



**Universidade Metodista de Piracicaba**  
**Faculdade de Ciências Exatas e da Natureza**  
**Mestrado em Ciência da Computação**

**RASTREAMENTO E VISUALIZAÇÃO DE TRAJETÓRIAS PARA  
TREINAMENTO COM REALIDADE AUMENTADA**

**FABIANO UTIYAMA**

**ORIENTADOR: PROF. DR. CLAUDIO KIRNER**

**PIRACICABA, SP**  
**2006**



**Universidade Metodista de Piracicaba**  
**Faculdade de Ciências Exatas e da Natureza**  
**Mestrado em Ciência da Computação**

**RASTREAMENTO E VISUALIZAÇÃO DE TRAJETÓRIAS PARA  
TREINAMENTO COM REALIDADE AUMENTADA**

**FABIANO UTIYAMA**

**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**

# **RASTREAMENTO E VISUALIZAÇÃO DE TRAJETÓRIAS PARA TREINAMENTO COM REALIDADE AUMENTADA**

AUTOR: FABIANO UTIYAMA

ORIENTADOR: PROF. DR. CLAUDIO KIRNER

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

Prof. Dr. Claudio Kirner - UNIMEP (Orientador)

Prof<sup>a</sup>. Dr<sup>a</sup>. Tereza Gonçalves Kirner - UNIMEP

Prof. Dr. Eduardo Marques - ICMC-USP

À

*Deus, por seu imenso amor*

Aos

*Meus pais, pelo incentivo*

Aos

*Irmãos e amigos, pelo companheirismo*

À

*Raquel, namorada e amiga de todas as horas*

## **AGRADECIMENTOS**

Ao Prof. Dr. Claudio Kirner, o auxílio e a confiança.

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

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

Aos colegas de curso as preciosas trocas de idéias e de conhecimentos.

---

---

## RESUMO

Com o crescente avanço da tecnologia e a necessidade de realizar treinamento nas diversas áreas do conhecimento, tornou-se importante a utilização de novas técnicas para visualização e interação nos ambientes de treinamento. Ambientes de Realidade Aumentada, envolvendo interface computacional dinâmica e dispositivos de interação mais intuitivos, vêm sendo usados em ferramentas de treinamento acadêmicas e comerciais para proporcionar uma experiência mais agradável e resultados satisfatórios quanto ao aprendizado. Este trabalho apresenta uma ferramenta de treinamento que favorece o aprendizado, através da visualização e comparação de trajetórias em ambientes de Realidade Aumentada.

**PALAVRAS-CHAVE:** Realidade Aumentada, Realidade Virtual, Treinamento.

---

---

---

## TRACKING AND VISUALIZATION OF TRAJECTORIES FOR TRAINING WITH AUGMENTED REALITY

### *ABSTRACT*

With the advance of the technology and the necessity to carry out training in several areas of the knowledge, it is necessary to use new techniques for visualization and interaction in training environments. Environments supported by Augmented Reality, with more dynamic computational interfaces and intuitive interaction devices, are being used both in academic and commercial training tools, providing more pleasant experience and satisfactory results related to learning. This work presents a training tool, suitable to learning activities, capable to show and compare visible trajectories in Augmented Reality environments.

**KEYWORDS:** Augmented Reality, Virtual Reality, Training.

---

## SUMÁRIO

<b>LISTA DE FIGURAS.....</b>	<b>V</b>
<b>LISTA DE QUADROS.....</b>	<b>VII</b>
<b>LISTA DE TABELAS.....</b>	<b>VIII</b>
<b>CAPÍTULO 1 - INTRODUÇÃO.....</b>	<b>1</b>
1.1. CARACTERIZAÇÃO DO PROBLEMA.....	1
1.2. MOTIVAÇÃO.....	2
1.3. DELIMITAÇÃO DO ASSUNTO.....	2
1.4. OBJETIVOS DA PESQUISA.....	3
1.5. METODOLOGIA.....	3
1.6. ESTRUTURA DA DISSERTAÇÃO.....	4
<b>CAPÍTULO 2 – REALIDADE VIRTUAL E REALIDADE AUMENTADA.....</b>	<b>5</b>
2.1. REALIDADE VIRTUAL.....	5
2.1.1. REALIDADE VIRTUAL IMERSIVA E NÃO-IMERSIVA.....	7
2.1.2. IMAGEM ESTEREOSCÓPICA.....	8
2.1.3. VRML.....	9
2.1.4. APLICAÇÕES DE REALIDADE VIRTUAL.....	12
2.1.4.1 MEDICINA.....	12
2.1.4.2 EDUCAÇÃO.....	14
2.1.4.3 VISUALIZAÇÃO DE DADOS.....	15
2.2. REALIDADE AUMENTADA.....	16
2.2.1 DISPOSITIVOS.....	18
2.2.2. ALGUNS TIPOS DE SISTEMAS DE REALIDADE AUMENTADA.....	20
2.2.2.1 REALIDADE AUMENTADA BASEADA EM CAPACETE COM VISÃO ÓPTICA.....	20
2.2.2.2 REALIDADE AUMENTADA BASEADA EM CAPACETE COM VISÃO POR VÍDEO.....	21
2.2.2.3 REALIDADE AUMENTADA BASEADA EM MONITOR.....	21
2.2.3. EXEMPLOS DE SISTEMAS DE REALIDADE AUMENTADA.....	22
2.2.3.1 STUDIERSTUBE.....	22
2.2.3.2 ARTOOLKIT.....	23
2.2.3.2.1 APLICAÇÕES COM ARTOOLKIT.....	26
2.2.3.3 MXR TOOLKIT.....	30
<b>CAPÍTULO 3 – TREINAMENTO COM REALIDADE AUMENTADA.....</b>	<b>33</b>
3.1. APLICAÇÕES PARA TREINAMENTO COM REALIDADE AUMENTADA.....	34
3.1.1. PLANEJAMENTO DE OPERAÇÕES DE EXPLORAÇÃO.....	34
3.1.2. SIMULADORES DE VÔO.....	34
3.1.3. MANUTENÇÃO.....	35
3.1.4. MÉDICA.....	36
3.2. FERRAMENTAS PARA TREINAMENTO COM REALIDADE AUMENTADA.....	37
3.2.1 MONTAGEM VIRTUAL USANDO TÉCNICA DE REALIDADE VIRTUAL.....	37
3.2.2 SISTEMA DE TREINAMENTO PARA REFINARIA DE PETRÓLEO.....	38
3.2.3 VIRTUAL REALITY FOR LEGO MINDSTORMS.....	39
3.2.4 AMBIENTE DE TREINAMENTO AUMENTADO PARA REABILITAÇÃO DA EXTENSÃO DOS DEDOS DAS MÃOS.....	40
3.2.5 ART – AUGMENTED REALITY FOR TRAINING.....	40
3.2.6 PROTÓTIPO PARA TREINAMENTO INDUSTRIAL COM REALIDADE AUMENTADA COLABORATIVA.....	42
3.2.7 ART – SISTEMA DE TREINAMENTO E ASSISTÊNCIA PARA MANUTENÇÃO INDUSTRIAL.....	42



3.2.8 SISTEMA PARA MONTAGEM DE MOBILIÁRIO .....	43
<b>CAPÍTULO 4 – RASTREAMENTO E VISUALIZAÇÃO DE TRAJETÓRIAS PARA TREINAMENTO COM REALIDADE AUMENTADA .....</b>	<b>46</b>
4.1. IMPLEMENTAÇÃO DO AMBIENTE AUMENTADO PARA TREINAMENTO .....	46
4.1.1. MODELAGEM DAS TRAJETÓRIAS .....	46
4.1.2. AMBIENTE DE PROGRAMAÇÃO .....	47
4.2. FERRAMENTA PARA TREINAMENTO COM REALIDADE AUMENTADA .....	49
4.2.1. EXIBIÇÃO DA TRAJETÓRIA PADRÃO .....	49
4.2.2. CAPTURA DA TRAJETÓRIA DO USUÁRIO .....	55
4.2.3. AVALIAÇÃO DE DESEMPENHO E APRENDIZAGEM .....	58
4.2.3.1 COMPARAÇÃO DA ROTA FINAL .....	59
4.2.3.2 INTERPOLAÇÃO .....	61
4.2.3.3 MEDIDAS DE DISPERSÃO – DESVIO PADRÃO .....	63
4.3. APLICAÇÕES .....	63
4.3.1. ANÁLISE DO EXPERIMENTO .....	65
4.3.2. POTENCIALIDADES .....	67
4.3.3. RESTRIÇÕES .....	68
<b>CAPÍTULO 5 – CONCLUSÃO .....</b>	<b>69</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>71</b>
<b>ANEXOS .....</b>	<b>78</b>
ANEXO 1 – PARTE DO PROTÓTIPO IMPLEMENTADO EM DELPHI .....	78
ANEXO 2 – PARTE DO PROTÓTIPO IMPLEMENTADO EM JAVA .....	83

## LISTA DE FIGURAS

FIGURA 1 – LUVA USADA EM REALIDADE VIRTUAL (GENERAL REALITY COMPANY, 2005).....	08
FIGURA 2 – CAPACETE USADO EM REALIDADE VIRTUAL (VREALITIES, 2005).....	08
FIGURA 3 - ESQUEMA DO FUNCIONAMENTO DA IMAGEM ESTEREOSCÓPICA (RAPOSO, 2004).....	09
FIGURA 4 - EXEMPPO DE DOIS OBJETOS, UM CUBO E UM CILINDRO, EM REALIDADE VIRTUAL .....	10
FIGURA 5 – SISTEMA DE REALIDADE VIRTUAL APLICADO EM CIRURGIAS (KISMET, 2006).....	13
FIGURA 6 – SISTEMA DE REALIDADE VIRTUAL APLICADO À ANATOMIA (ACKERMAN, 2006).....	14
FIGURA 7 – OPEN VIZ (WAYNE, 2006).....	15
FIGURA 8 – GRÁFICO DO 3D ACTIVE CHART (ACTIVE CHART, 2004) .....	16
FIGURA 9 – EXEMPLO DE REALIDADE AUMENTADA - VASO E CARRO INSERIDOS NO MUNDO REAL (KIRNER, 2004B) .....	17
FIGURA 10 – DISPOSITIVO GERADOR DE TEXTURA E GEOMETRIA (PINHO, 2002) ....	19
FIGURA 11 – DISPOSITIVO DE GERAÇÃO DE FORÇA (VIRTUAL TECHNOLOGY, 2006).....	19
FIGURA 12 – ESQUEMA DE VISUALIZAÇÃO EM REALIDADE AUMENTADA ATRAVÉS DO CAPACETE COM VISÃO ÓPTICA (AZUMA, 1997) (KIRNER, 2005B) .....	20
FIGURA 13 - ESQUEMA DE VISUALIZAÇÃO EM REALIDADE AUMENTADA ATRAVÉS DE CAPACETE COM VISÃO POR VÍDEO (AZUMA, 1997) (KIRNER, 2005B) .....	21
FIGURA 14 - ESQUEMA DE VISUALIZAÇÃO EM REALIDADE AUMENTADA ATRAVÉS DO MONITOR (AZUMA, 1997) (KIRNER, 2005B).....	22
FIGURA 15 - EXEMPLO DO STUDIERSTUBE (SCHMALSTIEG, 2002) .....	23
FIGURA 16 – ESTRUTURA DE ARQUIVOS DO ARTOOLKIT (ARTOOLKIT, 2004) .....	24
FIGURA 17 - EXEMPLO DE MARCADOR .....	25
FIGURA 18 – MARCADOR COM O OBJETO VIRTUAL ASSOCIADO (ARTOOLKIT, 2004) .....	25
FIGURA 19 - DIAGRAMA DE PROCESSAMENTO DAS IMAGENS NO ARTOOLKIT ADAPTADO DE KATO, 2000 .....	26
FIGURA 20 – PLACA DE CONTROLE ALTERANDO O OBEJTO VIRTUAL (SANTIN, 2004). .....	27
FIGURA 21 – SEQÜÊNCIA DE AÇÕES PARA O TRANSPORTE DO OBJETO VIRTUAL COM AUXÍLIO DA PÁ (SANTIN, 2004) .....	27
FIGURA 22 – EXEMPLOS DE USO DO VISUALIZADOR DE DADOS COM REALIDADE AUMENTADA (BUK, 2004) .....	29
FIGURA 23 – VISUALIZAÇÃO DINÂMICA (BUK, 2004) .....	30
FIGURA 24 – EXEMPLO DE INSERÇÃO DE PESSOAS NO AMBIENTE VIRTUAL (MXR TOOLKIT, 2006) .....	31
FIGURA 25 – FUNCIONAMENTO DO SISTEMA MXR TOOLKIT (MXR TOOLKIT, 2006). ..	31
FIGURA 26 – APLICAÇÃO PARA TREINAMENTO MILITAR (DONFREWARE, 2006) .	34
FIGURA 27 – SIMULADOR DE VÔO DA EMBRAER (EMBRAER, 2006) .....	35
FIGURA 28 – APLICAÇÃO PARA TREINAMENTO EM MANUTENÇÃO COM REALIDADE AUMENTADA (MICROVISION, 2006) .....	36

FIGURA 29 – APLICAÇÃO PARA TREINAMENTO NA ÁREA MÉDICA COM REALIDADE AUMENTADA (MICROVISION, 2006) .....	37
FIGURA 30 – APLICAÇÃO PARA TREINAMENTO NA ÁREA MÉDICA COM REALIDADE AUMENTADA (MICROVISION, 2006) .....	38
FIGURA 31 – VEÍCULO MONTADO EM UM AMBIENTE VIRTUAL (BROWER, 2004) ....	39
FIGURA 32 – VEÍCULO VIRTUAL SEGUINDO A TRAJETÓRIA EM ESPIRAL (BROWER, 2004).....	39
FIGURA 33 – SEGUNDO SISTEMA AUMENTADO PARA A REABILITAÇÃO DE PACIENTES (LUO, 2006) .....	40
FIGURA 34 – SETA INDICANDO AS PARTES RELACIONADAS (KUSTABORDER, 1999) .....	41
FIGURA 35 – EXPERIMENTO REALIZADO COM UM CAMINHÃO DE BRINQUEDO (KUSTABORDER, 1999) .....	41
FIGURA 36 – UM USUÁRIO TRABALHANDO COM UMA PLACA DE COMPUTADOR (ZHONG, 2005).....	42
FIGURA 37 – A PRIMEIRA IMAGEM MOSTRA UM USUÁRIO LOCAL, E A SEGUNDA IMAGEM UM USUÁRIO REMOTO, AMBOS VISUALIZANDO UM CHIP VIRTUAL SOBRE UMA PLACA (ZHONG, 2006).....	42
FIGURA 38 – SISTEMA DE TREINAMENTO PARA MANUTENÇÃO INDUSTRIAL (SCHWALD, 2006).....	43
FIGURA 39 – SISTEMA PARA MONTAGEM DE MOBILIÁRIO (ZAUNER, 2003) .....	44
FIGURA 40 – SEGUNDO MONTAGEM DE MOBILIÁRIO (ZAUNER, 2003) .....	44
FIGURA 41 – MARCADORES COM SEUS RESPECTIVOS OBJETOS VIRTUAIS UTILIZADOS NO PROTÓTIPO.....	47
FIGURA 42 – ESQUEMA DO FUNCIONAMENTO DO SISTEMA.....	47
FIGURA 43 – TELA DE CONTROLE .....	48
FIGURA 44 – TABELAS CRIADAS NO BANCO DE DADOS FIREBIRD .....	49
FIGURA 45 – FUNCIONAMENTO DA FERRAMENTA ARTOOLKIT COM A TÉCNICA DE ALTERAÇÃO DINÂMICA DE OBJETOS VIRTUAIS .....	50
FIGURA 46 – TELA DE CONTROLE – TRAJETÓRIA PADRÃO .....	53
FIGURA 47 – TRAJETÓRIA EXIBIDA COM ESFERAS .....	54
FIGURA 48 – TRAJETÓRIA EXIBIDA COM EXTRUSÃO.....	55
FIGURA 49 – MARCADOR UTILIZADO PARA REALIZAR O TREINAMENTO.....	55
FIGURA 50 – TELA DE CONTROLE – TRAJETÓRIA CAPTURADA .....	58
FIGURA 51 – TELA DE AVALIAÇÃO DO USUÁRIO.....	60
FIGURA 52 – INTERPOLAÇÃO ENTRE DOIS PONTOS .....	61
FIGURA 53 – INTERPOLAÇÃO, IDENTIFICANDO PONTOS CORRESPONDENTES .....	62
FIGURA 54 – CÂMERA, COLOCADA DE CIMA PARA BAIXO.....	62
FIGURA 55 – APLICAÇÃO EM UM JOGO DE DAMA .....	64
FIGURA 56 – TRAJETO POSSÍVEL EXIBIDO COM EXTRUSÃO.....	64
FIGURA 57 – OUTRO TRAJETO POSSÍVEL EXIBIDO COM EXTRUSÃO .....	64
FIGURA 58 – TRAJETO POSSÍVEL EXIBIDO COM ESFERAS .....	65
FIGURA 59 – SEGUNDO TRAJETO POSSÍVEL EXIBIDO COM ESFERAS.....	65
FIGURA 60 – EXPERIMENTO COM O USUÁRIO .....	66
FIGURA 61 – EXPERIMENTO COM O USUÁRIO – TELA DE AVALIAÇÃO DE DESEMPENHO .....	66

## LISTA DE QUADROS

QUADRO 1 – CÓDIGO REFERENTE À APLICAÇÃO MOSTRADA NA FIGURA 4 .....	11
QUADRO 2 – MÓDULO DE SOM .....	28
QUADRO 3 – CÓDIGO VRML, COM A MARCA #, PARA ATUALIZAÇÃO DOS DADOS DE TRANSLAÇÃO DO OBJETO VIRTUAL .....	50
QUADRO 4 – CÓDIGO PARA ATUALIZAÇÃO DO ARQUIVO VRML, COM DADOS DE TRANSLAÇÃO VINDOS DO BANCO DE DADOS.....	51
QUADRO 5 – CÓDIGO VRML, COM A MARCA #, PARA ATUALIZAÇÃO DOS DADOS DE TRANSLAÇÃO DO OBJETO VIRTUAL, UTILIZANDO EXTRUSÃO .....	52
QUADRO 6 – CÓDIGO ALTERADO NO ARTOOLKIT PARA EXIBIR A TRAJETÓRIA PADRÃO .....	54
QUADRO 7 – CÓDIGO PARA ATUALIZAÇÃO DO ARQUIVO VRML, UTILIZANDO EXTRUSÃO.....	57
QUADRO 8 – CÓDIGO DA TELA DE AVALIAÇÃO DO USUÁRIO .....	60

## LISTA DE TABELAS

<u>TABELA 1 - COMPARAÇÃO DOS TRAJETOS SEM TREINAMENTO E COM TREINAMENTO</u>	67
---	----

# CAPÍTULO 1

## 1. INTRODUÇÃO

### 1.1. CARACTERIZAÇÃO DO PROBLEMA

Percebe-se, atualmente, um momento de grandes mudanças, tanto na área tecnológica, como em outras diversas áreas (TAJRA, 2001). O conhecimento e a informação adquiriram grande importância na vida das pessoas. Desde criança, é necessário aprender, e para isto realiza-se treinamento, seja com a ajuda de um professor ou não.

O treinamento está profundamente associado ao cotidiano de todos os seres humanos. Quase todas as pessoas precisam ser treinadas para manusear um certo tipo de equipamento em uma empresa, para atualizar-se quanto a este mesmo equipamento, ou para aprender outras habilidades necessárias em seu dia a dia.

As aplicações computacionais, aliadas a diversas tecnologias como a Internet, vieram contribuir para que o treinamento se torne mais eficaz e interessante. Mesmo assim, quando realizado sem a presença de um instrutor, o treinamento pode tornar-se cansativo e pouco atraente.

A utilização de recursos avançados de áudio e vídeo, como interfaces tridimensionais, em ambientes computacionais mais intuitivos, contribuem para um treinamento mais atrativo. Um exemplo desta interface é a Realidade Virtual (RV) e suas variantes.

Uma das variantes é a Realidade Aumentada (RA), que consiste na inserção de elementos virtuais em cenas de um ambiente real (AZUMA, 1997).

Quando a interface de RA permite a utilização das mãos como elemento de interação, além de propiciar a manipulação desses objetos de

uma forma mais intuitiva, torna viável o desenvolvimento de inúmeras aplicações que beneficiam o treinamento humano, motivando mais o usuário para as tarefas a serem realizadas.

## **1.2. MOTIVAÇÃO**

A motivação para a realização deste trabalho é a busca por um meio de realizar um treinamento que possua interatividade, contribuindo para um aprendizado rápido e eficaz, utilizando-se de tecnologias computacionais.

O ser humano possui grande capacidade para assimilar informações na forma de imagens, pois está relacionado diretamente com atividades cognitivas (GERSHON, 1998; KIRNER, 2000). Também é mais intuitivo utilizar as próprias mãos para interagir em um ambiente computacional, pois é natural, atrativo e interessante. A união do sistema computacional com a capacidade humana de visualizar informações gráficas e interagir com estas informações, utilizando seu próprio corpo, permite a realização de treinamentos com resultados mais significativos.

Este trabalho aborda o treinamento de usuários em ambientes de Realidade Aumentada, usando visualização de trajetórias e captura de movimentos no ambiente real, permitindo a inserção do usuário em um ambiente potencializado e apropriado para seguir as instruções de seu treinamento.

## **1.3. DELIMITAÇÃO DO ASSUNTO**

O trabalho aborda uma ferramenta para treinamento de usuários em ambiente de Realidade Aumentada. Para o desenvolvimento, foram utilizados: banco de dados, linguagem de modelagem de objetos virtuais, linguagem de programação Delphi, e software de desenvolvimento do sistema de Realidade Aumentada.

#### **1.4. OBJETIVOS DA PESQUISA**

O objetivo deste trabalho é desenvolver técnicas inovadoras de apoio ao treinamento, que proporcionem um melhor aprendizado, através do uso de processos computacionais visuais, baseados em Realidade Aumentada, software de modelagem, visualização de dados e avaliação de desempenho.

Para isto, foi desenvolvido um protótipo em um ambiente de Realidade Aumentada, que exibe trajetórias, montadas com o uso de dados vindos de um banco de dados e representadas por objetos tridimensionais, para que o usuário possa segui-las, de acordo com as instruções visuais e/ou sonoras.

Com esse protótipo, pretende-se contribuir para uma melhoria nos processos de treinamento e aprendizado, facilitando a interação do usuário com o ambiente computacional.

As metas do projeto são:

- 1) apresentar uma forma inovadora de realizar um treinamento;
- 2) desenvolver uma ferramenta para treinamento em um ambiente de Realidade Aumentada.

#### **1.5. METODOLOGIA**

Para a execução deste trabalho, realizou-se uma revisão bibliográfica sobre: Realidade Virtual, Realidade Aumentada, treinamento em ambientes aumentados, Delphi, C e C++, VRML, software para desenvolvimento de aplicações de Realidade Aumentada, e banco de dados.

Em seguida, foi desenvolvido um protótipo simples, que implementa um ambiente de Realidade Aumentada, mostrando algumas trajetórias. Para isto, utilizou-se VRML (linguagem de Realidade Virtual), linguagem de programação Delphi e o software para desenvolvimento de aplicações de Realidade Aumentada ARToolKit (ARTOOLKIT, 2004). Esse



protótipo evoluiu para a pesquisa e busca de trajetórias pré-definidas em banco de dados, além do armazenamento de trajetórias produzidas pelo usuário.

## **1.6. ESTRUTURA DA DISSERTAÇÃO**

Este trabalho está dividido em seis partes. A primeira descreve uma introdução ao assunto. A segunda parte apresenta a fundamentação teórica das tecnologias, nas quais o trabalho está baseado, descrevendo os ambientes de Realidade Virtual e Realidade Aumentada. A terceira parte apresenta uma discussão sobre treinamento com Realidade Aumentada, mostrando exemplos de algumas aplicações e ferramentas existentes. Na quarta parte, é apresentado o projeto do ambiente de Realidade Aumentada voltado para treinamento, relatando o desenvolvimento da pesquisa e da ferramenta computacional. Por fim, são apresentadas a conclusão e as referências bibliográficas.

