



UNIVERSIDADE METODISTA DE PIRACICABA
FACULDADE DE CIÊNCIAS EXATAS E DA NATUREZA
MESTRADO EM CIÊNCIA DA COMPUTAÇÃO

AMBIENTE COLABORATIVO COM REALIDADE AUMENTADA

RÉRYKA RUBIA P. C. L. DA SILVA

ORIENTADOR: PROF. DR. CLAUDIO KIRNER

PIRACICABA, SP
2006



UNIVERSIDADE METODISTA DE PIRACICABA
FACULDADE DE CIÊNCIAS EXATAS E DA NATUREZA
MESTRADO EM CIÊNCIA DA COMPUTAÇÃO

AMBIENTE COLABORATIVO COM REALIDADE AUMENTADA

RÉRYKA RUBIA P. C. L. DA SILVA

ORIENTADOR: PROF. DR. CLAUDIO KIRNER

Dissertação apresentada ao Mestrado em Ciência da Computação, da Faculdade de Ciências Exatas e da Natureza, da Universidade Metodista de Piracicaba – UNIMEP, como requisito para obtenção do Título de Mestre em Ciência da Computação.

PIRACICABA, SP
2006

AMBIENTE COLABORATIVO COM REALIDADE AUMENTADA

AUTOR: RÉRYKA RUBIA P. C. L. DA SILVA

ORIENTADOR: CLAUDIO KIRNER

Dissertação de Mestrado defendida e aprovada em 19 de abril de 2006, pela Banca Examinadora constituída dos Professores:

Prof. Dr. Claudio Kirner (orientador)

UNIMEP

Prof. Dr. Marcio Merino Fernandes

UNIMEP

Prof. Dr. Jorge Luis e Silva

ICMC-USP

DEDICATÓRIA

*A Deus, meu Pai e Amigo de todas as horas,
pelo cuidado comigo; aos meus Pais,
Raimundo e Ivany, por sempre acreditarem
em mim; ao meu marido André pelo amor e
companheirismo; ao meu irmão Rylber e
amigos, pelo incentivo; e ao meu filho Érik
que, ao me olhar com aqueles olhinhos,
lembrava-me de que tudo valia a pena.*

AGRADECIMENTOS

Chegou a hora de agradecer àquelas pessoas que, de alguma forma, contribuíram para a realização deste trabalho. Pessoas que me incentivaram de diferentes maneiras. A todos vocês o meu sincero “MUITO OBRIGADO”!

A UNIMEP (Universidade Metodista de Piracicaba), pela oportunidade de fazer esse mestrado.

Ao Orientador Prof. Dr. Cláudio Kirner, pelo apoio, conhecimento passado e, principalmente pelo incentivo e exemplo.

Aos Professores do Mestrado em Ciência da Computação da FACEN – UNIMEP, pelos ensinamentos.

Ao colega Rafael Santin, pela ajuda no desenvolvimento das aplicações com Realidade Aumentada.

Aos colegas de curso, pela amizade e pelas preciosas trocas de idéias.

Aos meus amigos de Piracicaba, Ariane, Renata e principalmente Andréa e sua família pelo carinho e cuidado, me acolhendo em situações de necessidade.

AMBIENTE COLABORATIVO COM REALIDADE AUMENTADA

RESUMO

Colaboração pressupõe trabalhar junto, ou seja, um necessita da existência do outro e de uma interação com este outro. O trabalho cooperativo pode permitir a instauração de formas mais participativas na realização de tarefas conjuntas, onde, necessariamente, os parceiros interagem entre si, compartilhando informações que podem ser textuais, visuais, sonoras e até mesmo táteis acerca e sobre o objeto trabalhado. Este trabalho tem por objetivo a elaboração de um sistema de comunicação colaborativo em rede no qual os usuários podem estar face a face e mediado pelo computador, ou então, separados geograficamente utilizando uma nova tecnologia chamada Realidade Aumentada. Esta tecnologia é uma variação de um sistema de Realidade Virtual, que aumenta o mundo real com objetos virtuais gerados por computador. A Realidade Aumentada Colaborativa pode ser explorada em diversos ambientes, como em tarefas de aprendizado colaborativo, jogos em rede, potencializar a visualização e o estudo de dados complexos como sistemas de engenharia, arquitetura, sistemas médicos, entre outros. O suporte colaborativo do protótipo realizado permite que diversos tipos de aplicações rodem sobre ele. Como exemplo, foi desenvolvido um jogo parecido com o dominó onde os usuários utilizam as mãos para movimentar as peças do jogo deixando o ambiente bastante intuitivo.

PALAVRAS-CHAVE: Realidade Aumentada Colaborativa, Trabalho Cooperativo Suportado por Computador, Realidade Virtual, Realidade Aumentada.

AUGMENTED REALITY COLLABORATIVE ENVIRONMENTS

Abstract

Collaboration presupposes to work together, when a person needs the existence of the other to one interaction. The cooperative work stimulates more participative forms in the accomplishment of group tasks, where necessarily, the partners interact to each other, sharing textual, visual, sound information and even touch, related as supplied object. The objective of this work is the development of a collaborative system through the network, in which the users can accomplish interaction face the face or remote, though the computer, using, in both cases, a new technology denominated Augmented Reality to allow the manipulation of virtual objects, using the hands in an intuitive way. The Augmented Reality, considered a variation of the virtual reality, increases the real world with virtual objects modeled by computer, using some technological device. The collaborative Augmented Reality allows the local and remote user's interaction; it can be used in several environments (places), involving collaborative learning, games using the network, visualization and study of complex data as engineering systems, architecture, medical systems, among others. The collaborative support allows the development of many types of applications. To test the system, a game similar to domino was developed, where the users use the hands to move markers cards, representing the pieces of the game, with the help of the network support, letting the environment quite intuitive.

KEYWORDS: Collaborative Augmented Reality, Computer Supported Cooperative Work, Virtual Reality, Augmented Reality.

SUMÁRIO

LISTA DE FIGURAS	IX
LISTA DE TABELAS	XII
LISTA DE QUADROS	XIII
LISTA DE ABREVIATURAS E SIGLAS	XIV
CAPÍTULO 1	1
1. INTRODUÇÃO	1
1.1. CARACTERIZAÇÃO DO PROBLEMA	1
1.2. MOTIVAÇÃO	3
1.3. DELIMITAÇÃO DO ASSUNTO	3
1.4. OBJETIVOS DA PESQUISA.....	3
1.5. METODOLOGIA	4
1.6. ESTRUTURA DA DISSERTAÇÃO.....	5
CAPÍTULO 2	6
2. REALIDADE VIRTUAL E REALIDADE AUMENTADA	6
2.1. REALIDADE VIRTUAL	6
2.1.1. IDÉIA DE IMERSÃO, NÃO IMERSÃO	9
2.1.2. APLICAÇÕES DE RV.....	15
2.2. REALIDADE AUMENTADA	20
2.2.1. TIPOS DE SISTEMAS DE REALIDADE AUMENTADA.....	21
2.3. POR QUE REALIDADE MISTURADA?	24
2.4. APLICAÇÕES EM RA	25
2.4.1. APLICAÇÕES MÉDICAS	25
2.4.2. APLICAÇÕES EDUCACIONAIS.....	26
2.5. REALIDADE VIRTUAL X REALIDADE AUMENTADA.....	26
2.6. PROBLEMAS TÍPICOS DE UM SISTEMA DE REALIDADE MISTURADA	27
CAPÍTULO 3	29
3. TRABALHO COOPERATIVO SUPORTADO POR COMPUTADOR	29
3.1. ESPAÇO/TEMPO	29
3.1.1. PREVISIBILIDADE	30
3.1.2. TAMANHO DO GRUPO	31

3.2. TRABALHOS COLABORATIVOS COM REALIDADE AUMENTADA.....	32
3.2.1. SHAREDSPACE	32
3.3. CONSTRUCT3D.....	33
3.3.1. JOGOS EDUCACIONAIS EM AMBIENTES DE REALIDADE AUMENTADA	34
3.3.2. MARE - MULTIUSER AUGMENTED REALITY ENVIRONMENT	34
CAPÍTULO 4	37
4. MECANISMOS DE COMUNICAÇÃO.....	37
4.1. SOQUETES	37
4.1.1. SOQUETES DE BERKELEY	39
4.2. RPC - REMOTE PROCEDURE CALL.....	41
4.3. CORBA.....	43
CAPÍTULO 5.....	45
5. ARTOOLKIT	45
5.1. EXECUTANDO O SOFTWARE	48
CAPÍTULO 6	51
6. SUPORTE COLABORATIVO PARA APLICAÇÕES DE REALIDADE AUMENTADA	50
6.1. AMBIENTE COLABORATIVO	50
6.2. PRIMEIROS TESTES DA APLICAÇÃO.....	54
6.2.1. MANIPULAÇÃO DO AMBIENTE	56
6.3. TESTES DA APLICAÇÃO FINAL.....	65
6.3.1. MANIPULAÇÃO DO AMBIENTE	67
CAPÍTULO 7	68
7. CONCLUSÃO.....	68
7.1. TRABALHOS FUTUROS	70
REFERÊNCIAS BIBLIOGRÁFICAS.....	71
ANEXOS.....	76
ANEXO 1. EXEMPLO DO CÓDIGO VRML, DA PEÇA DE DOMINÓ DE COR AMARELA E AZUL.	77
ANEXO 2. EXEMPLO DE UM ARQUIVO “VRML.DATA”, COM NOVOS MARCADORES INSERIDOS.	78

LISTA DE FIGURAS

FIGURA 1 - EXEMPLO DE UMA CAVE (LABORATÓRIO DE SISTEMAS INTEGRÁVEIS, ESCOLA POLITÉCNICA DA USP, 2005).....	10
FIGURA 2. EXEMPLO DE LUVAS UTILIZADAS EM SISTEMAS DE RV (FIFTH DIMENSION TECHNOLOGIES, 2005).....	10
FIGURA 3. EXEMPLO DE UM CAPACETE USADO EM SISTEMAS DE RV (FIFTH DIMENSION TECHNOLOGIES, 2005).....	11
FIGURA 4. ÓCULOS ESTEREOSCÓPICOS (ÓCULOS CRYSTAL EYES - ABSOLUT TECHNOLOGIES, 2006).....	12
FIGURA 5. MONITOR TRIDIMENSIONAL. (MONITOR SYNTHAGRAM 222 - ABSOLUT TECHNOLOGIES, 2006).....	13
FIGURA 6. SISTEMA DE PROJEÇÃO EM ESTAÇÃO DE TRABALHO, LUVA E ÓCULOS ESTEREOSCÓPICOS (LABORATÓRIO DE RV – UNIMEP, 2006).....	13
FIGURA 7. SISTEMA DE PROJEÇÃO COM ÓCULOS POLARIZADOS (LABORATÓRIO DE RV – UNIMEP, 2006).....	14
FIGURA 8. SIMULAÇÃO DE CIRURGIA EM REALIDADE VIRTUAL (SABBATINI, 2006)...	16
FIGURA 9. VISÃO DOS MODELOS DO PROJETO PROFESSOR VIRTUAL (KIRNER, 1998).	18
FIGURA 10. AMBIENTE PARA TRATAMENTO DE PESSOAS QUE POSSUI MEDO DE DIRIGIR (MENEZES, 2005).	18
FIGURA 11. AMBIENTE PARA TRATAMENTO DE PESSOAS QUE POSSUI MEDO DE LUGARES FECHADOS (MENEZES, 2005).....	18
FIGURA 12. SISTEMA DELTA 9 SENDO UTILIZADO POR UM PARTICIPANTE DA FEIRA DA EMPRESA ABSOLUT TECHNOLOGIES (SVR, 2004).....	19
FIGURA 13. CAMINHO DE UM AMBIENTE REAL ATÉ UM AMBIENTE VIRTUAL (MILGRAM, 1994).....	20
FIGURA 14. APLICAÇÃO DE REALIDADE AUMENTADA.....	21
FIGURA 15. DIAGRAMA CONCEITUAL DO SISTEMA DE RA DE VISÃO ÓTICA DIRETA (AZUMA, 1997).	22
FIGURA 16. DIAGRAMA CONCEITUAL DE UM SISTEMA DE RA DE VISÃO DIRETA POR VÍDEO (AZUMA, 1997).....	23
FIGURA 17. DIAGRAMA DE UM SISTEMA DE RA DE VISÃO POR VÍDEO BASEADO EM MONITOR (BRAZ, 2000).	23
FIGURA 18. UTILIZANDO O SISTEMA AR WINDOW (MEDARPA, 2005).....	25
FIGURA 19. LIVRO: SÓLIDOS GEOMÉTRICOS COM REALIDADE AUMENTADA (APLICAÇÃO DINÂMICA), POR DANIELA AKAGUI E RAFAEL SANTIN (KIRNER, 2004).....	26

FIGURA 20. NA PARTE A, OS OBJETOS VIRTUAIS ESTÃO DESALINHADOS COM O MUNDO REAL. NA PARTE B OS DOIS MUNDOS ESTÃO SINCRONIZADOS (KIRNER, 2004B).	28
FIGURA 21. EFEITO DE OCLUSÃO (RAPOSO, 2004).	28
FIGURA 22 – CLASSIFICAÇÃO DE UM GROUPWARE QUANTO AO ESPAÇO E TEMPO (GRUDI, 1994)	30
FIGURA 23 – CLASSIFICAÇÃO TEMPO/ESPAÇO/PREVISÍVEL (GRUDIN, 1994).	31
FIGURA 24 – CLASSIFICAÇÃO TEMPO/ESPAÇO/TAMANHO (GRUDIN, 1994)	31
FIGURA 25. VISÃO DE QUEM ESTÁ DE FORA (BILLINGHURST, 1996).	33
FIGURA 26. VISÃO DO USUÁRIO NO SISTEMA SHAREDSPACE (BILLINGHURST, 1996).	33
FIGURA 27. ESTUDANTES TRABALHANDO COM O CONSTRUCT3D. (KAUFMANN, 2004).	34
FIGURA 28. AMBIENTE AUMENTADO DE PROJETOS DE ARQUITETURA (GASCUEL, 2002).	35
FIGURA 29 – AMBIENTE AUMENTADO DE VISUALIZAÇÕES CIENTÍFICAS (GASCUEL, 2002).	36
FIGURA 30. EXEMPLO DE UM OBJETO VIRTUAL (TORRE) ALINHADO COM A CENA REAL (HITLAB, 2004).	45
FIGURA 31. AMBIENTE DE REALIDADE AUMENTADA UTILIZANDO ARTOOLKIT (BILLINGHURST, 2000).	46
FIGURA 32. RESUMO DOS PASSOS REALIZADOS PELO ARTOOLKIT (HITLAB, 2004).	47
FIGURA 33. PADRÃO HIRO.	48
FIGURA 34. ESQUEMA GERAL DO AMBIENTE COLABORATIVO	51
FIGURA 35. ESQUEMA DE FUNCIONAMENTO DO AMBIENTE COLABORATIVO EM REDE.	52
FIGURA 36 – CLASSIFICAÇÃO DO SISTEMA QUANTO AO ESPAÇO E TEMPO.	53
FIGURA 37. CUBOS VIRTUAIS UTILIZADOS PARA O PRIMEIRO EXPERIMENTO.	55
FIGURA 38. USUÁRIO 1 NO AMBIENTE EM REDE.	55
FIGURA 39. USUÁRIO 2 NO AMBIENTE EM REDE.	55
FIGURA 40. COPIANDO UM OBJETO VIRTUAL COM O MARCADOR “PÁ”.	58
FIGURA 41. TRANSPORTANDO O OBJETO COM O MESMO MARCADOR.	58
FIGURA 42. FIXANDO O OBJETO ATRAVÉS DA TECLA F DO TECLADO.	60
FIGURA 43. MARCADOR DELETOR PRÓXIMO DO OBJETO A SER EXCLUÍDO.	62
FIGURA 44. CENA COM O OBJETO EXCLUÍDO.	62
FIGURA 45. INTERFACE DO SISTEMA CLIENTE PARA ATIVAR A CONEXÃO COM O SERVIDOR.	65
FIGURA 46. INTERFACE DO SISTEMA SERVIDOR RECEBENDO OS DADOS DOS SISTEMAS CLIENTES.	65

FIGURA 47. MARCADORES DA APLICAÇÃO FINAL	66
FIGURA 48. OBJETOS VIRTUAIS SOBRE SEUS MARCADORES, SIMULANDO UM JOGO DE DOMINÓS.	66
FIGURA 49. VISÃO DO JOGADOR QUE INSERIU A PEÇA BRANCA E AZUL.	66
FIGURA 50. VISÃO DO JOGADOR QUE INSERIU A PEÇA VERDE E BRANCA.	66

LISTA DE TABELAS

TABELA 1. PRIMITIVAS BÁSICAS PARA UM SERVIÇO DE TRANSPORTE SIMPLES.....	38
TABELA 2 AS PRIMITIVAS DE SOQUETES PARA TCP.	40

LISTA DE QUADROS

QUADRO 1. ALGORITMO QUE DEMONSTRA O RECEBIMENTO DOS DADOS PELO CLIENTE, ENVIADOS PELO SERVIDOR.	52
QUADRO 2A. CÓDIGO DA FUNÇÃO “CÓPIA” DO OBJETO VIRTUAL.	58
QUADRO 2B. CONTINUAÇÃO DO QUADRO 2A.	59
QUADRO 3A. CÓDIGO DA FUNÇÃO “FIXAR” O OBJETO VIRTUAL NA CENA.	60
QUADRO 3B. CONTINUAÇÃO DO QUADRO 3A.	61
QUADRO 4A. CÓDIGO DA FUNÇÃO “APAGAR” UM OBJETO VIRTUAL DA CENA.	62
QUADRO 4B. CONTINUAÇÃO DO QUADRO 4A.	63
QUADRO 4C. CONTINUAÇÃO DO QUADRO 4B.	64

LISTA DE ABREVIATURAS E SIGLAS

3D	Tridimensional
AV	Ambientes Virtuais
AVC	Ambientes Virtuais Colaborativos
CAVE	<i>Cave Automatic Virtual Environment</i>
CSCCL	<i>Computer Supported Collaborative Learning</i>
CSCW	<i>Computer Supported Cooperative Work</i>
DII	<i>Dynamic Invocation Interface</i>
DIVE	<i>Distributed Interactive Virtual Environments</i>
E/S	Entrada/Saída
FTP	<i>File Transfer Protocol</i>
HITLab NZ	<i>Human Interface Technology Laboratory – Nova Zelândia</i>
HITLab	<i>Human Interface Technology Laboratory</i>
HMD	<i>Head Mounted Display</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IDL	<i>Interface Definition Language</i>
IP	<i>Internet Protocol</i>
OMG	<i>Object Management Group</i>
ORB	<i>Object Request Broker</i>
RA	Realidade Aumentada
RAM	<i>Random Access Memory</i>
RM	Realidade Misturada
RV	Realidade Virtual
TCP	<i>Transmission Control Protocol</i>
UDP	<i>User Datagram Protocol</i>
VRML	<i>Virtual Reality Modeling Language</i>

CAPÍTULO 1

1. INTRODUÇÃO

1.1. CARACTERIZAÇÃO DO PROBLEMA

Vivemos em tempos em que muitas informações estão disponíveis e ao alcance de nossas mãos fator que aumenta a exigência de aprimorar o conhecimento em condições de temporalidade mais reduzidas.

