



Paulo Gallotti Rodrigues

**v-Glove: Uma proposta de dispositivo de interação para
aplicações imersivas de realidade virtual**

Dissertação de Mestrado

Dissertação apresentada ao Programa de Pós-Graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio, como requisito parcial para obtenção do grau de Mestre em Informática.

Orientador: Prof. Alberto Barbosa Raposo

Rio de Janeiro

Abril de 2011

Paulo Gallotti Rodrigues

**v-Glove: Uma proposta de dispositivo de interação para
aplicações imersivas de realidade virtual**

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico e Científico da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

Prof. Alberto Barbosa Rapodo

Orientador

Departamento de Informática – PUC-Rio

Prof. Marcelo Gattass

Departamento de Informática – PUC-Rio

Prof. Hugo Fuks

Departamento de Informática – PUC-Rio

Dr. Felipe Gomes de Carvalho

Tecgraf – PUC-Rio

Prof. José Eugenio Leal

Coordenador Setorial do Centro
Técnico Científico – PUC-Rio

Rio de janeiro, 1 de abril de 2011

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador

Paulo Gallotti Rodrigues

Graduado em Informática na PUC-Rio em 2007. Sua área de pesquisa atual é Computação Gráfica. Sua experiência profissional inclui mais de 9 anos atuando como analista de sistemas e arquiteto de software na área de telecomunicações.

Ficha Catalográfica

Rodrigues, Paulo Gallotti

v-Glove: uma proposta de dispositivo de interação para aplicações imersivas de realidade virtual / Paulo Gallotti Rodrigues; orientador: Alberto Barbosa Raposo. – 2011.
102 f.: il. (color.) ; 30 cm

Dissertação (mestrado)–Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática, 2011.
Inclui bibliografia

1. Informática – Teses. 2. Realidade virtual. 3. Interação 3D. 4. IHC. I. Raposo, Alberto Barbosa. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

Dedico este trabalho ao meu filho Eduardo,
que nascerá dentro de alguns meses.

Agradecimentos

A minha esposa Fernanda, pelo seu amor e companheirismo, pelas inúmeras contribuições durante o desenvolvimento deste trabalho e especialmente pelo suporte na difícil decisão de dedicar-me integralmente ao mestrado.

Aos meus pais por serem grandes exemplos de vida e por terem sempre me apoiado nas decisões mais importantes da minha vida.

Ao meu orientador Alberto Barbosa Raposo, pelo apoio e por toda sua dedicação na elaboração do trabalho, e pela oportunidade de vir para o Tecgraf.

Aos amigos Luciano Soares, Manuel Loaiza e Mathias Zug pelas contribuições na materialização das ideias e na construção do hardware necessário.

Ao amigo Marcus Alencar pela consultoria em Interação Humano-computador e pelo auxílio prestado durante a realização dos testes com os usuários.

Ao pessoal do SecondLab, em particular à Katia Cánepa, Débora Cardador e ao Eduardo Velloso pela grande ajuda no aprendizado do Lilypad Arduino.

Ao pessoal dos grupos do Galileu, Environ e Vídeo Interativo pela participação nos testes com usuários e pelas diversas sugestões e contribuições dadas durante a construção do trabalho.

Ao Tecgraf pelo suporte financeiro e pelo fornecimento de toda a infraestrutura necessária.

À Petrobras através do Ismael Santos, que apoiou a realização do trabalho e viabilizou minha vinda para o Tecgraf através do projeto Galileu.