# CAPÍTULO 2

## 2. REALIDADE VIRTUAL E REALIDADE AUMENTADA

### 2.1. REALIDADE VIRTUAL

Atualmente há um crescente número de aplicações em Realidade Virtual, nas diversas áreas do conhecimento. Segundo Trindade (1999), esse crescimento é devido a alguns fatores que muito contribuíram para o desenvolvimento da RV, como: colocação, no mercado, de computadores de uso pessoal com apreciáveis capacidades gráficas a custos reduzidos; aperfeiçoamento e desenvolvimento de novos dispositivos de interface, cada vez mais leves e portáteis e a crescente acessibilidade a este tipo de tecnologia; aparecimento, no mercado, de ferramentas de desenvolvimento de ambientes virtuais.

Dentre as muitas definições para Realidade Virtual, uma delas é a de Kirner (1999): *“É uma técnica avançada de interface, onde o usuário pode realizar imersão, navegação e interação em um mundo virtual sintético, gerado por computador, usando dispositivos multisensoriais, em tempo real”*.

Outra definição encontrada é a de que Realidade Virtual é uma forma das pessoas visualizarem, manipularem e interagirem com computadores e dados extremamente complexos (AUKSTAKALNIS, 1992).

Sherman (2003), define Realidade Virtual como um meio composto de simulações computacionais interativas, que sentem as posições e ações dos usuários e substituem ou aumentam a resposta a um ou mais sentidos, dando uma sensação de imersão mental ou presença na simulação – o mundo virtual.

De acordo com White (2002), a Realidade Virtual pode ser considerada como um protótipo ou uma realidade alternativa, onde é possível construir ambientes que não existem no mundo real.

Diante de tantas definições, pode-se dizer que Realidade Virtual é uma interface avançada tridimensional, onde o usuário pode interagir e navegar facilmente, utilizando-se de dispositivos multisensoriais, em tempo real.

Em Realidade Virtual são necessárias três características fundamentais: imersão, interação e envolvimento (KIRNER, 1999); (KNIGHT, 1998); (WLOKA, 1995); (KIRNER, 2000); (SHERMAN, 2003). Imersão significa sentir-se ativo dentro do ambiente, utilizando os sentidos de visão, audição e tato. A interação ocorre quando o usuário modifica instantaneamente o mundo virtual e suas ações sobre o mesmo. Envolvimento é o grau de engajamento do usuário em uma atividade no mundo virtual.

A Realidade Virtual difere de outras abordagens de conceitos computacionais tridimensionais por seus sistemas serem interativos em tempo real, ou seja, os usuários movimentam-se e interagem uns com os outros e com os objetos do ambiente (DIEHL, 2001); (WLOKA, 1995). A navegação é tridimensional (3D), ocorrendo nos eixos x, y e z, da mesma forma como ocorre no mundo real. Para visualizar as imagens em três dimensões, é necessário também lidar com os dados em três dimensões (x, y, z), o que afeta o modo como os dados são representados.

De acordo com BISHOP (1992), um sistema de Realidade Virtual envolve estudos e recursos ligados à percepção, hardware, software, interface do usuário e fatores humanos. Para o desenvolvimento de sistemas de Realidade Virtual é necessário ter conhecimento sobre dispositivos não convencionais de Entrada/Saída, 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 áreas.

O usuário de Realidade Virtual pode navegar livremente entre os objetos do ambiente, interagir e tomar decisões sobre as informações que estão sendo representadas pelos mesmos. O ambiente virtual, além de conter os objetos virtuais, também pode apresentar informações adicionais, como uma caixa de texto ou links para sites da Internet.

### **2.1.1. REALIDADE VIRTUAL IMERSIVA E NÃO-IMERSIVA**

Existem dois tipos de Realidade Virtual: a imersiva e a não-imersiva.

Os Sistemas Imersivos são o que se pode chamar de Realidade Virtual tradicional, de acordo com Pinho (2002). Neles, o usuário veste um conjunto de aparatos tecnológicos, como capacete de Realidade Virtual, luva, rastreador de posição e fones de ouvido. Com estes equipamentos, o usuário é “desligado do mundo real” e passa a visualizar, ouvir e sentir apenas os estímulos gerados por um sistema computacional. As ações do seu corpo são interpretadas como a única forma de entrada dos dados.

Os equipamentos utilizados em sistemas de Realidade Virtual Imersivos estão cada vez mais sofisticados, envolvendo luvas com sensores de tato e força (Figura 1) para dar a sensação de toque, peso, força, fazendo com que o usuário realmente sinta as mesmas sensações do mundo real no mundo virtual.

Os capacetes (Figura 2) viabilizam a visão estereoscópica, baseadas na apresentação de imagens levemente diferentes para cada olho. O cérebro humano recebe a imagem de cada olho e processa as informações, causando a impressão de profundidade.

A combinação dos sentidos de visão, audição e tato irá causar o efeito de percepção no mundo virtual.



**Figura 1 - Luva usada em Realidade Virtual (GENERAL REALITY COMPANY, 2005)**



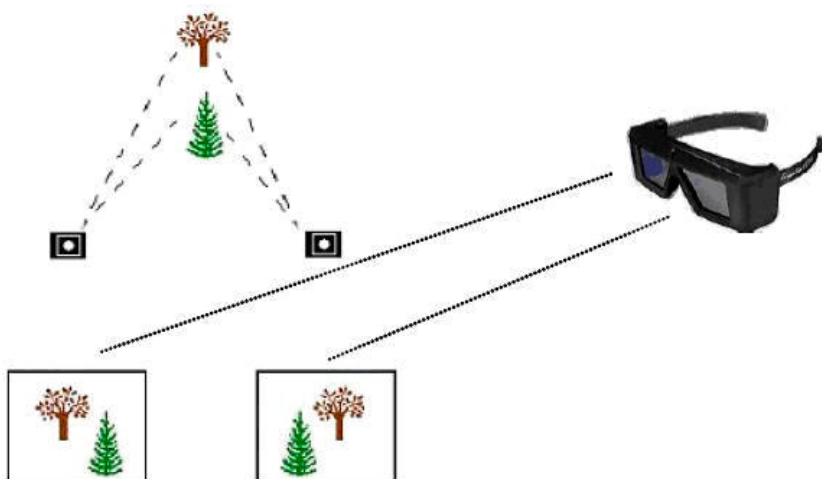
**Figura 2 - Capacete usado em Realidade Virtual (VREALITIES, 2005)**

Os sistemas não imersivos são aqueles nos quais o usuário vê o universo virtual por uma tela convencional de computador ou de uma projeção (PINHO, 2002). Nesta classe, enquadram-se hoje a grande maioria dos jogos eletrônicos e os sistemas interativos de navegação. O som destes ambientes é produzido por caixas de som colocadas à frente do usuário, como as que encontramos nos tradicionais kits multimídia. Alguns destes sistemas já possuem dispositivos especiais de interação como mouse 3D, luvas eletrônicas e óculos para visão estereoscópica. Nestes ambientes, o usuário deve manter-se olhando para a tela para poder ver o mundo virtual.

### **2.1.2. IMAGEM ESTEREOSCÓPICA**

Como mencionado anteriormente, os capacetes ou óculos especiais, utilizados na Realidade Virtual, possuem imagens estereoscópicas, ou seja, imagem com impressão de profundidade. O sistema visual humano é baseado em duas imagens diferentes, sendo uma do olho direito e outra do olho esquerdo, ou seja, cada olho vê a mesma imagem de pontos diferentes. Essas imagens são processadas no cérebro humano, e essa diferença de pontos de visão (disparidade binocular) é o que gera a impressão de profundidade (KNIGHT, 1998); (MALETIC, 2001).

Para formar imagens 3D no computador, são utilizados dois pontos de visão da mesma imagem, uma para o olho direito e outra para o olho esquerdo. A fusão dessas imagens resulta numa cena tridimensional, com a sensação de profundidade. Para visualizar essa imagem no computador, há a necessidade do uso de óculos especiais, pois o monitor irá apresentar essas imagens sobrepostas, estampadas na forma bidimensional. Os óculos especiais podem filtrar ou alternar a imagem do monitor para cada um dos olhos, sendo que cada uma irá ser processada no cérebro, gerando, então, a imagem tridimensional, como pode ser visto na Figura 3. Na fusão das imagens, o usuário a visualiza em estéreo, ou seja, com a impressão de profundidade da imagem apresentada (RAPOSO, 2004).



**Figura 3 - Esquema do funcionamento da imagem estereoscópica (RAPOSO, 2004)**

### **2.1.3. VRML**

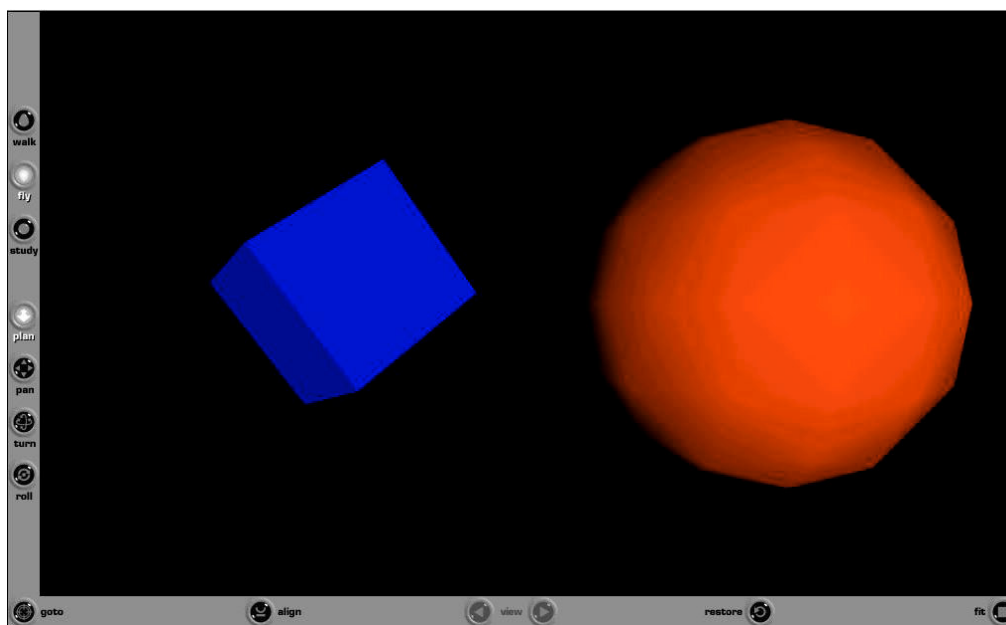
A linguagem VRML (Virtual Reality Modeling Language - Linguagem de Modelagem de Realidade Virtual) foi criada pelo Web3D Consortium para o desenvolvimento de aplicações para a Internet (INTERNATIONAL STANDARD ISO/IEC 14772-1,1997).

VRML é um formato de arquivo para descrever objetos e mundos tridimensionais, podendo ser acessada via Internet, Intranet e sistemas locais (INTERNATIONAL STANDARD ISO/IEC 14772-1, 1997); (DIEHL, 2001).

A proposta dessa linguagem é tornar-se padrão para gráficos e mídias 3D, sendo utilizada em várias áreas de aplicações, como apresentações multimídia, entretenimento, educação, medicina, dentre outras.

Essa linguagem permite descrição de hierarquias simples com o uso de formas primitivas, tais como cilindro, cubo e esfera. Recursos mais complexos podem ser definidos, como superfície de materiais, texturização e nível de detalhes. Os objetos também podem ser usados como conexão para outros objetos ou endereços da Internet. Outras características incluem a transformação de objetos, mudanças de pontos de vista, iluminação, renderização. etc. É possível também trabalhar com outros objetos de arquivos utilizados na Internet, como sons e animação.

Conforme pode ser visto na Figura 4, tem-se dois objetos, um cubo e um cilindro, no ambiente de Realidade Virtual.



**Figura 4 - Exemplo de dois objetos, um cubo e um cilindro, em Realidade Virtual**

O código referente à aplicação mostrada na Figura 4, é apresentado a seguir, no Quadro 1.

**Quadro 1 – Código referente à aplicação mostrada na Figura 4**

```

#VRML V2.0 utf8
Transform {
  children [
    Transform {
      translation 3 0 1
      children [
        Shape {
          geometry Sphere { radius 2.3 }
          appearance Appearance {
            material Material { diffuseColor 1 0 0 }
          }
        }
      ]
    }
    Transform {
      translation -2.4 .2 1
      rotation 0 1 1 .9
      children [
        Shape {
          geometry Box {}
          appearance Appearance {
            material Material { diffuseColor 0 0 1 }
          }
        }
      ]
    }
  ]
}

```

A linguagem pode ser escrita em qualquer editor de textos e visualizada em um *browser*, através de um plug-in, como o cortona, por exemplo. Existem também pacotes para desenvolvedores com programas que verificam a sintaxe e corrigem o código, além de ferramentas de autoria que geram código VRML.

Cada arquivo VRML (INTERNATIONAL STANDARD ISO/IEC 14772-1, 1997); (DIEHL, 2001) apresenta as seguintes propriedades:

- Estabelece as coordenadas para cada objeto definido no arquivo;
- Define e compõe um grupo 3D de objetos multimídia;
- Pode especificar conexões para outros arquivos e aplicações;



- Pode definir o comportamento dos objetos.

O VRML também pode ser utilizado com outras linguagens de programação, como Delphi ou Java, permitindo a troca de informações. Para a comunicação entre Java e VRML, pode ser implementada através da interface EAI (External Authoring Interface), de acordo com Morcrette (1999).

#### **2.1.4. APLICAÇÕES DE REALIDADE VIRTUAL**

As possibilidades de aplicações são muitas, sendo difícil prever onde os ganhos e os benefícios da Realidade Virtual serão mais significativos. O certo é que não haverá um único padrão nas interfaces de Realidade Virtual, de acordo com Pinho (2002). A tecnologia e as limitações de custos farão com que o tipo de aplicação defina o nível de sofisticação da tecnologia a ser aplicada. Neste cenário, muitas aplicações, na busca de soluções para problemas específicos, acabarão por gerar novos usos e soluções para problemas de outras áreas.

Com todas as novas tecnologias disponíveis, é tentador pensar que os novos conceitos envolvidos são a solução para todos os problemas. Se esta for a visão do pesquisador de Realidade Virtual, corre-se o risco de que soluções menos sofisticadas, mas com uma relação custo/benefício mais favorável, sejam esquecidas na ânsia de tomar-se a dianteira muito rapidamente.

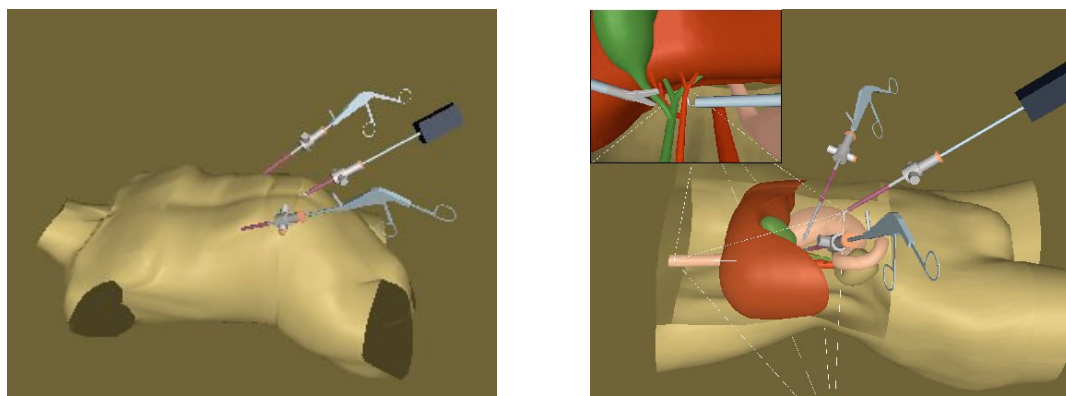
A seguir, serão apresentadas algumas aplicações de Realidade Virtual, em diversas áreas.

##### **2.1.4.1. MEDICINA**

Os sistemas de simulação produzem ambientes que buscam recriar experiências da vida real. É tradicional o uso de simuladores para treinamento de pilotos e de tripulações de barcos. Na área médica, entretanto, o uso desta tecnologia está iniciando.

Uma das primeiras experiências neste campo, que pretendeu ser tanto uma ferramenta de ensino, quanto de prática (treinamento) para procedimentos cirúrgicos (AUKSTAKALNIS, 1992) foi desenvolvida pelo Silas B. Hayes Army Community Hospital em Fort Old (CA). Partindo apenas de modelos da região abdominal, o sistema permite, apesar das imagens não terem alto grau de foto-realismo, uma interatividade excelente na manipulação dos órgãos virtuais. Pode-se, por exemplo, pegar e manipular instrumentos como bisturis e grampos virtuais e aplicá-los sobre o modelo. É possível também analisar em detalhe as relações anatômicas entre cada um dos órgãos e navegar ao redor ou dentro dos mesmos.

Outra área de aplicação da Realidade Virtual em cirurgias é nas chamadas CMI (Cirurgias Minimamente Invasivas) ou Cirurgias Laparoscópicas (Figura 5). Estes procedimentos, que nos EUA já chegaram ao patamar de 5 milhões de cirurgias por ano, segundo Pinho (2002), são realizadas fazendo-se um pequeno corte na pele do paciente e, através deste inserindo-se uma câmera de vídeo e instrumentos de corte. Durante a cirurgia o médico é obrigado a olhar para um monitor não podendo olhar para suas mãos. Esta difícil operação exige do médico um alto grau de destreza e uma capacidade de coordenação olho-mão bastante grande. Estas habilidades, por sua vez, só são obtidas com muito treinamento.

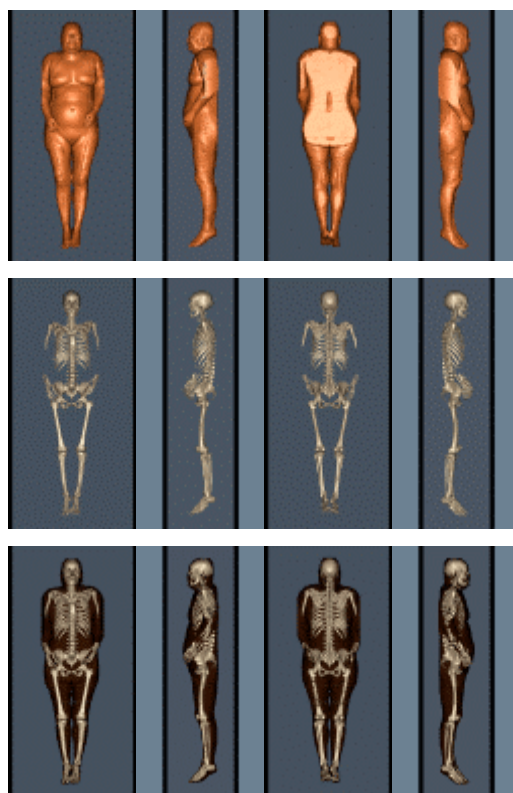


**Figura 5 – Sistema de Realidade Virtual aplicado em cirurgias (KISMET, 2006)**

Uma outra área da medicina, é o ensino da anatomia, que é basicamente ilustrativo. Além dos cadáveres, a ferramenta mais usada pelos

estudantes são os livros de folhas transparentes onde cada uma delas contém a imagem de uma parte do corpo humano, os chamados “atlas de anatomia” (PINHO, 2002).

De uma maneira bastante direta, um modelo virtual de um corpo humano pode substituir estes livros. Pensando nisto, a National Library of Medicine dos Estados Unidos, mantém o projeto Visible Human (ACKERMAN, 2000) que criou, em 1996, um modelo tridimensional de um corpo humano masculino a partir de imagens tomográficas e fatias reais (Figura 6). Em 1998 foi criado um modelo feminino.



**Figura 6 – Sistema de Realidade Virtual aplicado à Anatomia (ACKERMAN, 2006)**

#### **2.1.4.2. EDUCAÇÃO**

Educação é fundamentalmente um processo de exploração, de descoberta, de observação e de construção do conhecimento a partir desta interação com o mundo (PINHO, 2002).

Em aplicações educacionais, a potencialidade da Realidade Virtual está no fato de permitir que se explore alguns ambientes, processos ou objetos, não através de livros, fotos, filmes ou aulas, mas através da manipulação e análise virtual do próprio alvo do estudo.

A física é uma das áreas que mais se presta ao aprendizado por experimentação e observação de fenômenos. A idéia de um laboratório virtual de física é possibilitar a experimentação e a observação de fenômenos de uma forma que não é possível no mundo real.

### 2.1.4.3. VISUALIZAÇÃO DE DADOS

Com o acelerado crescimento dos tipos, das fontes e do volume das informações que estão sendo produzidas atualmente por cientistas, economistas, engenheiros ou executivos, existe uma grande demanda por novas formas de apresentação destes dados. Aliado a isto, os avanços na área de informática fazem do desenvolvimento de novas abordagens para a visualização da informação, uma área bastante promissora.

A Visualização Científica é usada em computação gráfica para auxiliar, com imagens, o entendimento de dados complexos em geral, em grande quantidade, mostrando conceitos científicos, resultados de simulações ou de coleta de dados (PINHO, 2002). A Figura 7 exhibe a ferramenta Open Viz, utilizada em visualização de informações e na tomada de decisões.

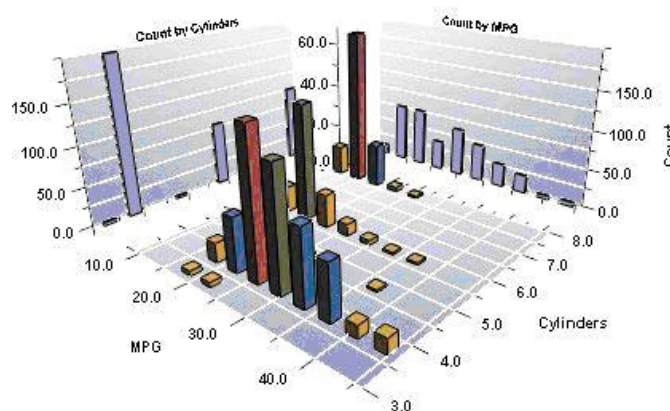


Figura 7 - Open Viz (WAYNE, 2006)

A Figura 8 mostra a ferramenta 3D Active Chart, que é um visualizador tridimensional de informações, no qual as informações são obtidas através de banco de dados, cuja extensão é dbf, ou seja, utiliza-se o banco de dados em Dbase (ACTIVE CHART, 2004).

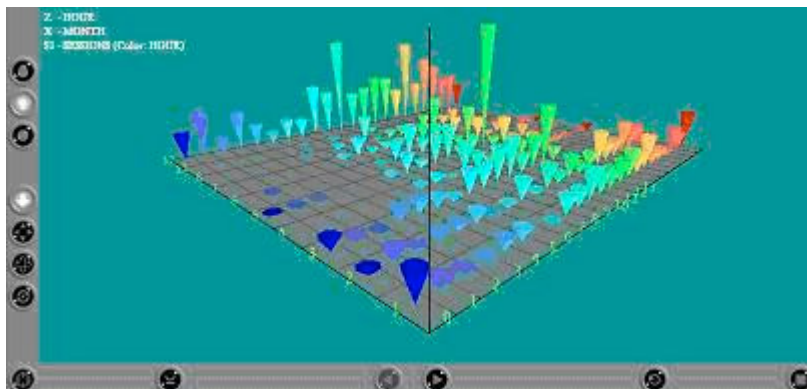


Figura 8 - Gráfico do 3D Active Chart (ACTIVE CHART, 2004)

## 2.2. REALIDADE AUMENTADA

De acordo com Azuma (1995), a Realidade Aumentada teve seu início a partir da Realidade Virtual sobreposta ao ambiente real, de forma que o usuário possa interagir com objetos virtuais imersos no ambiente real. Por meio dessa tecnologia, o usuário pode manipular, acrescentar ou remover objetos do ambiente (AHLERS, 1995; AZUMA, 2001).

As técnicas computacionais de Realidade Aumentada fazem com que o usuário manipule os objetos virtuais de maneira natural. O usuário recebe as imagens do mundo real por meio de câmeras colocadas no ambiente, e se vê imerso em uma representação gráfica, a qual combina imagens de vídeo do mundo real e imagens virtuais do sistema computacional, trazendo para o ambiente do usuário a visualização dos objetos do ambiente de Realidade Virtual.