O trabalho colaborativo vem crescendo juntamente com esta realidade. Ao se trabalhar em conjunto na realização de uma determinada tarefa podemos agrupar conhecimentos distintos, exigindo menos de cada indivíduo e mais do coletivo.

Dada a acentuada aceleração da dinâmica da sociedade contemporânea, praticamente em todas as áreas do conhecimento, observa-se que está se aprimorando, e muito, a ação do cooperativismo. Ninguém pode deter todo o conhecimento de uma área, mas pode ser especialista em uma fatia da mesma, enquanto detém o conhecimento de outra fatia. Devido a este fator, há a necessidade de gerar o trabalho cooperativo, no qual, todos ajudam a alcançar um objetivo.

Atualmente não são poucos os esforços para melhorar os meios de cooperação. Tecnicamente, surgem a cada instante, maneiras diferentes e melhores de comunicação entre pessoas que estão geograficamente separadas.

A Internet tem sido o suporte para a maioria dos meios de comunicação, utilizando sistemas como e-mails, *chats*, sites com recursos de áudio e vídeo que aproximam os usuários.

Com vistas a essa dinâmica social e tecnológica, este trabalho visa à elaboração de mais um sistema de comunicação colaborativo utilizando uma nova tecnologia chamada Realidade Aumentada (RA).

A Realidade Aumentada é uma variação de um sistema de Realidade Virtual que mistura objetos virtuais gerados por computador, com cenas do mundo real, proporcionando uma forma natural e lúdica na manipulação desses objetos virtuais.

A Realidade Aumentada vem despertando um grande interesse em pesquisadores do mundo inteiro, grupos de trabalho de várias áreas têm explorado esta nova tecnologia. Seu uso na engenharia, na medicina, na educação e em entretenimento, são alguns exemplos de áreas que se beneficiam da RA.

Sistemas de RA acrescidos de um sistema de suporte à colaboração permitiram a instauração de formas mais participativas na realização de tarefas conjuntas cujos parceiros, interagindo entre si, compartilham informações que podem ser textuais, visuais, sonoras e até mesmo táteis acerca e sobre o objeto trabalhado.

Para haver interação entre pessoas em um sistema computacional é necessário que uma comunicação seja estabelecida, portanto, o ambiente colaborativo suportado por computador deve possibilitar esta comunicação baseado em tecnologias de informação e de comunicação.

Este projeto possibilitará a interação entre pessoas, geograficamente distribuídas, sobre um ambiente de realidade aumentada no qual são utilizados recursos computacionais para viabilizar, intensificar ou mesmo armazenar os produtos decorrentes desta interação como imagens, mensagens, arquivos sonoros e outros.

O suporte colaborativo permitirá ainda que um ambiente de RA possa ser compartilhado entre dois ou mais usuários dispostos remotamente na rede de computadores.

1.2. MOTIVAÇÃO

A motivação para a realização deste trabalho foi elaborar um novo meio de comunicação interativo, dinâmico, estimulante e que possibilite a cooperação ou colaboração entre os seus usuários sobre um sistema de RA.

Utilizando a Realidade Aumentada teremos um ambiente altamente estimulante, pois, os sentidos visuais são os mais envolvidos, porém, usá-la como um meio de execução de uma determinada tarefa não tiraria o interessado da solidão, desestimulando o seu uso freqüente. Partindo deste princípio, visamos viabilizar a colaboração de um sistema de RA permitindo que múltiplos usuários dispostos remotamente (em diferentes máquinas) possam interagir entre si, utilizando a Realidade como a interface desta comunicação, visando ao baixo custo da aplicação e, conseqüentemente, aumentando o interesse pelo ambiente em diversas áreas de aplicações.

1.3. DELIMITAÇÃO DO ASSUNTO

Este trabalho discorrerá sobre a construção de um suporte colaborativo em rede, onde os usuários utilizarão formas intuitivas de manipulação do ambiente, possibilitando a colaboração entre usuários em um sistema de Realidade Aumentada. Para o desenvolvimento do protótipo foram utilizados conceitos sobre soquetes, linguagem de modelagem de objetos virtuais, linguagem de programação C++ e conhecimentos do software ARToolkit.

1.4. OBJETIVOS DA PESQUISA

Este trabalho tem como objetivo elaborar um sistema de Realidade Aumentada Colaborativo utilizando redes de computadores que viabilizem vários tipos de aplicações remotas. O sistema integrará técnicas de realidade aumentada, comunicação entre processos através de soquetes e manipulação de objetos tridimensionais.

O trabalho visa ao desenvolvimento de um protótipo, que abrange um sistema de RA encarregado de exibir dados tridimensionais e que, através de programação distribuída, permita enviar dados entre usuários. Com o ambiente aumentado colaborativo já em funcionamento, foram criadas formas de manipulação destes objetos virtuais nos quais os usuários podem movimentá-los e fixá-los na posição que desejarem.

Com isso o ambiente está pronto para receber vários tipos de aplicações como, por exemplo: um jogo, a construção de um ambiente educacional, de treinamento e quaisquer outros dados passíveis de serem mostrados pelo ARToolkit.

Os objetivos específicos pretendidos na pesquisa foram:

- a. Gerar um ambiente de realidade aumentada para aplicações colaborativas em rede;
- b. criar técnicas de manipulação dos objetos virtuais dispostos na cena;
- c. desenvolver uma aplicação colaborativa no ambiente como exemplo.

1.5. METODOLOGIA

Para realização deste trabalho foram feitas além de revisões bibliográficas em sites, livros e artigos sobre realidade virtual, realidade aumentada, trabalho colaborativo suportado por computador, ambientes colaborativos de realidade aumentada, software ARToolkit, C++ e programação distribuída; também realizaram-se algumas adaptações de softwares, desenvolvimentos de interações e implementação de exemplos testes.

Com isso foi possível avaliar o funcionamento do sistema e discutir suas vantagens, desvantagens, problemas e outras características.

Na realização dos testes do protótipo, foram utilizadas duas webcams Genius Modelo: VideoCam NB e dois computadores Pentium IV 2.8 Mhz, com 512 Mb de Memória RAM.

1.6. ESTRUTURA DA DISSERTAÇÃO

Este trabalho está dividido em seis capítulos.

O primeiro capítulo descreve uma introdução do trabalho com seu respectivo objetivo e metodologia.

O segundo, o terceiro, o quarto e o quinto capítulos apresentam a fundamentação teórica sobre realidade virtual e realidade aumentada, trabalhos cooperativos suportados por computador, mecanismos de comunicação em redes e, por fim, discorre sobre o Software ArToolkit.

O sexto capítulo apresenta a elaboração e a execução do protótipo de RA através do relato das técnicas e do equipamento computacional utilizados no seu desenvolvimento.

Por fim, são apresentadas a conclusão e as referências bibliográficas.

CAPÍTULO 2

2. REALIDADE VIRTUAL E REALIDADE AUMENTADA

2.1. REALIDADE VIRTUAL

A Realidade Virtual (RV) vem trazer ao uso do computador um novo paradigma de interface com o usuário. Neste paradigma o usuário não estará mais em frente ao monitor, mas sim, sentir-se-á dentro da interface. Com dispositivos especiais, a Realidade Virtual busca captar os movimentos do corpo do usuário (em geral braços, pernas, cabeça e olhos) (KIRNER, 2001) e, a partir destes dados, realizar a interação homem-máquina.

A interface com realidade virtual envolve um controle tridimensional altamente interativo de processos computacionais. O usuário entra no espaço virtual das aplicações e visualiza, manipula e explora os dados da aplicação em tempo real, usando seus sentidos, particularmente os movimentos naturais tridimensionais do corpo. A grande vantagem desse tipo de interface é que o conhecimento intuitivo do usuário a respeito do mundo físico pode ser transferido para manipular o mundo virtual. Para suportar esse tipo de interação, o usuário utiliza dispositivos não convencionais como capacete de visualização e controle, luva, e outros. Estes dispositivos dão ao usuário a impressão de que a aplicação está funcionando no ambiente tridimensional real, permitindo a exploração do ambiente e a manipulação natural dos objetos com o uso das mãos, por exemplo.

Podemos dizer que a RV é uma tecnologia de interface avançada entre um usuário e um sistema computacional. O objetivo dessa tecnologia é recriar ao máximo a sensação de realidade para um indivíduo, levando-o a adotar essa interação como uma de suas realidades temporais.

Para isso, essa interação é realizada em tempo real, com o uso de técnicas e de equipamentos computacionais que ajudem na ampliação do sentimento de presença do usuário dentro do ambiente virtual.

A Realidade Virtual é uma área de crescente avanço. Esse crescimento é devido a alguns fatores que muito contribuíram para o desenvolvimento da RV, como: a colocação no mercado de computadores de uso pessoal com boas capacidades gráficas e custos reduzidos, aperfeiçoamento e desenvolvimento de novos dispositivos de interfaces cada vez mais leves e portáteis o que também ajudam no avanço desta tecnologia.

Existem diversas definições na literatura em relação à Realidade Virtual. De maneira simplificada pode ser definida como a forma mais avançada de interface do usuário de computador até então disponível (HANCOCK, 1995).

Realidade virtual é uma forma de as pessoas visualizarem, manipularem e interagirem com computadores e dados extremamente complexos (AUKSTAKALNIS, 1992).

Trata-se de uma interface homem-computador que simula um ambiente real e permite aos participantes interagirem com os mesmos (LATTA & OBERG, 1994).

A realidade virtual também pode ser definida como o uso de computadores e interfaces homem-máquina, para criar o efeito de mundos tridimensionais que possuam objetos nas quais possam ser interativos, que respondam a um comando do usuário (BRYSON, 1996).

Este tipo de interface é altamente estimulante, pois pode proporcionar ao usuário estímulos visuais, sonoros e táteis onde ele explora dados desta aplicação, interagindo, em tempo real, com o mesmo.

A grande vantagem desse tipo de interface é que o conhecimento intuitivo do usuário a respeito do mundo físico pode ser

transferido para manipular o mundo virtual. Para suportar esse tipo de interação, o usuário utiliza dispositivos não convencionais como capacete de visualização e controle, e luvas de dados, as chamadas *datagloves* (EARNSHAW, 1995).

Situações antes impossíveis de se experimentar na vida real, com a RV se tornam possíveis. Como navegar em uma célula e ver as suas reações podendo também interagir com a mesma, mudando suas características, formas de visualização e inserir novos dados.

Existem dois tipos de envolvimento do homem com o mundo virtual: o passivo e o ativo. Um exemplo de envolvimento passivo é ler um livro ou ver televisão. O envolvimento ativo pode ser exemplificado, quando se participa de um jogo com um parceiro. A realidade virtual tem potencial para os dois tipos de envolvimento ao permitir a exploração de um ambiente virtual e ao propiciar a interação do usuário com um mundo virtual dinâmico (KIRNER, 1996).

A interação ocorre através de dispositivos que capturam e/ou mostram as ações do usuário, como capacetes, rastreadores, luvas, óculos, que proporcionam um envolvimento maior com o mundo, dando maior realismo à cena.

“Todos estes dispositivos dão ao usuário a impressão de que a aplicação está funcionando no ambiente tridimensional real, permitindo a exploração do ambiente e a manipulação natural dos objetos com o uso das mãos, por exemplo, para apontar, pegar, e realizar outras ações” (KIRNER, 1996).

Segundo Kirner (1996), para a elaboração de sistemas de RV é necessário ter algum domínio sobre dispositivos não convencionais de E/S, computadores de alto desempenho e boa capacidade gráfica, sistemas paralelos e distribuídos, modelagem geométrica tridimensional, simulação em tempo real, navegação, detecção de colisão, avaliação, impacto social, projeto de interfaces, e aplicações simples e distribuídas em diversas.

Existem muitas fontes textuais que abordam os conceitos sobre realidade virtual e como foi visto anteriormente, são várias as definições aceitas. Isto é devido, em parte, à natureza interdisciplinar da área e também à sua evolução, pois de uma maneira ou de outra, os sistemas de realidade virtual acabaram vindo de sistemas computacionais de mesa, simuladores e sistemas de teleoperação. (KIRNER, 1996).

2.1.1. IDÉIA DE IMERSÃO, NÃO IMERSÃO

A realidade virtual pode ser considerada imersiva e não imersiva. A idéia de imersão se dá ao fato de o usuário se sentir dentro do ambiente virtual. A realidade virtual imersiva é baseada no uso de capacetes ou de salas de projeção nas paredes, as chamadas CAVEs (*Cave Automatic Virtual Environment*). Além do fator visual, os dispositivos ligados com os outros sentidos também são importantes para o sentimento de imersão, como: som (BEGAULT, 1994), posicionamento automático da pessoa e dos movimentos da cabeça e controles de reação. A realidade virtual não imersiva se baseia no uso de monitores, mas se usados com dispositivos que exploram outros sentidos, como os óculos ou luvas, eles proporcionam algum grau de imersão.

2.1.1.1. Dispositivos de RV Imersiva

Os equipamentos utilizados em sistemas de Realidade Virtual Imersivos estão cada vez mais sofisticados. Um bom exemplo é a caverna digital ou CAVE (*Cave Automatic Virtual Environment*). A CAVE é uma pequena sala onde são projetados gráficos em três dimensões em suas paredes. Esses gráficos podem ser visualizados pelas pessoas que estão dentro da caverna e, ao utilizarem outros dispositivos de RV, os usuários exploram e interagem com objetos e pessoas virtuais do ambiente, permitindo aos mesmos a sensação de estarem mergulhados no mundo virtual.

O funcionamento da CAVE é viabilizado através de vários computadores integrados com aparelhos de projeção conectados a eles e cujas

imagens são projetadas nas paredes das salas. Um Projetor é usado para cada parede que se deseja enviar a imagem, como mostra a Figura 1.



Figura 1 - Exemplo de uma CAVE (Laboratório de Sistemas Integráveis, Escola Politécnica da USP, 2005).

Um outro equipamento utilizado em sistemas de Realidade Virtual são as luvas (Figura 2), que através de sensores, detectam e medem as flexões e abduções dos dedos. Os sensores podem ser mecânicos, ópticos ou híbridos, gerando a sensação de toque, peso ou força, aumentando o grau de imersão na cena.



Figura 2. Exemplo de luvas utilizadas em sistemas de RV (Fifth Dimension Technologies, 2005).

Por exemplo, ao utilizar uma luva que emite a sensação de peso, o usuário, ao manipular um objeto virtual “pesado”, sentirá o peso real

em suas mãos, tornando mais dificultoso movimentá-lo, como acontece na vida real.

Quando não se está em uma CAVE, podemos aumentar a sensação de imersão no mundo virtual utilizando os capacetes (Figura 3) ou HMD (*Head Mounted Display*), no qual, são exibidas, em duas pequenas telas (uma para cada olho), imagens de uma cena virtual. Ele possui um sensor de posição absoluta da cabeça (rastreador) e muitos deles também possuem fones estereofônicos.

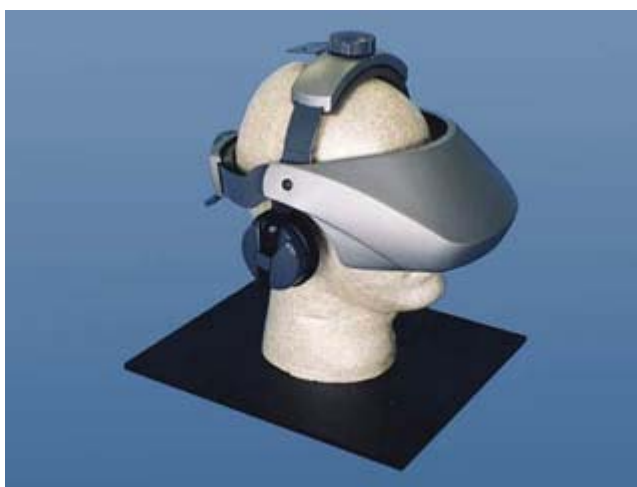


Figura 3. Exemplo de um capacete usado em Sistemas de RV (Fifth Dimension Technologies, 2005).

2.1.1.2. Dispositivos de RV Não Imersivos

Os sistemas não imersivos são aqueles nos quais o usuário vê o universo virtual por uma tela convencional de computador ou de projeção (PINHO, 2000). Embora, a imersão não seja total ainda assim pode-se sentir algum grau de imersão utilizando alguns equipamentos especiais para isso como: óculos estereoscópicos, monitores estereoscópicos, luvas especiais, rastreadores.

De acordo com a enciclopédia Wikipédia (2006), estereoscopia pode ser definida como:

“Um fenômeno natural que ocorre quando uma pessoa observa uma cena qualquer. A estereoscopia é a simulação de duas imagens da cena que são projetadas nos olhos em pontos de observação ligeiramente diferentes, o cérebro funde as duas imagens, e nesse processo, obtém informações quanto à profundidade, distância, posição e tamanho dos objetos, gerando uma sensação de visão de 3D”.

E os óculos estereoscópicos, de acordo com a mesma enciclopédia, tem o seguinte significado:

“São bastante comuns como instrumentos potencializadores da imersão. Os óculos estereoscópicos polarizados ou com lentes de cristal líquido possibilitam imersão parcial do usuário, que pode ter a sensação plena da tridimensionalidade do ambiente virtual”.

Estes óculos (Figura 4) reproduzem a maneira que a maioria das pessoas enxergam. No mundo real, cada olho possui uma perspectiva visual ligeiramente diferente de um objeto e é esta pequena diferença que fornece as sensações de profundidade no cérebro humano. Para a visualização de imagens 3D é necessário que existam duas imagens independentes, com uma perspectiva de cada olho. O sistema dos óculos permite a troca alternada das imagens em cada olho transformando a troca rápida de perspectiva na real sensação de profundidade. Esta troca entre o olho esquerdo e direito é feita em até 150 vezes por segundo, traduzindo-se em uma imagem 3D perfeita.



Figura 4. Óculos Estereoscópicos (Óculos CrystalEyes - Absolut Technologies, 2006).

Outro equipamento que gera um sistema parcial de imersão são os monitores estéreos (Figura 5). Eles têm as mesmas características dos

óculos, porém com estes monitores não é necessário o uso de óculos estereoscópicos, pois a própria tela do monitor é estéreo, gerando a sensação de profundidade.



Figura 5. Monitor tridimensional. (Monitor SynthaGram 222 - Absolut Technologies, 2006)

A Figura 6 ilustra um sistema semi-imersivo baseado em monitor. O usuário está utilizando um óculos estereoscópico e uma luva para manipular os dados tridimensionais que estão sendo mostrados no vídeo do computador.



