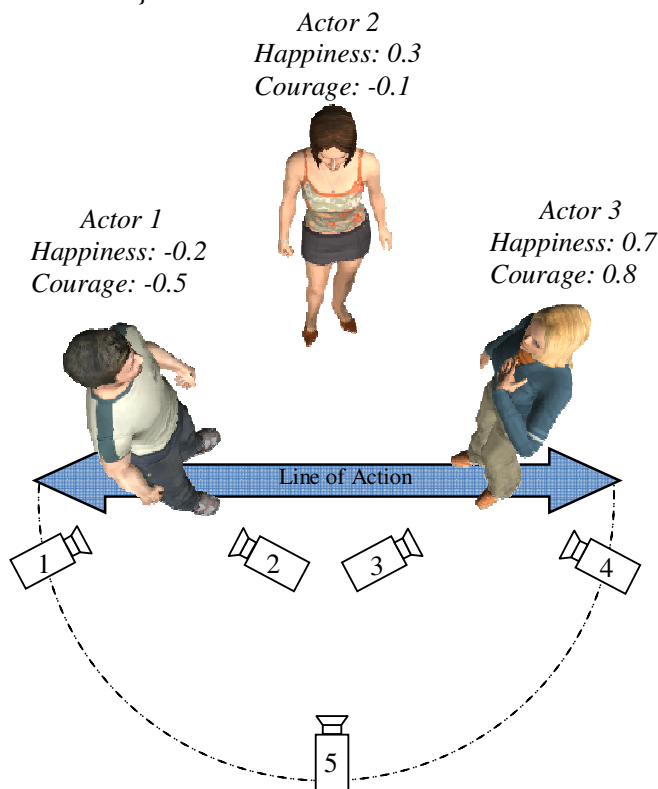


Figura 25 - Configuração de câmeras para a filmagem de uma cena de diálogo composta por três personagens.

Aplicando o conjunto de características na SVM, tem-se como resultado a classe na qual o conjunto de entrada foi classificado, ou seja, um rótulo representando a câmera que deve ser ativada. Para este exemplo, o resultado da SVM foi 3, ou seja, a câmera de rótulo 3 deve ser ativada para filmar a cena. Todas as possíveis tomadas para este exemplos podem ser vistas na Figura 26.



Câmera 1



Câmera 2



Câmera 3



Câmera 4



Câmera 5

Figura 26 - Possíveis tomadas para a filmagem de uma cena de diálogo composta por três personagens.

5.2.5 Diretor de fotografia

Durante o processo de criação de um filme, o diretor de fotografia tem a tarefa de interpretar a visão do diretor, extrair as emoções que cada cena deve expressar e então enfatizar visualmente essas emoções. Para esta tarefa, o diretor de fotografia utiliza diversos truques visuais que buscam afetar psicologicamente a maneira como os espectadores interpretam as cenas.

O processo de escolha do melhor efeito visual para enfatizar a emoção das cenas, assim como a seleção da melhor tomada para filmar uma cena, pode ser visto como um problema de classificação. É necessário um conjunto fixo de possíveis efeitos visuais (classes resultantes) e a escolha de qual efeito visual deve ser utilizado é resultante de algum padrão relativo às emoções dos atores (características de entrada). Deste modo o agente Diretor de Fotografia também faz uso de uma SVM treinada para selecionar o melhor efeito visual para enfatizar visualmente as emoções das cenas.

Para definir a estrutura da SVM utilizada pelo agente Diretor de Fotografia, primeiramente é necessário definir todas as possíveis classes nas quais as emoções dos atores poderão ser classificadas. Para isso, foram definidos os perfis de cenas apresentados na Tabela 5. Estes perfis de cena caracterizam um determinado tipo de emoção e identificam os efeitos visuais necessários para expressá-la. A escolha desses perfis emocionais foi baseada no círculo de emoções apresentado por Plutchik (2001), no qual o autor identifica 8 emoções básicas. Estas emoções foram utilizadas como base para a seleção dos 5 perfis de cena que foram considerados mais adequados para as cenas do cenário de testes do Logtell. A escolha dos efeitos visuais para os perfis emocionais foi feita com base nos estudos sobre iluminação, cores e efeitos visuais apresentados na sessão 3.6 e 3.7.

As características utilizadas pela SVM do agente Diretor de Fotografia, assim como as utilizadas pelas SVMs do agente Editor, têm relação com as emoções dos atores envolvidos nas cenas. Entretanto, o Diretor de Fotografia busca enfatizar a emoção da cena como um todo, e não somente de um determinado ator. As características escolhidas para serem usadas na SVM do Diretor de Fotografia são:

- Nível de intensidade emocional de felicidade/tristeza dos atores envolvidos na cena;

- Nível de intensidade emocional de coragem/medo dos atores envolvidos na cena;
- Nível de intensidade das relações (afeição) entre os atores envolvidos na cena;
- Nível de intensidade do clima do ambiente.

Tabela 5 - Perfis de efeitos visuais do agente Diretor de Fotografia.

Emoção	Efeito Visual
Cena Feliz	Luz brilhante na cor branca e com alta intensidade; Neblina distante; Céu ensolarado.
Cena Triste	Luz suave na cor cinza e com intensidade média; Neblina a uma distância média; Céu nublado.
Cena de Medo	Luz escura na cor cinza escuro e com baixa intensidade; Efeito <i>film grain</i> nas áreas mais escuras; Neblina densa; Céu noturno.
Cena de Raiva	Luz suave na cor vermelha e com intensidade média; Efeito suave de <i>motion blur</i> ; Céu de pôr-do-sol.
Cena de Tensão	Luz suave na cor vermelha escura e com baixa intensidade. Céu noturno.

A rede de emoções fornece os atributos de felicidade/tristeza, coragem/medo e afeição para cada um dos atores, entretanto o Diretor de Fotografia necessita do valor de emoção global da cena. Assim como o agente Editor, o Diretor de Fotografia também se depara com o problema de que cada cena pode ser composta por um número diferente de atores, porém, para este caso, a emoção de um único personagem pode não ter tanta influência no processo de classificação. Dessa maneira não é necessário um conjunto de SVMs como acontece no agente Editor. Uma única SVM é o suficiente para qualquer tipo de cena, porém isso torna necessária a utilização de algum cálculo para definir a emoção global das cenas.

Melo e Paiva (2005) propõem que o estado emocional de uma cena é definido pela média dos estados emocionais dos atores participante da cena. No entanto, esta abordagem pode não ser uma solução adequada. Supondo uma cena composta por dois atores com

estados emocionais opostos, a média será de um estado neutro, o que pode ser contraditório com a real emoção da cena.

Para determinar os níveis globais de emoção das cenas é proposto neste trabalho um método baseado na visão dos cineastas. A emoção da cena é estabelecida pela intensidade de emoção e a importância dos atores na dramatização. Em outras palavras, a emoção da cena é definida pelo ator protagonista com a maior intensidade emocional.

Considerando-se a_{hs} o conjunto de todos os valores de felicidade/tristeza que um ator a pode assumir ao longo de uma cena S , e $i \in [0...1]$ o fator de importância do ator na dramatização, e a função $maxintensity(set)$ que retorna o valor máximo absoluto (considerando valores positivos e negativos) de um conjunto de valores. A felicidade de um ator em uma cena S pode ser descrita como:

$$A_h(S) = maxintensity(a_{hs}) * i$$

Considerando uma cena composta por um vetor de atores Sa . A felicidade/tristeza da cena pode ser descrita como:

$$S_h(Sa) = maxintensity(A_h(Sa_h)) \quad (2)$$

A mesma formulação pode ser aplicada para avaliar a coragem/medo de uma cena. Considerando a_{cf} como o conjunto de todos os valores de coragem/medo ao longo de uma cena S . A coragem/medo de um ator pode ser descrita como:

$$A_c(S) = maxintensity(a_{cf}) * i$$

e a coragem/medo da cena como:

$$S_c(Sa) = maxintensity(A_c(Sa_c)) \quad (3)$$

Para determinar a afetividade da cena considera-se a média de afeto entre os atores envolvidos na cena. Considerando a_{af} como o conjunto de todos os valores de afetividade de um ator a_1 para um ator a_2 ao longo de uma cena S , a afetividade dos atores pode ser descrita como:

$$A_a(S) = \frac{1}{n} \sum_{i=0}^{n-1} affectivity(a_{cf})$$

e a afeição da cena como:

$$S_a(S) = \frac{1}{n_a} \sum_{i=0}^{n_a-1} affectivity(A_a(Sa_{cf})) \quad (4)$$

A última característica utilizada pela SVM do agente Diretor de Fotografia é o clima do ambiente. Esta característica é extraída diretamente do local onde a cena esta acontecendo. O cenário da dramatização é composto por 4 regiões (Figura 27). Cada região está dividida em várias sub-regiões (identificadas pelos quadrados claros na Figura 27) e cada uma possui um valor de clima emocional (entre -1 e +1). Valores próximos a +1 representam locais felizes e valores próximos a -1 representam locais assustadores.



Figura 27 - Mapa do ambiente 3D da dramatização do cenário de testes do Logtell.

Para realizar o treinamento da SVM foram simuladas diversas cenas e os perfis de cenas mais adequados foram selecionados baseando-se na emoção que a cena simulada deveria expressar. As características, juntamente com o perfil de efeitos visuais selecionado foram salvos em uma base de treinamento. O formato da base de treinamento é o mesmo utilizado pelo agente Editor. Esta base de treinamento é criada uma única vez e utilizada em todas as dramatizações futuras. Com a SVM treinada, o Diretor de Fotografia é capaz de atuar como um diretor de fotografia real e, com base na sua experiência anterior, selecionar em tempo real os melhores efeitos visuais para realçar as emoções das cenas.