A Figura 9 apresenta um exemplo de inserção de objetos virtuais no ambiente real, podendo-se adicionar objetos como um vaso e um carro virtual em uma mesa real, enriquecendo, dessa forma, o ambiente real.



**Figura 9 - Exemplo de Realidade Aumentada - vaso e carro inseridos no mundo real (KIRNER, 2004b)**

Para estar inserido no ambiente de Realidade Aumentada, o usuário pode utilizar capacetes ou visualizar a imagem através de um monitor de vídeo, da mesma maneira que a Realidade Virtual (KIRNER, 2005a).

O sistema de Realidade Aumentada pode utilizar os mesmos dispositivos de hardware (monitor de vídeo, capacetes, óculos especiais) usados em Realidade Virtual (FEINER, 2002), além de câmeras para capturar as imagens do mundo real.

Ao fazer uso desses dispositivos, é preciso que o usuário possua treinamento para tal, pois sua posição e orientação influenciam na organização das informações contidas na imagem visualizada. Koller (1997) apresenta alguns problemas que podem ocorrer, devido à colocação errada da câmera, da quantidade de luz incidente e de como as informações são inseridas e depois apresentadas ao usuário. O autor foca os problemas relacionados à câmera e apresenta sua solução através de cálculos matemáticos. A visualização, usando ferramentas como o ARToolkit, pode ocorrer naturalmente, pois está inserida no ambiente do próprio usuário.

A Realidade Aumentada pode ser aplicada em diversas áreas, como treinamento, medicina, manufatura, tele-robótica, área militar, engenharia, dentre outras.

### 2.2.1. DISPOSITIVOS

Além dos dispositivos vistos anteriormente (luva e capacete), alguns outros, como os dispositivos sonoros e os hápticos, colaboram para a imersão e o realismo em ambientes virtuais e aumentados.

A adição de sons, em qualquer processo de exibição de imagens que pretenda ser interativo, torna a exibição muito mais realista. Quando em uma tela, aparece no centro uma bola quicando, a agregação de um som mono, com a simples reprodução do som real sincronizado com o movimento da bola, já produz uma sensação de realismo muito maior para quem observa, segundo Pinho (2002). Em diversos casos, o estímulo auditivo não é somente uma forma de aumentar o realismo de uma simulação, uma vez que sua ausência pode tornar inviável uma aplicação.

A forma mais simples de geração de som tridimensional em um ambiente virtual é a disposição de alto-falantes ao redor do usuário e o correto acionamento destes conforme a necessidade de emissão do som. Esta alternativa, entretanto, pode necessitar de um processo demorado de calibração e ajuste para o correto posicionamento dos alto-falantes. Outra técnica, mais utilizada, foi desenvolvida no NASA AMES Auditory Display Project. O objetivo era a geração de sons tridimensionais a partir de fones de ouvido comuns (PINHO, 2002).

As sensações ditas *haptic* são um assunto bastante complexo que inclui mecanismos diferentes como percepção de tato, percepção de calor ou de frio, percepção de força, percepção de movimento e percepção de vibrações. A percepção tátil depende da sensibilidade cutânea. Esta sensibilidade inclui a capacidade do ser humano de detectar estímulos mecânicos e térmicos na superfície da pele. Esta detecção é feita através de receptores existentes na pele.

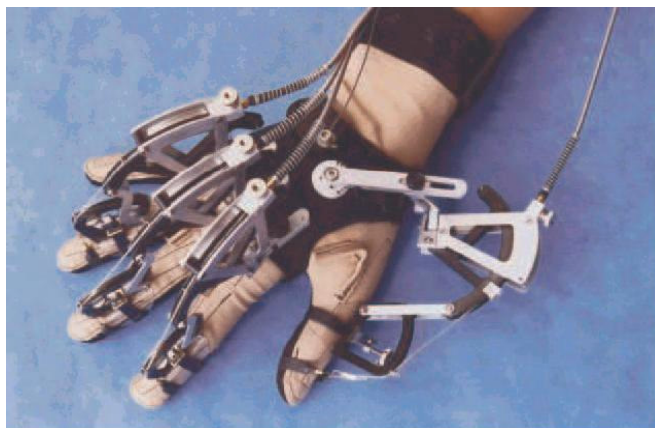
As percepções de força e de movimento estão intimamente relacionadas. Sua detecção é produzida por receptores nos músculos, juntas e tendões. Também chamada de percepção cinestésica, esta percepção

descreve a sensação que uma pessoa tem sobre sua mão quando a faz viajar através do ar ou quando sente um sopro sobre ela. A percepção de força depende da sensibilidade à oposição muscular que é capaz de resistir a forças mecânicas aplicadas sobre o corpo. Esta percepção, por sua vez, descreve a sensação que se tem quando se pressiona uma porta para tentar abri-la, ou quando se aperta uma fruta para avaliar se está madura ou não (PINHO, 2002).

Há várias tecnologias envolvidas na geração de tato ou na criação dos chamados monitores táteis, como o gerador de textura e geometria (Figura 10) e dispositivos de geração de força (Figura 11).



**Figura 10 – Dispositivo gerador de textura e geometria (PINHO, 2002)**



**Figura 11 – Dispositivo de geração de força (VIRTUAL TECHNOLOGY, 2006)**



## 2.2.2. ALGUNS TIPOS DE SISTEMAS DE REALIDADE AUMENTADA

A maneira como é visualizada a Realidade Aumentada indica alguns de seus tipos, conforme apresentado a seguir (AZUMA, 2001).

### 2.2.2.1. REALIDADE AUMENTADA BASEADA EM CAPACETE COM VISÃO ÓPTICA

O capacete com visão óptica recebe a imagem real diretamente do ambiente e projeta a imagem virtual nos olhos do usuário (KIRNER, 2005b). Um dos problemas no uso desse tipo de capacete é o fato da quantidade de luz recebida ser diferente daquela que existe no mundo real, pois o capacete reflete parte dessa luminosidade. O capacete recebe imagens do mundo real através de um sistema óptico, verifica a posição da cabeça do usuário, gera a cena virtual, posiciona-a adequadamente e a apresenta ao usuário, como mostrado no esquema da Figura 12 (AZUMA, 1997).

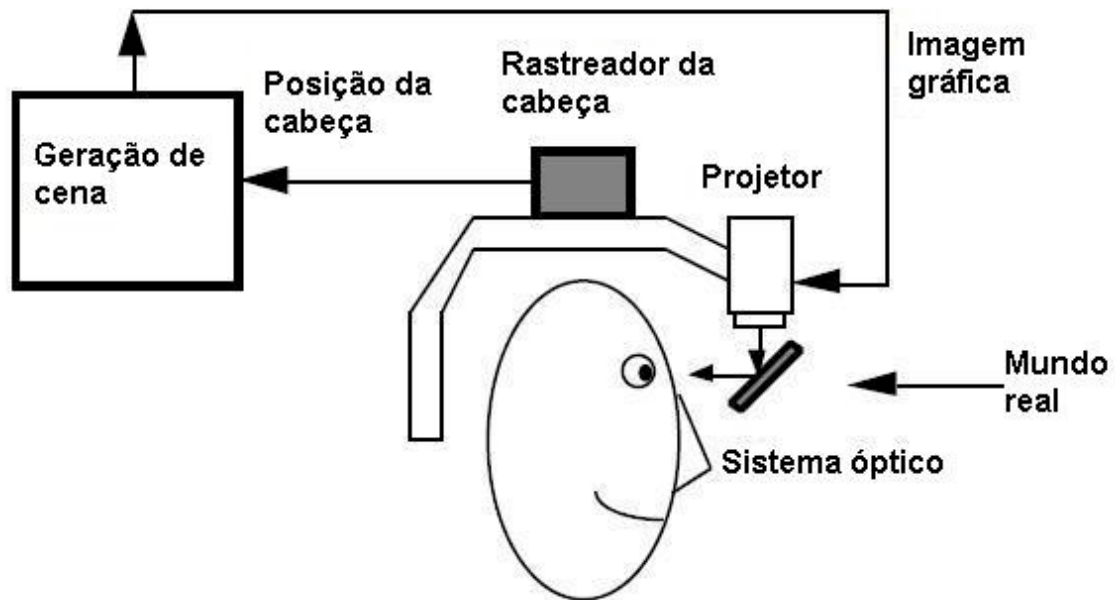


Figura 12 - Esquema de visualização em Realidade Aumentada através do capacete com visão óptica (AZUMA, 1997; KIRNER, 2005b)

### 2.2.2.2. REALIDADE AUMENTADA BASEADA EM CAPACETE COM VISÃO POR VÍDEO

Para visualizar a imagem por meio de vídeo, são utilizadas câmeras de vídeo que capturam as imagens reais, misturam essa imagem com elementos virtuais, utilizando processamento computacional, e as enviam para o sistema de visualização do usuário, conforme Figura 13 (KIRNER, 2005b).

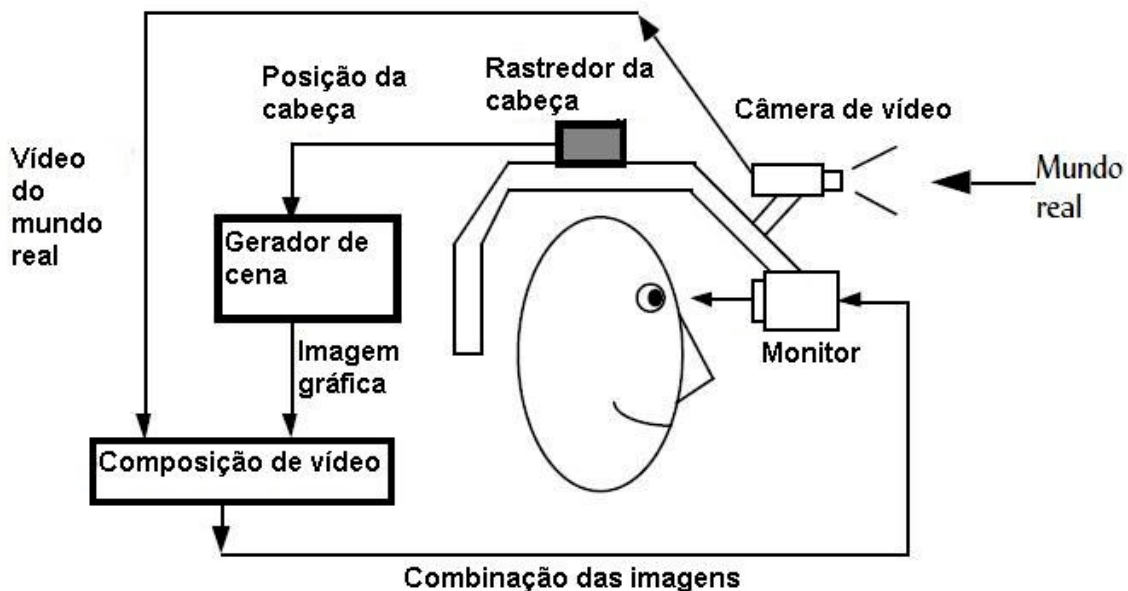


Figura 13 - Esquema de visualização em Realidade Aumentada através de capacete com visão por vídeo (AZUMA, 1997; KIRNER, 2005b)

### 2.2.2.3. REALIDADE AUMENTADA BASEADA EM MONITOR

No sistema de visão baseado em monitor, a imagem é gerada na estação de trabalho e apresentada em um monitor de alta resolução. Um conversor transforma a imagem gráfica recebida da câmera em um formato padrão de vídeo, fazendo também a mistura entre o sinal de vídeo e o sinal da câmera, gerando a fusão da imagem do vídeo e do gráfico em tempo real. O resultado é apresentado em um monitor de vídeo padrão (Figura 14). O mouse e o teclado são dispositivos para interface do usuário (KIRNER, 2005b).

Existe também um sistema de Realidade Aumentada utilizando projeção para a visualização do ambiente, no qual são utilizadas superfícies do ambiente real para apresentar os objetos virtuais sem equipamento auxiliar.

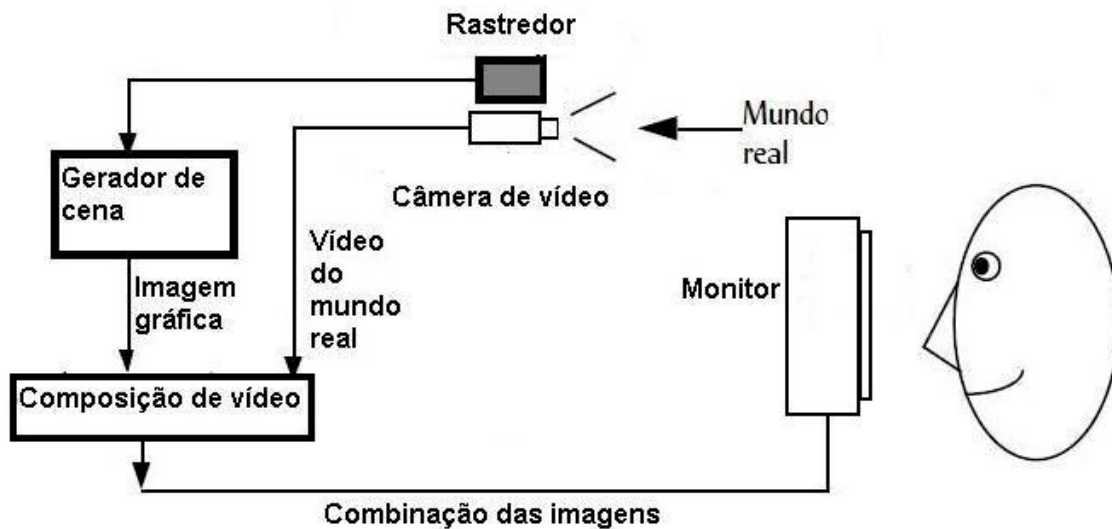


Figura 14 - Esquema de visualização em Realidade Aumentada através do monitor (AZUMA, 1997; KIRNER, 2005b)

### 2.2.3. EXEMPLOS DE SISTEMAS DE REALIDADE AUMENTADA

#### 2.2.3.1. STUDIERSTUBE

O sistema Studierstube, criado por Schmalstieg, é utilizado para estudar modelos tridimensionais em Realidade Aumentada, proporcionando uma interface de maior abrangência de visualização (BILLINGHURST, 2002); (SCHMALSTIEG, 2002). O sistema possui cinco características: usa objetos virtuais, utiliza Realidade Aumentada, cooperação, independência (cada usuário pode controlar seu próprio ponto-de-visão) e individualidade (os dados são apresentados separadamente para cada usuário). A Realidade Aumentada é utilizada para colocar as imagens virtuais no mundo real, através desse sistema.

A arquitetura desse sistema inclui a Realidade Aumentada colaborativa, permitindo que vários usuários possam usá-lo ao mesmo tempo. O Studierstube também pode ser utilizado através de redes, onde cada usuário trabalha na mesma visualização separadamente (SCHMALSTIEG, 2000).

Na Figura 15, são apresentados dois usuários examinando uma imagem tridimensional.



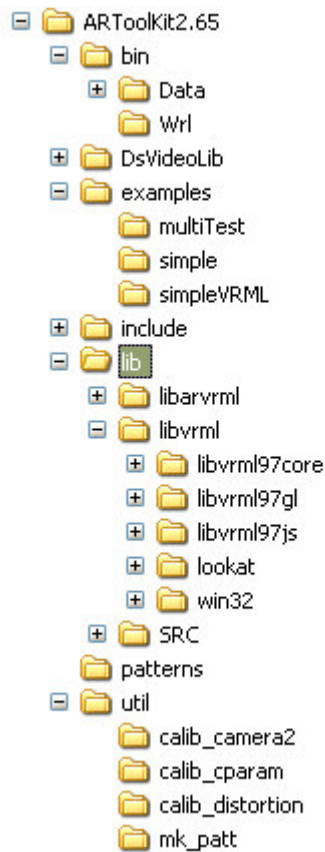
**Figura 15 - Exemplo do Studierstube (SCHMALSTIEG, 2002)**

#### **2.2.3.2. ARTOOLKIT**

O ARToolkit é uma biblioteca para desenvolver ambientes em Realidade Aumentada, onde o mundo virtual é projetado no mundo real. A imagem do usuário e as imagens virtuais devem coincidir para que o ambiente virtual e o real possam estar alinhados. Essa ferramenta possui algoritmos que calculam a posição da câmera, de acordo com a posição do ponto-de-vista do usuário (ARTOOLKIT, 2004); (KATO, 2000). Será dada maior ênfase a essa ferramenta, já que foi escolhida para o estudo.

A ferramenta ARToolkit é livre, de uso não comercial, sendo distribuída “open-source” sob licença GPL, ou seja, licença pública geral a qual garante que a ferramenta seja gratuita e livre, podendo inclusive ter seu código-fonte alterado. A versão utilizada neste trabalho é a 2.65 com VRML, e está disponível no endereço:

[http://www.hitl.washington.edu/research/shared\\_space/download/](http://www.hitl.washington.edu/research/shared_space/download/). A Figura 16 mostra a estrutura dos arquivos dessa ferramenta.

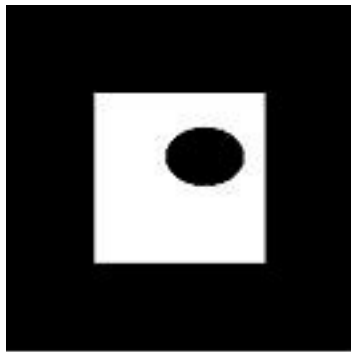


**Figura 16 - Estrutura de arquivos do ARToolkit (ARTOOLKIT, 2004)**

Para funcionar corretamente, são necessários uma câmera de vídeo e um dispositivo que mostre as imagens misturadas, podendo ser um monitor ou capacete de visualização (HMD). No caso do capacete de visualização, uma microcâmera de vídeo pode ser ajustada no capacete, apontando para o cenário na mesma direção em que os olhos apontariam, propiciando visão em primeira pessoa. No caso do monitor, uma *webcam* pode ser usada e colocada sobre o monitor, apontando para o usuário, propiciando

imagem de espelho, ou atrás do usuário, propiciando imagem em terceira pessoa.

Para capturar as imagens, o software ARToolkit utiliza marcadores de referência, em formato de quadrado, de forma que, para cada marcador possa ser associado um objeto virtual, conforme pode-se ver nas Figuras 17 e 18. É possível utilizar vários marcadores simultaneamente, pois cada marcador é associado a uma imagem separadamente.



**Figura 17 – Exemplo de marcador**



**Figura 18 – Marcador com o objeto virtual associado (ARTOOLKIT, 2004)**

Primeiramente, a câmera identifica os marcadores, convertendo a imagem em binário e calculando em seguida sua posição e orientação em relação à câmera. Com a orientação e a posição calculadas, os marcadores são comparados às imagens cadastradas previamente, associando os objetos virtuais aos respectivos marcadores. A renderização gera o vídeo de saída posicionando os objetos virtuais na cena real antes de serem apresentados aos usuários, através de seus dispositivos de visualização. A Figura 19 mostra o processo de reconhecimento do marcador e do cálculo de sua posição em relação à câmera. Em seguida, a imagem do cubo colorido (objeto virtual associado ao marcador) é alinhada ao marcador, sendo renderizada e apresentada no dispositivo de visualização (monitor de vídeo, óculos especial, por exemplo).

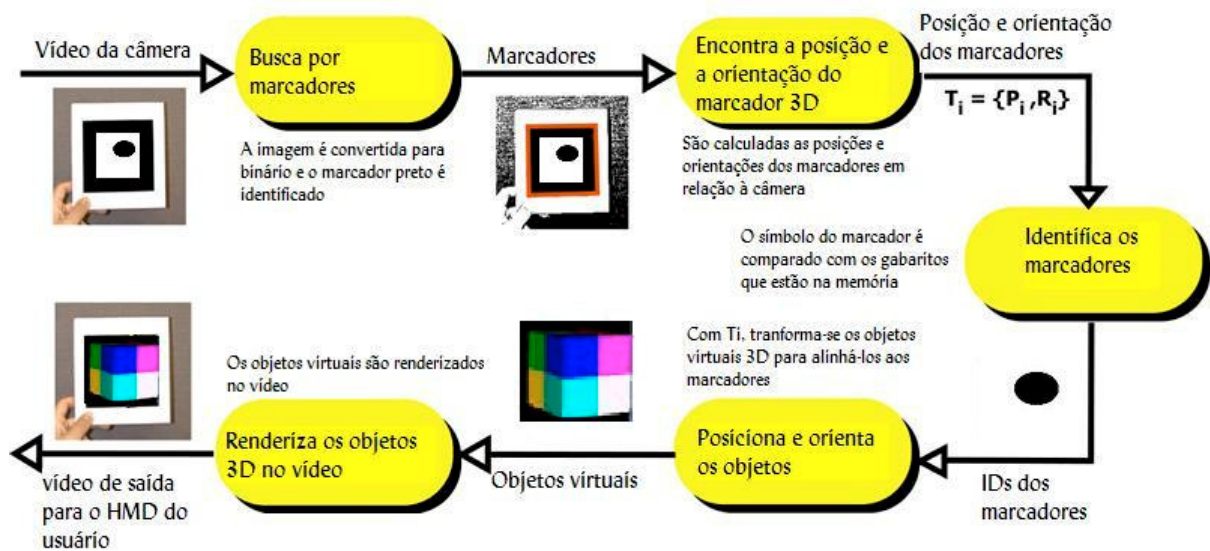


Figura 19 - Diagrama de processamento das imagens no ARToolkit (KATO, 2000)

### 2.2.3.2.1. APLICAÇÕES COM ARTOOLKIT

As possibilidades de aplicações com ARToolkit são diversas. Santin (2004), abordou em seu trabalho Ações Interativas em Ambientes de Realidade Aumentada com ARToolkit, novas formas de interação em ambientes virtuais, baseados no ARToolkit, para aplicações de montagem de layouts, por exemplo.

Sendo o ARToolKit uma ferramenta de código aberto, foi possível desenvolver modificações no código, referentes à interação da aplicação, como por exemplo: a alteração do objeto virtual, a partir da introdução de uma placa adicional de controle (Figura 20); a aplicação de uma placa que possibilita o transporte dos objetos virtuais entre as placas da cena, como se fosse uma pá, carregando os objetos virtuais de uma placa à outra (Figura 21), e o acionamento de som, quando um objeto virtual é colocado sobre a placa, podendo descrever o objeto, contar uma história ou dar instruções ao usuário.