Figura 6. Sistema de Projeção em Estação de Trabalho, luva e óculos estereoscópicos (Laboratório de RV – UNIMEP, 2006).

A Figura 7 mostra um sistema de RV que não é completamente imersivo. Neste sistema são utilizados óculos estereoscópicos onde as imagens são visualizadas em telas de projeção. Tal ambiente é útil em aplicações onde várias pessoas precisam observar a mesma imagem em estéreo.



Figura 7. Sistema de Projeção com óculos polarizados (Laboratório de RV – UNIMEP, 2006).

Estes óculos funcionam da seguinte maneira: exibe-se na tela a imagem correspondente a do olho esquerdo e bloqueia-se a visão do olho direito a seguir faz-se o contrário, ou seja, exibe-se a imagem do olho direito e bloqueia-se a visão do esquerdo; esse mecanismo é que chamamos de estereoscopia, ou seja, faz com que a imagem fique em três dimensões.

Portanto, para que um sistema de realidade virtual pareça mais realista não somente objetos simulados são inseridos, mas, outros artifícios são empregados, como por exemplo, a texturização dos objetos, a inserção de sons tanto ambientais quanto sons associados a objetos 3D, formas intuitivas de manipulação dos objetos, utilizando dispositivos específicos, e outras características que são essenciais na melhoria do ambiente (ARAÚJO, 1996).

2.1.2. APLICAÇÕES DE RV

Realidade Virtual pode ser aplicada nas mais variadas áreas do conhecimento e de maneira bastante diversificada. As aplicações que usam esta tecnologia se enquadram em vários campos como na Engenharia (Naval, Oceânica, Mecânica, Civil, Automobilística e Eletrônica), na medicina (simulações cirúrgicas, estudos em anatomia), nas ciências básicas (Astronomia, astrofísica, biologia e química) e finalmente no entretenimento (jogos, visualizações foto-realísticas e filmes interativos) (LSI –Laboratório de Sistemas Integráveis, Escola Politécnica da USP, 2005).

2.1.2.1. Aplicações Médicas

Na área médica a RV vem se desenvolvendo de uma maneira muito rápida e eficaz.

Já existem centenas de aplicações em Medicina e Biologia, particularmente para a área de ensino, que é uma vocação natural da RV em todo o mundo. A escassez de cadáveres e de pacientes para apoiar o ensino médico de anatomia, exames imagenológicos e invasivos, intervenções diagnósticas ou cirúrgicas, tem servido de incentivo para um maior desenvolvimento da RV na prática médica (SABBATINI, 2006).

Um bom exemplo de um sistema de RV médico dá-se quando um médico pode treinar cirurgias complexas e delicadas de microsuturas de vasos sanguíneos, usando um equipamento de realidade virtual que permite simular o uso de tesouras, pinças, e outros aparelhos médicos de verdade, sobre uma imagem virtual de um vaso sanguíneo (Figura 8). Eventualmente, essa aplicação poderá se transformar em uma telecirurgia, ou seja, uma cirurgia realizada à distância.

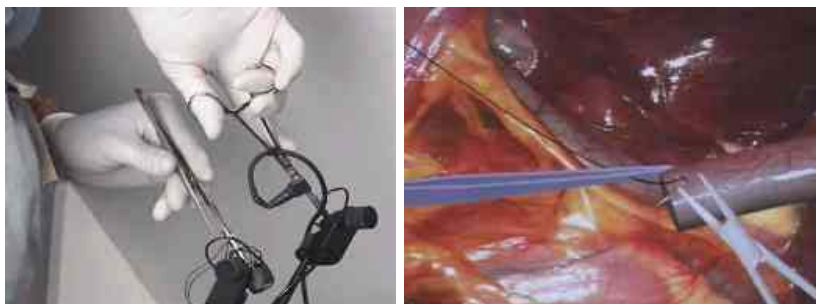


Figura 8. Simulação de cirurgia em realidade virtual (SABBATINI, 2006).

2.1.2.2. Aplicações Educacionais

Com a Realidade Virtual é possível estarmos em lugares distintos, mesmo que fisicamente não saíamos do mesmo local. Com ela se pode explorar dados que na vida real seria impossível ou muito difícil de conseguir, como por exemplo, átomos, o espaço e lugares subterrâneos ou ainda, manipular estruturas moleculares que na vida real só poderiam ser visualizadas. Ela nos permite explorar lugares onde seria muito difícil ou impossível de estarmos na vida real, seja pelo tamanho, pelo custo ou por este local não pertencer mais a nosso espaço atual.

Por isso a RV tem sido amplamente estudada para aplicações educacionais, dada sua capacidade de proporcionar aos estudantes formas mais participativas e intuitivas no processo de aquisição do conhecimento.

Vários autores concordam ao relatar que existem diversas razões para o uso a Realidade Virtual na educação dentre as quais destacamos (PANTELIDES, 1995):

- maior motivação dos estudantes (usuários);
- o poder de ilustração da realidade virtual para alguns processos e objetos é muito maior do que outras mídias;
- permite uma análise de muito perto;
- permite uma análise de muito longe;

- permite que pessoas com deficiência realizem tarefas que de outra forma não seriam possíveis;
- oportuniza experiências;
- permite que o aprendiz desenvolva o trabalho no seu próprio ritmo;
- não restringe o prosseguimento de experiências ao período da aula regular;
- permite que haja interação estimulando dessa forma participação ativa e coletiva do estudante.

Para Moran (2000):

"Educar é colaborar para que professores e alunos - nas escolas e organizações - transformem suas vidas em processos permanentes de aprendizagem... Uma mudança qualitativa no processo de ensino/aprendizagem acontece quando conseguimos integrar dentro de uma visão inovadora todas as tecnologias: as telemáticas, as audiovisuais, as textuais, as orais, musicais, lúdicas e corporais... É importante diversificar as formas de dar aula, de realizar atividades, de avaliar".

Como um exemplo podemos citar o Projeto Professor Virtual (KIRNER, 1998). Este sistema permite que o professor seja transportado para uma sala de aula remota e receber as informações da classe, podendo interagir com os alunos como se estivesse presente. Para isso foram utilizadas técnicas de realidade virtual, e em especial a telepresença, como mostra a Figura 9.

Além disso, o sistema oferecerá ferramentas adicionais que aumentarão as possibilidades de interação como, por exemplo, a identificação dos alunos (crachá virtual), prontuários e experimentos virtuais (laboratórios virtuais).

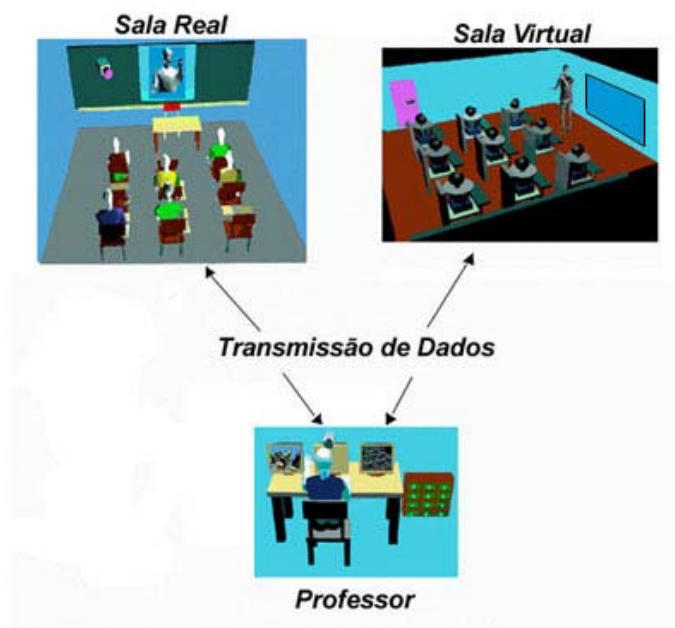


Figura 9. Visão dos modelos do Projeto Professor Virtual (KIRNER, 1998).

2.1.2.3. Terapia com Realidade Virtual

A Terapia com Realidade Virtual é uma técnica que permite a imersão e exposição gradual do cliente ao ambiente e ou situação temida e o equipamento que permite ao cliente vivenciar esta nova realidade é um capacete que envia as imagens e o áudio do ambiente virtual (Figuras 10 e 11). Como exemplo, podemos citar o trabalho da clínica do psicólogo Basileu Gomes de Menezes em Curitiba – PR (MENEZES, 2005).



Figura 10. Ambiente para tratamento de pessoas que possui medo de dirigir (MENEZES, 2005).



Figura 11. Ambiente para tratamento de pessoas que possui medo de lugares fechados (MENEZES, 2005).

Este sistema permite a:

- geração de estímulos para clientes com dificuldades para a criação de imagens mentais;
- geração de estímulos para clientes que não suportam a exposição à situação real;
- preservação da privacidade;
- geração de estímulos que ultrapassam as técnicas convencionais

2.1.2.4. Entretenimento

Várias aplicações na área de entretenimento estão sendo realizadas, como: turismos virtuais, jogos, cinema virtual, entre outros. O que você acharia de Conhecer o Rio de Janeiro, voando numa asa delta, mas sem sair do chão? Essa foi a proposta da empresa Absolut Technologies, cujo sistema está disponível tanto para venda como locação (Figura 12). O Delta 9 é direcionado para os segmentos de lazer como lojas de entretenimento, eventos, feiras, shoppings centers e aeroportos (Absolut Technologies, 2005). A Figura seguinte ilustra o Delta 9 em ação.



Figura 12. Sistema Delta 9 sendo utilizado por um participante da feira da empresa Absolut Technologies (SVR, 2004).

2.2. REALIDADE AUMENTADA

Realidade Aumentada (RA) é uma variação de Ambientes Virtuais (AV), ou Realidade Virtual (RV) como é mais comumente chamado. (AZUMA, 1997). É uma combinação da visão do ambiente real com o ambiente virtual (AZUMA, 1993).

De acordo com Azuma (1997), um sistema de realidade misturada possui três características principais:

- combina real com virtual;
- possui interação em tempo real;
- é visualizado em cenas tridimensionais.

Realidade Aumentada é uma particularização de um conceito mais geral denominado realidade misturada (KIRNER, 2005) e que consiste na sobreposição de ambientes reais e virtuais, em tempo real, através de técnicas de visão computacional.

A realidade misturada pode receber duas denominações: realidade aumentada e virtualidade aumentada. Temos a realidade aumentada quando os objetos reais presentes na cena são maioria, e virtualidade aumentada se caracteriza por possuir mais objetos virtuais que reais na cena.

Milgram et. al. (1994), descreve uma taxonomia sobre o caminho do Mundo Real se transformando em Mundo Virtual, mostrada na Figura 13.

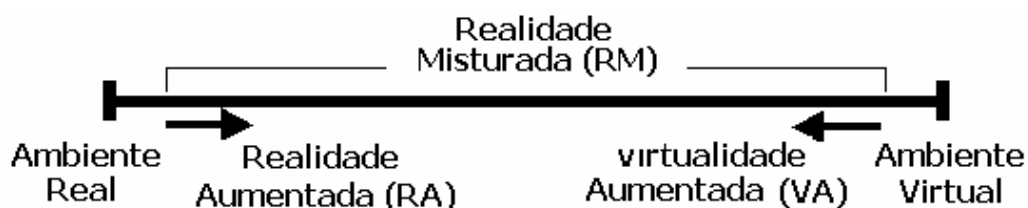


Figura 13. Caminho de um Ambiente Real até um Ambiente Virtual (Milgram, 1994).

Nas extremidades da figura 13 o autor apresenta o mundo real e o mundo virtual, à medida que os mundos se aproximam, um ambiente de realidade misturada é gerado, onde este pode ser dividido em dois: a Realidade Aumentada e a Virtualidade Aumentada. Quando o mundo misturado tem predominância de objetos reais, temos a Realidade Aumentada, quando os objetos virtuais são mais presentes e o mundo real diminui na cena, temos o que chamamos de Virtualidade Aumentada.

A Figura 14 mostra uma aplicação de realidade aumentada, onde pode ser observada a sobreposição dos objetos virtuais sobre um ambiente real.

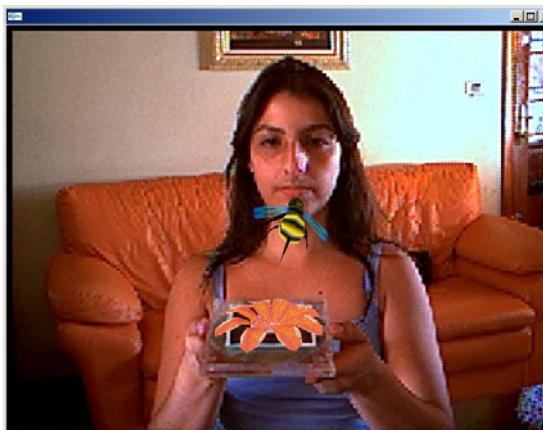


Figura 14. Aplicação de realidade aumentada.

2.2.1. TIPOS DE SISTEMAS DE REALIDADE AUMENTADA

Os sistemas de realidade aumentada podem ser classificados conforme o tipo de display utilizado (Azuma, 2001), envolvendo visão ótica ou visão por vídeo, dando origem a quatro tipos de sistema:

- 1) sistema de visão ótica direta;
- 2) sistema de visão direta por vídeo;
- 3) sistema de visão por vídeo baseado em monitor;
- 4) sistema de visão ótica por projeção.

2.2.1.1. Sistema de visão ótica direta

O sistema de realidade aumentada de visão ótica direta se utiliza de óculos ou capacetes, com lentes que permitem o recebimento direto da imagem real, ao mesmo tempo em que permitem a projeção de imagens virtuais ajustadas sobre o mundo real. A Figura 15 mostra um diagrama deste tipo de sistema.

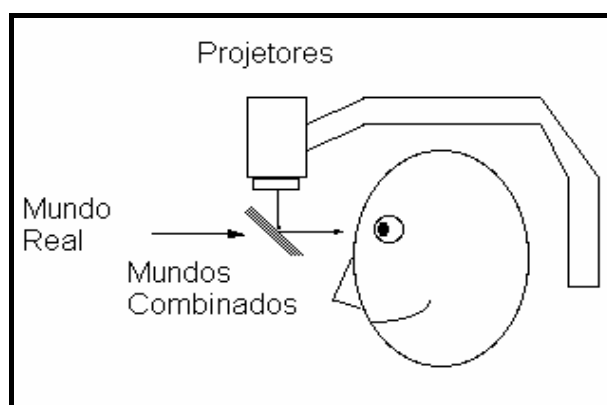


Figura 15. Diagrama conceitual do sistema de RA de visão ótica direta (AZUMA, 1997).

2.2.1.2. Sistema de visão direta por vídeo

O sistema de visão direta por vídeo é composto de um capacete de visualização com sistema de rastreamento de posição, sobre o qual é disposta uma câmera de vídeo, conforme a Figura 16 (BAJURA & NEUMANN, 1995). Nesse caso, a imagem real é obtida pela câmera de vídeo montada sobre o capacete, enquanto que a imagem virtual é gerada por um computador que considera o posicionamento do rastreador. Um misturador combina as duas imagens e mostra o resultado final ao usuário.

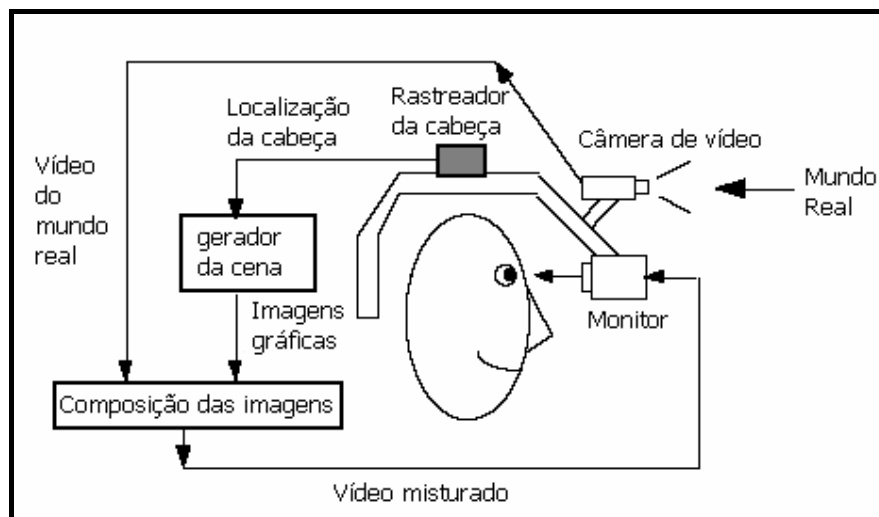


Figura 16. Diagrama Conceitual de um sistema de RA de visão direta por vídeo (AZUMA, 1997).

2.2.1.3. Sistema de visão por vídeo baseado em monitor

O Sistema de visão por vídeo baseado em monitor utiliza uma webcam acoplada ao computador que captura o ambiente real. Depois de capturado, o ambiente real é misturado com objetos virtuais gerados por computador e apresentado no monitor como mostra a Figura 17.

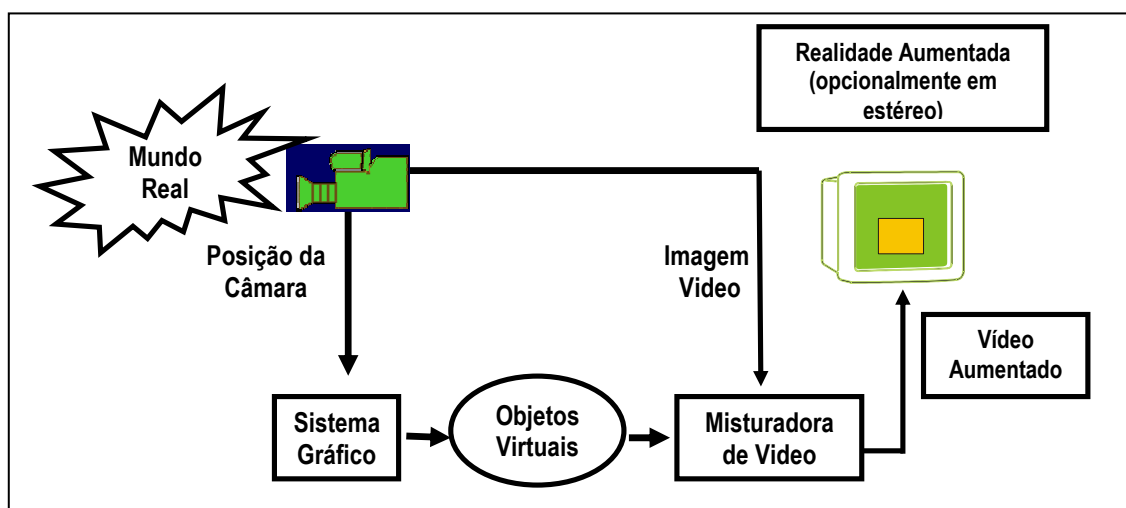


Figura 17. Diagrama de um sistema de RA de visão por vídeo baseado em monitor (BRAZ, 2000).