Durante o processo de dramatização, o agente Diretor de Fotografia deve selecionar, no início de cada cena, o melhor perfil de efeito visual para enfatizar a emoção da cena. Para isso, o agente recebe do Diretor uma mensagem informando sobre a nova cena que será dramatizada. Em seguida, utiliza a rede de emoção e relação para simular os eventos da cena com objetivo de gerar o conjunto de todas as emoções e das relações para cada ator ao longo da cena. Com este conjunto de dados, é possível extrair as características e aplicá-las à SVM a fim de classificá-las dentre os possíveis perfis de cenas.

Antes do Diretor de fotografia aplicar o efeito visual no ambiente, é necessário definir a forma como o efeito será simulado, podendo ser aplicado instantaneamente ou gradativamente. Isto é necessário, pois podem existir transições entre cenas ocorrendo em um mesmo local no ambiente, de modo que a próxima cena começa a ocorrer no mesmo local onde a anterior termina. Caso um novo efeito visual fosse aplicado instantaneamente a continuidade das cenas seria quebrada. Para tais situações o agente Diretor de Fotografia opta por aplicar o efeito visual de forma gradativa, assim tem-se a mudança do clima visual de forma suave e de modo que o espectador não percebe com facilidade a mudança. A escolha de qual transição utilizar é feita com base na análise do local onde a cena anterior terminou e de onde o primeiro evento da próxima cena irá ocorrer. Por fim, o Diretor de Fotografia aplica o efeito visual selecionado no ambiente.

Por exemplo, supondo o evento de alto nível “marry(Brian, Marian)”, que após ser expandido pelo agente Roteirista, incorpora uma cena de pedido de casamento do cavaleiro Brian para a princesa Marian. Considerando que esta cena está ocorrendo após Brian ter salvado a princesa do vilão e os eventos anteriores tenham aumentado a afeição da princesa pelo cavaleiro. O conjunto de micro-ações desta cena é descrito na Tabela 6.

O conjunto de valores de variação da felicidade/tristeza dos atores envolvidos na cena da proposta de casamento pode ser visto na Figura 28. Este conjunto de dados é calculado através da simulação dos eventos da cena na rede de relações emoções.

Tabela 6 - Sequência de eventos para a cena do pedido de casamento.

Código	Evento
1	Go(Marian, White_PalaceVillage)
2	LookToSmooth(Brian, Marian)
3	Go(Brian, Marian)
4	Talk(Brian, Marian, "Good day, my fair lady.")
5	LookToSmooth(Marian, Brian)
6	Talk(Marian, Brian, "Hello, my hero.")
7	Talk(Brian, Marian, "How do you find yourself this day?")
8	Talk(Marian, Brian, "Fine, and you?")
9	Talk(Brian, Marian, "Fine...")
10	Talk(Brian, Marian, "If you will answer me a question...")
11	LookAround(Marian)
12	Talk(Brian, Marian, "Will you marry me?")
13	LookToSmooth(Marian, Brian)
14	Talk(Marian, Brian, "Yes!")
15	Kiss(Brian, Marian)

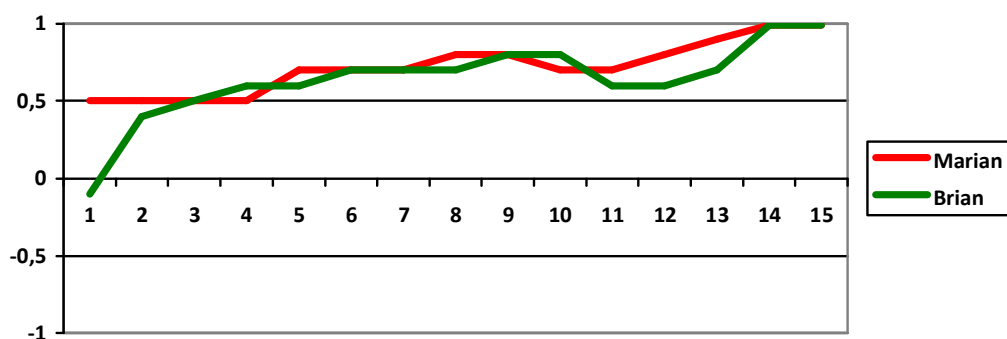


Figura 28 - Conjunto de valores de variação da felicidade/tristeza dos atores envolvidos na cena da proposta de casamento.

Com este conjunto de dados é possível calcular a felicidade da cena usando a equação 2. Na história de testes o fator de importância da Marian é 0.8 e o fator de importância do Brian é 0.9. Este fator é usado para garantir que a emoção dos atores principais terá mais influência sobre os atores menos importantes. No exemplo da cena do pedido de casamento o resultado do calculo da felicidade da cena é 0.9.

O mesmo pode ser feito para determinar o valor de coragem/medo da cena. O conjunto de valores da variação de coragem/medo dos atores na cena do pedido de casamento é mostrado na Figura 29. Aplicando estes valores a equação 3, tem-se o resultado da coragem/medo da cena sendo 0.72.

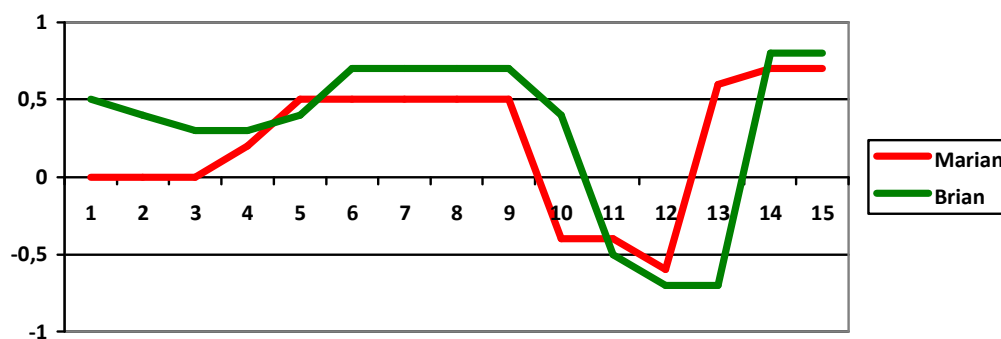


Figura 29 - Conjunto de valores de variação da coragem/medo dos atores envolvidos na cena da proposta de casamento.

O mesmo é feito para determinar a afeição da cena. Aplicando os valores de variação da afeição (Figura 30) dos atores envolvidos na cena a equação 4 tem-se como resultado a afeição da cena sendo 0.88.

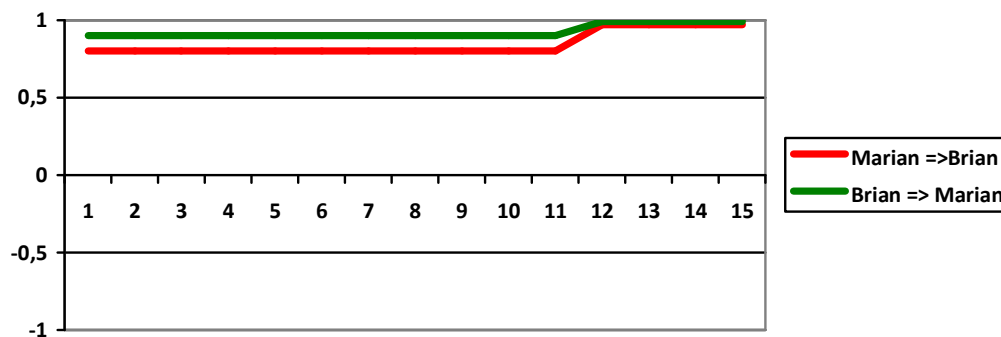


Figura 30 - Conjunto de valores de variação da afeição dos atores envolvidos na cena da proposta de casamento.

A última característica é o clima do ambiente. A característica é extraída do local onde a cena acontece. Na cena de exemplo, o valor é 0.8.

Com todas as características utilizadas pela SVM em mãos, o agente pode classificá-las dentre os possíveis perfis de efeitos visuais para decidir qual é o perfil mais adequado para a cena atual. No exemplo da cena do pedido de casamento o Diretor de Fotografia classificou a cena como uma "Cena Feliz" (de acordo com a Tabela 5).

Com o perfil selecionado, o agente finalmente pode ativar os efeitos visuais descritos pelo perfil de cena na dramatização. O resultado visual da cena é mostrado na Figura 31. As outras opções para esta mesma cena são apresentadas na Figura 32.



Figura 31 - Resultado visual da cena do pedido de casamento.



Cena Triste



Cena de Raiva



Cena de Medo



Cena de Tensão

Figura 32 - Outras opções de efeitos visuais da cena do pedido de casamento.

5.2.6 Diretor de música

Durante o processo de criação de um filme, o diretor de música tem a missão de usar o seu conhecimento sobre músicas para criar o clima e a atmosfera das cenas, de modo a reforçar as emoções expressas pela narrativa. Assim como o Editor, o Diretor de Música trabalha no filme após as filmagens, podendo escolher e avaliar quais as melhores músicas para as cenas. No modelo de dramatização proposto neste trabalho, o agente Diretor de Música deve escolher em tempo real, durante a dramatização, as melhores trilhas sonoras para cada cena.

O agente Diretor de Música é semelhante ao agente Diretor de Fotografia. Ambos possuem o objetivo de expressar emoções, para isso um manipula as músicas e o outro os efeitos visuais das cenas. O processo de escolha da melhor música para enfatizar a emoção das cenas, assim como a seleção do melhor efeito visual feita pelo Diretor de Fotografia, pode ser visto como um problema de classificação. Visto que é necessário um conjunto fixo de possíveis trilhas sonoras (classes resultantes) e a escolha de qual música deve ser utilizada é resultante de algum padrão relativo às emoções dos atores (características de entrada). Deste modo, o agente Diretor de Música também faz uso de uma SVM treinada para selecionar a melhor música para enfatizar as emoções das cenas.