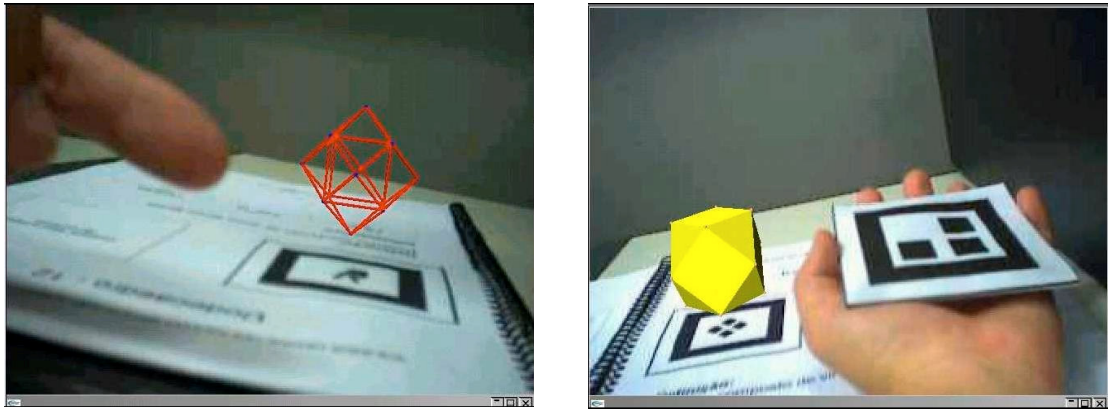


Figura 20 – Placa de controle alterando o objeto virtual (SANTIN, 2004)

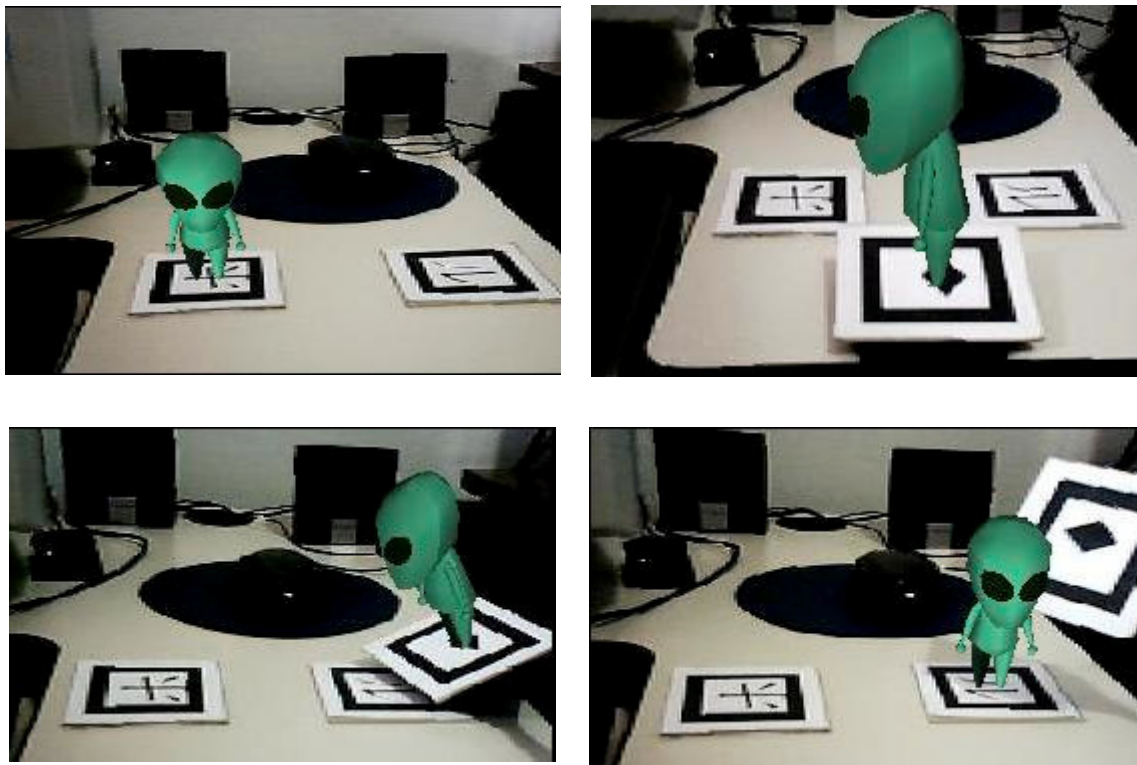


Figura 21 – Seqüência de ações para o transporte do objeto virtual com auxílio da pá (SANTIN, 2004)

O Quadro 2, mostra a alteração no ARToolkit para a inserção do módulo de som.



**Quadro 2 – Módulo de som**

```
void som(void )
{
    PlaySound(sound, NULL, SND_SYNC);
    ret=0;
    pthread_cancel (hThread);
    pthread_exit(NULL);
}

// Criação da Thread
if (ret==0)
{
    pthread_create(&hThread, NULL, (void (*)(void *))som, NULL);// Criação da Thread
    ret=1;
}
```

A combinação dessas técnicas de interação propicia ao usuário elementos interessantes para o exercício da criatividade na elaboração de aplicações.

Um outro trabalho, que aborda uma área de muita pesquisa, é o artigo Ambientes Colaborativos com Realidade Aumentada. O trabalho consiste no desenvolvimento de um sistema bastante simples, baseado no uso do ARToolkit, configurado para funcionar em rede, através do uso de sockets (GERALDI, 2005). Para isto, o ARToolkit foi modificado para importar e exportar posições, permitindo a colocação de objetos virtuais em posições recebidas e o envio das posições das placas marcadoras existentes no ambiente real.

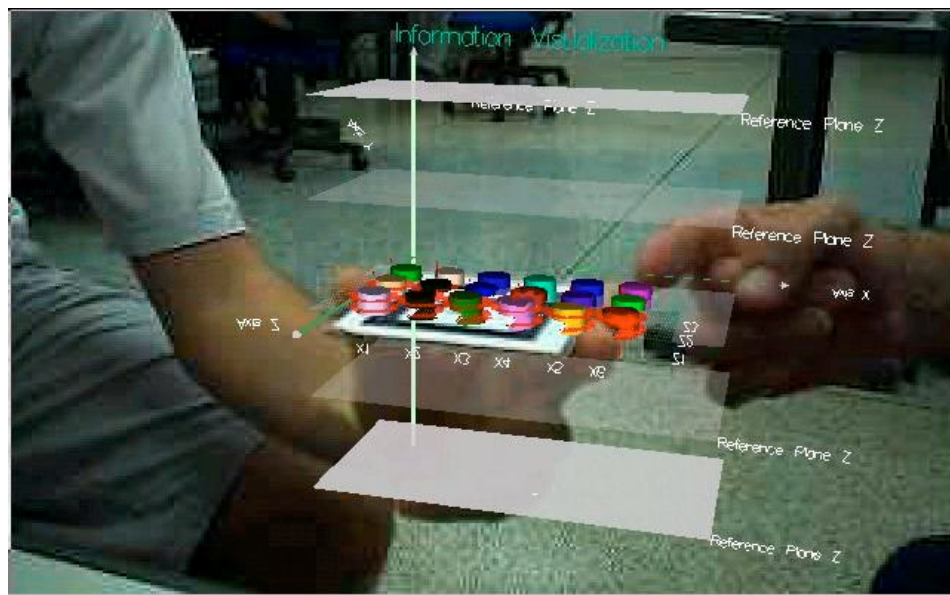
Por fim, um terceiro trabalho, com aplicações em tomadas de decisão e visualização de informações, é o artigo Visualização de Dados em Ambientes com Realidade Aumentada.

A representação de dados visa apresentar a informação ao usuários, de maneira simples e sintética. Uma das formas mais comuns de se fazer isto é através de gráficos discretos tridimensionais com múltiplos

atributos, como diagramas de barras dispostos no espaço e incrementados com elementos de animação, visuais e sonoros (BUK, 2004).

Embora a maior parte dos sistemas de visualização da informação se preocupem exclusivamente com a visualização em si, única, em janelas ou através de navegação tridimensional, a interação com a representação dos dados vem se tornando foco das atenções, uma vez que pode potencializar as aplicações, incrementando a análise e tomada de decisão por parte do usuário. Assim, gráficos podem ser mapeados em regiões clicáveis, sujeitos à interação com o usuário, fornecendo informações adicionais ou executando ações específicas. Limites visuais com mínimos, máximos, faixas de erro, etc. podem ser inseridos ou alterados graficamente pelo usuário.

As Figuras 22 e 23 mostram um visualizador de dados com Realidade Aumentada.



**Figura 22 – Exemplos de uso do Visualizador de Dados com Realidade Aumentada (BUK, 2004)**

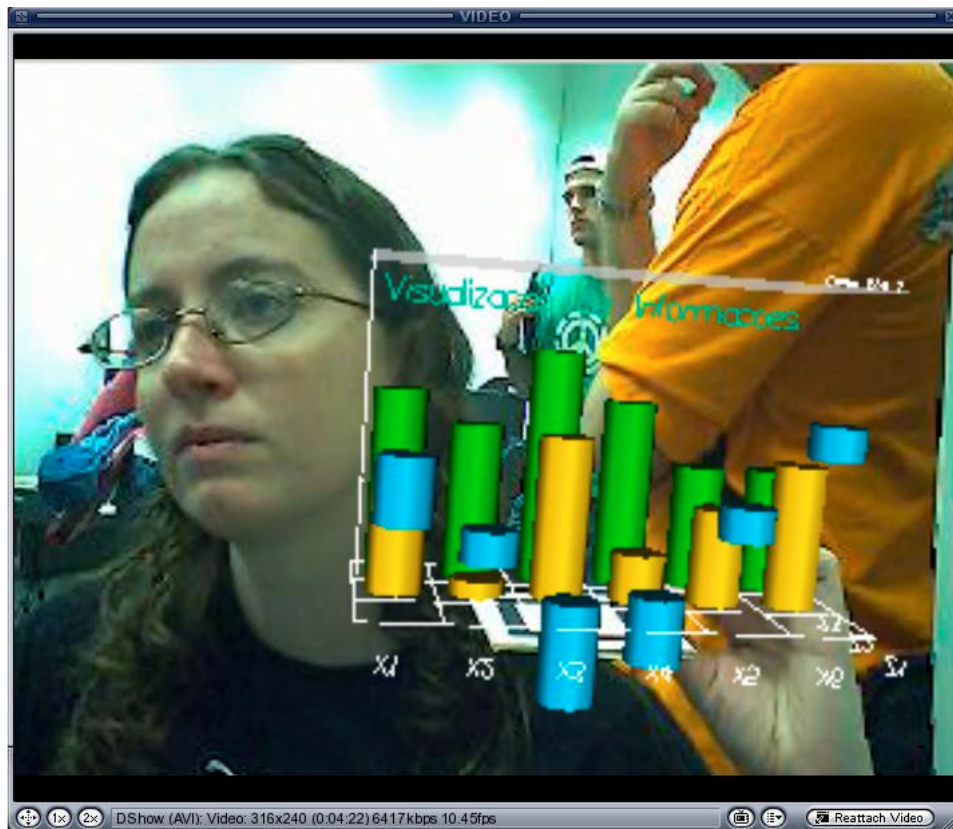


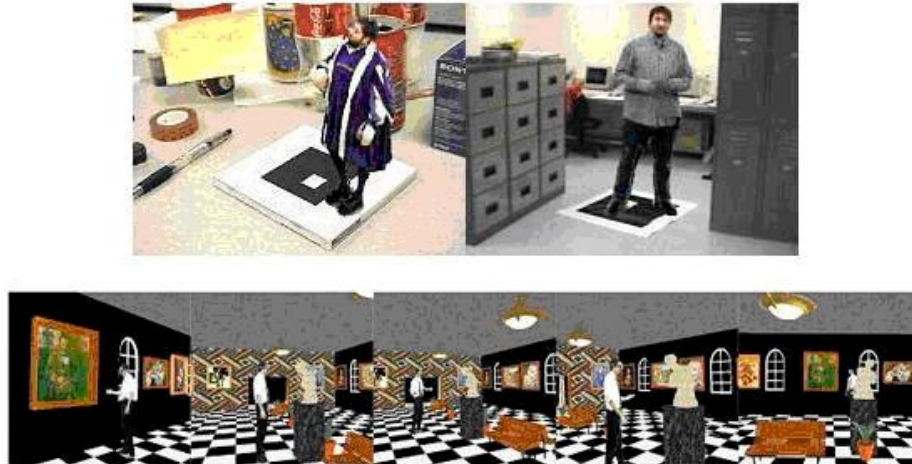
Figura 23 - Visualização Dinâmica (BUK, 2004)

### 2.2.3.3. MXR TOOLKIT

O sistema MXR Toolkit é um sistema aberto para fins não-comerciais, que consiste em uma biblioteca para ajudar a construir ambientes de Realidade Aumentada, possuindo várias rotinas para trabalhar imagens recebidas através de câmeras. Existem também soluções para técnicas de visualização com rotinas matemáticas e geométricas, as quais o desenvolvedor do ambiente pode acrescentar à biblioteca (MXR Toolkit, 2005).

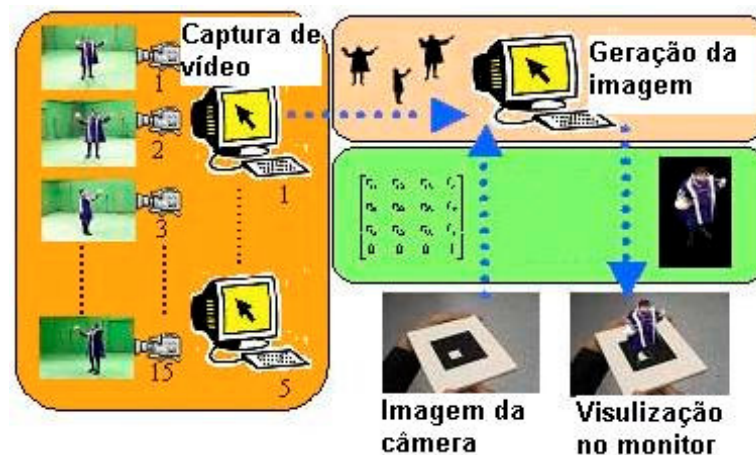
Nesse sistema, é possível inserir pessoas no ambiente virtual, conforme mostrado no exemplo da Figura 24. Para a manipulação das pessoas, primeiramente elas devem ser capturadas num estúdio, contendo várias câmeras de vídeo, fazendo tomadas do usuário em tempo real. Essas tomadas permitem a construção de um boneco virtual igual à pessoa, usando técnicas de reconstrução de objetos. O fato desse procedimento ter de ocorrer em tempo real exige um sistema computacional bastante potente. Esse

boneco, representando a pessoa, pode então ser levado a um ambiente de Realidade Aumentada e manipulado por outros usuários. Se a pessoa estiver usando um capacete de visualização, ela poderá ser colocada em um ambiente virtual, sendo vista por outros usuários e verá o cenário virtual como retorno, podendo com ele interagir.



**Figura 24 - Exemplos de inserção de pessoas no ambiente virtual (MXR Toolkit, 2006)**

Na Figura 25, pode-se ver como é feita a captura das imagens, onde inicialmente existem algumas câmeras que capturam a imagem da pessoa. Essas imagens são salvas e uma outra câmera captura a imagem do marcador. Então essas imagens são processadas no computador e finalmente misturadas, realizando então a inserção da pessoa no novo ambiente.



**Figura 25 - Funcionamento do sistema MXR Toolkit (MXR Toolkit, 2006)**

Também podem ser utilizados alguns artifícios para transportar a pessoa para outro ambiente virtual através do uso de uma pá, onde o usuário coloca a pá no ambiente atual, onde está a pessoa e a retira, inserindo-a então em outro ambiente totalmente diferente.

O próximo capítulo abordará conceitos e aplicações sobre treinamento apoiado por Realidade Virtual e Aumentada.

## CAPÍTULO 3

### 3. TREINAMENTO COM REALIDADE VIRTUAL E AUMENTADA

A Realidade Virtual vem ganhando espaço na área de treinamento, em função de sua versatilidade na visualização de modelos realistas de objetos e da possibilidade de exercitar um certo tipo de habilidade, sem correr os riscos inerentes do mundo real.

No entanto, há dois problemas, que vêm dificultando sua larga utilização: o primeiro é a necessidade de equipamentos especiais para a manipulação de objetos virtuais, como luvas, capacetes, rastreadores e projetores especiais, dentre outros; o segundo é a necessidade de acomodação do usuário no ambiente virtual, pois ele tem que ser transportado ao domínio da aplicação.

No sentido de minimizar esse problema, a área de treinamento vem recentemente utilizando Realidade Aumentada, que traz os objetos virtuais para o ambiente do usuário, deixando-o manipular esses objetos com as mãos ou dispositivos simples e não exigindo acomodação, pois ele usa o seu espaço já conhecido, tendo pleno domínio e segurança. Ainda aqui, o problema dos equipamentos persiste, mas o uso de rastreamento óptico, visão computacional e a manipulação direta dos objetos virtuais com as mãos, ou dispositivos mais simples como uma vareta com bolinhas (KIRNER, 2004a), vêm facilitando o trabalho e tornando-o mais acessível e de fácil utilização.

A indústria aeronáutica e automobilística utiliza Realidade Aumentada para resolver problemas de montagem e treinamento há alguns anos e muitos outros segmentos estão descobrindo essa aplicação. A área educacional é uma das que tem grande potencial de utilização da Realidade Aumentada, em função do elevado grau de informação visual animada que é possível de ser obtida.

### **3.1. APLICAÇÕES PARA TREINAMENTO COM REALIDADE VIRTUAL E AUMENTADA**

#### **3.1.1. PLANEJAMENTO DE OPERAÇÕES DE EXPLORAÇÃO**

Uma operação militar que envolva o envio de tropas a um local desconhecido dos soldados, como uma embaixada, por exemplo, pode ter uma chance muito maior de sucesso, se os soldados, antes, puderem caminhar pelo prédio e analisar os caminhos e obstáculos que terão de enfrentar. Uma análise deste tipo possibilita aos soldados um número muito maior de informações do que a análise de uma planta ou de fotos do local (PINHO, 2002).

A Figura 26 exibe um treinamento militar, utilizando ambiente de Realidade Virtual.



**Figura 26 – Aplicação para Treinamento Militar (DONFREWARE, 2006)**

#### **3.1.2. SIMULADORES DE VÔO**

Em vários casos, fazer com que uma pessoa treine uma certa tarefa pode ser muito difícil, muito caro e muitas vezes impossível. Um caso



clássico desta situação é o treinamento de pilotos em situações de perigo, como perda súbita de combustível ou um violento deslocamento de ar (PINHO, 2002).

Pensando nisto, a indústria aeronáutica tem criado excelentes simuladores de vôo, como o da Embraer (Figura 27). Um benefício grande que esses simuladores proporcionam é que eles permitem com que o piloto realize um número muito maior de vôos em um dia, do que conseguiria em um aparelho real.



**Figura 27 – Simulador de vôo da Embraer (EMBRAER, 2006)**

Nos últimos anos, alguns simuladores vêm sendo construídos, usando tecnologias de Realidade Virtual, tendo como principal vantagem a possibilidade de recriar a cabine, simulando um outro avião, sem precisar de uma nova estrutura física.

### **3.1.3. MANUTENÇÃO**

As pessoas que trabalham com manutenção de equipamentos sofisticados, costumam lidar com uma quantidade grande de manuais e fichas técnicas de produtos.



Com a evolução rápida, comum a este tipo de equipamento, a documentação, por sua vez, também deve ser atualizada. Buscando resolver este problema, muitos fabricantes estão distribuindo manuais e fichas em meio magnético. Este procedimento diminui, sem dúvida, a quantidade de papel utilizado. Entretanto, nem sempre é cômodo usar um computador para auxiliar a leitura da documentação, durante a manutenção.

Pensando nisto, um consórcio, envolvendo as empresas British Aerospace, Private Eye, Boeing e Virtual Vision, desenvolveu sistemas de Realidade Aumentada para auxiliar na manutenção de aviões e de turbinas (PINHO, 2002). Colocando um HMD, o técnico, que realiza a manutenção, pode deixar ao alcance de seus olhos, ao lado da imagem que ele está olhando, as páginas dos manuais que ele precisa.

A Figura 28 mostra um exemplo de aplicação para manutenção, com Realidade Aumentada.



**Figura 28 – Aplicação para Treinamento em Manutenção com Realidade Aumentada (MICROVISION, 2006)**

#### **3.1.4. MÉDICA**

A Realidade Aumentada tem sido muito utilizada na área médica, para visualização e treinamento em cirurgias, através de instruções virtuais que exibem os passos exigidos (Figura 29). Para este tipo de

treinamento, o profissional utiliza-se de sensores que indicam o local exato da cirurgia (BAJURA, 1993).



**Figura 29 – Aplicação para Treinamento na área médica com Realidade Aumentada (MICROVISION, 2006)**

## **3.2. FERRAMENTAS PARA TREINAMENTO COM REALIDADE VIRTUAL E AUMENTADA**

### **3.2.1. MONTAGEM VIRTUAL USANDO TÉCNICAS DE REALIDADE VIRTUAL**

O trabalho "Montagem Virtual usando Técnicas de Realidade Virtual" (CONNACHER, 2004), constitui-se num esforço de pesquisa para criar um ambiente de projeto com montagem virtual. Ele analisa os problemas de montagem virtual para projeto e manufatura e detalha as questões envolvidas com montagem virtual, incluindo o problema da trajetória e do armazenamento das informações em banco de dados.

### 3.2.2. SISTEMA DE TREINAMENTO PARA REFINARIA DE PETRÓLEO

Neste trabalho é apresentado um protótipo, utilizando Realidade Virtual, de um sistema para treinamento, abordando procedimentos de segurança em uma refinaria de petróleo. Detalha técnicas avançadas no treinamento de funcionários (HALLER, 1999), tendo como grandes vantagens, a possibilidade dos usuário navegarem e interagirem com as diversas partes da refinaria (Figura 30).

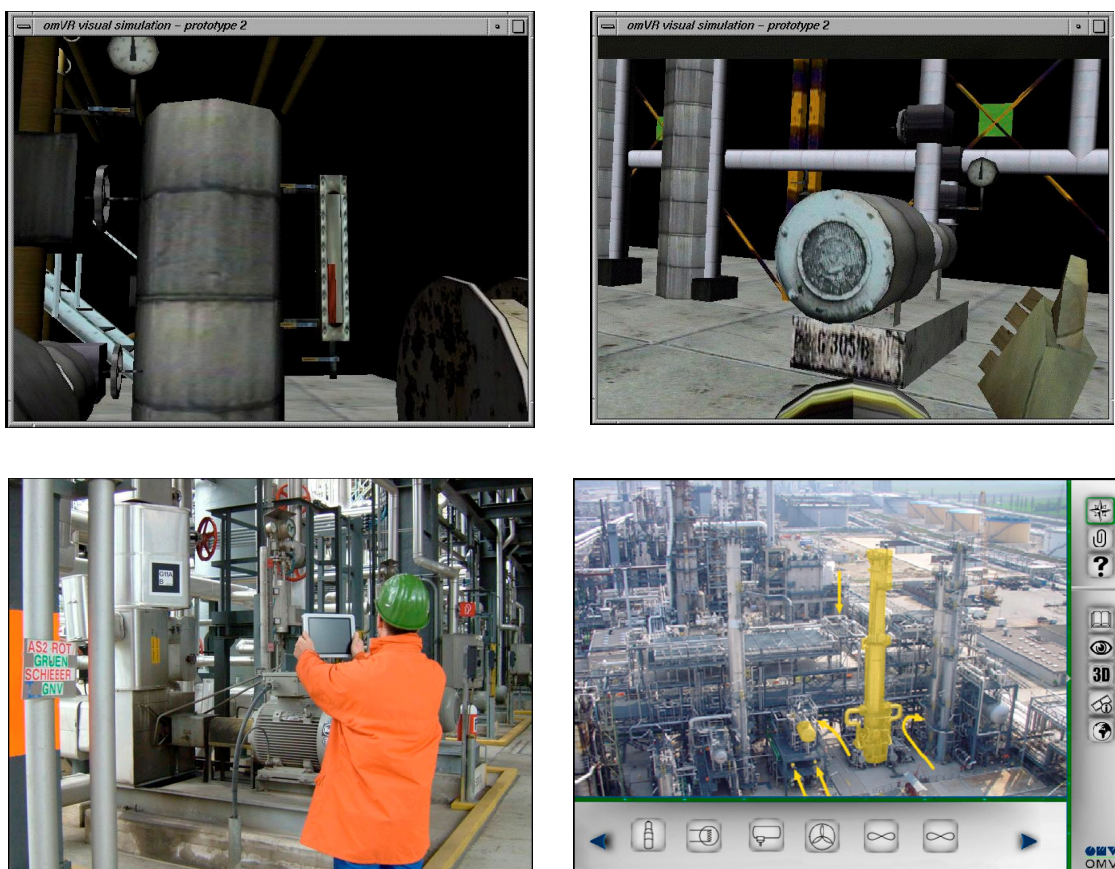


Figura 30 - Sistema de Treinamento em uma Refinaria de Petróleo (HALLER, 1999)

Este treinamento se torna bastante viável, quando analisados fatores como o grau de dificuldade e perigo de uma refinaria de petróleo, e até mesmo o custo deste treinamento no ambiente real.

### 3.2.3. VIRTUAL REALITY FOR LEGO MINDSTORMS

Um exemplo de sistema voltado para a área educacional, é o "Virtual Reality for Lego Mindstorms" (BROWER, 2004), que procura simular um kit de montagem real, permitindo a montagem de brinquedos mecânicos virtuais com o comportamento dos reais e sua utilização no mundo virtual. Com isto, espera-se que os estudantes aprendam os conceitos envolvidos com o processo de montagem e funcionamento dos brinquedos (Figura 31).