2.2.1.4. Sistema de visão ótica por projeção

O sistema de visão ótica por projeção utiliza superfícies do ambiente real, onde são projetadas imagens dos objetos virtuais, cujo conjunto é apresentado ao usuário que o visualiza sem a necessidade de nenhum equipamento auxiliar (KIRNER, 2005).

2.3. POR QUE REALIDADE MISTURADA?

Algumas das vantagens:

- Proporciona maior motivação dos usuários, pelo fato de explorar os sentidos visuais;
- O poder de ilustração para alguns processos e objetos é muito maior do que outras mídias, permitindo que o usuário veja a cena no ponto de visão que queira, com um simples movimento da cabeça (no caso de estar se utilizando HMD) ou com um simples movimento do objeto real onde o objeto virtual está inserido (no caso de estar utilizando uma câmera de filmar);
- Permite uma análise de muito perto e/ou de muito longe, um bom exemplo é poder analisar uma célula, nos seus detalhes, e o mais importante: poder interagir com este objeto;
- Estimula o usuário a realizar experiências em um ambiente sensorialmente atrativo;
- Altamente interativo;
- Permite que o usuário passivo torne-se ativo;
- Pode incentivar a colaboração;
- Permite a observação das áreas e dos eventos que seriam inviáveis por outros meios, tais como as cenas subaquáticas, históricas, reconstruções de locais arqueológicos.

A realidade misturada realça a percepção de um usuário e a interação com o mundo real, permitindo obter informações do objeto virtual

exposto ao qual não poderia diretamente detectar com seus próprios sentidos (AZUMA, 1997).

O objetivo final é criar um sistema que o usuário não possa distinguir o mundo real do virtual (VALLINO, 2002).

2.4. APLICAÇÕES EM RA

A aplicabilidade da Realidade Aumentada estende-se para diversas áreas do conhecimento. Assim, pode-se constar varias aplicações que empregam a RA para incrementar suas funcionalidades. Abaixo dispomos de alguns exemplos.

2.4.1. APLICAÇÕES MÉDICAS

Na área médica, muitas pesquisas têm sido realizadas sobre Realidade Aumentada. Muitas aplicações já foram desenvolvidas.

Vários pesquisadores desenvolveram uma estação de trabalho médica utilizando Realidade Aumentada, para visualização e interação com novos procedimentos de intervenção a pacientes. O sistema é chamado de AR Window, ou Janela de Realidade Aumentada. A Figura 18 ilustra a Janela sendo utilizada por um usuário. O médico faz interações sobre um boneco e através da janela aumentada é possível ver dados que se parecem com pacientes reais (MEDARPA, 2005).



Figura 18. Utilizando o sistema AR Window (MEDARPA, 2005).

2.4.2. APLICAÇÕES EDUCACIONAIS

A Realidade Aumentada vem sendo muito explorada na área de educação. Ela permite a fácil visualização e manipulação do objeto de estudo, reproduzindo os dados complexos sob a forma de objetos e textos tridimensionais, aumentando a capacidade de percepção do estudante, que passa a ser estimulado pela possibilidade de interação com a interface. Como exemplo podemos mencionar o livro "Sólidos Geométricos com Realidade Aumentada e ARToolKit" (KIRNER, 2004), onde os usuários podem visualizar os textos dos objetos geométricos e também de maneira tridimensional, o sistema também permite visualizar este mesmo objeto com características diferentes, como mostra a Figura 19.

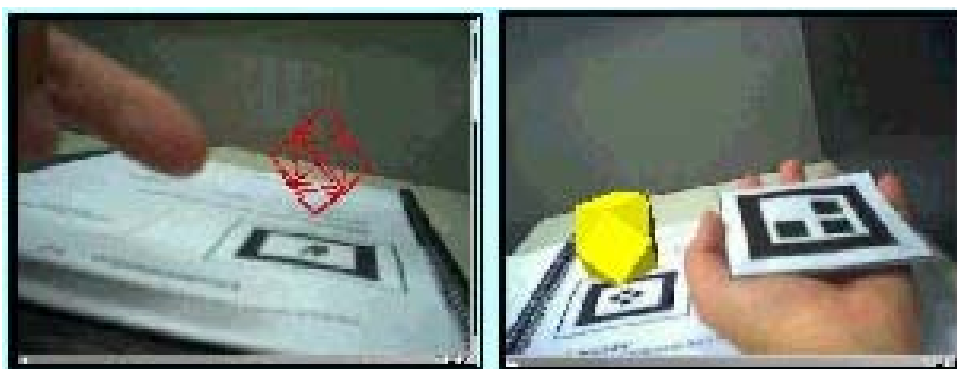


Figura 19. Livro: Sólidos Geométricos com Realidade Aumentada (aplicação dinâmica), por Daniela Akagui e Rafael Santin (KIRNER, 2004)

2.5. REALIDADE VIRTUAL X REALIDADE AUMENTADA

Realidade virtual pode ser definida como um sistema que permite as pessoas visualizarem, interagirem e imergirem em ambientes tridimensionais gerados por computador.

Nesta definição, podemos destacar três pontos importantes. Em primeiro lugar para se visualizar dados tridimensionais é requerido uma modelagem desses objetos e, para adquirir um alto grau de realismo é necessário um desempenho computacional elevado. No segundo ponto, o

mundo virtual é interativo; o usuário requer uma resposta em tempo real para que a interação possa ser algo interessante. O último ponto se refere à imersão do usuário neste ambiente virtual e, para se obter um alto grau de imersão, é necessário que ele esteja completamente mergulhado neste mundo controlado pelo computador, sem qualquer visão do mundo real.

A Realidade Aumentada não deixa o usuário totalmente imerso em ambiente virtual, permitindo que o participante visualize o ambiente real também. Em sistema de RA, o computador fica adquirindo imagens reais através de uma câmera de vídeo e calculando o local exato de sobrepor os objetos virtuais na cena. Esta técnica é a chamada visão computacional, o qual é feito o rastreamento do ambiente para encontrar algum padrão pré-estabelecido pelo programador.

Com as características mencionadas acima, podemos identificar quais as diferenças entre realidade virtual e realidade misturada. Uma diferença entre estes dois sistemas está no ato de imersão. Ao contrário da RV, a RM não exclui por completo o mundo real à volta do usuário, mas combina os dois mundos, dando a sensação que todos os objetos seriam reais. Partindo deste conceito, podemos dizer que um ambiente aumentado, se exige mais do computador, no sentido de que se está processando imagens e renderizando objetos virtuais, em tempo real, na cena. Em um sistema de realidade virtual não é necessário um mecanismo que combine as duas imagens (real e virtual), sendo que em RA se não houver um tratamento de sincronismo destas duas imagens, a sensação de realismo diminui.

2.6. PROBLEMAS TÍPICOS DE UM SISTEMA DE REALIDADE MISTURADA

Um dos problemas básicos que limitam, atualmente, as aplicações de realidade aumentada é o ato de sincronizar o mundo real com o virtual (AZUMA, 1997). Os objetos no mundo real e virtual devem ser alinhados corretamente, ou a ilusão que os dois mundos coexistem será comprometida. Podemos observar um ambiente misturado desalinhado na figura 20 parte A, e

na parte B os objetos virtuais sincronizados no ambiente no ambiente real, transmitindo uma sensação de realismo da cena.



A

B

Figura 20. Na parte A, os objetos virtuais estão desalinhados com o mundo real. Na parte B os dois mundos estão sincronizados (KIRNER, 2004b).

Outro problema encontrado pelos desenvolvedores é o conceito de oclusão. A oclusão é responsável em esconder partes de objetos, ou por inteiro, que estejam por de trás de outros. Com isto, observa-se a ordem dos objetos em termos de distância à câmera. Este efeito é ilustrado na Figura 21. Na imagem da esquerda a esfera está por trás do cubo na cena e na imagem da direita o cubo é que está por trás. Este efeito de oclusão é muito importante para dar realismo à cena. Imagine um avião passando por trás de uma montanha, se o efeito de oclusão não for tratado no ambiente, o avião passará pelo meio da montanha.

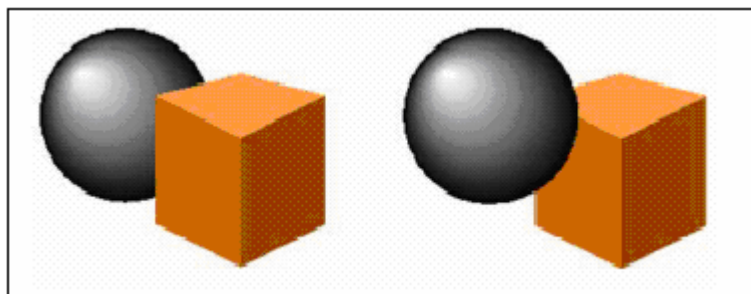


Figura 21. Efeito de oclusão (RAPOSO, 2004).

Capítulo 3

3. TRABALHO COOPERATIVO SUPORTADO POR COMPUTADOR

O termo *groupware* costuma ser usado quase como sinônimo de CSCW – *computer supported cooperative work* (trabalho cooperativo suportado por computador), porém alguns autores identificam uma tendência diferenciada no emprego destes termos. Enquanto CSCW é usado para designar a pesquisa na área do trabalho em grupo e como os computadores podem apoiá-lo, *groupware* é designado como o suporte tecnológico (*hardware/software*) ao trabalho colaborativo (GRUDIN, 1994). Assim, sistemas de correio-eletrônico, teleconferências, suporte a decisão e editores de texto colaborativos são exemplos de *groupware*. Podemos classificar um *groupware* em quatro tipos: quanto ao espaço/tempo, quanto à previsibilidade e quanto ao tamanho do.

3.1. ESPAÇO/TEMPO

Uma primeira classificação dos *groupwares* apresentada por DeSanctis e Gallupe em (DESANCTIS, 1987), se refere às noções de tempo e ao espaço sob as quais as interações são realizadas. Existe a interação onde os integrantes estão no mesmo local e no mesmo momento, como uma conversa face a face. Já uma troca de mensagens por correio eletrônico se dá em diferentes períodos de tempo e com as partes estando em diferentes ambientes. As quatro categorias possíveis são ilustradas na Figura 22.

	no mesmo momento	em momentos diferentes
no mesmo lugar	Interação Face a Face	Interação Assíncrona
em diferentes lugares	Interação Síncrona e Distribuída	Interação Assíncrona e Distribuída

Figura 22 – Classificação de um groupware quanto ao espaço e tempo (GRUDI, 1994)

Há quem critique esta classificação argumentando que os usuários requerem sistemas que satisfaçam aos quatro quadrantes de tempo/espaço. Isto é, o sistema deve ser flexível o suficiente para atender comunicações síncronas e assíncronas, entre usuários remotamente distribuídos ou não.

3.1.1. PREVISIBILIDADE

Grudin (1994) propõe uma nova classificação Espaço/Tempo como mostra a Figura 23, dividindo em duas a coluna que representa diferentes momentos e a linha que representa diferentes lugares. Passa-se a levar em conta se o lugar ou momento são previsíveis ou não. Isto é, pode-se prever ou não se uma atividade ocorrerá dentro de um dado intervalo de tempo ou em dado local. Por exemplo, ao enviar uma correspondência eletrônica, em geral aguarda-se uma resposta dentro de um tempo razoável, sendo esta uma atividade altamente previsível em relação aos fatores tempo e espaço. Por outro lado, a atividade de escrita colaborativa envolve lugares diferentes e previsíveis, e momentos diferentes e totalmente imprevisíveis. Por exemplo, dois escritores podem realizar a atividade de escrita em locais diferentes, mas cada um em seu local provável, enquanto que escolhem momentos totalmente aleatórios para realizar a tarefa.

		TEMPO		
		Mesmo	Diferentes mas previsíveis	Diferentes e imprevisíveis
L O C A L	Mesmo	Auxílio a reuniões	Deslocamento de Tarefas	Salas de grupos
	Diferentes mas previsíveis	Tele/Vídeo conferências	Correio Eletrônico	Edição Colaborativa
	Diferentes mas imprevisíveis	Seminários de Interação multicast	Bulletin Boards eletrônicos	Workflow

Figura 23 – Classificação Tempo/Espaço/Previsível (GRUDIN, 1994).

3.1.2. TAMANHO DO GRUPO

Uma outra extensão da classificação Tempo/Espaço, proposta por Nunamaker et al em (1991), considera importante o tamanho do grupo especialmente no contexto de sistemas de suporte a reuniões. Assim, cria-se mais uma dimensão na classificação Espaço/Tempo, transformando-a em Espaço/Tempo/Tamanho (veja a Figura 24).

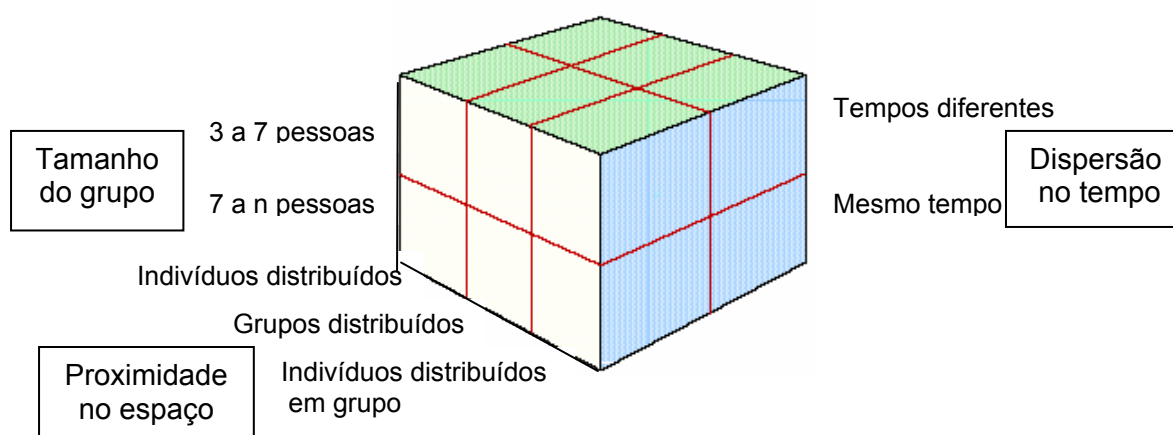


Figura 24 – Classificação Tempo/Espaço/Tamanho (GRUDIN, 1994)

O mundo moderno avança cada vez mais na direção do trabalho em grupo, seja pelo tamanho e complexidade crescente das tarefas, seja pela interdisciplinaridade das soluções. Os sistemas de groupware são respostas da ciência da computação para acompanhar esses avanços.

A área de CSCW continua em ascensão. A quantidade e a variedade de trabalhos gerados comprovam a crescente importância da área. Novas idéias, conceitos e aplicações surgem a cada conferência, a cada produto. A interdisciplinaridade do tema gera o aparecimento de trabalhos relacionados com a área em diversas conferências e publicações. Vertentes como CSCL (*Computer Supported Collaborative Learning* – Aprendizagem Colaborativa Suportada por Computador) surgem como consequência natural desse movimento.

3.2. TRABALHOS COLABORATIVOS COM REALIDADE AUMENTADA

Um ambiente virtual pode ser definido como uma interface homem-máquina que utiliza técnicas de computação gráfica possibilitando aos usuários visualizar, manipularem e interagirem em tempo real.

Ambientes Virtuais Colaborativos (AVC), segundo Churchill et. al. (2001), são ambientes virtuais que possibilitam aos usuários trabalharem juntos, cooperarem em uma mesma tarefa mesmo quando estão geograficamente separados, interagindo com objetos virtuais tridimensionais.

Um ambiente colaborativo de realidade aumentada refere-se a sistemas distribuídos onde pessoas podem interagir umas com as outras ou com os objetos virtuais.

3.2.1. SHAREDSPACE

Essa pesquisa, que desenvolveu o SharedSpace, possibilitou o desenvolvimento de um *browser* de internet tridimensional que permite a colaboração entre múltiplos usuários nesses *browsers*. Os usuários visualizam

as páginas da *web* flutuando no espaço (Figura 26) ao seu redor. A interface foi projetada para ser totalmente livre das mãos. As páginas são selecionadas através da visualização, uma vez selecionadas elas são unidas ao ponto de visão do usuário. Cada vez que uma página nova é carregada um novo objeto do browser é criado. Nesta aplicação os usuários possuem um HMD (*Head Mounted Display*) e um sensor eletromagnético que rastreia a posição da cabeça (Figura 25). Um programa de reconhecimento de voz é usado para permitir que os usuários carreguem e interajam com as páginas da *web* usando a entrada de voz (BILLINGHURST, 1996).

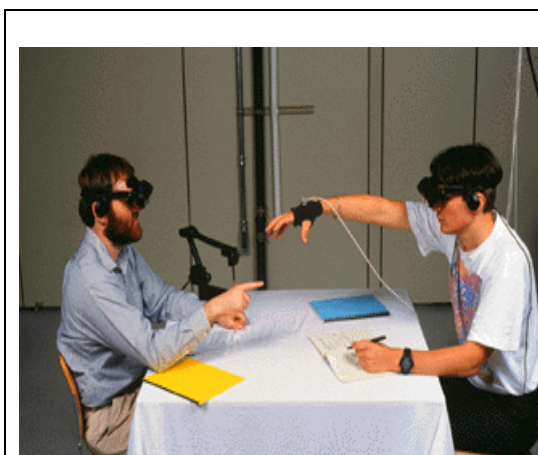


Figura 25. Visão de quem está de fora (BILLINGHURST, 1996).

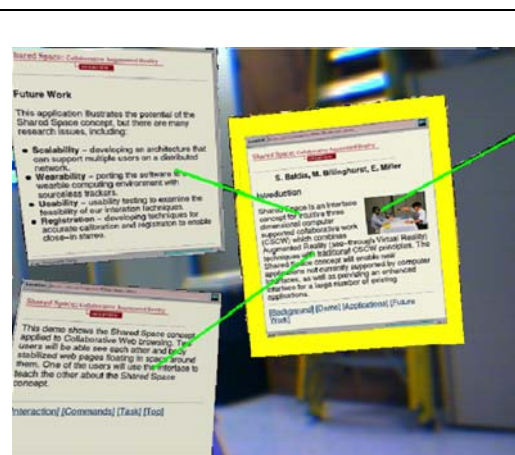


Figura 26. Visão do usuário no sistema SharedSpace (BILLINGHURST, 1996).

3.3. CONSTRUCT3D

Um sistema de Realidade Aumentada para educação da matemática geométrica é baseado no “Studierstube” que permite a colaboração em um sistema de realidade aumentada (SCHMALSTIEG, 1996). São usados o HMD e o PIP (*Personal Interaction Panel*), que disponibiliza aos usuários uma opção de menus que é selecionado através de uma caneta especial, que rastreia a posição e detecta qual menu foi selecionado. Estes dois dispositivos

possibilitam a interação tridimensional da cena, como mostra a Figura 27 (KAUFMANN, 2000).