Para definir a estrutura da SVM utilizada pelo agente Diretor de Música, primeiramente é necessário definir todas as possíveis classes nas quais as emoções dos atores poderão ser classificadas. Para isso foram definidos os perfis de cenas apresentados na Tabela 7. Estes perfis de cena caracterizam um determinado tipo de emoção e identificam o tipo de música necessária para expressá-la. A mesma abordagem utilizada para a escolha dos perfis de cena do Diretor de Fotografia foi adotada para definir os perfis do Diretor de Música. A escolha das características das trilhas sonoras para os perfis emocionais foi feita com base nos estudos sobre músicas apresentados na sessão 3.8.

As características escolhidas para serem usadas na SVM do Diretor de Música são as mesmas utilizadas pelo Diretor de Fotografia (nível de felicidade/tristeza, coragem/medo, relações dos atores envolvidos na cena e o clima do ambiente). A estrutura das SVMs do Diretor de Música e do Diretor de Fotografia é exatamente a mesma; ambos têm o objetivo de classificar a emoção que a cena deve expressar. Entretanto, se faz necessário a utilização de duas SVMs, pois o treinamento de cada um dos agentes deve ser feito por um profissional diferente. Em alguns casos, a visão de um diretor de fotografia pode ser diferente da visão do

diretor de música. Portanto, duas bases de treinamento são necessárias. Assim, cada agente tem uma SVM treinada especificamente para a sua tarefa.

Tabela 7 - Perfis de trilhas sonoras do agente Diretor de Música.

Emoção	Áudio
Cena Feliz	Trilhas sonoras em escala maior; ritmo rápido; tons agudos com grandes variações.
Cena Triste	Trilhas sonoras em escala menor; ritmo lento; faixa restrita de melodias.
Cena de Medo	Trilhas sonoras com ritmo rápido; dissonância; pequenas variações de melodia.
Cena de Raiva	Trilhas sonoras em escala menor; ritmo rápido; tons agudos.
Cena de Tensão	Trilhas sonoras em escala menor; melodias ascendentes; harmonias dissonantes.

O mesmo processo de treinamento realizado para o Diretor de Fotografia é também realizado para o Diretor de Música. Diversas cenas foram simuladas e os perfis de cenas mais adequados foram selecionados baseando-se na emoção que a cena simulada deveria expressar. As características, juntamente com o perfil de trilhas sonoras selecionado foram salvos em uma base de treinamento. Com a SVM treinada, o agente Diretor de Música é capaz de atuar como um diretor de música real e, com base na sua experiência anterior, selecionar em tempo real as melhores trilhas sonoras para realçar as emoções das cenas.

Durante o processo de dramatização, o agente Diretor de Música deve selecionar no início de cada cena a melhor trilha sonora para enfatizar a emoção da cena que será dramatizada. Para isso, no início das cenas o agente recebe do Diretor uma mensagem informando sobre a nova cena que será dramatizada, em seguida utiliza a rede de emoções e relações para simular os eventos da cena a fim de gerar o conjunto de todas as emoções e das relações ao longo da cena para cada ator. Com este conjunto de dados, o Diretor de Música extrai as características utilizadas pela SVM e as classifica dentre as possíveis classes a fim de determinar a música mais adequada para cena.

Considerando a cena do pedido de casamento utilizada anteriormente para exemplificar o agente Diretor de Fotografia, cujos eventos são mostrados na Tabela 6. O

mesmo processo de simulação dos eventos da cena utilizando a rede de relações e os cálculos das emoções globais da cena realizados pelo Diretor de Fotografia para extrair as características utilizadas pela SVM, também são realizados pelo Diretor de Música. Após a extração das características, o Diretor de Música faz a classificação do perfil de áudio mais adequado para a cena. Após a classificação, o Diretor de Música executa a trilha sonora descrita no perfil de áudio selecionado. No caso da cena do pedido de casamento, o perfil de trilha sonora selecionado foi o de “Cena Feliz”. Infelizmente não é possível demonstrar no papel o resultado produzido pelo Diretor de Música.

6 RESULTADOS

Para validar a arquitetura proposta nesta dissertação, foram realizados testes de performance e precisão para cada um dos agentes que utilizavam SVMs para a previsão de soluções, pois estes são os agentes que demandam mais processamento e necessitam de uma avaliação de precisão para garantir que as SVMs estão adequadamente treinadas. Por fim, foi realizado também um teste geral de aceitação para avaliar o quanto a utilização dos conceitos de cinematografia melhorou visualmente a dramatização das histórias.

Todos os testes de performances foram realizados em um Intel Core 2 Quad 2.40 GHz, 4 GB de memória, utilizando um único núcleo para processar as SVMs.

6.1 Validação do agente Editor

Para a validação do agente Editor, primeiramente foi realizado o teste de desempenho de treinamento para avaliar o tempo necessário para treinar as SVMs. Para isso, foi realizado o treinamento de todas as SVMs utilizando para cada uma, um número variado de exemplos de treinamento.

A Tabela 8 mostra o resultado do teste de desempenho de treinamento de todas as SVMs do agente Editor utilizando as bases de treinamento variando de 10 a 55 exemplos. O mesmo resultado pode ser visto graficamente na Figura 33. É importante ressaltar que este tempo de treinamento é o somatório do tempo necessário para treinar as 9 SVMs do agente Editor. Além disso, este processo é executado somente uma vez durante a inicialização da dramatização.

Tabela 8 - Tempo necessário para o treinamento das 9 SVMs do agente Editor utilizando bases de treinamento de tamanho variado.

Número de Exemplos de Treinamento	10	25	35	45	55
Tempo necessario para o Treinamento (<i>segundos</i>)	1,79	2,37	3,37	4,05	4,52

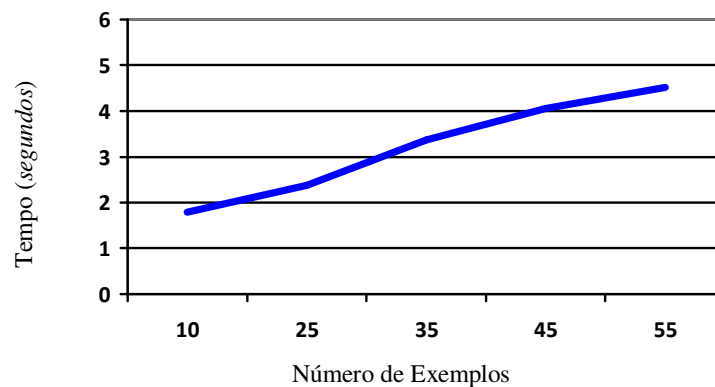


Figura 33 - Tempo necessário para o treinamento das 9 SVMs do agente Editor utilizando bases de treinamento de tamanho variado.

O segundo teste realizado no agente Editor foi o de mensuração do tempo necessário para a classificação de um conjunto de dados e a previsão do resultado. Este é o ponto mais crítico, pois o processo deve ser executado em tempo real durante a dramatização das cenas. No caso do agente Editor este resultado é utilizado para selecionar a tomada da câmera mais adequada para a cena que está sendo dramatizada e qualquer atraso na classificação pode resultar na perda de algum evento que estava enquadrado na tomada anterior.

Para realizar o teste de desempenho de classificação, cada SVM foi treinada com um número de exemplos de treinamento e então utilizadas para prever as melhores tomadas em uma sequência de 6 cenas que totalizam aproximadamente 40 tomadas diferentes. Para cada uma das tomadas, o tempo necessário para a classificação foi calculado. A Tabela 9 mostra o resultado do teste de desempenho de previsão do agente Editor utilizando bases de treinamento variando de 10 a 55 exemplos e o tempo correspondendo a média de todas as SVMs. O mesmo resultado pode ser visto graficamente na Figura 34.

Tabela 9 - Tempo necessário para a classificação utilizando as SVMs do agente Editor com bases de treinamento de tamanho variado.

Número de Exemplos de Treinamento	10	25	35	45	55
Tempo necessario para a Previsão (μs)	30.3	38.6	51.4	60.1	63.9

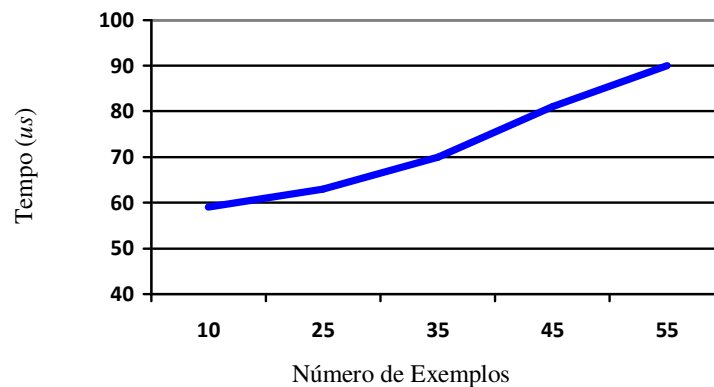


Figura 34 - Tempo necessário para a classificação utilizando as SVMs do agente Editor com bases de treinamento de tamanho variado.

Ambos os testes de desempenho demonstram a viabilidade da aplicação das SVMs em tempo real. Por fim, resta somente o teste de precisão para a validação completa do agente Editor. O teste de precisão visa mensurar a porcentagem de acertos da SVM na classificação de dados desconhecidos. Para realizar este teste, para cada SVM utilizada pelo Editor, foram criadas 5 bases de treinamento com diferentes números de exemplos e, para cada uma, uma base de teste com metade do número de exemplos da base de treinamento correspondente. As bases de treinamento foram utilizadas para treinar as SVMs e os exemplos das bases de teste foram classificados. Os acertos e erros foram então computados. A Tabela 10 mostra o resultado deste teste com as bases de treinamento variando de 10 a 55 exemplos. A porcentagem da precisão corresponde à média dos resultados obtidos para as diferentes SVMs. O mesmo resultado pode ser visto graficamente na Figura 35.