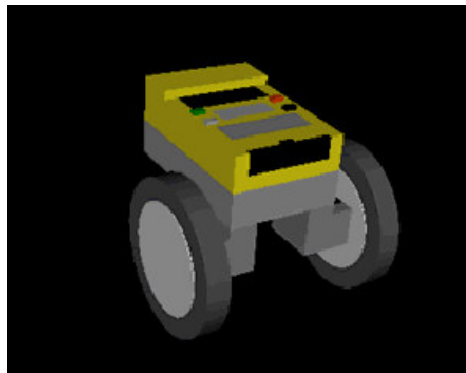


Figura 31 - Veículo montado em um ambiente virtual (BROWER, 2004)

Uma das aplicações do trabalho, foi fazer com que um veículo virtual se movesse em torno de um espiral, do centro para a extremidade, como mostra a Figura 32.

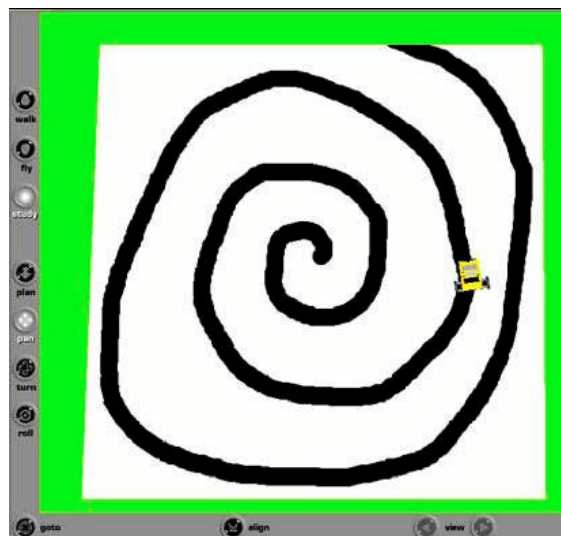


Figura 32 - Veículo virtual seguindo a trajetória em espiral (BROWER, 2004)

### **3.2.4. AMBIENTE DE TREINAMENTO AUMENTADO PARA REABILITAÇÃO DA EXTENSÃO DOS DEDOS DAS MÃOS**

Utilizando uma luva com sensor de força, o autor aborda a importância da Realidade Aumentada para reabilitação de pacientes com problemas nas extensões dos dedos das mãos. Pessoas com problemas na manipulação de objetos com as mãos, pode-se utilizar de sistemas aumentados, como o abordado neste artigo, que contribuam para a reabilitação dos movimentos dos dedos. (Figura 33).

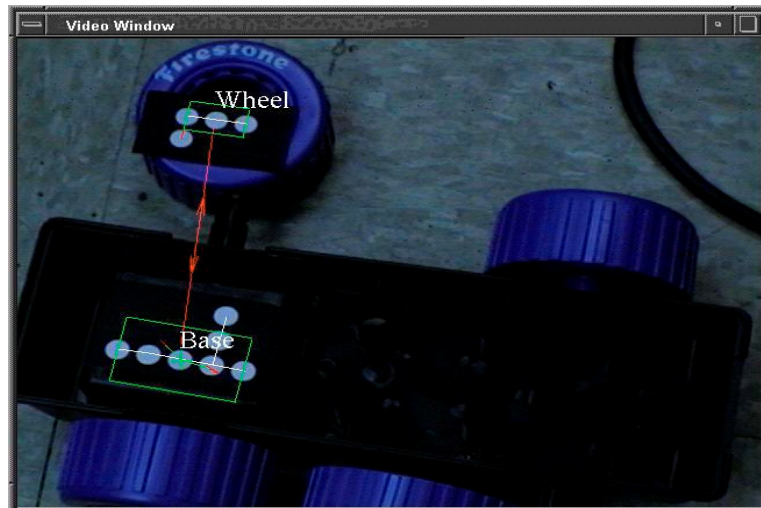


**Figura 33 - Sistema Aumentado para a Reabilitação de Pacientes (LUO, 2006)**

### **3.2.5. ART – AUGMENTED REALITY FOR TRAINING**

O projeto ART - Augmented Reality for Training (KUSTABORDER, 1999), tem por objetivo ampliar a capacidade de treinamento e usa visão computacional, animação, monitoração e avaliação das etapas do processo. O sistema usa visualização no monitor, vídeo e som, além da sobreposição de modelos gráficos 3D gerados com OpenGL (Figura 34).





**Figura 34 - Seta indicando as partes relacionadas (KUSTABORDER, 1999)**

Numa das experiências realizadas, verificou-se como o treinamento com Realidade Aumentada, para aumento de produtividade de funcionários, pode ser realizado com um tempo menor do que se fossem utilizados métodos convencionais. A Figura 35 exibe o conjunto de peças utilizadas no experimento. O caminhão de brinquedo possui 30 peças, que são montadas utilizando-se uma chave de fenda e uma chave de encaixe, seguindo instruções virtuais.



**Figura 35 – Experimento realizado com um caminhão de brinquedo (KUSTABORDER, 1999)**

### 3.2.6. PROTÓTIPO PARA TREINAMENTO INDUSTRIAL COM REALIDADE AUMENTADA COLABORATIVA

Este trabalho aborda uma ferramenta para treinamento, com Realidade Aumentada, em um ambiente colaborativo de aprendizagem. (Figura 36). A colaboração permite que vários usuários compartilhem e troquem informações, ajudando-os mutuamente nas tarefas a serem realizadas (Figura 37).



Figura 36 – Um usuário trabalhando com uma placa de computador (ZHONG, 2005)



Figura 37 – A primeira imagem mostra um usuário local, e a segunda imagem um usuário remoto, ambos visualizando um chip virtual sobre uma placa (ZHONG, 2006)

### 3.2.7. SISTEMA DE TREINAMENTO E ASSISTÊNCIA PARA MANUTENÇÃO INDUSTRIAL

Este artigo aborda o uso da Realidade Aumentada como forma de contribuir para um melhor treinamento em tarefas complexas de manutenção industrial, sem a necessidade de extensos manuais e uma

documentação em mãos do equipamento. Juntamente com o ambiente real, são geradas informações adicionais virtuais, no visor de um capacete, para que o usuário possa segui-las, realizando assim a tarefa desejada (Figura 38).



Figura 38 - Sistema de Treinamento para Manutenção Industrial (SCHWALD, 2006)

### 3.2.8. SISTEMA PARA MONTAGEM DE MOBILIÁRIO

Outra ferramenta para treinamento com Realidade Aumentada foi desenvolvida para montagem de mobiliário (ZAUNER, 2003). O sistema usa marcadores reais para o posicionamento dos objetos virtuais, tem demonstrações animadas e permite o uso de dispositivo manual de ajuste de posição dos objetos virtuais. Além disso, o sistema permite o desenvolvimento de passos de treinamento, através de uma interface de autoria com elementos próprios para a montagem de mobiliário (Figura 39).



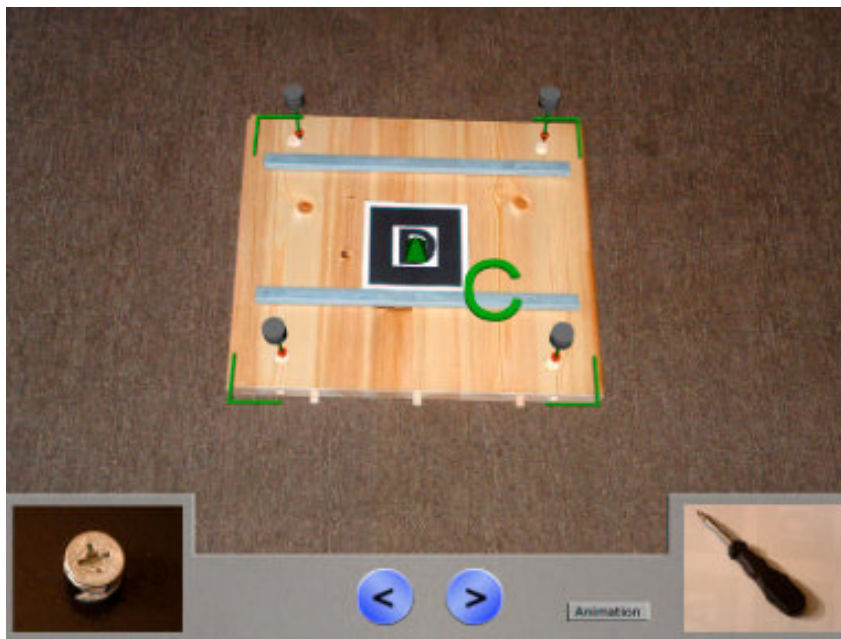


Figura 39 - Sistema para Montagem de Mobiliário (ZAUNER, 2003)

Para a montagem, pode-se utilizar a opção animação, que exibe uma interpolação e mostra a colocação correta de uma placa, de forma simples e muito intuitiva (Figura 40).



Figura 40 - Montagem de Mobiliário (ZAUNER, 2003)

Percebe-se, portanto com esta amostra de trabalhos, que tanto a Realidade Virtual, quanto a Realidade Aumentada, podem ajudar em muito as pessoas na tarefa de treinamento e ensino/aprendizagem.

O próximo capítulo apresentará um protótipo de uma ferramenta, desenvolvido para apoiar o treinamento.

# CAPÍTULO 4

## **4. RASTREAMENTO E VISUALIZAÇÃO DE TRAJETÓRIAS PARA TREINAMENTO COM REALIDADE AUMENTADA**

Este capítulo apresenta o protótipo de uma ferramenta computacional visual para treinamento. Utiliza-se de um ambiente aumentado, onde objetos tridimensionais exibirão trajetórias padrões pré-determinadas, que deverão ser seguidas pelo usuário, de acordo com as instruções presentes no monitor.

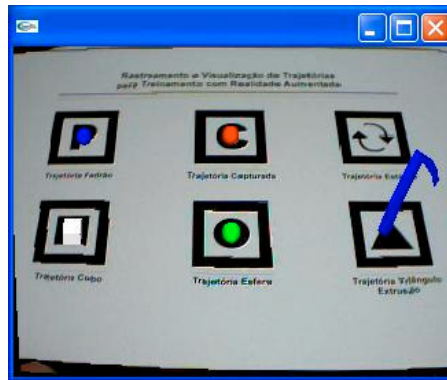
A seguir, são apresentadas as ferramentas e linguagens utilizadas para a implementação do sistema protótipo.

### **4.1. IMPLEMENTAÇÃO DO AMBIENTE DE REALIDADE AUMENTADA PARA TREINAMENTO**

#### **4.1.1. MODELAGEM DAS TRAJETÓRIAS**

Os objetos, que representam as trajetórias, foram modelados em VRML, uma linguagem para modelagem de objetos e ambientes de Realidade Virtual. Foram utilizados diversos objetos, como esferas e cubos de diversas cores, para proporcionar um fácil entendimento quanto ao treinamento e uma experiência agradável ao usuário.

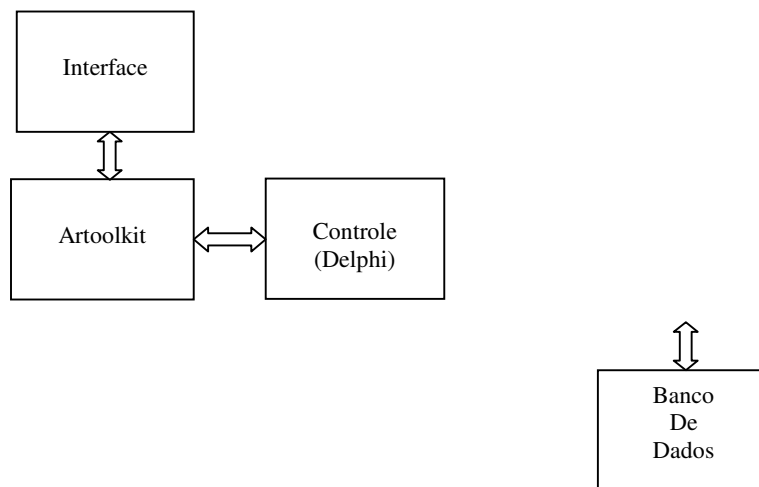
A Figura 41 ilustra os objetos virtuais utilizados para o desenvolvimento da ferramenta.



**Figura 41 - Marcadores com seus respectivos objetos virtuais utilizados no protótipo**

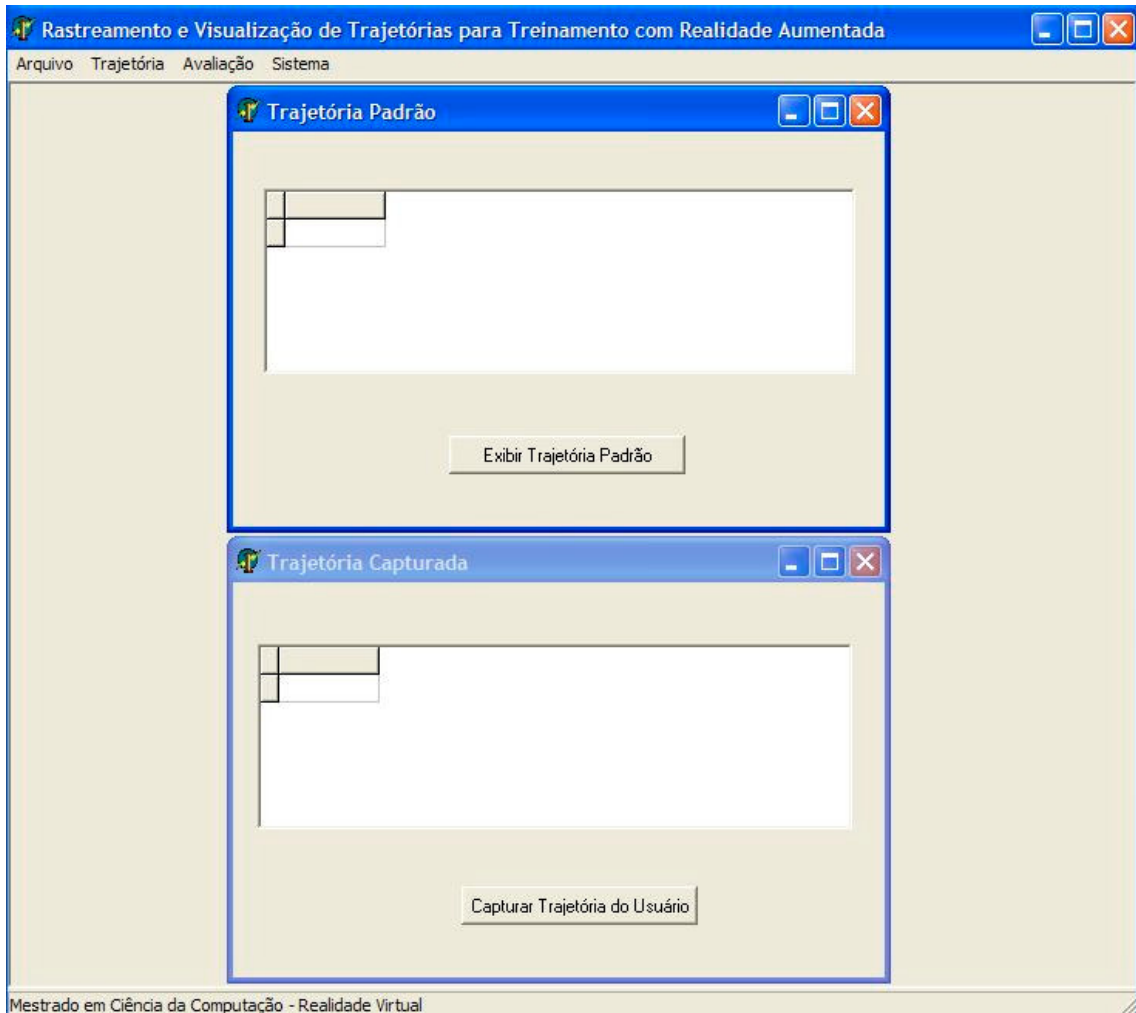
#### 4.1.2. AMBIENTE DE PROGRAMAÇÃO

A linguagem Delphi foi utilizada para a comunicação entre o banco de dados e o ARToolkit. A Figura 42 mostra um esquema do funcionamento da ferramenta. Os dados da trajetória padrão são enviados ao ARToolkit através da Tela de Controle, que também recebe os dados da trajetória capturada, realizada pelo usuário, e os envia ao banco de dados.



**Figura 42 - Esquema do funcionamento do sistema**

A Figura 43, mostra a tela de controle, implementada em Delphi.



**Figura 43 - Tela de controle**

Um sistema de gerenciamento de banco de dados (SGBD) pode ser visto como um sistema computadorizado de armazenamento de arquivos, consistindo em um repositório ou recipiente para uma coleção de dados computadorizados (DATE, 2000). Normalmente, utiliza-se da linguagem SQL (Struct Query Language) para manipular as informações do banco.

O banco de dados utilizado no desenvolvimento do protótipo foi o Firebird (FIREBIRD, 2006), por ser um sistema robusto e seguro. Nele, foram criadas duas tabelas: uma com os dados de orientação da Trajetória Padrão, que será exibida para que o usuário possa segui-la durante o treinamento, e uma outra com os dados de captura da trajetória realizada pelo usuário, durante seu treinamento (Trajetória Capturada). As duas tabelas são utilizadas,

posteriormente, para se gerar uma avaliação quanto ao desempenho do usuário. A Figura 44 apresenta as duas tabelas construídas no Firebird.

ID_TRA_PADRAO	X	Y	Z
1	0,00	0,00	0,00
2	1,00	1,00	0,00
3	2,00	2,00	0,00
4	3,00	3,00	0,00
5	4,00	4,00	0,00
6	5,00	5,00	0,00
7	6,00	6,00	0,00
8	7,00	7,00	0,00
9	8,00	8,00	0,00
10	9,00	9,00	0,00
11	10,00	10,00	0,00
12	11,00	11,00	0,00
13	12,00	12,00	0,00
14	13,00	11,00	0,00
15	14,00	10,00	0,00
16	15,00	9,00	0,00
17	16,00	8,00	0,00
18	17,00	7,00	0,00
19	18,00	6,00	0,00
20	19,00	5,00	0,00
21	20,00	4,00	0,00
22	21,00	3,00	0,00
23	22,00	2,00	0,00
24	23,00	1,00	0,00

ID_TRA_CAPTURADA	X	Y	Z
1	0,00	0,00	0,00
2	1,00	1,00	0,00
3	1,50	2,00	0,00
4	3,50	3,05	0,03
5	4,00	3,95	0,00
6	5,00	5,00	0,20
7	6,70	6,00	0,10
8	7,30	7,00	0,00
9	8,00	8,01	0,00
10	9,00	9,00	0,00
11	10,00	10,00	0,00
12	10,80	11,00	0,00
13	12,00	12,00	0,15
14	13,00	11,02	0,12
15	14,00	9,89	0,06
16	15,00	9,00	0,00
17	16,00	8,00	0,00
18	16,85	7,00	0,00
19	17,98	5,97	0,02
20	19,00	5,00	0,00
21	20,00	4,00	0,00
22	20,95	3,02	0,00
23	22,00	2,00	0,01
24	23,00	1,00	0,00

Figura 44 - Tabelas criadas no banco de dados Firebird

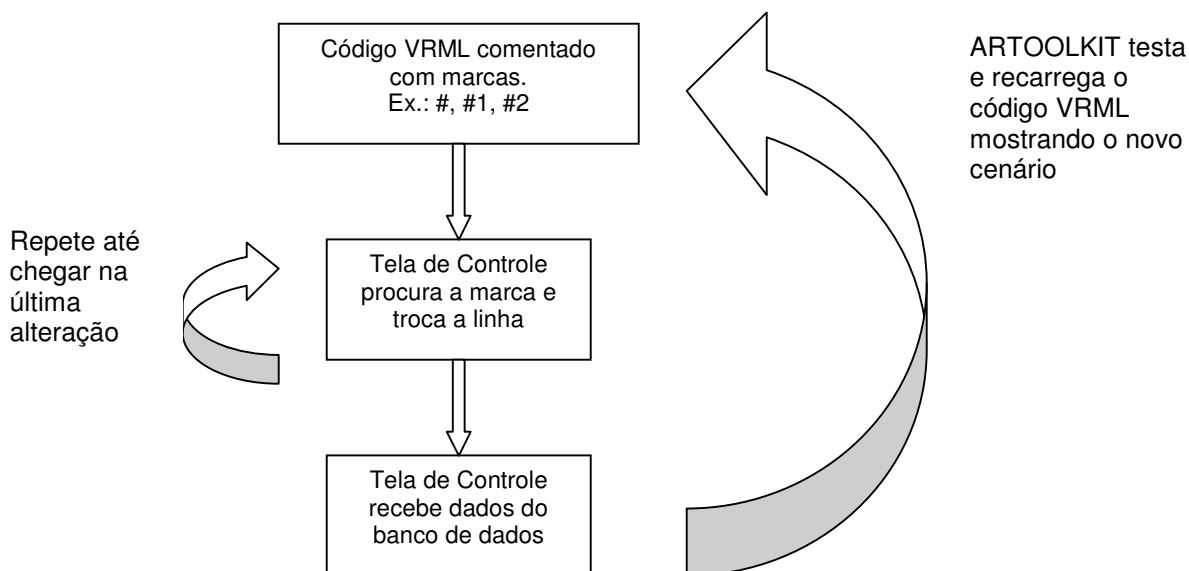
## 4.2. FERRAMENTA PARA TREINAMENTO COM REALIDADE AUMENTADA

A seguir, são apresentadas as soluções adotadas para os principais elementos da ferramenta para treinamento.

### 4.2.1. EXIBIÇÃO DA TRAJETÓRIA PADRÃO

Para exibir a trajetória padrão, armazenada em um banco de dados, foi utilizada a técnica de alteração dinâmica de objetos virtuais em ambientes de Realidade Aumentada, citada por Santin e Kirner (2005). Esta técnica consiste em modificar o objeto virtual carregado no ARToolkit, durante o tempo de execução, permitindo a alteração das propriedades do objeto

virtual, como a translação, com dados vindos do banco de dados, como mostra a Figura 45.



**Figura 45 - Funcionamento da ferramenta ARToolkit com a técnica de alteração dinâmica de objetos virtuais**

A seguir, consta no Quadro 3 um exemplo de código VRML, mostrando a linha que contém a marca para que a tela de Controle, quando acionada, realize a atualização dos dados, e no Quadro 4, o código em delphi da tela de controle, que atualiza o arquivo VRML e exibe a trajetória padrão, com esferas.