Figura 27. Estudantes trabalhando com o Construct3D. (KAUFMANN, 2004)

3.3.1. JOGOS EDUCACIONAIS EM AMBIENTES DE REALIDADE AUMENTADA

Neste trabalho foi desenvolvida uma aplicação remota de caráter colaborativo. O jogo da velha foi o ambiente escolhido para gerar a colaboração e a comunicação entre os sistemas foi implementada através de soquetes e do software ARToolkit. O tabuleiro era carregado em cima de um marcador e cada peça colocada pelos usuários tinha seu próprio marcador (KIRNER, 2005).

3.3.2. MARE - MULTIUSER AUGMENTED REALITY ENVIRONMENT

É um ambiente de realidade aumentada onde os usuários se posicionam ao redor de uma mesa visualizando e manipulando dados tridimensionais. Um método simples e diferente foi desenvolvido para a integração dos elementos no ambiente cujo erro máximo obtido foi de 1 cm, valor suficiente para aplicações que não são de caráter crítico. A estabilidade e a exatidão de Optotrack (rastreador utilizado) geraram um resultado bem melhor que do antigo sistema, baseado em rastreador magnético. O objetivo foi gerar um ambiente dinâmico, estendível e de fácil utilização (GASCUEL, 2002).

A parte colaborativa é muito importante neste sistema, tendo todo o aparato de hardware e software. O usuário pode tanto visualizar quanto manipular os objetos virtuais com uma caneta especial que está sendo rastreada, porém esta caneta também pode ser utilizada para escrever alguma anotação em uma caderneta.

É um ambiente genérico, ou seja, passível de ser usado entre várias áreas citadas e testadas pelos desenvolvedores que são:

Engenharia: o engenheiro pode discutir sobre um protótipo virtual durante uma reunião, selecionar uma área específica para obter mais informações e visualizar o objeto em ângulos diferentes.

Arquitetura: O arquiteto pode apresentar o projeto virtual ou real, como por exemplo, um planejamento urbano, durante uma reunião diretamente sobre a mesa. O cliente pode modificar interativamente o projeto através de um método intuitivo e simples da interação (Figura 28).

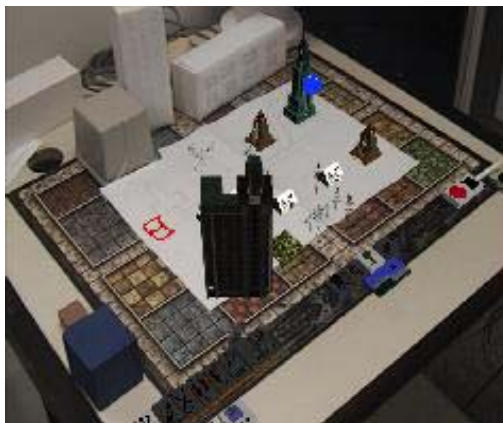


Figura 28. Ambiente aumentado de projetos de arquitetura (GASCUEL, 2002).

Visualização científica: O químico durante uma reunião pode usar a possibilidade da interação 3D com moléculas e também usar algum material real, esboçando o esquema do elemento (Figura 29).



Figura 29 – Ambiente aumentado de visualizações científicas (GASCUEL, 2002).

Jogos: Jogos reais podem ser adicionados de características virtuais. O coordenador do jogo pode adicionar elementos virtuais, em tempo real e modificar totalmente o rumo do jogo.

Enfim, este sistema pode abrir várias oportunidades de uso da RA mostrando uma nova forma de visualizar informações, interagir e colaborar com elas.

CAPÍTULO 4

4. MECANISMOS DE COMUNICAÇÃO

4.1. SOQUETES

Um soquete é um mecanismo de comunicação entre processos que serve como um ponto final da comunicação. O soquete serve para viabilizar a comunicação em uma rede de TCP/IP e fornece uma estrutura de comunicação entre os processos executados tanto localmente como em nós de uma rede (SCRIMGER, 2002). A utilização de soquetes torna única cada conexão cliente/servidor e elimina a necessidade de gravar os dados na unidade de disco do destinatário durante cada transação.

O soquete se utiliza da camada de transportes para a transmissão de dados. A função da camada de transportes é promover uma transferência de dados confiável e econômica entre a máquina de origem e a máquina de destino, independente das redes físicas em uso no momento. Graças à camada de transporte, os programas aplicativos podem ser desenvolvidos utilizando-se um conjunto de primitivas padrão, onde podem funcionar em uma ampla variedade de redes, sem a preocupação de se lidar com interfaces de sub-redes diferentes e com uma transmissão não-confiável (TANENBAUM, 2003), ou seja, se todas as redes reais fossem perfeitas e se todas tivessem as mesmas primitivas de serviço e ainda a garantia de nunca mudar, provavelmente a camada de transporte não seria necessária. Porém, na vida real ela cumpre a função primordial de tornar as camadas superiores do projeto imunes à tecnologia e às imperfeições da sub-rede.

Para permitir que os usuários tenham acesso ao serviço de transporte, a camada de transporte deve oferecer algumas operações a

programas aplicativos, ou seja, uma interface de serviço de transporte (*Application Program Interface* – API). Cada serviço de transporte tem sua própria interface.

Para se ter um exemplo genérico de um serviço de transporte, considere as cinco primeiras primitivas apresentadas na Tabela 1. Essa interface de transporte é uma estrutura básica, mas denota o que uma interface de transporte orientada a conexões deve fazer. Ela permite aos programas aplicativos estabelecer, enviar/receber dados e encerrar uma conexão, o que é suficiente para muitas.

PRIMITIVA	SIGNIFICADO
LISTEN	Esperar um contato de um processo que queira se conectar.
CONNECT	Tentar ativamente estabelecer uma conexão.
SEND	Enviar informações/dados.
RECEIVE	Receber informações/dados
DISCONNECT	Pedir encerramento da conexão.

Tabela 1. Primitivas básicas para um serviço de transporte simples.

De acordo com Scrimger (2002) os soquetes APIs populares são:

- soquetes Berkeley:
- *transport Layer Interface* – TLI (Interface de Camada de Transporte);
- *transport Independent Remote Procedure Calls* – TI-RPCS (Procedimento de chamadas remotas de transporte independente);
- soquetes Windows (Winsock).

4.1.1. SOQUETES DE BERKELEY

A interface dos soquetes Berkeley foi inicialmente utilizada em várias versões do UNIX. Essas primitivas são amplamente usadas em programação para a Internet (TANENBAUM, 2003). Resumidamente elas seguem o mesmo modelo do exemplo das primitivas utilizadas em um serviço de transporte simples, mas oferecem mais recursos e maior flexibilidade.

Os soquetes Berkeley fornecem uma interface que é fácil de utilizar e de implementar. Com os soquetes Berkeley, há duas maneiras pelas quais os aplicativos podem acessar os serviços de uma rede de transportes:

- **Acesso Orientada a Conexão** – Um canal virtual é mantido entre o remetente e o receptor. A entrega de dados é confirmada ou é negada com a ajuda de reconhecimentos ou de reconhecimentos negativos. Se os dados não forem entregues ao destinatário, inicia-se uma retransmissão qualquer dos dados perdidos ou as camadas superiores são informadas da falha. As camadas superiores então podem tomar as ações corretivas necessárias. Serviços orientados a conexão são mantidos pelo TCP.

- **Acesso sem conexão** – Os dados são enviados ao destinatário sem esperar por reconhecimentos. Se todos os dados ou parte deles são perdidos durante uma transmissão, o remetente não terá conhecimento sobre isso. O *User Datagram Protocol* (UDP) fornece serviços sem conexão.

Na Tabela 2 são mostradas as primitivas com seus respectivos significados de soquetes para TCP. As quatro primeiras primitivas são executadas pelo servidor nesta mesma ordem.

PRIMITIVA	SIGNIFICADO
SOQUETE	Criar um novo ponto para conexão, alocando espaço de tabela na entidade de transporte.
BIND	Anexar um endereço local a um soquete, por onde os dados passarão.
LISTEN	Fazer o servidor ficar esperando por uma conexão; mostra o tamanho da fila.
ACCEPT	Bloquear o servidor até que seja requisitada uma transferência de dados por um cliente.
CONNECT	Fazer o cliente tentar ativamente uma conexão.
SEND	Enviar dados através da conexão.
RECEIVE	Receber dados da conexão.
CLOSE	Encerrar a conexão.

Tabela 2 As primitivas de soquetes para TCP.

No trabalho proposto, utilizou-se o conceito do soquete de Berkeley, onde um servidor é executado e fica esperando uma conexão de um cliente. Com a conexão do cliente o servidor está pronto para receber os dados. Estes dados são o identificador do objeto virtual e suas posições naquele momento; estas informações são passadas para o servidor que atualiza o ambiente com o novo objeto virtual e sua respectiva posição. O servidor faz o mesmo processo, enviando o identificador e as posições objeto virtual disposto na sua cena e envia estes dados para o cliente que recebe e atualiza a sua cena com o novo objeto virtual e sua respectiva posição.

4.2. RPC - REMOTE PROCEDURE CALL

A chamada de procedimento remoto (RPC) é um método bem conhecido de transferência de controle de um processo para outro. Tal aplicação torna as aplicações de rede muito mais fáceis de programar e mais familiares. A intenção do RPC é tornar-se transparente para o programador, de maneira que não exista diferença entre um procedimento local ou remoto. Todo o procedimento de chamada remota é escondido por procedimentos, com códigos de rede, chamados de *stubs* (TANENBAUM, 2003).

De acordo com Scrimger (2002) o RPC não contém qualquer código de comunicação. Como resultado é independente dos seguintes itens:

- plataformas e hardware de comunicação;
- protocolos de comunicação;
- sistemas operacionais;
- seqüências de chamadas que precisariam utilizar um software subjacente de comunicação.

Podemos identificar alguns passos ilustrando a chamada remota:

(1) Cliente chama o *stub* cliente. O *stub* cliente "empacota" os argumentos necessários para o procedimento remoto, coloca-os no formato padrão e elabora uma ou mais mensagens.

(2) As mensagens produzidas pelo *stub* cliente são enviadas usando uma chamada ao sistema operacional.

(3) As mensagens são transferidas para sistema remoto usando os protocolos UDP ou TCP.

(4) O *stub* servidor em estado de espera recebe a mensagem da rede do sistema operacional remoto.

(5) O *stub* servidor executa uma chamada de procedimento local, possivelmente usando as mensagens recebidas do cliente.

(6) O procedimento retorna um valor ou responde ao *stub* servidor.

(7) O *stub* converte a resposta para mensagens de rede e envia-as ao sistema operacional do servidor para transmissão remota ao cliente.

(8) A mensagem é enviada pela rede para o sistema operacional do cliente.

(9) A *stub* cliente, em estado de espera, lê a mensagem resposta do sistema operacional local.

(10) A resposta, já convertida, é repassada a rotina do cliente.

A RPC, apesar de todos os pontos positivos, para que ela funcione bem e corretamente são necessárias algumas restrições. De acordo com Tanenbaum (2003) existem alguns problemas que podem dificultar a implementação de RPC, abaixo são listados alguns deles:

- iso de parâmetros de ponteiros. Normalmente, a passagem de um ponteiro a um procedimento não é um problema. O procedimento chamado pode usar um ponteiro do mesmo modo que o chamador utiliza, porque ambos procedimentos convivem no mesmo espaço de endereços virtuais. Com a RPC, a passagem de ponteiros é impossível, porque o cliente e o servidor estão em espaços de endereços diferentes.

- falta de especificação do tamanho de um vetor. Na linguagem C, por exemplo, é perfeitamente válido escrever um procedimento que calcula o produto interno de dois vetores (arrays), sem especificar o tamanho de cada um. Cada um deles poderia terminar com um valor especial conhecido apenas pelo procedimento de chamada e pelo procedimento chamado. Nestas circunstâncias, é essencialmente impossível para o *stub*

cliente empacotar os parâmetros: ele não tem como determinar o tamanho desses parâmetros.

- um outro problema se relaciona ao uso de variáveis globais. Normalmente, o procedimento chamador e o procedimento chamado podem se comunicar usando variáveis globais, além de parâmetros. Se o procedimento chamado for agora deslocado para uma máquina remota, o código falhará, porque as variáveis globais não serão mais compartilhadas.

Problemas como estes não impossibilitam o uso da RPC (ela é amplamente utilizada), mas são necessárias algumas restrições para fazê-la funcionar bem na prática.

4.3. CORBA

CORBA é a especificação de uma arquitetura desenvolvida para ambientes distribuídos e heterogêneos e que utiliza a tecnologia de orientação a objetos. (OMG, 2006) Essa especificação foi criada pela **OMG** (*Object Management Group*). A OMG é uma organização internacional da indústria de software fundada em maio de 1989 suportada por centenas de membros. A carta de princípios da organização inclui o estabelecimento de diretrizes na indústria e especificações de gerenciamento de objetos para fornecer uma estrutura comum para desenvolvimento de aplicações. O objetivo primário é alcançar sistemas baseados em objetos em ambientes distribuídos e heterogêneos com características de reusabilidade, portabilidade e interoperabilidade.

Assim sendo, a CORBA é uma especificação de uma arquitetura que permite as aplicações fazerem solicitações a objetos, de uma forma transparente e independente, indiferente à linguagem, sistema operacional ou considerações de localização.

Tal arquitetura define um módulo intermediário ("barramento") entre clientes e servidores, o Object Request Broker (ORB). Neste modelo,

objetos clientes requisitam serviços às implementações de objetos (Servants) através de um ORB. O ORB é responsável por todos os mecanismos requeridos para encontrar o objeto, preparar a implementação de objeto para receber a requisição, e executá-la. O cliente vê a requisição de forma independente de onde o objeto está localizado, qual linguagem de programação ele foi implementado, ou qualquer outro aspecto que não está refletido na interface do objeto.

Para isso, a CORBA utiliza a IDL OMG (*Interface Definition Language*) como uma forma de descrever interfaces. OMG IDL é uma linguagem puramente declarativa baseada em C++. Por ser uma linguagem declarativa, a IDL não possui nenhuma estrutura algorítmica ou de variáveis. São descritos em IDL apenas os tipos, as constantes e as operações necessárias para especificar uma interface de objeto. Isso garante que os componentes em CORBA sejam autodocumentáveis, permitindo que diferentes objetos, escritos em diferentes linguagens, possam interoperar através das redes e de sistemas operacionais.

Para fazer uma requisição, o cliente pode utilizar rotinas *stub* IDL que definem os objetos de acordo com as operações que podem ser executadas e seus parâmetros. Essas rotinas são geradas na compilação da descrição da interface do objeto ao qual se destina o pedido. Entretanto, pode-se utilizar a interface de invocação dinâmica (DII), permitindo assim, acesso aos seus serviços em tempo de execução. Sendo necessário para tanto, a inclusão da interface do objeto ao repositório de interfaces. Um Repositório de Interfaces é uma base de dados que contém interfaces OMG IDL. Seus serviços basicamente permitem o acesso, armazenagem e atualização dessas interfaces.

Tanto a invocação dinâmica (através da DII) quanto a estática (através de rotinas *stub* IDL) satisfazem a mesma semântica de requisição, e o receptor da mensagem não sabe dizer como o pedido foi invocado.

CAPÍTULO 5

5. ARTOOLKIT

O software ARToolkit é uma biblioteca escrita na linguagem C e C++ que permite aos programadores desenvolver facilmente aplicações de realidade aumentada. Esta biblioteca utiliza técnicas de visão computacional para calcular a posição e orientação da câmera relativa a um marcador em tempo real (dentro de um tempo aceitável, ou seja, passando a sensação ao usuário que os objetos virtuais estão sendo mostrados no mesmo momento dos objetos reais). O programador pode então usar esta informação para desenhar objetos tridimensionais exatamente alinhados, como nos mostra a Figura 30, onde a torre virtual está absolutamente inserida no mundo real, transmitindo a sensação de que seja real. Ela não está flutuando no espaço, está alinhada com a mesa, da mesma forma que os outros objetos reais dispostos na cena estão.

O ARToolKit foi desenvolvido primeiramente pelo Dr. Hirokazu Kato da Universidade de Osaka, Japão, e é apoiado pelo *Human Interface Technology Laboratory* (HITLab) da Universidade de Washington, e pelo HITLab NZ da Universidade de Canterbury, Nova Zelândia. Como o software está disposto na rede com o seu código fonte aberto, outras pessoas já modificaram o ARToolKit, conforme a sua necessidade.



Figura 30. Exemplo de um objeto virtual (torre) alinhado com a cena real (HITLab, 2004).

O segredo da “mágica” do ARToolkit, está no rastreamento dos quadrados pretos usados como marcadores. Seguem abaixo, os passos que o ARToolkit realiza ao misturar os dois mundos (BERRE, 2006):

1- a câmera captura o vídeo do mundo real e emite-o ao software do computador.

2- a aplicação procura todas as formas quadradas nos *frames* de vídeo capturados.

3- se um quadrado for encontrado, o software se utiliza fórmulas matemáticas para calcular a posição da câmera relativa ao quadrado preto.

4- uma vez que a posição da câmera é sabida, um objeto virtual é inserido na mesma posição.

5- este objeto virtual é renderizado na cena e sobreposto no marcador.

6- o resultado final é mostrado no HMD, ou mesmo na tela do monitor, então, o usuário vê os dois mundos misturados. A Figura 31 exhibe o resultado final. E a Figura 32 sumariza as etapas mencionadas acima.

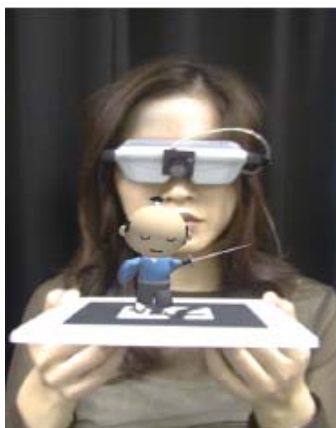


Figura 31. Ambiente de Realidade Aumentada utilizando ARToolkit (BILLINGHURST, 2000).