Tabela 10 - Taxa de reconhecimento das SVMs do agente Editor utilizando bases de treinamento de tamanho variado.

Número de Exemplos de Treinamento	10	25	35	45	55
Precisão	92%	94.6%	96.5%	98%	98.6%

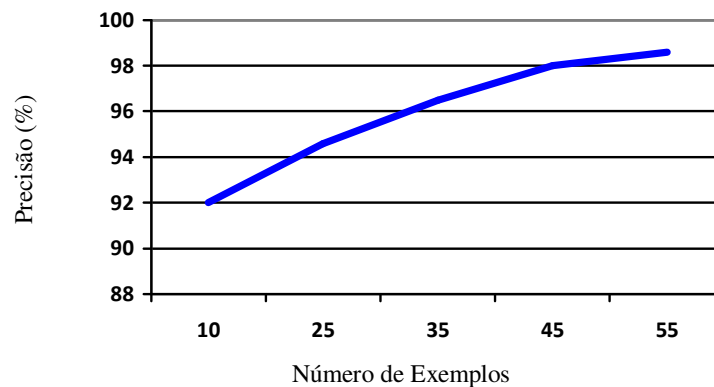


Figura 35 - Taxa de reconhecimento das SVMs do agente Editor utilizando bases de treinamento de tamanho variado.

É visível que o custo computacional cresce quase linearmente com o número de exemplos da base de treinamento. Mais exemplos resultam em alta precisão, mas por outro lado diminuem o desempenho de treinamento e reconhecimento. Menos exemplos de treinamento resultam em alto desempenho, porém diminuem a precisão do reconhecimento. Entretanto, mesmo com a maior base de treinamento utilizada nos testes, o desempenho, tanto de treinamento como de reconhecimento, pode ser considerado excelente e permite a utilização das SVMs em tempo real para selecionar as melhores tomadas para filmar as cenas que estão sendo dramatizadas pelos atores.

6.2 Validação dos agentes Diretores de Fotografia e Música

O agente Diretor de Fotografia e o agente Diretor de Música, apesar de serem treinados por bases de treinamento diferentes, possuem SVMs constituídas pela mesma estrutura. Portanto, é possível avaliar os dois agentes em conjunto.

O primeiro teste realizado para validar os agentes foi o de desempenho de treinamento. A mesma metodologia de teste utilizada no agente Editor foi adotada, com o diferencial dos agentes utilizarem somente uma SVM. O resultado do teste com a base de treinamento variando de 25 a 65 exemplos pode ser visto numericamente na Tabela 11 e graficamente na Figura 36.

Tabela 11 - Tempo necessário para o treinamento da SVM do agente Diretor de Fotografia/Diretor de Música utilizando bases de treinamento de tamanho variado.

Número de Exemplos de Treinamento	25	35	45	55	65
Tempo necessário para o Treinamento (<i>ms</i>)	199	264	374	451	502

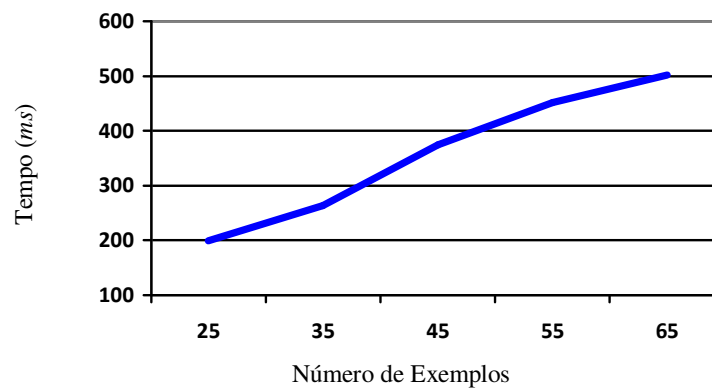


Figura 36 - Tempo necessário para o treinamento da SVM do agente Diretor de Fotografia/Diretor de Música utilizando bases de treinamento de tamanho variado.

O segundo teste realizado foi o de mensuração do tempo necessário para a classificação de um conjunto de dados em tempo real. Diferente do agente Editor, nos agente Diretor de Fotografia e Diretor de Música este ponto acaba não sendo crítico para o sistema, pois o processo é executado somente uma vez no início de uma cena e mesmo que o processo usasse mais tempo, a dramatização estaria em um processo de transição de cenas, o que não acarretaria em perda de informações visuais.

Para realizar o teste de desempenho de classificação, a SVM dos agentes foi treinada com um determinado número de exemplos de treinamento e então utilizada para prever os melhores perfis emocionais em uma sequência de 6 cenas. No início de cada cena o perfil de efeitos visuais e músicas são classificados, e o tempo necessário para a classificação é então calculado. A Tabela 12 mostra o resultado do teste de desempenho de previsão utilizando bases de treinamento variando de 55 a 65 exemplos. O mesmo resultado pode ser visto graficamente na Figura 37.

Tabela 12 - Tempo necessário para a classificação utilizando a SVM do agente Diretor de Fotografia/Diretor de Música com bases de treinamento de tamanho variado.

Número de Exemplos de Treinamento	25	35	45	55	65
Tempo necessário para o Previsão (μs)	21.1	28.7	39.1	51.9	59.1

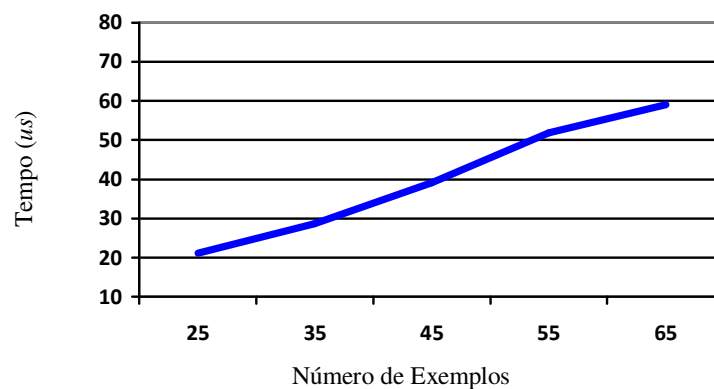


Figura 37 - Tempo necessário para a classificação utilizando a SVM do agente Diretor de Fotografia/Diretor de Música com bases de treinamento de tamanho variado.

O último teste realizado para validar os agentes é o de precisão de classificação. A mesma metodologia de teste utilizada no agente Editor foi adotada. A Tabela 13 mostra o resultado do teste com as bases de treinamento variando de 25 a 65 exemplos. O mesmo resultado pode ser visto graficamente na Figura 38.

Tabela 13 - Taxa de reconhecimento da SVM do agente Diretor de Fotografia/Diretor de Música utilizando bases de treinamento de tamanho variado.

Número de Exemplos de Treinamento	25	35	45	55	65
Precisão	93%	94.6%	95.4%	96%	96.8%

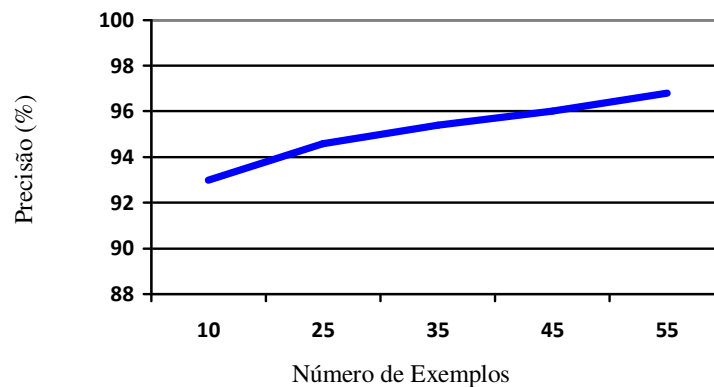


Figura 38 - Taxa de reconhecimento da SVM do agente Diretor de Fotografia/Diretor de Música utilizando bases de treinamento de tamanho variado.

Pode-se concluir que, assim como no agente Editor, o custo computacional das SVMs do Diretor de Fotografia e Diretor de Música cresce quase linearmente com o número de exemplos da base de treinamento. Todos os testes aqui apresentados demonstram a aplicabilidade das SVMs utilizadas pelos agentes em tempo real durante dramatização.

6.3 Validação do modelo de dramatização

Para uma avaliação geral do modelo de dramatização, foi realizado um comparativo visual da nova versão do sistema de dramatização utilizando o modelo proposto neste trabalho, com a primeira versão do Logtell (Pozzer, 2005). Também foi feita a mesma comparação com uma versão intermediária do novo sistema utilizando a nova *engine* gráfica, porém somente com parte dos conceitos de cinematografia. O resultado visual das comparações pode ser visto na Tabela 14.

Tabela 14 - Comparativo visual do modelo de dramatização apresentado neste trabalho com duas versões anteriores do sistema.