**Quadro 3 – Código VRML, com a marca #, para atualização dos dados de translação do objeto virtual**

```

#VRML V2.0 utf8
Transform{
# 1 modifica
translation 0.0 0.0 0.0
children [
Shape {
appearance Appearance {
material Material { diffuseColor 0 1 0 }}
geometry Sphere { } }}
  
```

**Quadro 4 – Código para atualização do arquivo VRML, com dados de translação vindos do banco de dados**

```

procedure TForm2.BitBtn1Click(Sender: TObject);
var
  Texto: TextFile;
begin {insere texto no arquivo}
  dm.consulta.close;
  dm.consulta.sql.clear;
  dm.consulta.sql.add('select * from trajetoria_padrao order by id_tra_padrao');
  dm.consulta.Prepare;
  dm.consulta.open;
  dm.consulta.first;
  while not dm.consulta.eof do
  begin
    If FileExists ('C:\ARToolKit2.6VRML\bin\Wrl\teste.wrl') then
      DeleteFile ('C:\ARToolKit2.6VRML\bin\Wrl\teste.wrl');
    AssignFile(Texto,'C:\ARToolKit2.6VRML\bin\Wrl\teste.wrl');
    Rewrite(texto);
    Append(Texto);
    Writeln(Texto, '#VRML V2.0 utf8');
    Writeln(Texto, 'Transform {}');
    Writeln(Texto, '# 1 modifica');
    Write(Texto, 'translation ');
    Writeln(Texto,          dm.consulta.fieldbyname('x').asString          +          '          '+
dm.consulta.fieldbyname('y').asString + ' '+dm.consulta.fieldbyname('z').asString);
    Writeln(Texto, 'children []');
    Writeln(Texto, 'DEF MAT2 Transform {}');
    Writeln(Texto, 'children []');
    Writeln(Texto, 'Shape {}');
    Writeln(Texto, 'appearance Appearance {}');
    Writeln(Texto, 'material DEF MAT1 Material {}');
    Writeln(Texto, 'diffuseColor 0 0 1');
    Writeln(Texto, '}'); Writeln(Texto, '}');
    Writeln(Texto, 'geometry Sphere { }');
    Writeln(Texto, '}'); Writeln(Texto, ']'); Writeln(Texto, '}'); Writeln(Texto, ']'); Writeln(Texto, '}');
    CloseFile(Texto);
    sleep(1000);
    dm.consulta.next;
  end; end;

```

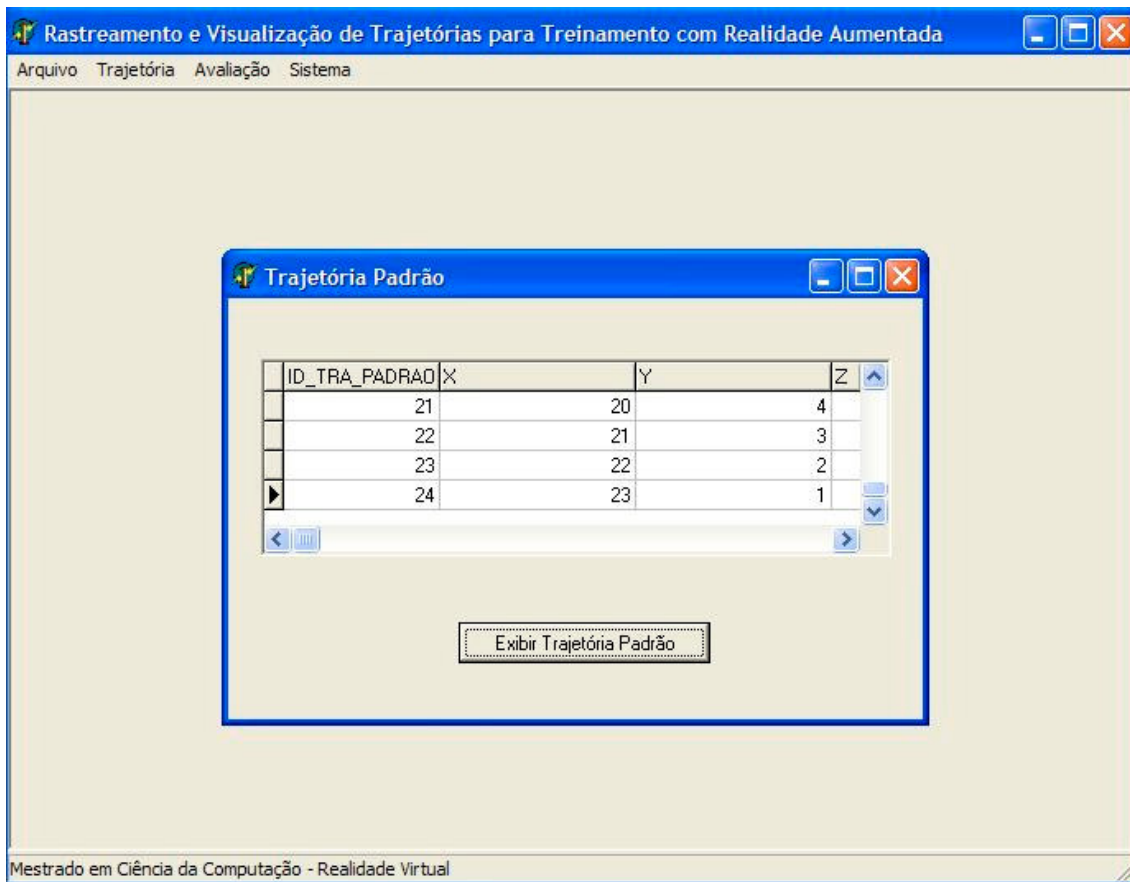


No Quadro 5, consta um exemplo de código VRML, utilizando extrusão, mostrando a linha que contém a marca para que a tela de Controle, quando acionada, realize a atualização dos dados. O código em delphi da tela de controle, que atualiza o arquivo VRML e exibe a trajetória padrão, utilizando extrusão, pode ser visto no Anexo 1.

**Quadro 5 - Código VRML, com a marca #, para atualização dos dados de translação do objeto virtual, utilizando extrusão**

```
#VRML V2.0 utf8
Shape
{
  geometry Extrusion
  {
    ccw TRUE
    convex FALSE
    solid TRUE
    beginCap TRUE
    endCap TRUE
    crossSection [0 0, 1 1, 1 0, 0 0]
    # 1 modifica
    spine [ 10 0 0, 11 -1 3, 12 -2 5, 13 -3 7, 14 -4 9, 15 -5 7, 16 -6 5, 17 -7 3, 18 -8 0]
  }
  appearance Appearance
  {
    material Material
    {
      diffuseColor 0 0 1
    }
  }
}
```

A Figura 46 mostra a tela de controle, para exibir a trajetória padrão. O botão Exibir Trajetória Padrão faz com que os dados do banco de dados sejam visualizados na tela, e atualizem a cena aumentada.



**Figura 46 – Tela de Controle de uma Trajetória Padrão**

Para o desenvolvimento desta etapa do sistema, foram necessárias modificações em arquivos, como o arvrml e o simplevrml, ambos contidos no ARToolkit. Essas modificações consistiram na inserção de rotinas nesses códigos, a fim de fazer com que o ARToolkit fizesse a re-leitura do arquivo WRL. Feita a modificação no código WRL, como alteração da translação, por exemplo, com informações vindas do banco de dados. A re-leitura desse objeto implica na leitura desse código modificado, possibilitando, dessa maneira, a visualização da alteração no objeto virtual, durante a execução do ARToolkit.

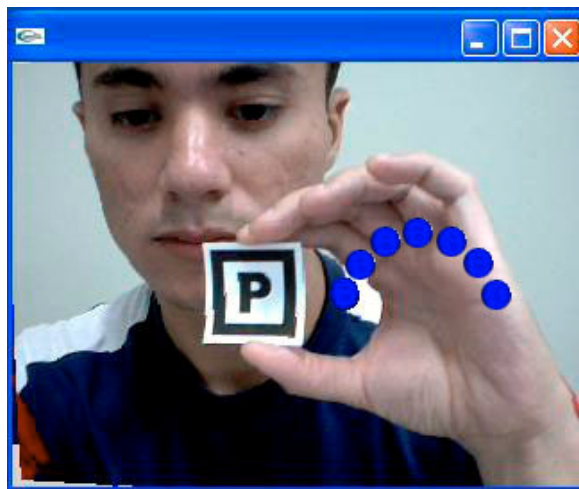
O Quadro 6 mostra o código alterado no ARToolkit, no arquivo simplevrml, para exibir a trajetória padrão.

**Quadro 6 – Código alterado no ARToolkit para exibir a trajetória padrão**

```
inter++;  
    if(inter==1)  
        {  
            modifica_Objeto(object[i].id,object[i].name);  
            inter=0;  
        }  
  
// Função modifica_Objeto  
  
void modifica_Objeto(int id,char *name){  
    FILE *fp;  
        arVrml97ReloadFile(id,name) ... }  
}
```

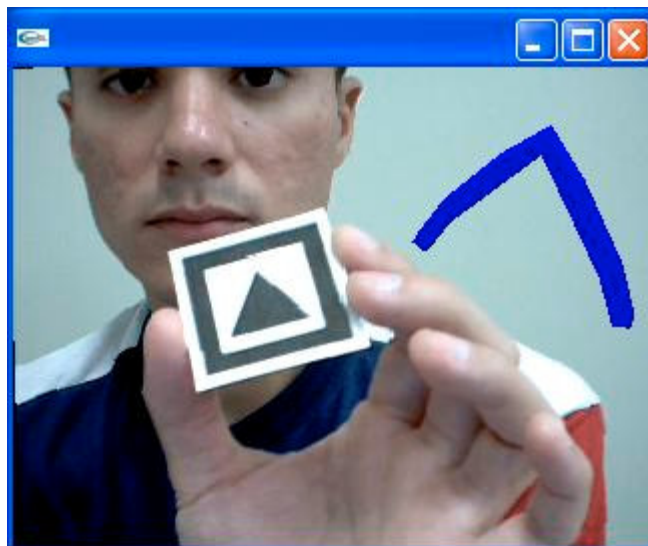
Como mostrado anteriormente, a trajetória padrão poderá ser exibida de diversas formas, enriquecendo ainda mais o ambiente computacional:

- Com cubos ou esferas, que ficarão estáticos e sempre visíveis ao usuário, como mostra a Figura 47;



**Figura 47 - Trajetória exibida com esferas**

- Através de extrusão, com animação. A trajetória é exibida aos poucos, seguindo a trajetória determinada, até chegar ao final, reiniciando o processo logo após o término, como apresenta a Figura 48;

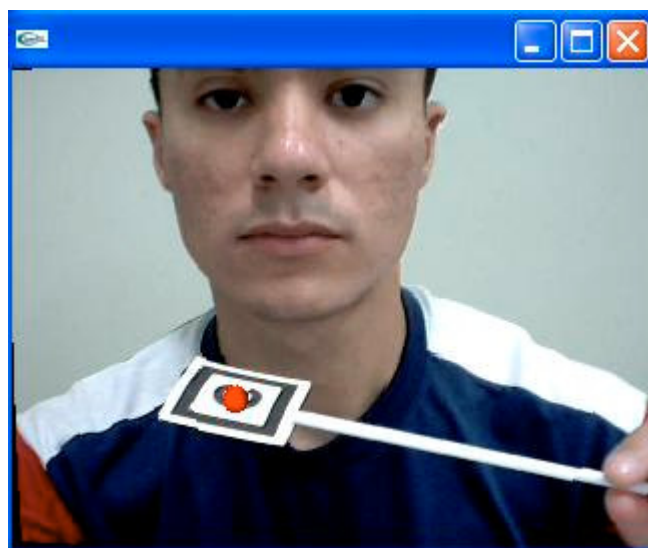


**Figura 48 - Trajetória exibida com extrusão**

- Utilizando setas, que indicarão a trajetória ao usuário. Estas setas poderão conter efeitos, para uma melhor interação.

#### **4.2.2. CAPTURA DA TRAJETÓRIA DO USUÁRIO**

Visualizando a trajetória padrão, o usuário deverá seguir as instruções visuais, para realizar o treinamento desejado. A Figura 49 mostra o marcador utilizado para realizar o treinamento.



**Figura 49 - Marcador utilizado para realizar o treinamento**

A câmera, necessária para a captura do ambiente real e da trajetória do usuário, deve estar estática, sem sofrer alterações quanto ao foco. O único movimento deve ser realizado pela placa do usuário, que fará o movimento do objeto virtual associado a ela. Os dados de translação do movimento do usuário serão então armazenados no banco de dados.

Para armazenar uma trajetória no banco de dados, está sendo utilizado rastreamento de marcadores (LAHR, 2004) e uma outra técnica, que consiste no armazenamento otimizado de pontos, inspirada no comportamento do *Dead Reckoning* (MORCLETTE, 1999).

O rastreamento de marcadores consiste numa alteração do ARToolkit, para que sejam informadas as posições X, Y e Z do marcador, de acordo com sua movimentação. Ao realizar o treinamento, as posições de translação do movimento do usuário serão capturadas, enviadas para um arquivo texto, e posteriormente armazenadas no banco de dados através da tela de controle.

O Quadro 7 apresenta o código alterado no ARToolkit, no arquivo `simplevrml`, para capturar a trajetória realizada pelo usuário.

O uso de técnicas de captura de pontos inspiradas no *Dead Reckoning* permite armazenar somente pontos discretos de uma trajetória, evitando sobrecarregar o banco de dados. Estes pontos, sendo ligados por segmentos de retas, irão compor o trajeto.

O código em Delphi, da tela de controle, para armazenar no banco de dados as posições X, Y e Z, capturadas após o usuário realizar o trajeto, ou seja, o treinamento, pode ser visto no Anexo 1.

**Quadro 7 – Código para atualização do arquivo VRML, utilizando extrusão**

```

/* draw the the AR objects */
int draw( ObjectData_T *object, int objectnum )
{
    int i;
    double gl_para[16];
    /* calculate the viewing parameters - gl_para */
    for( i = 0; i < objectnum; i++ ) {
        if( object[i].visible == 0 ) continue;
            argConvGlpara(object[i].trans, gl_para);

            // Trecho do código alterado para capturar as posições X, Y e Z
            x=gl_para[12];
            y=gl_para[13];
            z=gl_para[14];

            draw_object( &object[i], gl_para );
        }
        return(0);}

// Função modifica_Objeto

void modifica_Objeto(int id,char *name){
    FILE *fp;

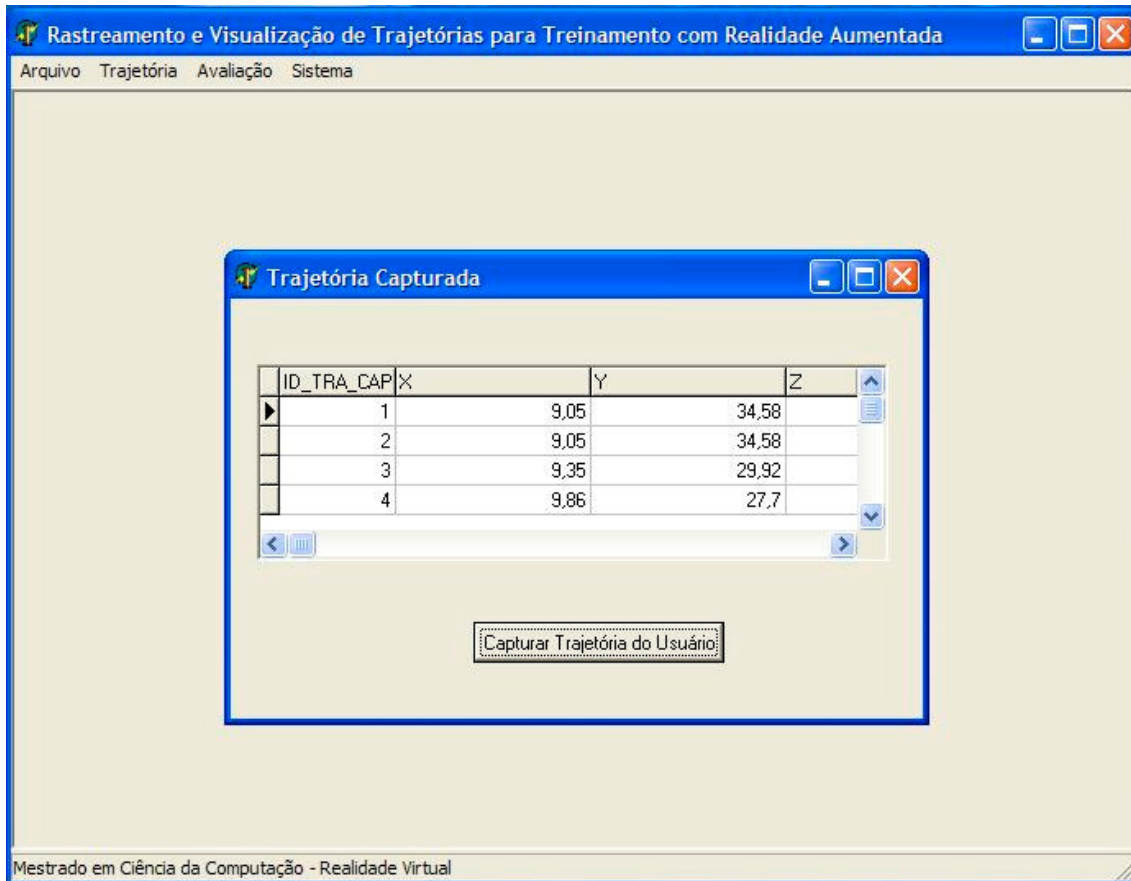
        arVrml97ReloadFile(id,name);

        // Trecho do código alterado para verificar se é a placa de captura e enviar as
        posições X, Y e Z para um arquivo texto, que posteriormente são enviadas ao banco de dados
        if( id == 1 )
            {
                fp=fopen("coordenadas.txt","a+");
                fprintf(fp,"%f\n %f\n %f\n", x, y, z);
                fclose(fp);
            }
}

```

A Figura 50 mostra a tela de controle para capturar a trajetória realizada pelo usuário. O botão Capturar Trajetória do Usuário faz com que os

dados de posição do trajeto do usuário sejam armazenados no banco de dados.



**Figura 50 – Tela de Controle de uma Trajetória Capturada**

#### **4.2.3. AVALIAÇÃO DE DESEMPENHO E APRENDIZAGEM**

Nesta última etapa, será gerada uma avaliação do usuário. A trajetória padrão é comparada com a trajetória capturada, ambas armazenadas no banco de dados, obtendo-se assim uma avaliação quanto ao treinamento.

Esta avaliação pode ser feita, utilizando diversas formas de análise:

- Verificando se o usuário alcançou a rota final proposta no treinamento;

- Comparando ambos os trajetos, e verificando as diferenças físicas das trajetórias;
- Averiguando o tempo de execução da trajetória, em relação ao previsto pelo trajeto padrão;
- Verificação espacial e temporal, em conjunto.

A forma de análise utilizada neste trabalho foi a comparação de ambos os trajetos, verificando se o usuário alcançou a rota final proposta no treinamento.

#### **4.2.3.1. COMPARAÇÃO DA ROTA FINAL**

Nesta fase do protótipo, foram verificadas a última posição da trajetória padrão com a última posição da trajetória capturada, comparando-se o último registro, na posição X, da tabela que armazena o trajeto padrão, com o último registro, também na posição X, da tabela que armazena o trajeto capturado, realizado pelo usuário.

Para isto é necessário que a câmera esteja perfeitamente ajustada, para que o ponto de partida da trajetória padrão seja o mesmo do trajeto realizado pelo usuário. A Figura 51 mostra a tela de avaliação, comparando as posições X, Y e Z.

O primeiro gráfico, trajetórias do eixo X, mostra o trajeto padrão e o trajeto realizado pelo usuário no eixo X. O segundo, exibe o trajeto padrão e o trajeto realizado pelo usuário no eixo Y. E o terceiro gráfico, o trajeto padrão e o trajeto realizado pelo usuário no eixo Z. A preocupação, nesta fase da implementação, foi fazer uma análise somente do eixo X, verificando se os pontos finais do trajeto são os mesmos. Assim é possível saber se o usuário atingiu, através do seu trajeto, a posição final desejada.



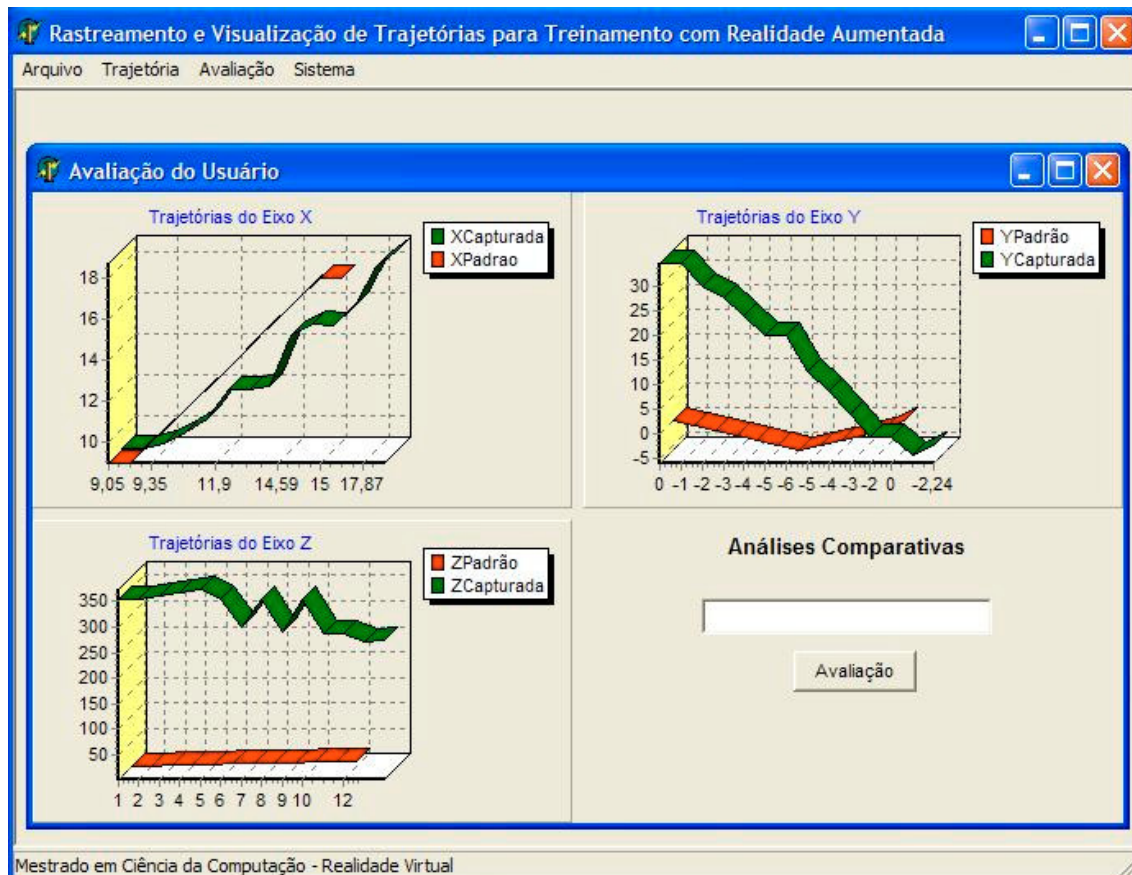


Figura 51 – Tela de Avaliação do Usuário

A seguir é mostrado o código da tela de avaliação, de acordo com o Quadro 8.

Quadro 8 – Código da tela de avaliação do usuário

```

procedure Tf1.Button1Click(Sender: TObject);
var a1, a2 : string;
begin
  dm.consulta.open; dm.consulta.last;
  dm.consulta2.open; dm.consulta2.last;
  a1 := dm.consulta.fieldbyname('x').asstring;
  a2 := dm.consulta2.fieldbyname('x').asstring;
  if a1 = a2 then e1.text := 'Trajetória Correta!'
  else
    e1.text := 'Trajetória Incorreta!'
end;

```

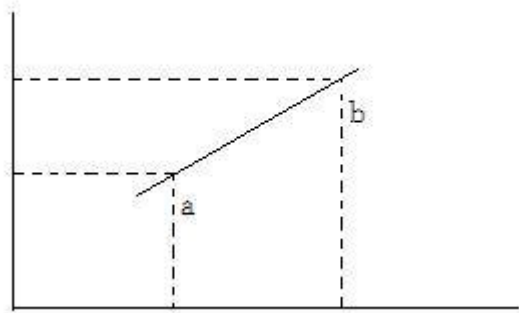
Uma outra forma de se avaliar o desempenho do usuário, é a comparação de ambos os trajetos, verificando as diferenças físicas das trajetórias, utilizando interpolação.

#### 4.2.3.2. INTERPOLAÇÃO

A interpolação de pontos de translação de um trajeto, consiste em adicionar novos pontos à ele, baseado nos pontos existentes, aumentando a quantidade total de pontos, formando assim, o trajeto desejado. Esta abordagem é adequada para se comparar duas trajetórias.

É necessário que se compare uma trajetória com a outra, ou seja, compare pontos equivalentes entre duas trajetórias. Um maneira de verificar a equivalência dos pontos, é fazer a interpolação, conseguindo assim, os dados dos dois trajetos.

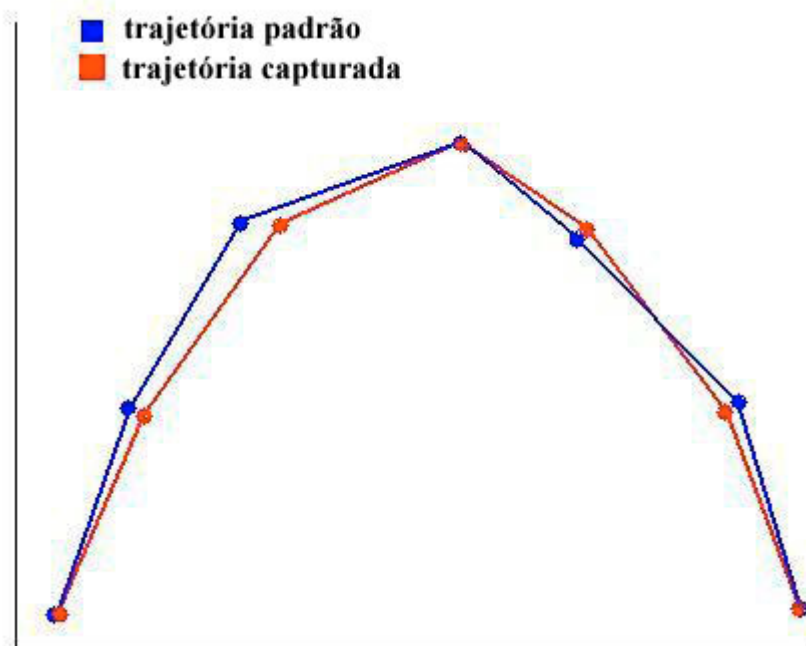
A interpolação linear é uma linha que se ajusta a dois pontos. A Figura 52 mostra a interpolação linear entre os pontos a e b.