Resumo

Rodrigues, Paulo Gallotti; Raposo, Alberto Barbosa. **v-Glove: Uma proposta de dispositivo de interação para aplicações imersivas de realidade virtual**. Rio de Janeiro, 2011. 102p. Dissertação de Mestrado – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Dispositivos de interação tradicionais como mouse e teclado não se adaptam adequadamente a aplicações imersivas, uma vez que sua utilização nesse tipo de ambiente não é ergonômica, já que o usuário pode estar em pé ou até mesmo em movimento. Além disso, utilizando o modelo atual de interação para esse tipo de aplicação (baseado em *wands* e mouses 3D), o usuário se vê obrigado a realizar diversas mudanças de contexto a cada vez que necessita realizar uma tarefa não suportada no modo imersivo, especialmente a entrada de símbolos. Essas mudanças constantes de contexto da imersão para o WIMP (*Windows, Icons, Menus and Pointers*) introduzem uma ruptura no modo de interação do usuário com a aplicação. O objetivo deste trabalho é explorar as possibilidades de uso de um dispositivo que mapeia uma interface de toque em um ambiente imersivo de realidade virtual. Foi desenvolvida uma luva para interação com ambientes imersivos de realidade virtual 3D (v-Glove) com duas funcionalidades principais: rastreamento da posição referente ao dedo indicador do usuário no espaço e a geração de uma vibração na extremidade do dedo indicador no momento em que este atinge a localização de uma área plana mapeada no espaço de interação. Foram feitos testes quantitativos e qualitativos com usuários para avaliar a v-Glove, comparando-a com um mouse 3D tradicionalmente utilizado em ambientes imersivos.

Palavras-chave

Realidade Virtual; Interação 3D; IHC.

Abstract

Rodrigues, Paulo Gallotti; Raposo, Alberto Barbosa (Advisor). **v-Glove: Proposing an interaction device for immersive virtual reality applications**. Rio de Janeiro, 2010. 102p. MSc Dissertation – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Traditional interaction devices such as mouse and keyboard don't adapt to immersive applications, since their use in this kind of environment isn't ergonomic, because the user may be standing or in movement. Moreover, in the current interaction model for this kind of application (based on wands and 3D mice), the users have to change context every time they need to execute a non-immersive task, specially the symbolic input. These constant context changes from immersion to WIMP (Windows, Icons, Menus and Pointers) introduce a rupture in the user interaction with the application. The objective of this work is to explore the use possibilities of a device that maps a touch interface in a virtual reality immersive environment. We developed a glove for interaction in 3D virtual reality immersive environments (v-Glove), which has two main functionalities: tracking of the position of the user's forefinger in the space and the generation of a vibration in the fingertip when it reaches an area mapped in the interaction space. We performed quantitative and qualitative tests with users to evaluate v-Glove, comparing it with a 3D mouse used in immersive environments.

Keywords

Virtual Reality; 3D Interaction; HCI.

Sumário

1	Introdução	14
2	Trabalhos Relacionados	17
2.1.	Hardware	17
2.1.1.	Sistema de rastreamento de dedos da A.R.T.	17
2.1.2.	Feedback tátil em aplicações imersivas de realidade virtual	19
2.2.	Software	20
2.2.1.	Tela de toque virtual	20
2.2.2.	Teclado 3D de realidade aumentada	23
2.2.3.	<i>Widgets</i> 3D	24
2.3.	Considerações	26
3	v-Glove: um dispositivo de interação para aplicações imersivas de realidade virtual	27
3.1.	Princípio de funcionamento	27
3.2.	Visão geral	27
3.3.	Luva	28
3.3.1.	Micro controlador Arduino	31
3.3.2.	Antena e modem XBee	31
3.3.3.	Feedback tátil	32
3.3.4.	Marcadores retroreflexivos	33
3.4.	Câmeras de rastreamento	34
3.4.1.	Posicionamento das câmeras	35
3.4.2.	Iluminador infravermelho	37
3.5.	Estações de controle	38
3.5.1.	Estação de controle de rastreamento e feedback tátil	38
3.5.2.	Estação de controle da aplicação	39
3.6.	Sistema da Cave	40
3.7.	Arquitetura de software	41
3.7.1.	Menu Rendering Service	42
3.7.2.	Wii Remote Connector	43
3.7.3.	VR Proxy	45
3.7.4.	Tactile Feedback Connector	47
3.7.5.	Configuration Manager	49
4	Estudo de Caso: Environ	50
4.1.	Adaptações no Environ	52