Primeira versão do Logtell (Pozzer, 2005)	Versão do Logtell utilizando à nova engine gráfica	Nova versão com o modelo proposto nesta dissertação
		
reduce_protection(Marian, White_Palace)		
		
go(Draco, White_Palace)		
		
attack(Draco, White_Palace)		
		
kidnap(Draco, Marian)		

 brian attacks the dragon's castle	 Brian attacks the Red Castle	
attack(Brian, Red_Castle)		
 brian fights against draco	 Brian fights against draco	
fight(Brian, Draco)		
 brian frees marian	 Brian frees Marian	 Brian says: Princess! I am here to save you!
free(Brian, Marian)		
 brian and marian get married	 Brian and Marian get married	 Brian says: How do you find yourself this day?
marry(Brian, Marian)		

É perceptível que a primeira versão do Logtell (Pozzer, 2005) possui uma qualidade gráfica inferior às novas versões. Além disso, também é possível observar a ausência de princípios cinematográficos nas cenas, todas as cenas da narrativa são filmadas de longe e em

ângulos altos. Na versão utilizando a nova *engine* gráfica é visível a melhoria na qualidade gráfica, porém ainda se pode observar que algumas cenas são prejudicadas por oclusões e cenas filmadas de longe, além da ausência de clima emocional nas cenas. Já na versão utilizando o modelo de dramatização apresentado neste trabalho, é possível perceber o uso de ângulos de câmera mais adequados e que exploram mais a dramaticidade das cenas, percebe-se também a variação da iluminação do ambiente usada para criar o clima emocional das cenas.

7 CONCLUSÃO

Este capítulo apresenta as considerações gerais sobre a presente pesquisa, as contribuições mais relevantes e os trabalhos futuros recomendados.

7.1 Considerações gerais

Neste trabalho foi apresentado um modelo de dramatização de histórias baseado em agentes cinematográficos autônomos para storytelling interativo. Os agentes foram modelados seguindo princípios e conceitos definidos pela cinematografia, de modo que estes pudessem enfatizar o conteúdo dramático das cenas, apresentando-as de modo atraente e envolvente para os espectadores.

O modelo apresentado é composto por seis agentes inspirados na cinematografia:

- O agente Roteirista, que tem como objetivo gerenciar e gerar as cenas da dramatização com base em eventos de alto nível provenientes do planejador IPG (Ciarlini, 1999);
- O agente Diretor, que tem por objeto gerenciar toda a narrativa, interpretar as cenas criadas pelo Roteirista e solicitar a dramatização de ações ao atores;
- O agente Cameraman, cujo objetivo é propor possíveis tomadas de câmera para filmar as cenas que são dramatizadas;
- O agente Editor, que seleciona as tomadas mais adequadas para filmar as cenas da dramatização, com base nas tomadas sugeridas pelo Cameraman;
- O agente Diretor de Fotografia, que busca enfatizar as emoções da dramatização, criando efeitos visuais, manipulando a iluminação e cores das cenas; e
- O agente Diretor de Música, cujo objetivo é expressar as emoções das cenas através de músicas.

A definição destes agentes, e da arquitetura como um todo, é resultado de uma extensa pesquisa sobre cinematografia. Buscou-se integrar ao modelo os principais conceitos de cinematografia apresentados pela literatura, principalmente no que diz respeito à manipulação das câmeras e à organização da equipe de filmagem durante a criação de um filme.

Um dos principais desafios encontrados durante o desenvolvimento deste trabalho foi a subjetividade de como a cinematografia é tratada em alguns aspectos. Não existem padrões ou regras pré-definidas sobre como as cenas devem ser filmadas. Cada diretor ou até mesmo integrantes da equipe de filmagem têm uma forma diferente de tratar de suas funções. Neste ponto fica claro que o estilo de direção de um diretor pode muitas vezes determinar o resultado final de um filme, tanto é que espectadores entusiastas por cinema conseguem muitas vezes identificar o diretor de um filme apenas assistindo a algumas cenas do mesmo. O uso de técnicas de aprendizado supervisionado, mais especificamente de SVMs, foi a metodologia adotada para se lidar com a subjetividade existente na cinematografia. Desta forma, deixa-se a cargo do próprio método de classificação a função de encontrar um padrão existente em um conjunto de exemplos de treinamento. Além disso, possibilita-se que estes exemplos de treinamento possam ser criados com auxílio de um profissional cineasta, garantido que o processo de classificação seja baseado na visão de um cineasta do mundo real.

O modelo de agentes cinematográficos apresentado neste trabalho destina-se a ambientes de storytelling interativo baseados em roteiro (*plot-based*), porém o mesmo poderia ser facilmente adaptado para abordagens baseadas em personagens (*character-based*). Neste caso, não seria necessário um agente Roteirista, visto que a história é gerada através da própria interação entre os atores autônomos. Os outros agentes podem ser mantidos, porém nesta abordagem, seria função dos próprios agentes analisarem a interação dos atores e identificar os eventos que estão acontecendo no ambiente, para então executar o posicionamento das câmeras, seleção de tomadas e escolha de efeitos audiovisuais para as cenas. Além de storytelling interativo, o modelo proposto neste trabalho também poderia ser expandido para jogos, porém visando atingir uma área conhecida como “espectadores de jogos”, visto que o modelo foca na dramaticidade e não possui nenhum método que privilegie a jogabilidade.

Pode-se concluir que o modelo apresentado neste trabalho atendeu a todas as expectativas iniciais. A melhora no resultado visual da dramatização é notória, porém ainda existem muitos outros aspectos que devem ser pesquisados para que se possa ter um sistema de dramatização que venha a ser considerado “perfeito”, com um resultado final comparável a um filme.

7.2 Contribuições

- **Modelo de dramatização baseado em agentes cinematográficos autônomos:** A principal contribuição deste trabalho é o modelo de dramatização baseado em agentes diretores autônomos, que utilizam conhecimentos cinematográficos para a dramatização de histórias com ênfase no conteúdo dramático.
- **Método automático para a seleção das melhores tomadas para a filmagem de cenas em tempo real:** O método utilizado pelo agente Editor para a seleção das melhores tomadas para filmar as cenas em tempo real utilizando SVMs, treinadas com conhecimento cinematográfico, é outra contribuição deste trabalho. Na literatura analisada, não foi encontrado nenhuma outra solução para este problema.
- **Método automático para a seleção dos melhores efeitos audiovisuais para enfatizar as emoções de cenas em storytelling interativo:** Outra contribuição deste trabalho é o método utilizado pelos agentes Diretores de Fotografia e Música para a seleção automática dos melhores efeitos audiovisuais para enfatizar as emoções das cenas. Os trabalhos relacionados que buscam enfatizar as emoções das cenas utilizam outras abordagens para a aplicação destes efeitos. Desta maneira, o método proposto neste trabalho também pode ser considerado uma contribuição.
- **Novo método para o cálculo da emoção global de uma cena:** Quando não se tem previamente a informação de qual emoção uma cena deve expressar, é necessário que esta informação seja extraída de algum lugar. Neste trabalho, foi proposto um método para o cálculo das emoções globais de uma cena baseando-se somente no estado emocional dos personagens que compõem a cena.
- **Aplicação de support vector machines como método de inteligência artificial em storytelling interativo/jogos:** Este trabalho introduziu o uso de SVMs como método de inteligência artificial para storytelling interativo. Foi demonstrado que SVMs podem ser utilizadas com sucesso, para determinadas tarefas, em aplicações de tempo real, como é caso de dramatizações de storytelling interativo e jogos.

7.3 Publicações

Durante o desenvolvimento desta dissertação, foram publicados os seguintes artigos com os resultados parciais obtidos:

- ACE 2009 – “Virtual Cinematography Director for Interactive Storytelling”. In: International Conference on Advances in Computer Entertainment Technology, Athens, Greece, 2009, p. 263-270.
- SBGames 2009 – “Support Vector Machines for Cinematography Real-Time Camera Control in Storytelling Environments” In: VIII Simpósio Brasileiro de Jogos e Entretenimento Digital, Rio de Janeiro, 2009, p. 176-184, October 2009.

Com os resultados finais, pretende-se publicar novos artigos em conferências ou periódicos internacionais.

7.4 Trabalhos futuros

No decorrer do desenvolvimento desta dissertação, muitas idéias foram consideradas. Entretanto, com o objetivo de manter o foco nos principais aspectos da cinematografia, algumas destas não foram incorporadas ao modelo proposto. A seguir, são apresentadas sugestões de pesquisas futuras para este trabalho.

- **Explorar a dramaticidade dos atores:** Atualmente, um dos principais problemas da dramatização do Logtell é a falta de vida dos personagens. Por mais que a câmera, iluminação e música estejam apropriadas, os atores continuam possuindo um conjunto fixo de animações que não parecem naturais. Um dos trabalhos futuros sugeridos é a exploração de formas para melhorar a dramaticidade destes atores. Uma abordagem baseada em animação comportamental pode ser uma das alternativas para solucionar estes problemas. Alguns outros aspectos nesta mesma linha de pesquisa que podem ser explorados são:
 - Expressão das emoções dos atores através de animações comportamentais;
 - Expressões faciais para enfatizar as emoções dos atores;

- Interação comportamental entre os atores e o ambiente.

- **Controle de eventos em paralelo pelo agente Diretor:** A versão atual do Logtell gera sequências ordenadas de eventos; não existe a necessidade de se tratar eventos que possam estar acontecendo simultaneamente em locais diferentes. Entretanto, para tornar o sistema mais robusto, seria interessante que o agente Diretor incorporasse um método para o controle e dramatização de eventos em paralelo. Sugere-se como trabalho futuro, primeiramente, uma reestruturação do modelo de geração de histórias do Logtell, de modo que este permita a geração de histórias com eventos em paralelo. Em seguida, seria possível a expansão do agente Diretor para que ele pudesse controlar a dramatização dos eventos em paralelo.

Uma possível solução para o controle da dramatização de eventos em paralelo seria a utilização de um classificador, que pudesse decidir qual é o evento mais importante para ser mostrado a cada momento durante a dramatização. Em alguns momentos, podem existir eventos importantes para o entendimento da história acontecendo simultaneamente, neste caso o Diretor deveria alternar entre a exibição dos eventos ou então pausar a execução de um, enquanto mostra o outro. Em outros casos, pode ser que um evento em paralelo seja somente a caminhada de um ator, neste caso não seria necessário mostrar todo o percurso do ator para o espectador.