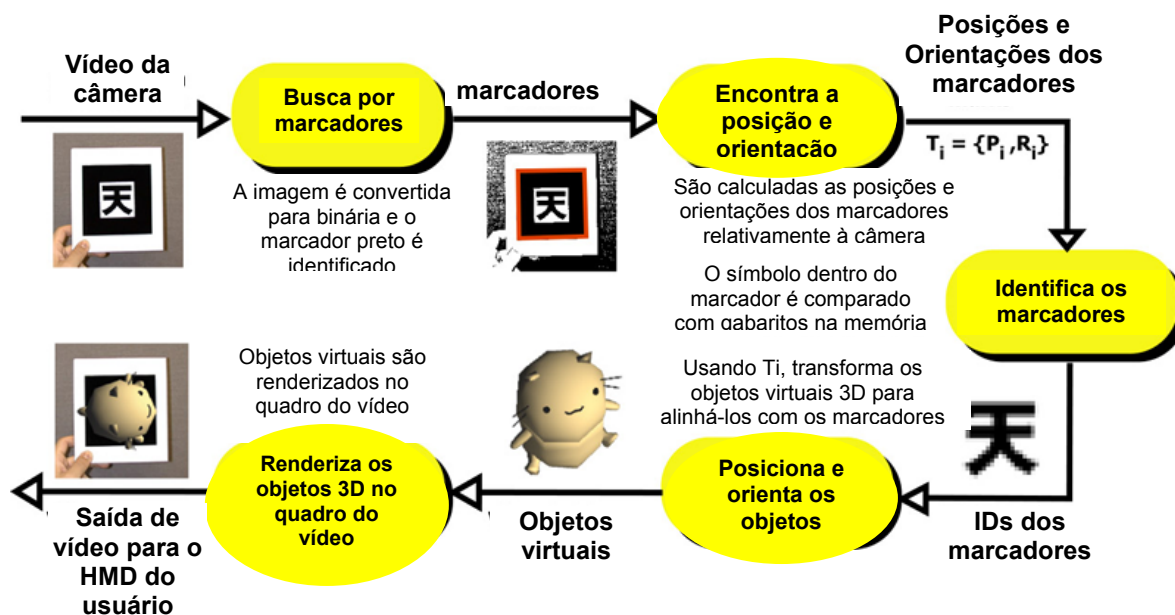


Figura 32. Resumo dos passos realizados pelo ARToolkit (HITLab, 2004).

O laboratório desenvolvedor do ARToolkit (HITLab), define algumas características importantes deste software, que seguem abaixo:

- necessita apenas de uma simples câmera de vídeo para rastrear a posição/orientação do marcador;
- o código que rastreia os quadrados pretos (marcadores) é simples;
- possibilita a utilização de vários marcadores;
- código fácil para a calibração da câmera;
- rápido o bastante para aplicações em tempo real de RA;
- distribuições para Sistemas Operacionais Sgi IRIX, OS, Linux, MacOS e Windows;
- código fonte aberto.

5.1. EXECUTANDO O SOFTWARE

Os executáveis ficam armazenadas na pasta bin do ARToolkit.

Exemplo: C:\ARToolkit\bin

São eles:

- simpleVRML
- simple
- multiTest
- mk_patt
- calib_camera2

Alguns marcadores já vêm como padrões no próprio programa (Figura 33). Eles estão em: “C:\ARToolkit\patterns”



Figura 33. Padrão Hiro.

Também é possível criar os seus próprios marcadores, tirando a palavra Hiro e colocando qualquer outro desenho dentro do quadrado preto. Este novo marcador terá que ser cadastrado no ARToolkit. Abaixo há a descrição dos passos de cadastramento:

1º - faça um novo marcador;

2º - execute o programa mk_patt;

3º - coloque a placa que deseja cadastrar em frente à câmera, quando visualizar o marcador envolto de uma linha verde e vermelha, clique com o mouse sobre o vídeo, a imagem ficará estática e depois inserir o nome que deseja para aquele marcador;

4º - Um arquivo do tipo "dat" será criado na pasta c:\ARToolkit\bin, basta remove-lo para a pasta Data;

5º - Abrir o arquivo vrml_data, que esta dentro da pasta Data e modifica-lo, inserindo a quantidade de padrões cadastrados e apontar o objeto VRML específico para cada marcador. Esses arquivos VRML têm que estar dentro da pasta WRL.

Para visualizar os objetos virtuais sobre os marcadores, basta executar o arquivo simpleVRML.

CAPÍTULO 6

6. SUPORTE COLABORATIVO PARA APLICAÇÕES DE REALIDADE AUMENTADA

Este capítulo apresenta o projeto que contempla a construção de um protótipo que integre um ambiente de realidade aumentada tridimensional com ferramentas de apoio ao trabalho colaborativo. Este protótipo deverá constituir um ambiente apropriado tanto para pesquisas em computação gráfica (visualização de dados, interação tridimensional) como para pesquisas educacionais e trabalhos colaborativos.

6.1. AMBIENTE COLABORATIVO

O ambiente colaborativo foi desenvolvido utilizando uma programação distribuída via soquetes. . A escolha dessa forma de comunicação se deve ao fato da facilidade de integração com o software ARToolkit.

O ambiente é Cliente/Servidor escrito de forma dedicada, ou seja, a construção de um único servidor onde todos os clientes estão conectados a ele, enviando atualizações de tempos em tempos para todos os mesmos.

A Figura 34 ilustra a estrutura geral de funcionamento do protótipo.

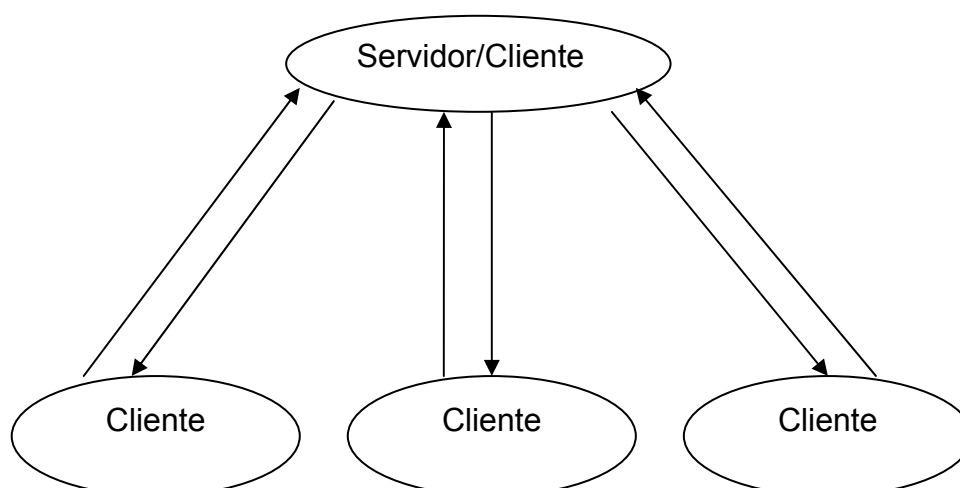


Figura 34. Esquema geral do ambiente colaborativo

Neste ambiente, quando um cliente fixa um objeto virtual na cena do ambiente aumentado, a identificação do objeto e sua posição são enviadas ao servidor, que atualiza o seu mundo (se ele também for um cliente) e repassa essas informações a todos os outros clientes, que atualizando a cena, permitem a todos os participantes visualizar o mesmo ambiente aumentado.

A conexão entre os usuários é realizada primeiramente ativando o servidor, deixando-o escutando a rede. Após isso, o cliente que quiser se conectar deve saber previamente o IP do servidor, para depois requisitar conexão.

O soquete está implementado externamente ao ARToolkit, porém algumas modificações, no código ARToolkit, foram necessárias. O usuário ao fixar o objeto na cena, o sistema cliente faz a leitura do ARToolkit das posições x, y e z e do identificador VRML do objeto virtual fixado, escrevendo estes dados em um arquivo texto. O soquete cliente envia o conteúdo do arquivo para o soquete servidor, que por sua vez, reenvia para todos os outros clientes, inclusive para o próprio cliente que o enviou os dados. O ARToolkit compara as posições recebidas. Se forem iguais ao do objeto local, significa que foi ele quem mandou o objeto, então o software descarta os

dados recebidos, porém se forem diferentes, o ARToolkit lê este arquivo e atualiza a cena local, alocando o objeto no lugar indicado. Abaixo está o código que possibilita esta ação descrita acima. O quadro 1 mostra o algoritmo do envio de um objeto virtual pela rede.

```
// Procedimento onde a aplicação cliente recebe as informações enviadas
// pelo servidor
Receber a posição do objeto e o identificador VRML
Escrever os dados em um arquivo texto

// Procedimento onde o ARToolkit atualiza a cena
Ler os dados do arquivo
Comparar se o objeto enviado existe em sua cena e se está na mesma
posição
Se verdadeiro
    Não faz nada
Se falso
    Posiciona o objeto no local indicado
```

Quadro1. Algoritmo que demonstra o recebimento dos dados pelo cliente, enviados pelo servidor.

A Figura 35 demonstra a comunicação do protótipo.

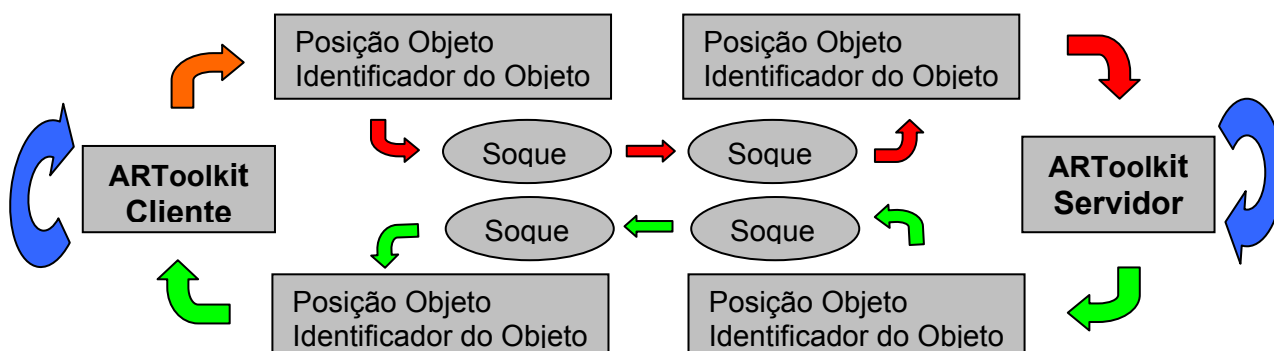


Figura 35. Esquema de funcionamento do ambiente colaborativo em rede.

Agora que vimos como a aplicação funciona, podemos defini-la como sendo um *groupware*. Como vimos no Capítulo 3, *groupware* é a tendência que estuda o suporte tecnológico (*hardware/software*) ao trabalho colaborativo (GRUDIN, 1994). Portanto podemos classificar este trabalho como sendo um *groupware*, pois implementa um suporte (*software*) para trabalhos colaborativos em rede de realidade aumentada.

Em relação a classificação quanto ao espaço e tempo vamos rever a Figura 36 do capítulo mencionado acima. Esta figura ilustra que podem existir quatro tipos de *groupware*, vamos identificar onde o sistema poderá se enquadrar.

	no mesmo momento	em momentos diferentes
no mesmo lugar	Interação Face a Face	Interação Assíncrona
em diferentes lugares	Interação Síncrona e Distribuída	Interação Assíncrona e Distribuída

Figura 36 – Classificação do sistema quanto ao espaço e tempo.

Claro que para identificar onde o protótipo se enquadra precisaríamos saber qual a aplicação que estaria sendo utilizada. Por exemplo, se tivéssemos uma aplicação que simule um jogo da velha ficaríamos com o local azul e vermelho, pois teríamos que jogar em um mesmo momento, porém poderíamos estar face a face ou em lugares diferentes.

Se estivéssemos construindo uma cidade virtual e os usuários estivessem distribuídos e não houvesse a necessidade de seqüência de ações, a aplicação se enquadraria na parte pintada de verde, pois um construtor poderia inserir objetos na cena em momentos diferentes que de outros.

Um exemplo de aplicação para a parte pintada em amarelo seria uma interação educacional, onde os alunos estariam no mesmo local,

porém manipulariam e/ou visualizariam o ambiente em momentos distintos um dos outros.

Enfim, com isso concluímos que o suporte colaborativo em rede de realidade aumentada poderia ser utilizado em vários tipos de aplicações, cabendo a cada usuário colocar sua criatividade em ação.

Abaixo são descritas algumas vantagens e desvantagens, percebidas no sistema.

Vantagens:

1 – facilidade na criação do algoritmo de comunicação;

2 – ideal para pequenas aplicações com pouca troca de mensagens. O servidor é dedicado a servir. Para ambientes com poucos clientes, onde não há sobrecargas de tarefas, o servidor se porta muito bem.

Desvantagens:

1 – causa queda do desempenho se for grande o fluxo de E/S, pois o sistema é centralizado;

2 – baixa escalabilidade. Não se recomenda para aplicações com grandes números de participantes, pois o servidor pode ser tornar um “gargalo”.

6.2. PRIMEIROS TESTES DA APLICAÇÃO

Os objetos virtuais foram modelados em VRML, uma linguagem que permite descrever, através de elementos geométricos, ambientes computacionais (mundos) pelos quais, o usuário pode navegar e com cujos componentes o visitante pode interagir (Silicon Graphics, 2004).

No primeiro protótipo, foram implementados quatro cubos em VRML, para realizar os primeiros testes da aplicação em rede. A Figura 37 ilustra os cubos virtuais sobre os seus respectivos marcadores.

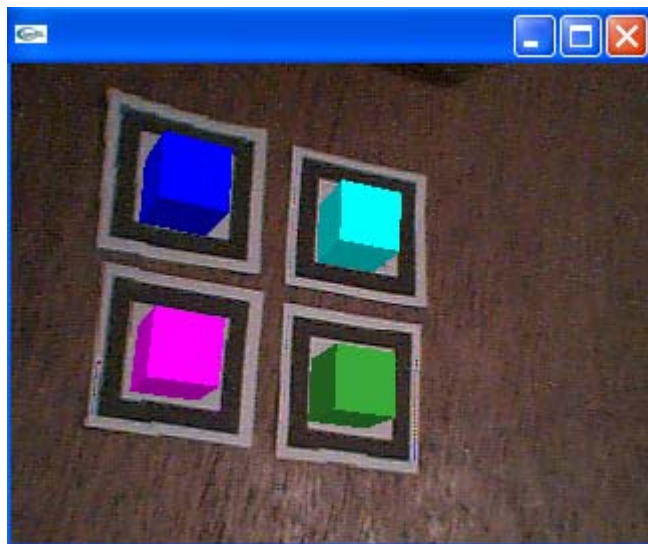


Figura 37. Cubos virtuais utilizados para o primeiro experimento.

No primeiro teste da aplicação, todos os *hosts* eram Clientes e Servidores, era necessário iniciar a aplicação Cliente e também a aplicação Servidora na mesma máquina, e portanto, somente dois usuários eram possíveis dentro do ambiente. Os testes da aplicação em rede foram realizados com êxito, as Figuras 38 e 39, ilustram o funcionamento da colaboração.

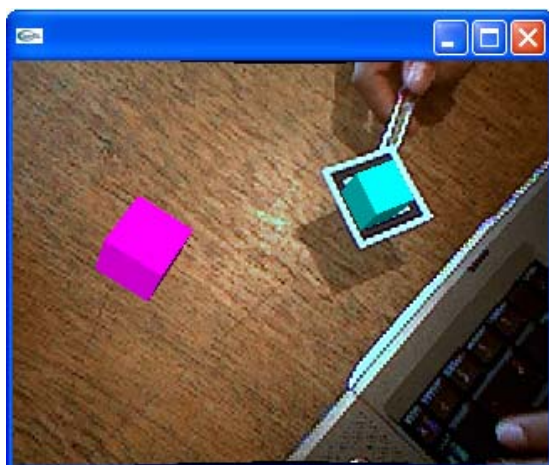


Figura 38. Usuário 1 no ambiente em rede.

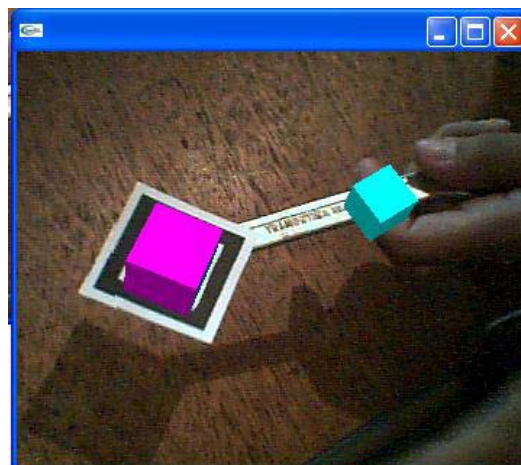


Figura 39. Usuário 2 no ambiente em rede.

Cada usuário enxerga a sua placa com o seu respectivo objeto inserido a ela, mas não enxerga a placa dos outros integrantes, apenas os objetos virtuais.

6.2.1. MANIPULAÇÃO DO AMBIENTE

Algumas novas características no software ARToolkit, além da possibilidade de estar em rede, também foram adicionadas para realmente haver a colaboração em um ambiente. Apenas estar em rede, não significa que o ambiente passa a ser colaborativo. Como já foi dito, para haver a colaboração na realização de uma determinada tarefa, é necessária a troca de informações uns com os outros; estas informações podem ser textuais, sonoras ou visuais.

A colaboração deste ambiente se dá pelas formas de manipulação instauradas no sistema. Poderíamos fazer algumas perguntas para descobrir como os integrantes interagirão entre si e com o ambiente. Por exemplo:

- 1) Os participantes poderão conversar entre si? Como?
- 2) De que forma, os usuários irão interagir com o ambiente?
- 3) Haverá restrições no acesso aos objetos do ambiente? Quais?

Respondendo:

1) Não há nenhum suporte implementado no sistema, que utilize a comunicação sonora ou textual, através do computador. Os participantes apenas poderão conversar entre si, se estiverem face a face, ou se integrarem o uso de tecnologias de comunicação existentes atualmente, como *chats*, videoconferência e outros.

2) Os usuários poderão interagir com o ambiente de forma bastante intuitiva. Eles poderão manipular os objetos previamente cadastrados no ARToolkit, através de uma “pá”. Esta “pá” tem a função de copiar os objetos

que estão cadastrados e transportá-los por onde desejar na cena. Também foi implantada a função de fixar, onde o usuário poderá fixar esse objeto que está sobre a “pá”, onde desejar, apenas acionando a tecla F do teclado. Além destas duas funções essenciais, houve a necessidade de construir uma outra “pá”, na qual tem a função de apagar algum objeto inserido na cena.

3) Sim, haverá restrições. O participante poderá copiar, transportar e apagar apenas os objetos que pertencem a ele, não podendo manipular as peças do outro integrante.

As peças do “jogo” devem ser previamente cadastradas pelo ARToolkit, isso é normal em todas as aplicações que se utilizam o ARToolkit. Mas as placas que possibilitam copiar, transportar e apagar, não fazem parte do software, portanto houve algumas alterações em seu código para que estas funcionassem.

O primeiro marcador a ser cadastrado é considerado a “pá”, ou seja, aquele que se comporta como um mecanismo de cópia e de transporte dos objetos virtuais dispostos na cena. Logo em seguida tem que ser cadastrado a “pá” que tem a função de apagar, para somente depois cadastrar os objetos que serão usados na aplicação.