**Figura 52 – Interpolação entre dois pontos**

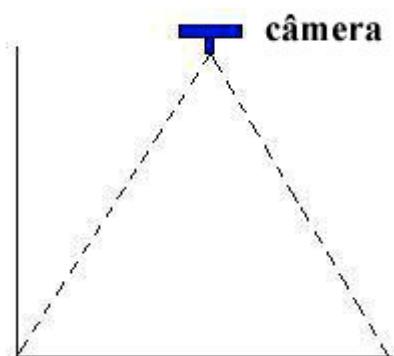
O primeiro passo é determinar pontos correspondentes entre as duas trajetórias, num plano cartesiano, através dos eixos X e Y.

A Figura 53 mostra duas trajetórias, a padrão (azul) e a capturada (vermelho), com seus pontos correspondentes.



**Figura 53 – Interpolação, identificando pontos correspondentes**

Pode-se ignorar um dos eixos, se a câmera estiver colocada exatamente na vertical, capturando as imagens que estão em dois eixos, abaixo dela, como mostra a Figura 54.



**Figura 54 – Câmera, colocada de cima para baixo**

Em seguida pode-se medir a distância entre os pontos dos dois trajetos e por fim, fazer a média de dispersão.

Para algumas aplicações que necessitam comparar duas trajetórias, esta é uma forma eficaz de avaliação.

#### **4.2.3.3. MEDIDAS DE DISPERSÃO: DESVIO PADRÃO**

Consiste na comparação da dispersão física dos dados (comparação de ambos os trajetos em relação a um ponto médio), de acordo com Barbetta (2004).

Pode ser utilizado para avaliar o usuário em aplicações que verifiquem sua deficiência motora. Se os dados da Trajetória Capturada forem muito mais dispersos do valor médio, do que os dados da Trajetória Padrão, pode-se, por exemplo, chegar à conclusão de que o usuário possui alguma deficiência.

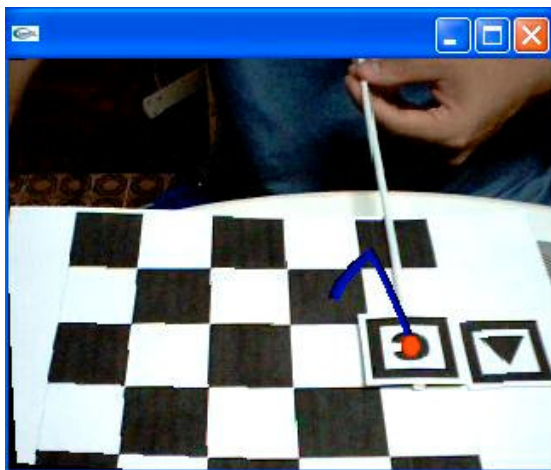
#### **4.3. APLICAÇÕES**

A aplicação desse projeto está ligada a sistemas que necessitem de algum tipo de treinamento, como jogos, manufatura, medicina, e outros.

Um exemplo de aplicação se destina a reconhecimento de assinaturas de clientes, como forma de segurança, em alguns ambientes comerciais. Como uma assinatura é uma trajetória, pode-se criar um ambiente em que o usuário possa assinar com suas próprias mãos algum tipo de documento, e o sistema faça o reconhecimento de sua assinatura, ou seja, do trajeto realizado.

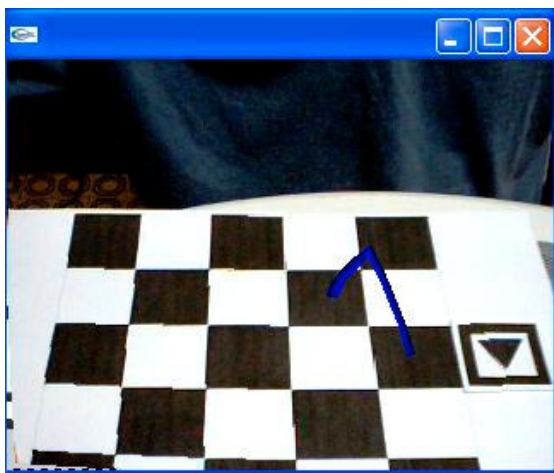
Uma outra aplicação, se destina a treinamento de alunos em auto escolas. Pode-se criar um sistema que mostre ao aluno a trajetória que deve seguir durante suas aulas ou avaliações. Ao efetuar o trajeto, o aluno, juntamente com seu instrutor, poderá obter do sistema a avaliação de seu desempenho.

A Figura 55 mostra a aplicação da ferramenta em um jogo de dama.

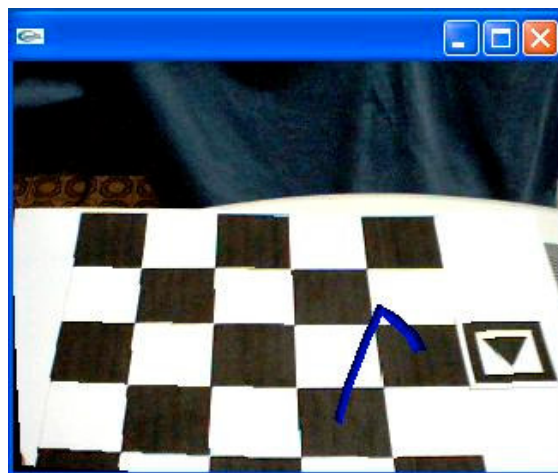


**Figura 55 - Aplicação em um jogo de dama**

O sistema exibe ao usuário as possíveis trajetórias e este, com um marcador, realiza um dos trajetos possíveis, de acordo com as Figuras 56 e 57.

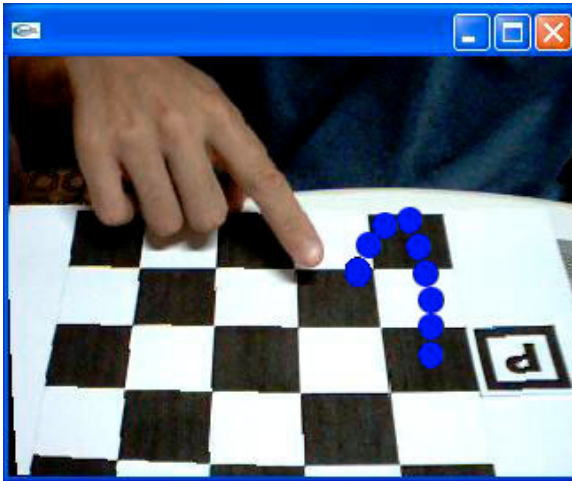


**Figura 56 – Trajeto possível, exibido com extrusão**

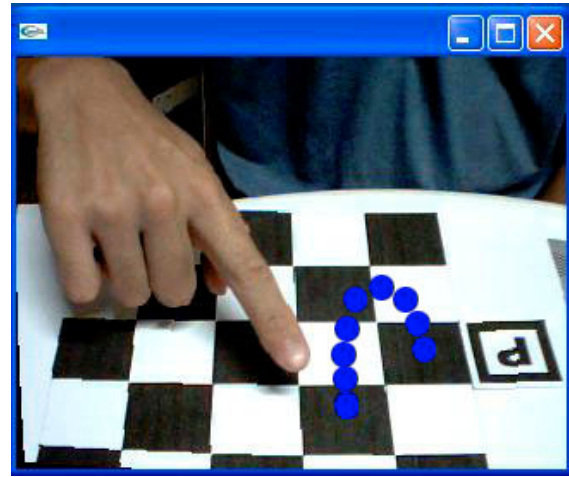


**Figura 57 – Outro trajeto possível, exibido com extrusão**

As Figuras 58 e 59 exibem os trajetos possíveis, utilizando esferas para demonstrar a trajetória.



**Figura 58 – Trajeto possível, exibido com esferas**



**Figura 59 – Segundo trajeto possível, exibido com esferas**

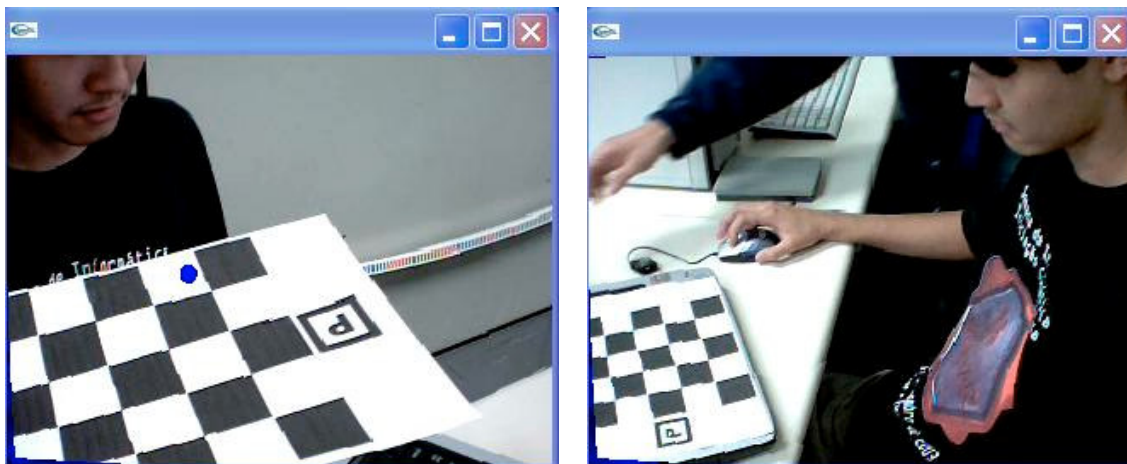
O sistema de treinamento pode ser utilizado em qualquer aplicação que seja beneficiada com a interatividade existente, sendo possível capturar os cenários do mundo real, sobrepondo a eles as trajetórias tridimensionais que representam as rotas a serem seguidas pelo usuário para a execução de operações.

Além disso, as aplicações em Realidade Aumentada permitem visualizar trajetórias em ambientes tridimensionais, acrescidas de outros elementos para facilitar o treinamento, como informações, sob a forma de textos explicativos, ou sons e narrações, induzindo o usuário aos procedimentos do treinamento.

#### **4.3.1. ANÁLISE DO EXPERIMENTO**

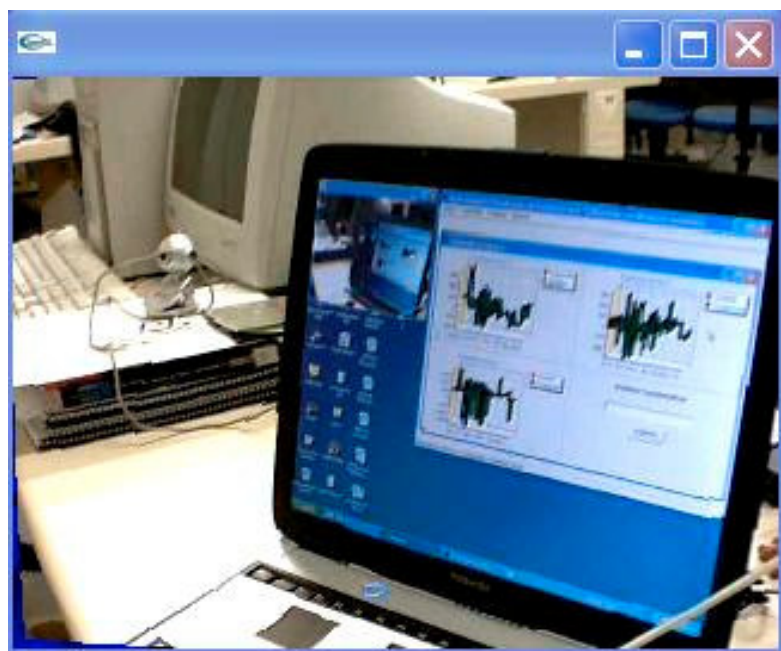
Foram realizados, por fim, alguns testes na ferramenta, para avaliar a contribuição da mesma para um treinamento eficaz. Foram medidas a trajetória sem treinamento e com treinamento, realizadas pelo usuário, e comparando-as, para se analisar os resultados finais. A Figura 60 mostra a utilização da ferramenta pelo usuário.





**Figura 60 – Experimento com o usuário**

A Figura 61 exibe a tela de avaliação, após o usuário realizar sua trajetória.



**Figura 61 – Experimento com o usuário – Tela de avaliação de desempenho**

Foram tiradas as seguintes conclusões, de acordo com a Tabela 1:

Tabela 1 – Comparação dos trajetos sem treinamento e com treinamento

Trajetória sem treinamento	Trajetória com treinamento
<ul style="list-style-type: none"> <li>- Sem instrução</li> <li>- Pouco envolvimento e interesse por parte do usuário</li> <li>- Trajeto realizado corretamente, por um usuário que conhece o jogo</li> <li>- O usuário não possui muitas opções de configuração, para interagir e realizar o treinamento</li> </ul>	<ul style="list-style-type: none"> <li>- Instrução tridimensional</li> <li>- Maior envolvimento e interesse do usuário pelo treinamento, devido às cores e animações</li> <li>- Trajeto realizado com maior perfeição</li> <li>- O usuário modifica o tipo de trajetória padrão na realização do treinamento</li> </ul>

#### 4.3.2. POTENCIALIDADES DA FERRAMENTA

O sistema projetado, implementando um ambiente de Realidade Aumentada, pode ter sua funcionalidade ampliada com um uso mais intensivo de sons e marcações sob a forma de textos no ambiente, bem como com o uso de cores, texturas e pulsações para indicar pontos a serem tocados ou trajetórias a serem seguidas. O sistema pode também ser usado como um sistema de autoria de atividades de treinamento, na medida em que pode armazenar trajetórias para serem mostradas ou comparadas posteriormente. Com o avanço da Educação à Distância, o sistema apresenta alto potencial de uso para gerar habilidades mecânicas, principalmente em cursos, onde isto é exigido, como educação física e fisioterapia, sem contar os cursos de treinamento propriamente ditos.

Foi realizada uma experiência com trajetórias tridimensionais, usando o software ARToolkit modificado. Com uma *webcam* ligada ao computador e o software ARToolkit modificado, foi possível associar os objetos que representam o trajeto à uma marca previamente cadastrada. Colocando-se a placa com a marca no campo de visão da webcam, o sistema insere o trajeto



sobre a placa, com dados vindos do banco de dados, podendo assim ser realizado um treinamento pelo usuário.

O sistema ARToolkit modificado tem a capacidade de reconhecer diferentes marcas em diferentes placas ao mesmo tempo, podendo trabalhar com vários objetos no ambiente computacional. Além disso, ele pode também utilizar tipos de trajetórias diferentes com o mesmo cartão marcador, permitindo que se explore a mesma atividade de formas diferentes, usando seqüência de esferas, linhas com extrusão, trajetórias pulsadas, trajetórias sonorizadas, etc.

#### **4.3.3. RESTRIÇÕES**

Uma das restrições do sistema é a necessidade de configuração, no banco de dados, da trajetória padrão.

Uma outra restrição está relacionada com a captura da trajetória realizada pelo usuário. O sistema pode armazenar muitos dados no banco de dados, devido a trajetetos não previstos, realizados pelo usuário, tornando a avaliação menos precisa.

Além disso, deve-se estar atento ao tamanho das placas utilizadas, pois, se forem muito grandes prejudicam a visualização, e se forem muito pequenas podem não ser reconhecidas pelo sistema, restringindo seu uso nos dois casos. Outra preocupação está em não se ocultar com as mãos, ou outro objeto, o marcador com seu símbolo, pois isto faria a trajetória desaparecer. Uma maneira de atenuar este problema é usar um cubo com o marcador nas várias faces, de forma que a oclusão de uma delas, deixando outra visível, mantenha a trajetória visível.

O próximo capítulo apresentará as conclusões dessa dissertação de mestrado.

# CAPÍTULO 5

## 5. CONCLUSÃO

Este trabalho abordou o projeto e o desenvolvimento de um sistema de Realidade Aumentada para apoiar atividades de treinamento de uma maneira mais natural.

A partir dos resultados obtidos com os experimentos, pôde-se verificar que a Realidade Aumentada possui grande capacidade para a implementação de ambientes tridimensionais sobrepostos com o mundo real, facilitando o entendimento do usuário, por meio de imagens que podem ser manipuladas naturalmente.

O treinamento do usuário é um importante passo para a realização de tarefas que, em muitos casos, pode ser facilitada com um ambiente de Realidade Aumentada.

A Realidade Aumentada tem a vantagem de apresentar uma interface mais amigável ao usuário, contribuindo com interações mais simples e naturais do usuário com o sistema de visualização. Por meio desse sistema, o usuário pode utilizar muito mais sua intuição, pois realiza os mesmos movimentos naturais que são utilizados no ambiente real.

A ferramenta desenvolvida incorporou recursos de Realidade Aumentada, otimizando a interação do usuário com os objetos virtuais, que representam as trajetórias e permitindo seu uso em atividades de treinamento.

A principal contribuição do trabalho refere-se ao desenvolvimento da ferramenta e à sua utilização para estimular o usuário a reproduzir trajetórias e avaliar seus movimentos, em relação a trajetórias de referência previamente cadastradas.

Para os desdobramentos do trabalho, a ferramenta pode ser ajustada para usar outros tipos de banco de dados, como o SQL, por exemplo, ou outros que são mais comuns em empresas de grande porte.

Atualmente, o sistema funciona no ambiente operacional Windows XP. Sugere-se fazer uma versão para ser utilizada em outros sistemas operacionais, como o Linux, que é um sistema operacional que está em constante crescimento e utilização, por ser um software livre.

Encontram-se em andamento estudos para implementar um módulo que faça a captura de uma trajetória sem a necessidade de marcadores, substituindo assim o uso desse módulo no ARToolkit.

## REFERÊNCIAS BIBLIOGRÁFICAS

ACKERMAN, M. J. **Visible Human.** Disponível em: <http://www.crd.ge.com/~lorensen/vw/#thevisiblewoman> Acesso em: 01/04/2006.

ACTIVE CHART. Disponível em: <http://www.rinvoice.com/activechart.htm> Acesso em: 2/04/2006.

AHLERS, K. H. et al. Distributed Augmented Reality for Collaborative Design Applications. In: EUROGRAPHIC'S, 1995. **Proceedings**. Maastricht, NL, 1995. p. 3-14.

AHLERS, M. A. VR Based Safety Training in a Petroleum Refinery. In: EUROGRAPHIC'S, 1999. Áustria, 1999. p. 1-3.

**ARTOOLKIT.** Disponível em: [http://www.hitl.washington.edu/research/shared\\_space/](http://www.hitl.washington.edu/research/shared_space/). Acesso em: 12/06/2004.

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

AZUMA, R. et al. Recent Advances in Augmented Reality. **IEEE Computer Graphics and Applications**, v .21, n.6, p. 34-47, 2001.

AZUMA, R. et al. A Survey of Augmented Reality. Hughes Research Laboratories, Malibu, CA, 1997.

AZUMA, R. T. A Survey of Augmented Reality. In: COMPUTER GRAPHICS - SIGGRAPH '95, Los Angeles, 1995. Proceedings, Course Notes #9: Developing Advanced Virtual Reality Applications. Los Angeles, 1995. p. 1-38.

BAJURA, M. et al. Camera Calibration for Video See-Through Head-Mounted Display. UNC Chapel Hill Department of Computer Science technical report, Julho, 1993.

BARBETTA, P. A. **Estatística para Cursos de Engenharia e Informática**. São Paulo: Atlas, 2004.

BILLINGHURST, M.; KATO, H. Collaborative Augmented Reality. **Communications of the ACM**, p. 64-70, 2002.

BISHOP, G. et al. Reseach Directions in Vu Environments. **Computer Graphics - ACM**, 26(3):153-177, Agosto, 1992.

BROWER, T. et al. **Virtual Reality for Lego Mindstorms**. Disponível em: <<http://www.princeton.edu/~tbrower/Indie/VirtualRealityforLegoMindstorms.pdf>> Acesso em: 02/04/2004.

BUK, C.; KIRNER, C. Visualização de Dados em Ambientes com Realidade Aumentada. In: Workshop de Realidade Aumentada – WRA2004, I, Piracicaba: Universidade Metodista de Piracicaba, 2004. **Anais**. Piracicaba, 2004. p. 45-48.

CONNACHER, H. et al. **Virtual Assembly using Virtual Reality Techniques**. Disponível em: <<http://www.nist.gov/msidlibrary/doc/cadtech95.pdf>> Acesso em: 02/04/2004.

DATE, C. J. **Introdução a Sistemas de Banco de Dados**. Rio de Janeiro: Campus, 2000.

DIEHL, S. **Distributed Virtual Worlds: Foundations and Implementation Techniques Using VRML, Java, and CORBA**. Berlin: Springer, 2001, 166 páginas.

**DONFREWARE**. Disponível em: < <http://www.donfreeware.com/armys.jpg>>. Acesso em: 01/05/2006.

**EMBRAER**. Disponível em: <<http://www.embraer.com.br/portugues/content/empresa/technology.asp>>. Acesso em: 01/05/2006.

FEINER, S. K. Augmented Reality: A New Way of Seeing. **Scientific American**, 24/04/2002. Disponível em:

<[http://www.sciam.com/print\\_version.cfm?articleID=0006378C-CDE1-1CC6-B4A8809EC588EEDF](http://www.sciam.com/print_version.cfm?articleID=0006378C-CDE1-1CC6-B4A8809EC588EEDF)> Acesso em: 15/10/2004.

**FIREBIRD**. Disponível em: <<http://www.comunidade-firebird.org/>>. Acesso em: 25/01/2006.

**GENERAL REALITY COMPANY**. Disponível em: <<http://www.genreality.com>> Acesso em: 26/07/2004.

GERALDI, T.; KIRNER, C. A. Ambientes Colaborativos com Realidade Aumentada. In: Workshop de Realidade Aumentada – WRA2004, I, Piracicaba: Universidade Metodista de Piracicaba, 2004. **Anais**. Piracicaba, 2004. p. 13-16.

GERSHON, N. et al. Information Visualization. In: interactions, Jg. 1998, ACM Press, New York, NY, vol. 5, ISSN:1072-5520, p. 9-15.

INTERNATIONAL STANDARD ISO/IEC 14772-1:1997. **The Virtual Reality Modeling Language**, 1997. Disponível em: <[http://www.web3d.org/x3d/specifications/vrml/ISO\\_IEC\\_14772-All/](http://www.web3d.org/x3d/specifications/vrml/ISO_IEC_14772-All/)> Acesso em: 06/10/2004.

KATO, H. et al. **ARToolKit 2.33 Documentation**. Seattle, 2000.

KIRNER, C. Realidade Virtual: Dispositivos e Aplicações. In: Escola Regional de informática da SBC Regional Sul - VII ERI, SBC, Londrina, Chapecó, Novo Hamburgo, 1999. **Anais**. Londrina, Chapecó, Novo Hamburgo, 1999, p. 135-158.

KIRNER, C. (2004a). Mãos Colaborativas em Ambientes de Realidade Misturada. In: Workshop de Realidade Aumentada – WRA2004, 1º, Piracicaba, 2004. **Anais**. Piracicaba: Universidade Metodista de Piracicaba, 2004. p. 45-48.

KIRNER, C. (2005a). Visualização de Informações em Ambientes de Realidade Aumentada. In: Workshop de Realidade Aumentada, II, Piracicaba, 2005.

**Anais.** Piracicaba: Universidade Metodista de Piracicaba, São Paulo. 2005. p. 68-71.

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. (2005b). Aplicações Educacionais em Ambientes Colaborativos com Realidade Aumentada. **XVI Simpósio Brasileiro de Informática na Educação**, Juiz de Fora – MG, 2005.

KIRNER, T. G. et al. Development Of An Information Visualization Tool Using Virtual Reality. In: ACM Symposium on Applied Computing – SAC'2000, 15<sup>th</sup>, Como, 2000. **Proceedings**. Como, 2000, p. 604-607.