- **Integração com o modelo de dramatização não-determinística:** Em Dória (2009), é apresentado um modelo para controlar o processo de dramatização de histórias no Logtell. Esse modelo baseia-se na associação um autômato não-determinístico a cada evento da história. Estes autômatos são utilizados para descrever micro-ações para os eventos, garantindo maior variedade de histórias e maior flexibilidade para o controle do tempo de cada evento. O modelo apresentado por Dória (2009) pode ser utilizado para substituir o processo de expansão da história, que atualmente é realizada pelo agente Roteirista. A integração dos dois sistemas foi iniciada no momento que as micro-ações de baixo nível, interpretadas pelo agente Diretor, foram disponibilizadas para a camada de geração de histórias. Deste modo, teoricamente os sistemas estariam aptos a ser facilmente integrados, porém mais testes devem ser realizados para a completa integração dos dois sistemas.

- **Aprendizado dos agentes em tempo real:** Atualmente, o treinamento dos agentes Editor, Diretor de Fotografia e Diretor de Música é realizado em uma etapa anterior à dramatização. Um trabalho futuro a ser analisado, é o treinamento destes agentes em tempo real durante a dramatização, de modo que os agentes possam adaptar-se às preferências dos espectadores. Para isso, seria necessária alguma forma de os espectadores avaliarem as escolhas dos agentes, e estas avaliações poderem ser guardadas na base de treinamento dos agentes, para que as SVMs possam ser treinadas novamente com exemplos de treinamento fornecidos pelos próprios espectadores.

REFERÊNCIAS

ABRAMS, N.; UDRIS, J.; BELL, I. **Studying Film**. Londres, Inglaterra: Hodder Education, 2001.

AMERSON, D; KIME, S. **Real-time Cinematic Camera Control for Interactive Narratives**. Working notes of the AAAI Spring Symposium on Artificial Intelligence and Interactive Entertainment. Stanford, California, 2000.

AYLETT, R.; DIAS, J.; PAIVA, A. An affectively driven planner for synthetic characters. In: **Proceedings of 16th International Conference on Automated Planning and Scheduling**, Cumbria, Inglaterra, p. 2-10. 2006.

ARIJON, D. **Grammar of the Film Language**. Los Angeles, Estados Unidos: Silman-James Press, 1976.

BATES, J.; LOYALL, B.; REILLY, S. An Architecture for Action, Emotion, and Social Behavior. In: **Proceedings of the 4th European Workshop on Modeling Autonomous Agents in a Multi-Agent World**, Pittsburgh, Estados Unidos, p. 55-68, 1992.

BOURNE O.; SATTAR A. Applying Constraint Weighting to Autonomous Camera Control. In: **Proceedings of the Fifth Artificial Intelligence and Interactive Digital Entertainment**. Marina Del Ray, California, Estados Unidos. p. 3-8, 2005.

BROWN, B. **Cinematography: Theory and Practice: Image Making for Cinematographers, Directors, and Videographers**. Burlington, Estados Unidos: Focal Press, 2002.

BURELLI, P.; JHALA, A. Dynamic Artificial Potential Fields for Autonomous Camera Control in 3D environments. In: **Proceedings of the Artificial Intelligence and Interactive Digital Entertainment (AIIDE'09)**, Stanford, Estados Unidos, 2009.

CAI, T.; HOFMANN, T. Hierarchical document categorization with support vector machines. In: **Proceedings of the ACM 13th Conference on Information and Knowledge Management**. Washington, Estados Unidos, p. 78-87, 2004.

CAI, Y.; MIAO, C.; TAN, A.H.; SHEN, Z. A Hybrid of Plot-Based and Character-Based Interactive Storytelling. In: **Proceedings of the 2nd International Conference of E-Learning and Games**. Hong Kong, p. 260-273, 2007.

CAMANHO, M. M.; CIARLINI, A. E. M.; FURTADO, A. L.; POZZER, C. T.; FEIJÓ, B. A Model for Interactive TV Storytelling. In: **VIII Brazilian Symposium on Digital Games and Entertainment**. Rio de Janeiro, Brasil, 2009.

CAVAZZA, M.; CHARLES, F.; MEAD, S. Character-based interactive storytelling. **IEEE Intelligent Systems**. Volume 17, Issue 4, p. 17-24, 2002.

CHANG, C.; LIN, C. LIBSVM: A Library for Support Vector Machines, 2001. Disponível em: <<http://www.csie.ntu.edu.tw/~cjlin/libsvm>>. Acessado em: 14 jun. 2010.

CHARLES, F.; CAVAZZA, M. Exploring the Scalability of Character-Based Storytelling. In: **Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent System**. New York, Estados Unidos, p. 872-879, 2004.

CHARLES, F.; LUGRIN, J.; CAVAZZA, M.; MEAD, S. Real-time camera control for interactive storytelling. In: **Proceedings of the Game-On**. Londres, Inglaterra, 2002.

CHRISTIE, M.; NORMAND, J.-M. A semantic space partitioning approach to virtual camera control. In **Annual Eurographics Conference, Computer Graphics Forum**. Dublin, Ireland, p. 247-256. 2005.

CIARLINI, A. E. M. **Geração Interativa de Enredos**. Tese (Doutorado em Informática) - Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 1999.

CIARLINI, A. E. M.; POZZER, C. T.; FURTADO, A. L.; FEIJÓ, B. A logic-based tool for interactive generation and dramatization of stories. In: **Proceedings of the International Conference on Advances in Computer Entertainment Technology**, Valencia, Spain, p. 133-140, 2005.

BYVATOV, E.; FECHNER, U.; SADOWSKI, J.; SCHNEIDER, G. Comparison of Support Vector Machine and Artificial Neural Network Systems for Drug/Nondrug Classification, **Journal of Chemical Information and Computer Sciences**, vol.43, p. 1882-1889, 2003.

COZIC, L. **Automated Cinematography for Games**. Master's Thesis - Lansdown Centre for Electronic Arts, School of Arts, Middlesex University. Londres, Inglaterra, 2007.

CRAWFORD, C. **Chris Crawford on Interactive Storytelling**. Berkeley, Estados Unidos: New Riders, 2004.

DAVIS, R. **Complete Guide to Film Scoring**. Boston, Estados Unidos: Berklee Press, 2000.

DORIA, T. R.; CIARLINI, A. E. M.; ANDREATTA, A. A Nondeterministic Model for Controlling the Dramatization of Interactive Stories. In: **Proceedings of the 2nd ACM Workshop on Story Representation, Mechanism and Context**. Vancouver, Canada, p. 21-26, 2008.

DRUCKER, H.; WU, D.; VAPNIK, V. Support Vector Machines for Spam Categorization. **IEEE Transactions on Neural Networks**. Volume 10, Número 5, p. 1048-1054, 1999.

DRUCKER, S. M.; ZELTZER, D. CamDroid: A system for implementing intelligent camera control, In: **Proceedings of the 1995 Symposium on Interactive 3D Graphics**. Monterey, Estados Unidos, p. 139-144, 1995

EL-NAQA, I.; YANG, Y.; WERNICK, M. N.; GALATSANOS, N. P.; NISHIKAWA, R. M.; A Support Vector Machine Approach for Detection of Microcalcifications. **IEEE Transactions on Medical Imaging**. Volume 21, Número 12, p. 1552-1563, 2002.

EL-NASR, M. S. **Automatic Expressive Lighting for Interactive Scenes**. PhD. Thesis - . Computer science Department, Northwestern University. Chicago, Estados Unidos, 2003.

EL-NASR, M. S. Interactive Narrative Architecture Based on Filmmaking Theory. **International Journal on Intelligent Games and Simulation**. Volume 3, issue 1, 2004.

EL-SATTAR, H. K. H. A new framework for plot-based interactive storytelling generation. In: **Proceedings of the Fifth International Conference on Computer Graphics, Imaging and Visualization**. Penang, Malaysia, p. 317-322. 2008.

FRIEDMAN, D.; FELDMAN, Y. Knowledge-Based Formalization of Cinematic Expression and its Application to Animation. In: **Proceedings of the 2002 Eurographics**. Saarbrücken, Alemanha, p. 163-168, 2002.

FURTADO, A. CIARLINI, A. Operational Characterization of Genre in Literary and Real-Life Domains. In: **Proceedings ER'99 Conceptual Modeling Conference**, Paris, France, p. 460-474, 1999.

GABRIELSSON, A.; LINDSTRÖM, E. **The Influence of Musical Structure on Emotional Expression**. Oxford University Press, New York, 2001.

GRASBON, D.; BRAUN, N. A morphological approach to interactive storytelling. In: **Proceedings of the 2001 Living in Mixed Realities**. Sankt Augustin, Alemanha, p. 337-340, 2001.

GUNN, S. **Support Vector Machines for Classification and Regression**. Technical Report, University of Southampton, Hampshire, Inglaterra, 1998. Disponível em: < <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.10.7171&rep=rep1&type=pdf>>. Acessado em: 14 jun. 2010.

HALPER, N.; HELBING, R.; STROTHOTTE, T. A camera engine for computer games: Managing the trade-o between constraint satisfaction and frame coherence. In: **Proceedings of the 2001 Eurographics**, Volume 20, p. 174-183, 2001.

HAWKINS, B. **Real-Time Cinematography for Games**. Hingham, Estados Unidos: Charles River Media, 2005.

HE, L.; COHEN, M. F.; SALESIN, D. H. The Virtual Cinematographer: A Paradigm for Automatic Real-Time Camera Control and Directing. In: **Proceedings of 23rd International Conference on Computer Graphics and Interactive Techniques**. New York, Estados Unidos, p. 217-224, 1996.