As Figuras 40 e 41 mostram a “pá” de cópia e transporte em ação no ambiente. O usuário coloca a pá próxima da placa que contém o objeto virtual que queira manipular e após um curto espaço de tempo (30 interações), o identificador da placa passa a ser atribuído a pá, agora a pá possui o mesmo objeto da placa podendo movimentá-lo para onde desejar.

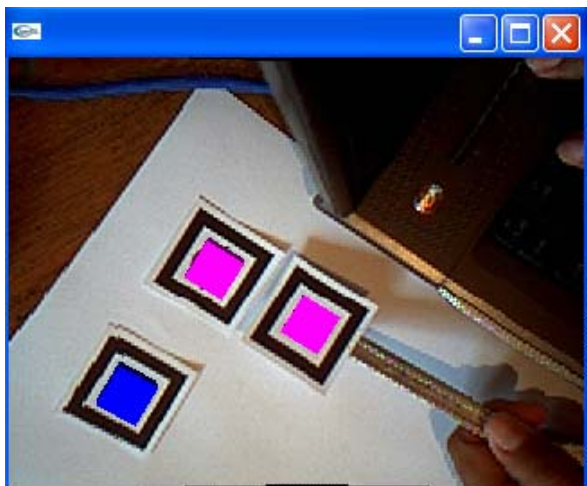


Figura 40. Copiando um objeto virtual com o marcador “pá”.

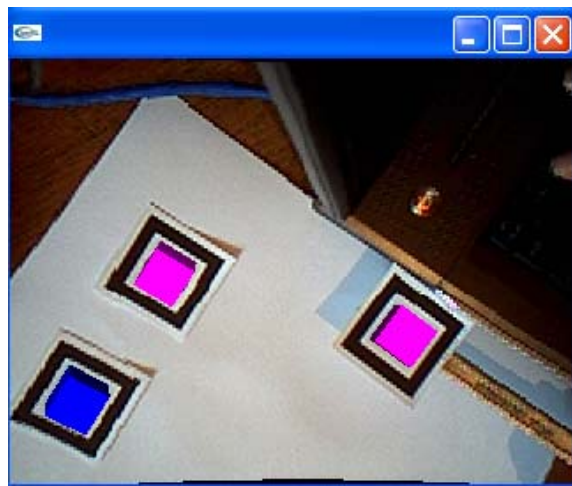


Figura 41. Transportando o objeto com o mesmo marcador.

O Quadro 2 ilustra o Código Fonte de uma parte do arquivo simpleVRML.cpp, do ARToolkit, onde a placa “pá” copia o objeto virtual da placa mais próxima à ela.

```
static int mkrCopia( ObjectData_T * object,int objectnum ){
    /* compara as posições de placas e objetos com relação à placa 0 a
    fim de
    copiar os objetos virtuais*/

    double mat1[3][4], mat2[3][4];
    static double x=-1, y=-1;
    double bufx=-1,bufy=-1;
    static int inter,last;
    int mkr =-1;
    ObjectData_T *aux=NULL ;
    ObjectData_T *aux_erase=NULL ;

    arUtilMatInv(object[0].trans, mat1);

    for(int ind = 2; ind < objectnum; ind++ ) {
        if( object[ind].visible == 0 ) continue;

        arUtilMatMul(mat1, object[ind].trans, mat2);
```

Quadro 2a. Código da função “cópia” do objeto virtual.

```

        x=mat2[0][3];
        if(x<0)
            x=x*(-1);

        y=mat2[1][3];
        if(y<0)
            y=y*(-1);

        if(x<10||y<10){
            if(bufx==-1&&bufy==-1){
                bufx=x;
                bufy=y;
                mkr=ind;
            }
            else if(x<bufx||y<bufy)    {
                bufx=x;
                bufy=y;
                mkr=ind;
            }
        }
    }

    if(mkr==last&&mkr!=-1){
        inter++;
    }
    else
        inter=0;

    if(inter==30){
        inter=0;
        object[0].vrmI_id=mkr;
        x=-1;
        y=-1;
    }
    else
        mkrTransp(objeto);

    last=mkr;

return(0);
}

```

Quadro 2b. Continuação do quadro 2a.

É possível manter o objeto da pá em qualquer posição fixando os pontos x, y e z, para isso uma lista foi implementada que armazena esses pontos a cada ação de fixar, esta ação é realizada a partir da tecla “f” do teclado, ao pressionar a tecla f o objeto é fixado naquele ponto, como mostra a Figura 42.

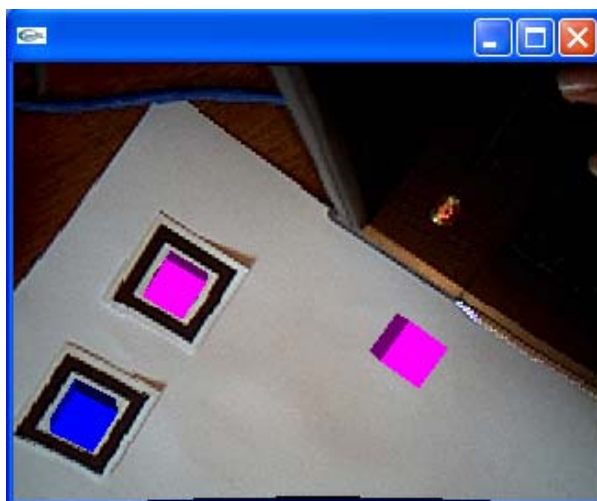


Figura 42. Fixando o objeto através da tecla F do teclado.

No momento que o objeto é fixado o identificador Vrml da pá é removido, ficando assim, sem objetos, dessa maneira a pá fica livre para poder pegar outros objetos.

O Quadro 3 ilustra o Código Fonte, onde é realizado a função de fixar um objeto na cena.

```
static void keyEvent( unsigned char key, int x, int y)
{
    /* quit if the ESC key is pressed */
    if( key == 0x1b ) {
        printf("*** %f (frame/sec)\n", (double)count/arUtilTimer());
        cleanup();
        exit(0);
    }
    if( key == 'f'){
        fix();
    }
}
```

Quadro 3a. Código da função “fixar” o objeto virtual na cena.

```

void fix(){
/*fixa o objeto da placa 0 */

ObjectData_T *aux=NULL ;
objeto = (ObjectData_T *)malloc( sizeof(ObjectData_T) );
objeto[0] =object[0];
object[0].vrmI_id=0;
if(objeto[0].vrmI_id>1){

if(inicio==NULL){
    inicio=objeto;
    aux=inicio;
    aux->tipo=0;
    aux->prox=NULL;

}

else{
    aux=inicio;
    while(aux!=NULL){
        if(aux->prox==NULL){
            aux->prox=objeto;
            aux=objeto;
            aux->tipo=0;
            aux->prox=NULL;

        }
        aux=aux->prox;
    }
}
}
}
}
}

```

Quadro 3b. Continuação do quadro 3a.

Para apagar os objetos fixos, basta deixar a “pá” deletora, próxima ao objeto em questão, após 30 interações será feita a exclusão do objeto, como mostra as Figuras 43 e 44.

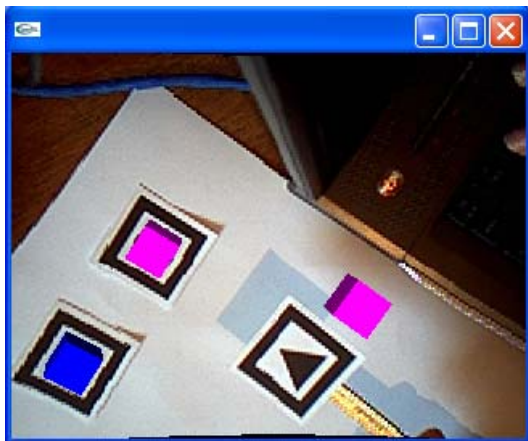


Figura 43. Marcador deletor próximo do objeto a ser excluído.

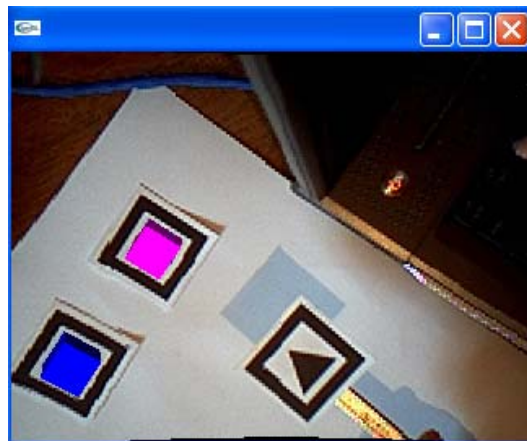


Figura 44. Cena com o objeto excluído.

O quadro 4 ilustra o trecho de código onde é realizada a função Apagar.

```
static int mkrApagador( ObjectData_T * objeto ){
    /* compara as posições de placas e objetos com relação à placa 1
    para apagar os objetos */

    double mat1[3][4], mat2[3][4];
    static double x, y;
    double bufx=-1,bufy=-1;
    static int inter,last;
    int mkr =-1;
    ObjectData_T *aux=NULL ;
    ObjectData_T *aux_erase=NULL ;

    arUtilMatInv(object[1].trans, mat1);
    aux=inicio;
```

Quadro 4a. Código da função “apagar” um objeto virtual da cena.

```

        if(aux!=NULL){
        while(aux!=NULL){
        arUtilMatMul(mat1, aux->trans, mat2);
        x=mat2[0][3];
        if(x<0)
            x=x*(-1);
        y=mat2[1][3];
        if(y<0)
            y=y*(-1);

        if(x<10||y<10){

            if(bufx==-1&&bufy==-1){
                bufx=x;
                bufy=y;
                mkr=aux->vrml_id;
                aux_erase=aux;
            }

            else if(x<bufx||y<bufy)    {
                bufx=x;
                bufy=y;
                mkr=aux->vrml_id;
                aux_erase=aux;
            }

        }
        aux=aux->prox;
    }
}

if(mkr==last&&mkr!=-1){
    inter++;
}
else
    inter=0;

if(inter==30){
    inter=0;
    printf("0");
    aux=inicio;
}
if(aux!=NULL&&aux_erase!=NULL){
    if(aux!=aux_erase){
        while(aux!=aux_erase&&aux_erase!=NULL){

```

Quadro 4b. Continuação do quadro 4a.

```

if(aux->prox==aux_erase){
    aux->prox=aux_erase->prox;
    delete(aux_erase);
    aux_erase=NULL;
}
aux=aux->prox;
}
else{
    if(aux->prox!=NULL){
        inicio=aux->prox;
        aux=NULL;
        delete (aux_erase);
        aux_erase=NULL;
    }
    else{
        inicio=NULL;
        aux_erase=NULL;
        delete(aux);
        aux=NULL;
    }
}
}
}

last=mkcr;

```

Quadro 4c. Continuação do quadro 4b.

Como já foi mencionado, o Quadro 1 cliente só pode modificar os seus objetos. Os objetos que estão sendo enviados pelos outros clientes, ainda não sofrem nenhum tipo de intervenção.

As mudanças que foram realizadas no software ARToolkit, foram programadas sobre o ambiente de desenvolvimento Microsoft Visual C++ 6.0. A interface Cliente/Servidor foi realizada sobre o ambiente Builder C++ 6.0, por permitir uma programação visual, como mostram as Figuras 45 e 46.

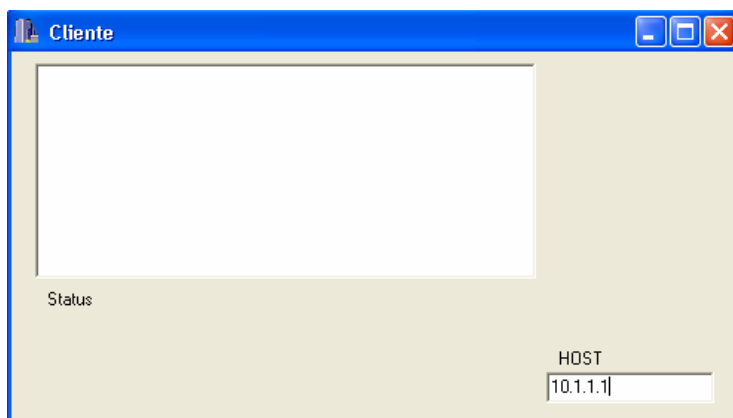


Figura 45. Interface do sistema Cliente para ativar a conexão com o servidor.

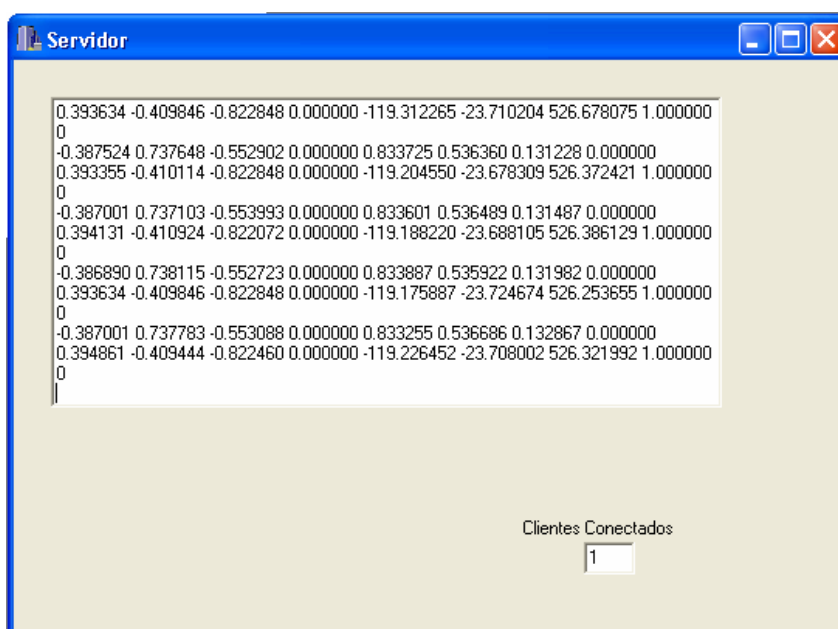


Figura 46. Interface do sistema Servidor recebendo os dados dos sistemas clientes.

6.3. TESTES DA APLICAÇÃO FINAL

Para testes de uma aplicação final, foi desenvolvido um jogo parecido com o dominó, onde os participantes podem jogar de forma colaborativa e em rede. Os objetos virtuais também foram modelados em VRML. As Figuras 47 e 48 mostram respectivamente os marcadores utilizados na aplicação e com os objetos virtuais sobre eles.



Figura 47. Marcadores da aplicação final

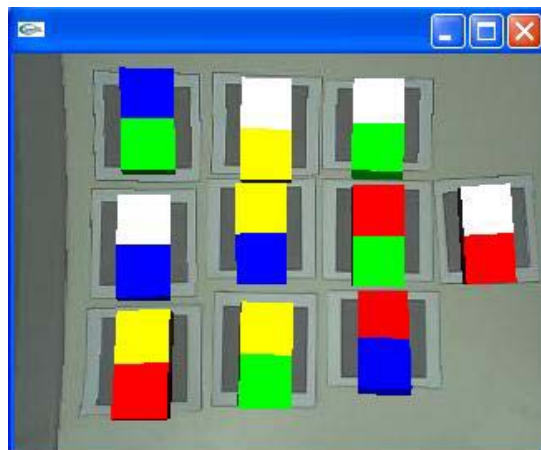


Figura 48. Objetos virtuais sobre seus marcadores, simulando um jogo de dominós.

.Esta aplicação não tem o objetivo de criar fielmente o jogo de dominós, com todas as suas regras. Este exemplo foi utilizado apenas para demonstrar as funcionalidades do sistema, podendo ser modificado conforme a necessidade do usuário.

A aplicação consiste em várias peças, cada uma com duas cores, o próximo jogador tem que inserir uma peça com a cor correspondente da peça jogada anteriormente pelo outro jogador.

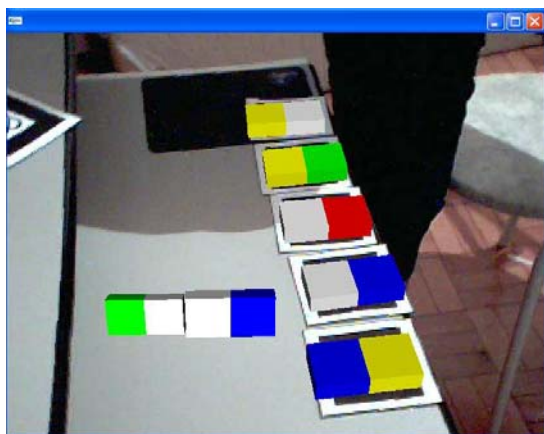


Figura 49. Visão do jogador que inseriu a peça Branca e Azul.

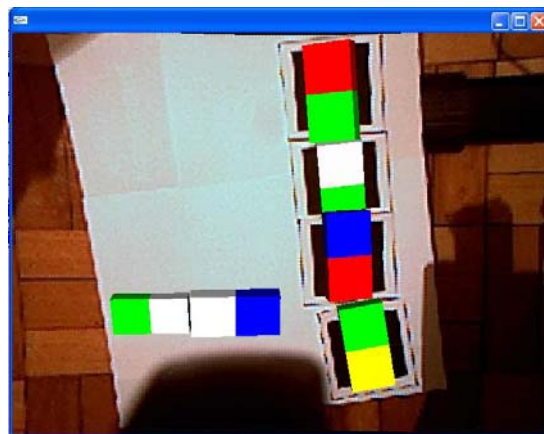


Figura 50. Visão do jogador que inseriu a peça Verde e Branca.

Nesta etapa, foi implementado um único servidor, que serve a todos os clientes conectados a ele. Portanto, agora, pode-se jogar com mais de dois jogadores. Os testes da aplicação em rede foram realizados com êxito.

As figuras 49 e 50 ilustram dois ambientes, onde os usuários jogam em rede. Cada jogador possui a mesma visão da cena, porém com peças diferentes para jogar. Cada jogador enxerga somente os objetos dele mesmo e os objetos fixados pelos outros participantes, pois somente são enviados pela rede, os objetos que foram fixados no ambiente. Na figura 49 o jogador possui 5 peças do jogo, portanto, somente vê as suas peças e as peças jogadas (2). Na figura 50 o jogador possui apenas 4 peças de dominó, portanto, em seu espaço de visão somente interage com as suas peças e com as duas peças jogadas.

6.3.1. MANIPULAÇÃO DO AMBIENTE

A manipulação dos objetos na aplicação final se dá da mesma maneira que no primeiro protótipo. Utilizando as pás (cópia e deleção) e a ação Fixar (apertando a tecla “F” do teclado) para a interação com o ambiente.

Após a implementação do protótipo, foi possível perceber que este ambiente se mostra muito interessante e eficiente para realizar interações com as mãos, ou seja, de uma maneira muito intuitiva, abrindo um leque de possibilidades em aplicações de várias atividades dinâmicas em grupos.