KISMET. **The Karlsruhe Endoscopic Surgery Trainer**. Disponível em: <[http://iregt1.iai.fzk.de/TRAINER/mic\\_trainer1.html](http://iregt1.iai.fzk.de/TRAINER/mic_trainer1.html)> Acesso em: 01/04/2006.

KNIGHT, C. Virtual Reality for Visualization. **Computer Science Technical Report**, 1998. Disponível em: <<http://vrg.dur.ac.uk/papers/papersearch.php3?year=ALL>> Acesso em: 12/04/2003.

KOLLER, D. et al. Real-time {Vision-Based} Camera Tracking for Augmented Reality Applications. In: Symposium on Virtual Reality Software and Technology, ACM, Lausanne – Switzerland, 1997. **Proceedings**. Lausanne – Switzerland, 1997. p. 87-94.

KUSTABORDER, J.; SHARMA, R. Experimental Evaluation of Augmented Reality for Assembly Training. **Proc. of the Second IEEE and ACM International**. Workshop on Augmented Reality, I WAR, 1999.

LAHR, P.; LOURENÇO, P. C.; DAINESE, C. A. Rastreamento em Realidade Aumentada com ARToolkit e Marcadores. In: Workshop de Realidade

Aumentada – WRA2004, I, Piracicaba: Universidade Metodista de Piracicaba, 2004. **Anais**. Piracicaba, 2004. p. 37-40.

LUO, X. **An Augmented Reality Training Environment for Post-Stroke Finger Extension Rehabilitation**. Disponível em: <<http://www.evl.uic.edu/files/pdf/XLuo-ICORR05.pdf>>. Acesso em: 01/03/2006.

MALETIC, J. I.; LEIGH, J.; MARCUS, A. Visualizing Software in an Immersive Virtual Reality Environment, In: ICSE' 01 Workshop on Software Visualization, Toronto, Ontario, 2001. **Proceedings**. Toronto, Ontario, 2001. p. 49-54.

**MICROVISION**. Disponível em: <<http://www.mvis.com>>. Acesso em: 05/04/2006.

MORCLETTE, C. **VRML Generation Tools for Visualization of Database Content in three dimensions**. Massachusetts Institute of Technology, 1999.

**MXR Toolkit**. Disponível em <[http://155.69.54.110/SOFTWARE/FILES/SOFTWARE/MXR.Tool.Kit/All\\_document/HTM\\_format/MXRToolkit.htm](http://155.69.54.110/SOFTWARE/FILES/SOFTWARE/MXR.Tool.Kit/All_document/HTM_format/MXRToolkit.htm)>. Acesso em 10/12/2005.

PINHO, M. S. **Tópicos Especiais em Computação Gráfica**. Faculdade de Informática – PUCRS, 2002.

RAPOSO, A. B.; SZEMBERG, F.; GATTASS, M.; CELES, W. Visão Estereoscópica, Realidade Virtual, Realidade Aumentada e Colaboração. In: Congresso da Sociedade Brasileira de Computação, XXIV, V.2, XXIII JAI – Livro Texto, SBC, Brasil, 2004, cap. 7. **Anais**. Brasil, 2004. p. 289-331.

SANTIN, R.; KIRNER, C. Ações Interativas em Ambientes de Realidade Aumentada com ARToolKit. In: Workshop de Realidade Aumentada – WRA2004, I, Piracicaba: Universidade Metodista de Piracicaba, 2004. **Anais**. Piracicaba, 2004. p. 26-30.

SANTIN, R.; KIRNER, C. Alteração Dinâmica de Objetos Virtuais em Realidade Aumentada. In: Workshop de Realidade Aumentada – WRA2005, II, Piracicaba: Universidade Metodista de Piracicaba, 2005. **Anais**. Piracicaba, 2005. p. 37-40.



SCHMALSTIERG, D. et al. Bridging multiple user interface dimensions with augmented reality. In: IEEE and ACM International Symposium on Augmented Reality (ISAR'00). Munich, Germany, 2000. **Proceedings**. Munich, Germany. 2000. p. 20-29.

SCHMALSTIERG, D. et al. The Studierstube Augmented Reality Project. Teleoperators and Virtual Environments, Vol. 11, No. 1, MIT Press, 2002. pp. 33-54.

SCHWALD, B. **An Augmented Reality System for Training and Assistance to Maintenance in the Industrial Context**. Disponível em: <[http://wscg.zcu.cz/wscg2003/Papers\\_2003/l23.pdf](http://wscg.zcu.cz/wscg2003/Papers_2003/l23.pdf)> Acesso em: 01/03/2006.

SHERMAN, W. R.; CRAIG, A. B. **Understanding Virtual Reality**. 2ª Ed. San Francisco - CA: Morgan e Morgan, 2003.

TAJRA, S. F. **Informática na Educação. Novas Ferramentas Pedagógicas para o Professor da Atualidade**. São Paulo, Érica, 2001.

TRINDADE, J. **Realidade Virtual, conceitos e aplicações**. Escola Superior de Tecnologia e Gestão, Instituto Politécnico da Guarda, Portugal, 1999.

**VIRTUAL TECHNOLOGY**. Disponível em: <<http://www.virtex.com>> Acesso em: 05/04/2006.

ZAUNER, J. et al. Authoring of a Mixed Reality Instructor for Hierarchical Structures. Proc. of the Second IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR: 2003 - 10 pp.

ZHONG, W. Z. **Collaborative Augmented Reality: A Prototype for Industrial Training**. Disponível em: <<http://www.discover.uottawa.ca/publications/queensApril07.pdf>> Acesso em: 01/03/2006.

WAYNE, R. **The Folks at AVS Get It**, Software Development Magazine. Disponível em: <[http://www.avs.com/software/soft\\_b/openviz/index.html](http://www.avs.com/software/soft_b/openviz/index.html)> Acesso em: 05/04/2006.

WHITE, J. **Virtual Reality and the Built Environment**. Great Britain: Architectural Press, 2002. 160 páginas.

WLOKA, M. M. Interacting with Virtual Reality. **In Virtual Environments and Product Development Processes**. Chapman and Hall, 1995.

## ANEXOS

### ANEXO 1 – PARTE DO PROTÓTIPO IMPLEMENTADO EM DELPHI

#### CÓDIGO PARA ATUALIZAÇÃO DO ARQUIVO VRML, UTILIZANDO EXTRUSÃO

```

procedure TForm4.BitBtn1Click(Sender: TObject);
var
  Texto: TextFile;
  Auxiliar: String[200];
begin {insere texto no arquivo}
  dm.consulta.close;
  dm.consulta.sql.clear;
  dm.consulta.sql.add('select * from trajetoria_padrao order by id_tra_padrao');
  dm.consulta.Prepare;
  dm.consulta.open;
  screen.cursor := crdefault;
  dm.consulta.first;
  while not dm.consulta.eof do
  begin
    If FileExists ('C:\ARToolKit2.6VRML\bin\Wr\teste5.wrl') then
      DeleteFile ('C:\ARToolKit2.6VRML\bin\Wr\teste5.wrl');
    AssignFile(Texto,'C:\ARToolKit2.6VRML\bin\Wr\teste5.wrl');
    Rewrite(texto);
    Append(Texto);
    Writeln(Texto, '#VRML V2.0 utf8');
    Writeln(Texto, 'Shape {');
    Writeln(Texto, 'geometry Extrusion { ');
    Writeln(Texto, 'ccw TRUE ');
    Writeln(Texto, 'convex FALSE');
    Writeln(Texto, 'solid TRUE ');
    Writeln(Texto, 'beginCap TRUE');
    Writeln(Texto, 'endCap TRUE ');
    Writeln(Texto, 'crossSection [0 0, 1 1, 1 0, 0 0]');
    Writeln(Texto, '# 1 modifica');
    Write(Texto, 'spine [ ');
    Auxiliar := Auxiliar + dm.consulta.fieldbyname('x').asstring + ' ' +
dm.consulta.fieldbyname('y').asstring + ' '+dm.consulta.fieldbyname('z').asstring + ' ';

```

```

    Write(Texto, Auxiliar);
    Writeln(Texto, ' ');
    Writeln(Texto, '}');
    Writeln(Texto, 'appearance Appearance { ');
    Writeln(Texto, 'material Material { ');
    Writeln(Texto, 'diffuseColor 0 0 1');
    Writeln(Texto, ' } }');
    CloseFile(Texto);
    sleep(500);
    dm.consulta.next;
end;
end;

```

### **CÓDIGO PARA CAPTURA DO TRAJETO REALIZADO PELO USUÁRIO**

```

procedure TForm3.BitBtn1Click(Sender: TObject);
var
    TXT : TextFile;
    Entradax,entraday,entradaz: String;
    x: integer;
Begin
    screen.cursor := crhourglass;
    dm.texecuta.starttransaction;
    dm.executa.close;
    dm.executa.sql.clear;
    dm.executa.sql.add('delete from trajetoria_capturada');
    dm.executa.execquery;
    dm.texecuta.commit;

    AssignFile(TXT,'C:\ARToolKit2.6VRML\bin\coordenadas.txt');
    Reset(TXT);
    ReadLn(TXT,Entradax);
    ReadLn(TXT,Entraday);
    ReadLn(TXT,Entradaz);
    Eid.text := '1';
    dm.texecuta.starttransaction;
    dm.executa.close;
    dm.executa.sql.clear;

```

```

dm.executa.sql.add('insert into trajetoria_capturada (id_TRA_CAP,x,y,z) values
(:var0,:var1,:var2,:Var3)');
dm.executa.params[0].value := eid.text;
dm.executa.params[1].value := entradax;
dm.executa.params[2].value := entraday;
dm.executa.params[3].value := entradaz;
dm.executa.execquery;
dm.texecuta.commit;
Eid.text := '2';
While not Eof(TXT) do
Begin
  ReadLn(TXT,Entradax);
  ReadLn(TXT,Entraday);
  ReadLn(TXT,Entradaz);
  if (trim(entradax)<>") and
    (trim(entraday)<>") and
    (trim(entradaz)<>") then
  begin
    if compara(entradax,entraday,entradaz) = true then
    begin
      dm.texecuta.starttransaction;
      dm.executa.close;
      dm.executa.sql.clear;
      dm.executa.sql.add('insert into trajetoria_capturada (id_TRA_CAP,x,y,z) values
(:var0,:var1,:var2,:Var3)');
      dm.executa.params[0].value := eid.text;
      dm.executa.params[1].value := entradax;
      dm.executa.params[2].value := entraday;
      dm.executa.params[3].value := entradaz;
      dm.executa.execquery;
      dm.texecuta.commit;
    end;
  end;
  eid.text := inttostr(strtoint(eid.text) + 1);
End;
CloseFile(TXT);
dm.consulta2.close; dm.consulta2.open;
screen.cursor := crdefault;
//showmessage('Concluído!');
End;

```

```

function tform3.compara(mx: string; my: string; mz: string): boolean;
var x: integer;
    sx,sy,sz: string;
begin
    sx := "";
    sz := "";
    sy := "";
    for x := 1 to length(mx) do
        begin
            if copy(mx,x,1) <> '.' then
                sx := sx + copy(mx,x,1)
            else break;
        end;
    for x := 1 to length(my) do
        begin
            if copy(my,x,1) <> '.' then
                sy := sy + copy(my,x,1)
            else break;
        end;
    for x := 0 to length(mz) do
        begin
            if copy(mz,x,1) <> '.' then
                sz := sz + copy(mz,x,1)
            else break;
        end;
    if eid.text <> '1' then
        begin
            if eid.text = '18' then
                begin
                    //showmessage
                end;
            if (ex.text=sx) and (ey.text=sy) and (ez.text=sz) then
                begin
                    result := false;
                end
            else
                begin
                    ex.text := sx;
                    ey.text := sy;

```

```
        ez.text := sz;  
        result := true;  
    end;  
end  
else  
begin  
    ex.text := sx;  
    ey.text := sy;  
    ez.text := sz;  
    result := true;  
end;  
end;
```

## ANEXO 2 – PARTE DO PROTÓTIPO IMPLEMENTADO EM JAVA

### CONEXÃO COM O BANCO

```
package banco;

import java.sql.Connection;
import java.sql.DriverManager;

public class Banco {
    private static Banco banco;
    private static Connection conexao;

    private Banco(){ //Como o construtor é private não é possível
        //instanciar com new, tem que usar o getInstance()
    }

    public void finalize(){
        desconecta();
    }

    public static Banco getInstance(){ //Garante que só UMA CONEXÃO
        if (banco==null){ //seja criada
            banco=new Banco();
            conecta();
        }
        return banco;
    }

    private static void conecta(){
        try{
            if (conexao==null){
                Class.forName("org.firebirdsql.jdbc.FBDriver");//
                conexao=DriverManager.getConnection(
                    "jdbc:firebirdsql:localhost/3050:/MESTRADO.GDB", "sysdba", "masterkey");
                System.out.println("Banco conectado...");
            }
        } catch(Exception ex){
            System.out.println("Erro conectando.\n"+ex);
        }
    }
}
```



```

}

private void desconecta(){
    try{
        conexao.close();
        System.out.println("Banco desconectado...");
    } catch(Exception ex){
        System.out.println("Erro desconectando.\n"+ex);
    }
}

public Connection getConnection(){
    return conexao;
}
}

```

### **TELA DE CONTROLE**

```

package visao;
import banco.DaoTrajetoriaCapiturada;
import banco.DaoTrajetoriaPadrao;
import entidades.TrajetoriaCapiturada;
import entidades.TrajetoriaPadrao;
import java.io.BufferedReader;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;
import javax.swing.JOptionPane;

public class LerArquivo extends javax.swing.JFrame {
    DaoTrajetoriaCapiturada daoCapiturada = null;
    DaoTrajetoriaPadrao daoPadrao = null;

    public LerArquivo() {
        initComponents();
    }
}

```

```
daoCapiturada = new DaoTrajetoriaCapiturada();
daoPadrao = new DaoTrajetoriaPadrao();
}

public int quantidadeLinhaArquivo(){
    int q = 0;
    try{
        FileReader fileReader = new FileReader(tfDiretorioCapturar.getText());
        BufferedReader arquivo = new BufferedReader(fileReader);

        while(arquivo.readLine() != null){
            q++;
        }

        arquivo.close();

    }catch(IOException e){
        System.out.println("Exception:"+e);
    }
    return q;
}

public void deletarRegistros(){
    ArrayList todosObjetos = new ArrayList();
    todosObjetos = daoCapiturada.listaObjetos(null);
    TrajetoriaCapiturada objeto;
    for(int i=0;i<todosObjetos.size();i++){
        objeto = (TrajetoriaCapiturada)todosObjetos.get(i);
        daoCapiturada.deletar(objeto);
    }
}

public void deletarRegistrosIguais(){
    ArrayList arrayObjetosCapturada1 = new ArrayList();
    TrajetoriaCapiturada objeto1;
    arrayObjetosCapturada1 = daoCapiturada.listaObjetos(null);

    ArrayList arrayObjetosCapturada2 = new ArrayList();
    TrajetoriaCapiturada objeto2;
```

```

arrayObjetosCapturada2 = daoCapturada.listaObjetos(null);

for(int i=0;i<arrayObjetosCapturada1.size();i++){
    if(i < arrayObjetosCapturada1.size()-1){
        objeto1 = (TrajetoriaCapturada)arrayObjetosCapturada1.get(i);
        int x1 = converteParaInteiro(objeto1.getX());
        int y1 = converteParaInteiro(objeto1.getY());
        int z1 = converteParaInteiro(objeto1.getZ());

        objeto2 = (TrajetoriaCapturada)arrayObjetosCapturada2.get(i+1);
        int x2 = converteParaInteiro(objeto2.getX());
        int y2 = converteParaInteiro(objeto2.getY());
        int z2 = converteParaInteiro(objeto2.getZ());

        if((x1 == x2)&&(y1 == y2)&&(z1 == z2)){
            daoCapturada.deletar(objeto1);
        }else{
            System.out.println("Objeto "+objeto1.getIdTraCap()+" Diferente de objeto
"+objeto2.getIdTraCap());
        }
    }
}

public int converteParaInteiro(double d){

    String s = ""+d;

    String valor = "";
    int i = 0;
    while(i < s.length()){
        if(String.valueOf(s.charAt(i)).equals(".")){
            i = s.length();
        }else{
            valor = valor + s.charAt(i);
        }
        i++;
    }
    System.out.println("inteiro:"+Integer.parseInt(valor));
    return Integer.parseInt(valor);
}

```

```
}

```

```
// <editor-fold defaultstate="collapsed" desc=" Generated Code ">
private void initComponents() {
    lbDiretorio = new javax.swing.JLabel();
    tfDiretorioCapturar = new javax.swing.JTextField();
    jPanel2 = new javax.swing.JPanel();
    btCapturar = new javax.swing.JButton();
    jScrollPane1 = new javax.swing.JScrollPane();
    taTrajetoria = new javax.swing.JTextArea();
    btExibir = new javax.swing.JButton();
    tfDiretorioExibir = new javax.swing.JTextField();
    lbDiretorio1 = new javax.swing.JLabel();
    btEstatistica = new javax.swing.JButton();

    getContentPane().setLayout(null);

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setTitle("Visualizar e Capturar Trajet\u00f3rias");
    lbDiretorio.setText("Diretorio:");
    getContentPane().add(lbDiretorio);
    lbDiretorio.setBounds(50, 20, 70, 20);

    getContentPane().add(tfDiretorioCapturar);
    tfDiretorioCapturar.setBounds(120, 20, 260, 19);

    jPanel2.setLayout(new java.awt.GridLayout(1, 2, 15, 0));

    btCapturar.setText("Capturar Trajet\u00f3ria do Usu\u00e1rio");
    btCapturar.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btCapturarActionPerformed(evt);
        }
    });

    jPanel2.add(btCapturar);

    getContentPane().add(jPanel2);
    jPanel2.setBounds(120, 50, 260, 23);

```

```

jScrollPane1.setViewportViewView(taTrajetoria);

getContentPane().add(jScrollPane1);
jScrollPane1.setBounds(50, 90, 390, 260);

btExibir.setText("Exibir Trajet\u00f3ria Padr\u00e3o");
btExibir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btExibirActionPerformed(evt);
    }
});

getContentPane().add(btExibir);
btExibir.setBounds(120, 400, 260, 23);

tfDiretorioExibir.setText("C:\\ARToolKit2.6VRML\\bin\\Wr\\teste.wrl");
getContentPane().add(tfDiretorioExibir);
tfDiretorioExibir.setBounds(120, 370, 260, 19);

lbDiretorio1.setText("Diretorio:");
getContentPane().add(lbDiretorio1);
lbDiretorio1.setBounds(50, 370, 70, 20);

btEstatistica.setText("Avalia\u00e7\u00e3o");
btEstatistica.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btEstatisticaActionPerformed(evt);
    }
});

getContentPane().add(btEstatistica);
btEstatistica.setBounds(395, 50, 100, 23);

java.awt.Dimension screenSize = java.awt.Toolkit.getDefaultToolkit().getScreenSize();
setBounds((screenSize.width-512)/2, (screenSize.height-484)/2, 512, 484);
}
// </editor-fold>

```

```

private void btEstadisticaActionPerformed(java.awt.event.ActionEvent evt) {
    ArrayList todosObjetosCapturada = new ArrayList();
    todosObjetosCapturada = daoCapiturada.listaObjetos(null);
    TrajetoriaCapiturada objetoCapturada;

    ArrayList todosObjetosPadrao = new ArrayList();
    todosObjetosPadrao = daoPadrao.listaObjetos(null);
    TrajetoriaPadrao objetoPadrao;
    double resultadoX = 0;
    double resultadoY = 0;
    double resultadoZ = 0;
    double rsFinalX = 0;
    double rsFinalY = 0;
    double rsFinalZ = 0;
    for(int i=0;i<todosObjetosPadrao.size();i++){
        objetoPadrao = (TrajetoriaPadrao)todosObjetosPadrao.get(i);
        objetoCapturada = (TrajetoriaCapiturada)todosObjetosCapturada.get(i);

        System.out.println("Objeto Capturada:"+objetoCapturada.getIdTraCap());
        System.out.println("Objeto Padrao:"+objetoPadrao.getIdTraPadrao());

        resultadoX = resultadoX +
            (objetoCapturada.getX()-objetoPadrao.getX())*(objetoCapturada.getX()-
objetoPadrao.getX());

        resultadoY = resultadoY +
            (objetoCapturada.getY()-objetoPadrao.getY())*(objetoCapturada.getY()-
objetoPadrao.getY());

        resultadoZ = resultadoZ +
            (objetoCapturada.getZ()-objetoPadrao.getZ())*(objetoCapturada.getZ()-
objetoPadrao.getZ());

    }

    rsFinalX = resultadoX / (todosObjetosPadrao.size()-1);
    rsFinalY = resultadoY / (todosObjetosPadrao.size()-1);
    rsFinalZ = resultadoZ / (todosObjetosPadrao.size()-1);
    taTrajetoria.setText("");
}

```

```

String mostrarResultado;
mostrarResultado = "Variância X: "+rsFinalX+"\nVariância Y: "+rsFinalY+"\nVariância Z:
"+rsFinalZ;
taTrajetoria.setText(mostrarResultado);
}

private void btExibirActionPerformed(java.awt.event.ActionEvent evt) {
try{
ArrayList todosObjetos = new ArrayList();
todosObjetos = daoPadrao.listaObjetos(null);
TrajetoriaPadrao objeto;
for(int i=0;i<todosObjetos.size();i++){
objeto = (TrajetoriaPadrao)todosObjetos.get(i);
FileWriter arquivo = new FileWriter(tfDiretorioExibir.getText());
PrintWriter saida = new PrintWriter(arquivo);
saida.println("#VRML V2.0 utf8");
saida.println("Transform{");
saida.println("# 1 modifica");
saida.print("translation ");
saida.println(""+objeto.getX()+" "+objeto.getY()+" "+objeto.getZ());
saida.println("children [");
saida.println("Shape {");
saida.println("appearance Appearance {");
saida.println("material Material {");
saida.println("diffuseColor 0 1 0");
saida.println("}");
saida.println("geometry Sphere { }");
saida.println("}");
saida.close();

int j = 0;
while(j < 10500){
System.out.println(""+j);
j++;
}

System.out.println("Mudado");
}
taTrajetoria.setText("");
}catch(IOException e){

```

```

        taTrajetoria.setText("Exception:"+e);
    }
}

private void btCapturarActionPerformed(java.awt.event.ActionEvent evt) {
    try{
        FileReader fileReader = new FileReader(tfDiretorioCapturar.getText());
        BufferedReader entrada = new BufferedReader(fileReader);

        deletarRegistros();

        int quantidadeLinha = 0;
        quantidadeLinha = quantidadeLinhaArquivo();
        System.out.println("Quantidade Linha:"+quantidadeLinha);

        int quantidade = 0;
        quantidade = quantidadeLinha/3;

        String trajetoria = "";
        String[][] dados = new String[quantidade][3];
        double valorConvertido;
        int j;
        TrajetoriaCapiturada objeto = new TrajetoriaCapiturada();
        for(int i=0;i<quantidade;i++){
            j = 0;
            while(j < 3){
                dados[i][j] = entrada.readLine();
                valorConvertido = Double.parseDouble(dados[i][j]);
                if(j == 0)
                    objeto.setX(valorConvertido);
                if(j == 1)
                    objeto.setY(valorConvertido);
                if(j == 2)
                    objeto.setZ(valorConvertido);
                j++;
            }
            objeto.setIdTraCap(i+1);
            daoCapiturada.inserir(objeto);
        }
    }
}

```



```

        deletarRegistrosIguais();

        taTrajetoria.setText(trajetoria);
        entrada.close();

    }catch(IOException e){
        taTrajetoria.setText("");
        taTrajetoria.setText("Exception:"+e);
    }
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new LerArquivo().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JButton btCapturar;
private javax.swing.JButton btEstatistica;
private javax.swing.JButton btExibir;
private javax.swing.JPanel jPanel2;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JLabel lbDiretorio;
private javax.swing.JLabel lbDiretorio1;
private javax.swing.JTextArea taTrajetoria;
private javax.swing.JTextField tfDiretorioCapturar;
private javax.swing.JTextField tfDiretorioExibir;
// End of variables declaration

}

```