HERMANN, G. P. R. **Controle Automático de Câmera em Ambientes Virtuais Dinâmicos**. Dissertação (Mestrado em Informática) - Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2005.

HMEIDI, I.; Hawashin B.; El-Qawasmeh E. Performance of KNN and SVM classifiers on full word Arabic articles. **Advanced Engineering Informatics**, Volume 22, p. 106-111, 2008.

Katz, S. **Film Directing Shot by Shot: Visualizing from Concept to Screen**. Los Angeles, Estados Unidos: Michael Wiese, 1991

KELSO, M. T.; WEYHRAUCH, P.; BATES, J. Dramatic Presence. **Presence: The Journal of Teleoperators and Virtual Environments**, Volume 2, Número 1, MIT Press, Estados Unidos, 1993.

KNEAFSEY, J. **Virtual Cinematography for Computer Games**, Master's Thesis -, Institute of Technology Blanchardstown, Dublin, Irlanda, 2006.

LEBOWITZ, M. Creating Characters in a Story-telling Universe. **Poetics**. Volume 13, Número 3, 1984.

LOBRUTTO, V. **The Filmmaker's Guide To Production Design**. New York, Estados Unidos: Allworth Press, 2002.

MASCELLI, J. **The Five C's of Cinematography: Motion Picture Filming Techniques**. Los Angeles, Estados Unidos: Silman-James Press, 1965.

MATEAS, M. **Interactive Drama, Art, and Artificial Intelligence**. Ph.D. Thesis - School of Computer Science, Carnegie Mellon University, Pittsburgh, Estados Unidos, 2002.

MATEAS, M.; STERN, A. Structuring content in the facade interactive drama architecture. In: **Proceedings of the First Artificial Intelligence and Interactive Digital Entertainment Conference**, Marina del Rey, Estados Unidos, p. 93-98. 2005.

MATEAS, M.; STERN, A. Façade: An Experiment in Building a Fully-Realized Interactive Drama. **Game Developers Conference**, p. 4-8, 2003.

MEEHAN, J. Tale-Spin, an interactive program that writes stories. **Proceedings of the Fifth International Joint Conference on Artificial Intelligence**. Cambridge, Estados Unidos, p. 91-98, 1977.

MELO, C.; PAIVA, A. Environment Expression: Telling Stories Through Cameras, Lights and Music. **International Conference on Virtual Storytelling**. Strasbourg, França, p. 129-132, 2005.

MILLER, W. **Screenwriting for Narrative Film and Television**, Athens, Estados Unidos: Virgin Books, 1988.

MILLERSON, G., **Lighting for Television and Film**. Woburn, Estados Unidos: Focal Press, 1991.

NEWMAN, R. **Cinematic Game Secrets for Creative Directors and Producers**. Oxford, Inglaterra: Focal Press, 2008.

O'BRIEN, M.; SIBLEY, N. **The Photographic Eye: Learning to See with a Camera**. Worcester, Estados Unidos: Sterling, 1995.

PAIVA, A.; MACHADO, I.; PRADA, R. Heroes, villains, magicians, ... Dramatis personae in a virtual story creation environment. In: **Proceedings of intelligent user interfaces**. Santa Fé, Novo Mexico, Estados Unidos, p. 129-136, 2001.

PLUTCHIK, R. The nature of emotions. **American Scientist**, Volume 89, p. 344-350, 2001.

POZZER, C. T. **Um Sistema para Geração, Interação e Visualização 3D de Histórias para TV Interativa**. Tese (Doutorado em Ciência da Computação), Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brasil, 2005.

PROPP, V. **Morphology of the Folktale**. Austin, Texas: University of Texas Press, 2nd edition, 1968.

SALLES, F. Como se faz Cinema - Funções e equipe. Disponível em: <http://www.mnemocine.art.br/index.php?option=com_content&view=article&id=102:fazercinema1&catid=34:tecnica&Itemid=67>. Acessado em: 14 de jun. de 2010.

SCHWARTZ, H. P. M. **Intelligent Decision Making for Camera Views and Behaviours**. Master's Thesis, Media Technology and Games, University of Copenhagen. Copenhagen, Dinamarca, 2009.

SCIENTIFIC AMERICAN MAGAZINE, **Special issue on digital entertainment**, Volume 283, 2000.

SI, M.; MARSELLA, S.C.; RIEDL, M. Integrating plot-centric and character-centric designs: An mixed-initiative framework for interactive drama. In: **Proceedings of the 4th Conference on Artificial Intelligence and Interactive Digital Entertainment**, Palo Alto, California. 2008.

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

SZILAS, N. The mutiny: an interactive drama on IDtension. In: **Proceedings of the 3rd ACM International Conference on Digital Interactive Media in Entertainment and Arts**. New York, Estados Unidos, p. 539-540, 2008.

THOMPSON, R.; BOWEN, C. **Grammar of the Shot**. Woburn, Estados Unidos: Focal Press, 2009.

TOMLINSON, B.; BLUMBERG, B.; NAIN, D. Expressive Autonomous Cinematography for Interactive Virtual Environments, In: **Proceedings of the fourth international conference on Autonomous Agents**, Barcelona, Spain, p. 317-324, 2000.

TURNER, S. **Minstrel: A Computer Model of Creativity And Storytelling**. PhD Thesis - Computer Science Department, University of California, Estados Unidos 1992.

UNITY3D, Unity: Game Development Tool. Disponível em: <<http://unity3d.com/>>. Acessado em: 14 jun. 2010.

VAPNIK, V. **The Nature of Statistical Learning Theory**. New York, Estados Unidos: Springer, 1995.

VERLET, L. Computer Experiments on Classical Fluids. **PhysRev**, Volume 159, Número 98, 1967.

WANG, X.; ZHONG, Y. Statistical Learning Theory and State of the Art in SVM. In: **Proceedings of the 2nd IEEE International Conference on Cognitive Informatics**. Londres, Inglaterra, p. 55-60, 2003.

YOUNG, M. An Overview of the Mimesis Architecture: Integrating Intelligent Narrative Control into an Existing Gaming Environment. In: **Working Notes of the AAAI Spring Symposium on Artificial Intelligence and Interactive Entertainment**. Palo Alto, Estados Unidos, 2001.

APÊNDICE A - Support Vector Machine

Support Vector Machine, ou simplesmente SVM é uma poderosa metodologia para resolver problemas de aprendizagem de máquina proposta por Vapnik (1995). Consiste em um método de aprendizado supervisionado que tenta encontrar a maior margem para separar as diferentes classes de dados. Permite a classificação de conjuntos de dados linearmente separáveis e não linearmente separáveis. No caso de dados não linearmente separáveis o SVM emprega funções de Kernel de forma a mapear os dados de entrada para um espaço hiperdimensional e arranja os dados de modo que métodos lineares possam ser aplicados.

O SVM pertence à classe de algoritmos de aprendizado de máquina baseados em métodos de aprendizado supervisionado. Métodos de aprendizado supervisionado buscam deduzir uma função de classificação a partir de um conjunto de dados de treinamento. Os dados de treinamento são normalmente formados por vetores de características e um resultado desejado para aquele conjunto de características. A saída da função de classificação consiste de um rótulo de classe representando o resultado da classificação do conjunto de características de entrada. O objetivo de um método de aprendizado supervisionado é criar uma função (utilizando uma série de exemplos de treinamento) que classifique corretamente qualquer conjunto de características de entrada válidos.

A essência do SVM é a construção de um hiperplano ótimo, de modo que ele possa separar diferentes classes de dados com a maior margem possível. A Figura mostra um hiperplano ótimo separando a classe de quadrados da classe de estrelas. Como pode ser visto, existem muitos outros hiperplanos que poderiam separar as duas classes, no entanto, o SVM busca a melhor solução, o hiperplano ótimo.

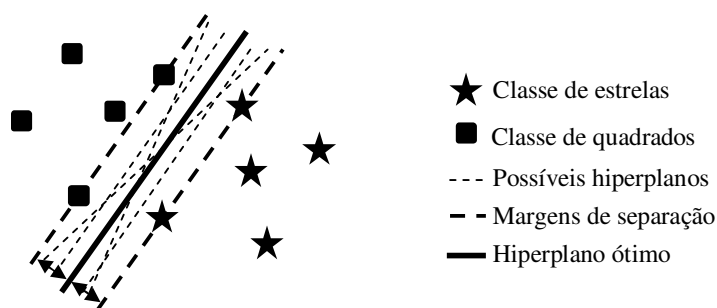


Figura A.1 - Hiperplano ótimo separando duas classes de dados.

Supondo um conjunto de dados de treinamento $(x_1, y_1), \dots, (x_l, y_l)$, onde cada exemplo $x_i \in R^n$ pertence à classe $y_i \in \{-1, +1\}$. O hiperplano de divisão entre as classes pode ser expresso como:

$$\omega \cdot x + b = 0$$

e as margens que separam as classes como:

$$\omega \cdot x + b = +1$$

$$\omega \cdot x + b = -1$$

onde,

- ω é um vetor de peso;
- b é um bias;
- x é um ponto no espaço R^n .