CAPÍTULO 7

7. CONCLUSÃO

O mundo moderno avança cada vez mais na direção do trabalho em grupo, seja pelo tamanho e complexidade crescente das tarefas, seja pela interdisciplinaridade das soluções. Os sistemas de groupware são respostas da ciência da computação para acompanhar esses avanços. A RA consiste em uma nova forma de visualização que pode ser útil em várias aplicações e em diversas áreas. É uma ferramenta de versátil visualização pela possibilidade oferecida de interação entre o usuário e os objetos analisados.

Fundindo estes dois conceitos, o trabalho colaborativo com sistemas de realidade aumentada, temos um novo grande leque de possibilidades para pesquisas como se pode verificar através deste trabalho.

A construção de um Ambiente Colaborativo de Realidade Aumentada foi o objetivo desta pesquisa. Pode-se verificar com este trabalho que a colaboração em um ambiente de Realidade Aumentada, torna a tarefa realizada muito mais estimulante. A visualização de dados tridimensionais misturada com o mundo real já provoca, por si só, a imaginação e o interesse para realizar até mesmo tarefas simples. Com este trabalho não só o tridimensional foi explorado, mas a colaboração entre pessoas e com o ambiente 3D. O que antes se fazia sozinho e poder-se-ia tornar-se enfadonho, neste ambiente, o trabalho em grupo permite aos usuários do sistema interagir com o ambiente e também com outros participantes, estimulando o cumprimento dos objetivos da aplicação utilizada.

Neste trabalho foi desenvolvido um suporte computacional para garantir que um sistema de realidade aumentada, utilizando o software ARToolkit, possa ser compartilhado com vários usuários dispostos sobre uma rede de computadores. Apesar de o sistema estar em rede, isso não quer dizer

que o mesmo permita a colaboração entre os seus usuários sobre uma tarefa. Portanto, foram desenvolvidas formas naturais de interação com o ambiente que favorecem o trabalho colaborativo.

Foi criado para verificar as funcionalidades do sistema um jogo imitando o dominó, ao invés das peças conterem uma seqüência de bolinhas, as peças do “dominó virtual” serão associadas usando cores. Cada jogador terá que colocar a peça que contenha a cor da peça jogada anteriormente pelo outro jogador. O objetivo desta aplicação não foi gerar um jogo com regras claras e definidas, mas demonstrar como o sistema suporta a colaboração em rede. Portanto, o ambiente suporta vários tipos de aplicações, não somente o dominó, permitindo que os usuários possa utilizá-lo para fins educacionais, científicos, estimular a colaboração, até mesmo para entretenimento, como foi o exemplo deste trabalho.

Um servidor dedicado foi construído com o intuito de receber as peças inseridas por cada jogador e atualizar a cena de todos os outros. Cada usuário poderá manipular as peças que são suas, ou seja, aquelas que estão sendo geradas pelo seu computador. Essas manipulações são: copiar um objeto (uma peça do dominó), transportar este objeto, fixá-lo no local desejado e apagá-lo quando quiser.

Com este ambiente pode-se concluir:

- a arquitetura de comunicação, utilizando soquetes externos, se mostrou eficaz quando não há grandes volumes de dados trafegando na rede.
- o suporte colaborativo pode ser utilizado em várias outras aplicações, incrementando-o conforme a necessidade do usuário.
- as formas de manipulação instauradas no sistema garantiram a interação intuitiva.

7.1. TRABALHOS FUTUROS

O sistema pode ser melhorado para garantir que a rede suporte a escalabilidade, ou seja, não sofrer alterações com o aumento de usuários, ou expandido, integrando novos sistemas que permitam a comunicação através de outras tecnologias, como textuais e sonoras, melhorando ainda mais a interface de colaboração. A melhoria do sistema passa incondicionalmente, entre outros fatores, como, por exemplo, modificar a forma dos soquetes se comunicarem, ao invés de os dados de E/S serem escritos em arquivos textos, eles poderiam ser enviados diretamente para o ARToolkit, diminuindo o acesso à rede. Atualmente o sistema fica em um *loop* lendo os dados armazenados nestes arquivos, mesmo não havendo alterações.

Outra melhoria que deixaria o ambiente mais intuitivo seria a troca de marcadores pelas próprias mãos do usuário e a câmera rastrearía os dedos substituindo assim o uso dos marcadores.

8. REFERÊNCIAS BIBLIOGRÁFICAS

ABSOLUT TECHNOLOGIES. **Visualizações Inteligentes em Realidade Virtual**. Acessado em: 10/03/06. Disponível em :<www.abs-tech.com>.

ARAÚJO, R. B. **Especificação e análise de um sistema distribuído de realidade virtual**. Tese (Doutorado), Departamento de Engenharia de Computação e Sistemas Digitais, Escola Politécnica da Universidade de São Paulo, 1996.

AUKSTAKALNIS, S.; BLATNER, D. **Silicon Mirage: The Art and Science of Virtual Reality**. Peachpit Press, Berkeley, CA, 1992.

AZUMA, R. T. (1993). **Tracking Requirements for Augmented Reality**. *Communications of the ACM*, 36(7):50-51, July 1993.

AZUMA, R. T. (1997). **A Survey of Augmented Reality**. *Teleoperators and Virtual Environments*, 355-385, 1997.

AZUMA, R. T. et al. (2001). **Recent advances in augmented reality**. *Computer Graphics and Applications*, IEEE, 21(6):34-47, Nov/Dec 2001.

BAJURA, M., NEUMANN, U. **Dynamic Registration Correction in Video-Based Augmented Reality Systems**. *IEEE Computer Graphics & Applications*, 15(5):52-60, Sept. 1995

BEGAULT, D. R. **3-D Sound for Virtual Reality and Multimedia**. *Academic Press*, Cambridge, MA, 1994.

BENFORD, S. D.; GREENHALGH, C.M. **Collaborative Virtual Environments** *Communications of the ACM*, Vol. 44 No. 7 p. 79. ACM Press, MA, Julho 2001.

BERRE, A.; SYVERINS, T.; SALOMONSEN, M. et al. **ARToolKit for Dummies**. Acessado em: 01/03/06 Disponível em: <<http://rasmus.uib.no/~st02204/HCI/ARTUserDoc.pdf>>.

BILLINGHURST, M et al. **Shared space: An augmented reality interface for computer supported collaborative work**. In Proc. of Collaborative Virtual Environments Workshop '96, Nottingham, Great Britain, 1996.

BILLINGHURST, M.; CAMPBELL, D. H.; CHINTHAMMIT, W. **Magic Book: Exploring Transitions in Collaborative AR Interfaces**. SIGGRAPH 2000 Emerging Technologies Proposal, 2000.

BOMAN, D. K. **International Survey: Virtual Environment research**, *IEEE Computer*, 28(6):57-65, June 1995.

BRAZ, J. **Realidade Aumentada para além da realidade virtual**. Exposição Olhares - 2000, EST - Setúbal, 2000. Acessado em: 01/03/06. Disponível em: <<http://ltodi.est.ips.pt/jbraz/curriculum.html#artigos>>.

BRYSON, S. **Virtual reality in scientific visualization**, *Computer e Graphics*, 17(6):679-685, 1996.

CHURCHILL, E.F.; SNOWDON, D.; MUNRO, A. **Collaborative Virtual Environments**. Digital Places and Spaces for Interaction. London, UK: Springer Verlag, 2001.

COLAVITTI, F. **A explosão da realidade virtual: A tecnologia que está revolucionando a medicina, a indústria e a diversão**. Acessado em 01/03/06. Disponível em: <http://revistagalileu.globo.com/Galileu/0,6993,ECT803835-1938,00.html>.

DESANCTIS, G., GALLUPE, B.. (1987) **A foundation for the study of group decision support systems**. *Management Science*, v. 33, n. 5. p. 589-609.

EARNSHAW, R. A. et al. **Virtual Reality applications**. *Academic Press*, 328 pp., London, 1995.

EBERSPÄCHER, H, F.; VASCONCELOS, C. D.; JAMUR, J. H.; ELEUTERIO, M. A. **Eureka: um ambiente de aprendizagem cooperativa baseado na Web para Educação à Distância**. *Simpósio Brasileiro de Informática na Educação (SBIE)*, Curitiba, 1999.

Enciclopédia WIKIPÉDIA. **Óculos Estereoscópicos**. Acessado em: 01/03/06. Disponível em: http://pt.wikipedia.org/wiki/%C3%93culos_estereosc%C3%B3picos.

FIFTH DIMENSION TECHNOLOGIES. **Produtos de Realidade Virtual**. Acessado em 01/03/06. Disponível em: <http://www.5dt.com/products.html>.

GASCUEL, J. D.; GRASSET, R. **MARE: Multiuser Augmented Reality Environment on Real Table Setup**. ACM SIGGRAPH Conference Abstracts and Applications, 2002 [Versão Eletrônica]. Acesso em: 01/03/06. Disponível em: <http://artis.imag.fr/Publications/2002/GG02>

GRUDIN, J. (1994). **Computer-Supported Cooperative Work: History and Focus**. *Computer*, maio/1994.

GRUDIN, J. (1994b). **Groupware and Social Dynamics: Eight Challenges for Developers**. *Communications of the ACM*, 1994.

HAGSAND, O. et al. **Introduction to DIVE**. Acessado em 01/03/06. Disponível em: <http://www.sics.se/dce/dive/online/diveinfo.html>.

HAGSAND, O. **KTH Home Page**. Acessado em 01/03/06. Disponível em: <http://www.imit.kth.se/~olofh/>.

HANCOCK, D. **Viewpoint: Virtual Reality in Search of Middle Ground**, *IEEE Spectrum*, 32(1):68, Jan 1995.

HITLab. **ARToolkit**. Acessado em 01/03/06. Disponível em: <http://www.hitl.washington.edu/research/artoolkit/>.

JOHNSON, A.; et. al. **The NICE Project: Learning Together in a Virtual World**. Proceedings of VRAIS '98: Atlanta, 1998.

KATO, H.; BILLINGHURST, M. **ARToolKit Download Page**. Acessado em: 01/03/06. Disponível em: <http://www.hitl.washington.edu/research/shared_space/download/>.

KAUFMANN, H. et al. **Construct3D: A Virtual Reality Application for Mathematics and Geometry Education**. Institute of Computer Graphics and Algorithms, Vienna University of Technology, 2000.

KAUFMANN, H. **Geometry Education With Augmented Reality**. 2004. 179f. Universidade de Tecnologia de Viena, Viena, Austria.

KIRNER, C. (1999). **Realidade Virtual, dispositivos e aplicações**. Faculdade de Informática. Marília: SP, 1999.

KIRNER, C. (2004). **Mãos Colaborativas em Ambientes de Realidade Misturada**. I Workshop de Realidade Aumentada. Piracicaba: SP, 2004.

KIRNER, C.; PINHO, M. S. (1996). **Uma Introdução à Realidade Virtual**. SIBGRAPI - Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens. Campos do Jordão, SP. Minicurso. Outubro, 1996. Acessado em: 01/03/06. Disponível em: <<http://grv.inf.pucrs.br/Pagina/TutRV/tutrv.htm#sumario2.2>>.

KIRNER, C.; TORI, R. (2004b). Introdução à Realidade Virtual, Realidade Misturada e Hiper-Realidade. In: Claudio Kirner; Romero Tori. (Ed.). **Realidade Virtual: Conceitos, Tecnologia e Tendências**. 1ed. São Paulo: Editora SBC, 2004, capítulo 1, v. 1, p. 3-20.

KIRNER, C.; ZORZAL, E. R. (2005). **Jogos Educacionais em Ambiente de Realidade Aumentada**. WRA – II Workshop de Realidade Aumentada, 2005.

KIRNER, C.; AKAGUI, D. (2004c). **Desenvolvimento de Aplicações de Realidade Aumentada com ARToolKit**. 2004.

KIRNER, C.; DIZERÓ, W.J., VINCENTIN, V.J. (1998). **Professor Virtual: A Realidade Virtual como Suporte ao Ensino à Distância**. Anais do VI Workshop de Ensino em Informática, Belo Horizonte, Agosto de 1998.

KIRNER, C.; PINHO, M. S.; (2001). **Uma Introdução à Realidade Virtual**. Acessado: 01/03/06. Disponível em: <<http://grv.inf.pucrs.br/Pagina/TutRV/tutrv.htm>>

Laboratório de RV – UNIMEP. (2006). Acessado em: 10/04/06. Disponível em: <<http://www.unimep.br/~ckirner/projetos.htm>>.

LABORATORY FOR SCIENTIFIC VISUAL ANALYSIS. Acessado em: 10/2005. Disponível em: <<http://www.sv.vt.edu/>>

LATTA, J. N.; OBERG, D. J. **A conceptual virtual reality mode**, *IEEE Computer Graphics & Applications*, pp. 23-29. Jan, 1994.

LOGAN, T. **Médicos usam coração virtual em 3D para tratar bebês**. BBC Brasil. Disponível em: <<http://www.netmarkt.com.br/noticia2004/2461.html>>. Acessado em: 10/04/06.

LUCENA, C. J.; MILIDIÚ, R.; BLOIS, M.; TORRES, V.; CARVALHO, G. R. **O Aulanet e as novas tecnologias de informação aplicadas À educação baseada na WEB**, *VI congresso Internacional de educação a distância*.

MEC. **Visão**. Acessado em: 01/02/04. Disponível em: <www.mec.gov.br/semtec/ensmed/ftp/visao.pdf>.

MEDARPA. **Medical Augmented Reality for Patients**. Acessado em 01/03/06. Disponível em: <<http://www.medarpa.de/englisch/index.html>>

MENEZES, B. G. **Terapia com Realidade Virtual**. Acessado em: 15/03/05. Disponível em: <<http://www.basileu-menezes.psc.br/>>.

MILGRAM, P. KISHINO, F. **A Taxonomy of Mixed Reality Visual Displays**. *IEICE Transactions on Information Systems* E77-D (12): 1321-1329, 1994.

NUNAMAKER, J. F. *et al.* (1991) **Electronic meeting and decision rooms. Readings in groupware and computer-supported cooperative work**, *Communications of the ACM*, v. 34, n. 7. p. 40-61.

OMG - Object Management Group. Acessado em: 01/03/06. Disponível em: <<http://www.omg.org>>.

PANTELIDES, V. **Reasons to use Virtual Reality in Education**. *VR in the Schools*, vol. 1, no. 1, jun. 1995.

PERUZZA, A.P.P.M. DISSERTAÇÃO DE MESTRADO. TÍTULO. "**Suporte para Navegação em Ambientes Virtuais Distribuídos**", 1998. Acessado em: 01/03/06. Disponível em: <www.dc.ufscar.br/posgrad/discentes/diss77.htm>.

PINHO, M. S. **Técnicas de Interface em Ambientes Virtuais Colaborativos**. UFRGS, Curso de Pós Graduação em Ciência da Computação. Porto Alegre: RS, 2000. Acessado em 10/11/03. Disponível em: <<http://www.inf.ufrgs.br/~pinho/Proposta.htm>>.

RAPOSO, A.B.; et. al. (2004). **Visão Estereoscópica, Realidade Virtual, Realidade Aumentada e Colaboração**. Disponível em: <www.tecgraf.puc-rio.br/publications/artigo_2004_visao_estereoscopica_realidade_virtual.pdf>. Acessado em: 12/02/06.

RODELLO, I. A. et al. **Análise dos Principais Fatores para o Desenvolvimento de Sistemas Distribuídos de Realidade Virtual**. *Proceedings of the 4o SBC Symposium on Virtual Reality* (2001), Florianópolis, 314-323.

SABBATINI, R.M.E. **Realidade Virtual no Ensino Médico**. Núcleo de Informática Biomédica Universidade Estadual de Campinas - Campinas, Brasil. Disponível em: <<http://www.informaticamedica.org.br/informaticamedica/n0202/sabbatini.htm#Conclusões>>. Acessado em: 10/04/06.

SCHMALSTIEG, D.; FUHRMANN, A.; SZALAVÁRI, Z.; GERVAUTZ, M. **Studierstube – An Environment for Collaboration in Augmented Reality**. Extended abstract in Proceedings of Collaborative Virtual Environments, Nottingham, UK, 1996.

SCRIMGER, R. **TCP/IP – A Bíblia**. Rio de Janeiro: Campus, 2002.

SILICON GRAPHICS - <http://www.sgi.com>

STEREOGRAPHICS – <<http://www.stereographics.com>>

SUN Microsystems. <<http://www.sun.com/index.xml>>

SVR, 2004. **Delta 9**. SENAC – SP.

TANENBAUM, A. S. **Redes de Computadores**. 4 ed. Rio de Janeiro: Elsevier, 2003.

VALLINO, J. **Introduction to Augmented Reality**. Departamento de Engenharia de Software e Instituto de Tecnologia. Rochester: NY, 2002. Disponível em: <<http://www.se.rit.edu/~jrv/research/ar>>

VIRTUAL RESEARCH – <<http://virtualresearch.com>>

VIRTUAL TECHNOLOGIES - <<http://www.immersion.com>>

VRML. **Specification**. Disponível em: <<http://www.web3d.org/x3d/specifications/vrml/>>. Consultado em: 01/09/03.

WEBCT “**Started with WebCT**”, <<http://www.webct.com/quickstart/Getting>> Started with WebCT (consultado em 20/05/2002).

WIKIPÉDIA. Óculos Estereoscópicos. Acessado em 01/03/06. Disponível em: <http://pt.wikipedia.org/wiki/%C3%93culos_estereosc%C3%B3picos>

ZYDA, M. J. **Networking Virtual Environments**. Califórnia, 1996.

ANEXOS

ANEXO 1. EXEMPLO DO CÓDIGO VRML, DA PEÇA DE DOMINÓ DE COR AMARELA E AZUL.

```
#VRML V2.0 utf8
Transform {
  translation  -1 0 0
  children [
Shape {
  appearance Appearance {
    material Material {
      diffuseColor 1 1 0
    }
  }
  geometry Box {
    size 2 0.5 2
  }
}}
Transform {
  translation  1 0 0
  children [
Shape {
  appearance Appearance {
    material Material {
      diffuseColor 1 0 0
    }
  }
  geometry Box {
    size 2 0.5 2
  }
}]
}]}
```

ANEXO 2. EXEMPLO DE UM ARQUIVO “VRML.DATA”, COM NOVOS MARCADORES INSERIDOS.

```
#the number of patterns to be recognized  
4
```

```
#pattern 1  
VRML Wrl/dominoAmV.dat  
Data/peca1  
80.0  
0.0 0.0
```

```
#pattern 2  
VRML Wrl/dominoAmVm.dat  
Data/peca2  
80.0  
0.0 0.0
```

```
#pattern 3  
VRML Wrl/dominoVmA.dat  
Data/peca3  
80.0  
0.0 0.0
```

```
#pattern 4  
VRML Wrl/dominoVBr.dat  
Data/peca4  
80.0  
0.0 0.0
```