Este conjunto de vetores é separado pelo hiperplano ótimo se, e somente se, ele é separado sem erro e a distância entre o vetor mais próximo e o hiperplano é máxima. O hiperplano de separação pode ser descrito da seguinte forma:

$$\begin{cases} \omega \cdot x_i + b \geq +1, & \text{if } y_i = +1 \\ \omega \cdot x_i + b \leq -1, & \text{if } y_i = -1 \end{cases} \quad (\text{A.1})$$

ou equivalentemente:

$$y_i (\omega \cdot x_i + b) \geq 1, \quad i = 1, \dots, l$$

O hiperplano ótimo é aquele que satisfaz as condições e minimiza a função: $\frac{1}{2} \|\omega\|^2$

Vapnik (1995) mostrou que, para executar esta minimização, temos de maximizar a seguinte função no que diz respeito à variável α_i :

$$W(\alpha) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j (x_i \cdot x_j)$$

sujeito a $0 \leq \alpha_i, \quad i = 1, \dots, l$ e $\sum_{i=1}^l \alpha_i y_i$

Estes x_i s com $0 < \alpha_i$ são denominados *support vectors*. Os *support vectors* estão localizados nas margens de separação e são geralmente um pequeno subconjunto do conjunto dos dados de treinamento denotados por X_{SVM} .

Para um vetor desconhecido x_i , a sua classificação corresponde em encontrar:

$$f(x) = \text{sign} \left(\sum_{x_i \in X_{SVM}} \alpha_i y_i (x \cdot x_i) + b \right)$$

onde:

$$\omega = \sum_{x_i \in X_{SVM}} \alpha_i y_i x_i$$

é a soma é sobre estes *support vectors* diferentes de zero com $0 < \alpha_i$. Em outras palavras, este processo corresponde à conclusão de que lado do hiperplano o vetor desconhecido pertence.

No entanto, na maioria dos casos a classificação não é tão simples. Muitas vezes estruturas mais complexas são necessárias para a criação de um hiperplano de separação ótima. Por exemplo, na Figura A.1 a separação exige uma curva, o que é muito mais complexo do que uma simples linha.

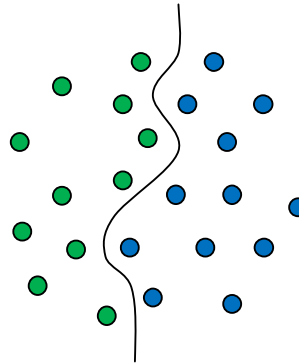


Figura A.2 – Conjunto de dados não linearmente separáveis.

No caso em que os dados não são linearmente separáveis, o SVM utiliza dois métodos para construir o hiperplano ótimo. Primeiramente permite erros de treinamento. Além disso o SVM transforma o espaço original de característica de entrada em um espaço de dimensão

mais alta através de uma função de $\phi(x)$. Neste espaço de dimensão mais alta é possível que as características possam ser linearmente separadas Wang and Zhong (2003). Neste caso o problema pode ser descrito como:

$$\min \frac{1}{2} \|\omega\|^2 + C \sum_{i=1}^l \xi_i \quad (\text{A.2})$$

sujeito a:

$$y_i(\omega \cdot \phi(x_i) + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad i = 1, \dots, l, \quad C > 0$$

Um termo de penalidade $C \sum_{i=1}^l \xi_i$ é utilizado na função para considerar os erros de treinamento. Se os dados forem linearmente separáveis o problema (A.2) retorna para a equação (A.1) e todos os ξ_i serão zero. Pode-se equivalentemente maximizar $W(\alpha)$, mas a restrição agora será $0 \leq \alpha_i \leq C$ em vez de $0 \leq \alpha_i$:

$$W(\alpha) = \sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i=1}^l \sum_{j=1}^l \alpha_i \alpha_j y_i y_j (\phi(x_i) \cdot \phi(x_j)) \quad (\text{A.3})$$

sujeito a:

$$0 \leq \alpha_i \leq C, \quad i = 1, \dots, l \quad \text{e} \quad \sum_{i=1}^l \alpha_i y_i = 0$$

O produto interno no espaço de dimensão mais alta deve ser substituído por alguma função especial de Kernel. Uma função de Kernel recebe dois pontos do espaço de entradas e computa o produto escalar desses dados no espaço de características:

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$$

Alguns dos Kernels mais populares são os Polinomiais, os Gaussianos ou RBF (*Radial-Basis Function*) e os Sigmoidais (Tabela A.1).

Tabela A.1 – Lista dos Kernels mais populares.

Kernel	Função $K(x_i, x_j)$
Polinomial	$(\delta(x_i \cdot x_j) + k)^d$
Gaussiano	$\exp(-\sigma \ x_i - x_j\ ^2)$
Sigmoidal	$\tanh(\delta(x_i \cdot x_j) + k)$

Para ilustrar o processo de treinamento de uma SVM não linear será utilizado o problema de classificação do operador XOR. A tabela A.2 mostra os exemplos de treinamento e a figura A.3 mostra estes exemplos em um plano 2D. Não existe nenhuma função linear para separar este conjunto de dados.

Tabela A.2 – Exemplos de treinamento para o problema do operador XOR.

Vetor de Característica x	Resultado y
$(-1, -1)$	-1
$(-1, +1)$	+1
$(+1, -1)$	+1
$(+1, +1)$	-1

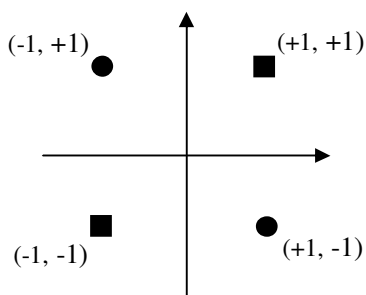


Figura A.2 – Conjunto de dados não linearmente separáveis do operador XOR.

Considerando a função Kernel:

$$K(x, x_j) = (1 + x \cdot x_j)^2$$

Considerando $x = (x_1 \cdot x_2)$ e $x_i = (x_{i1} \cdot x_{i2})$, a função Kernel pode ser expressa em termos de monomios de varias ordens:

$$K(x, x_j) = 1 + x_1^2 x_{j1}^2 + 2x_1 x_2 x_{j1} x_{j2} + x_2^2 x_{j2}^2 + 2x_1 x_{j1} + 2x_2 x_{j2}$$

O vetor de entrada x introduzido no espaço de características é então deduzido sendo:

$$\varphi(x) = 1, x_1^2, \sqrt{2}x_1 x_2, x_2^2 + \sqrt{2}x_1 + \sqrt{2}x_2$$

Com base nesta função de mapeamento, a função objetivo pode ser derivada da equação (A.3), como mostrado a seguir:

$$\begin{aligned} \varphi(\alpha) &= \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 \\ &- \frac{1}{2}(9\alpha_1^2 - 2\alpha_1\alpha_2 - 2\alpha_1\alpha_3 + 2\alpha_1\alpha_4 \\ &+ 9(9\alpha_2^2 - 2\alpha_2\alpha_3 - 2\alpha_2\alpha_4 + 9\alpha_3^2 - 2\alpha_3\alpha_4 + \alpha_4^2)) \end{aligned}$$

Otimizando $\varphi(\alpha)$ em relação aos multiplicadores de Lagrange, gera-se o seguinte conjunto de equações:

$$9\alpha_1 - \alpha_2 - \alpha_3 + \alpha_4 = 1$$

$$-\alpha_1 + 9\alpha_2 + \alpha_3 - \alpha_4 = 1$$

$$-\alpha_1 + \alpha_2 + 9\alpha_3 - \alpha_4 = 1$$

$$9\alpha_1 - \alpha_2 - \alpha_3 + 9\alpha_4 = 1$$

Assim, os valores ideais dos multiplicadores de Lagrange são:

$$9\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = \frac{1}{8}$$

Este resultado indica que os quatro vetores de entrada são vetores de suporte e o valor ótimo de $\varphi(\alpha)$ é:

$$Q(\alpha) = \frac{1}{4}$$

e

$$\frac{1}{2}\|w\|^2 = \frac{1}{4}, \quad \text{ou} \quad \|w\| = \frac{1}{\sqrt{2}}$$

A partir da equação (A.3) encontra-se o vetor de peso ideal:

$$w = \frac{1}{8} [-\varphi(x_1) + \varphi(x_2) + \varphi(x_3) - \varphi(x_4)]$$

$$w = \frac{1}{8} \left[- \begin{bmatrix} 1 \\ 1 \\ \sqrt{2} \\ 1 \\ -\sqrt{2} \\ -\sqrt{2} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ -\sqrt{2} \\ 1 \\ -\sqrt{2} \\ \sqrt{2} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ -\sqrt{2} \\ 1 \\ \sqrt{2} \\ -\sqrt{2} \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ \sqrt{2} \\ 1 \\ \sqrt{2} \\ \sqrt{2} \end{bmatrix} \right] = + \begin{bmatrix} 0 \\ 0 \\ -\frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Neste caso, o hiperplano ótimo pode ser descrito como:

$$w \cdot \varphi(x) = \begin{bmatrix} 0 & 0 & \frac{-1}{\sqrt{2}} & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x_1^2 \\ \sqrt{2}x_1x_2 \\ x_2^2 \\ \sqrt{2}x_1 \\ \sqrt{2}x_2 \end{bmatrix} = 0$$

O qual pode ser reduzido a:

$$-x_1x_2 = 0$$

O hiperplano ótimo $-x_1x_2 = 0$ é a solução para o problema do operador XOR. Ele faz com que o resultado seja $y=1$ para ambos os pontos de entrada $x_1 = x_2 = 1$ e

$x_1 = x_2 = -1$, e $y = -1$ para ambos os pontos de entrada $x_1 = 1, x_2 = -1$ e $x_1 = -1, x_2 = 1$. A figura A.3 representa os quatro pontos no espaço de características transformado.

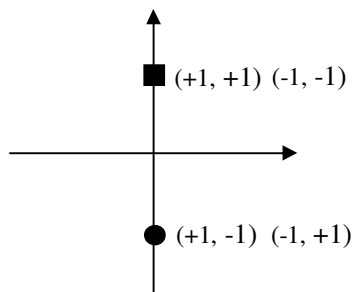


Figura A.3 – Problema do operador XOR no espaço de características transformado.