4.1.1. Interface de menus e <i>widgets</i>	52
4.1.2. Cursor do mouse	53
4.2. Experimento	54
4.2.1. Formato do Estudo	54
4.2.2. Grupos de Usuários	59
4.2.3. Perfil dos Participantes	59
4.2.4. Equipamentos	60
4.2.5. Teste Piloto	61
4.3. Resultados dos Testes	61
4.3.1. Sucesso da Tarefa (<i>task success</i>)	62
4.3.2. Duração da tarefa (<i>task time</i>)	62
4.3.3. <i>Métricas de situações de usabilidade (issue-based metrics)</i>	63
4.3.3.1. Frequência	63
4.3.3.2. Prioridade	64
4.3.3.3. <i>Rating</i>	64
4.3.3.4. Categoria	64
4.3.4. Métricas auto-relatadas (<i>self-reported metrics</i>)	69
4.3.4.1. Análise quantitativa	69
4.3.4.2. Análise qualitativa - sugestões de melhorias	71
4.4. Discussões	74
4.4.1. Comparação dos resultados das métricas de usabilidade e impacto do perfil dos usuários	74
4.4.2. Impacto do feedback tátil na v-Glove	76
4.4.3. Impacto do aprendizado na avaliação	77
5 Conclusão	79
5.1. Trabalhos futuros	80
5.1.1. Teste da v-Glove com tarefas de interação 3D	81
5.1.2. Ajuste na escala de cores do feedback visual	81
5.1.3. Suporte multi-toque e de gestos	81
5.1.4. Interação com as duas mãos	81
5.1.5. Alteração no Posicionamento das câmeras	82
5.1.6. Clicar através do movimento de pinçar entre os dedos	82
5.1.7. Plano de toque virtual relativo à localização do usuário	82
6 Referências Bibliográficas	83
Apêndice A – Estrutura de um programa Arduino e código fonte comentado	87
Apêndice B	90
Apêndice C – Material para Observação de Uso	92
C.1 Termo de Consentimento	92

C.2 Questionário inicial – Levantamento de perfil do participante	93
C.3 Formulário de acompanhamento dos testes	94
C.4 Questionários pós-testes	95
Apêndice D – Análise da variância (ANOVA)	97
D.1 ANOVA no desempenho das tarefas	97
D.2 ANOVA no questionário pós-testes	100

Lista de Figuras

Figura 2.1: Sistema de rastreamento da A.R.T. [ART Tracking 2011].	18
Figura 2.2: Luva para rastreamento de dedos da A.R.T. com 3 e 5 dedos [ART Fingertracking 2011].	18
Figura 2.3: Tactile Feedback at the Finger Tips [Scheibe, Moehring e Froehlich 2007].	20
Figura 2.4: Inicialização do teclado virtual estático [Tosas e Li 2004].	22
Figura 2.5: Virtual Touch Screen for Mixed Reality [Tosas e Li 2004].	23
Figura 2.6: ARKB: 3D Augmented Reality Keyboard [Lee e Woo 2003].	24
Figura 3.1: Visão geral da solução proposta	28
Figura 3.2: Primeiro protótipo da v-Glove	29
Figura 3.3: Diagrama esquemático do circuito da luva [Tune Glove 2011]	30
Figura 3.4: Micro-controlador Lilypad Arduino 328 [SparkFun 2011]	31
Figura 3.5: Modem XBee 1mW Chip Antenna [SparkFun 2011]	32
Figura 3.6: Lilypad Vibe Board [SparkFun 2011]	33
Figura 3.7: (a) Tecido refletivo 3M 8910 [3M 2011], (b) Tecido refletivo aplicado à v-Glove	33
Figura 3.8: Wii Remote [Wii Remote 2011]	35
Figura 3.9: Posicionamento das câmeras IR em relação ao usuário	36
Figura 3.10: Iluminador infravermelho	37
Figura 3.11: Estação de controle de rastreamento e feedback tátil com o modem XBee	39
Figura 3.12: Cave montada no CENPES da Petrobras	41
Figura 3.13: Diagrama de arquitetura do software	42
Figura 3.14: Diagrama de classes do Menu Rendering Service	43
Figura 3.15: Diagrama de classes do Wii Remote Connector	44
Figura 3.16: Diagrama de sequência da recepção de um evento de infravermelho	44
Figura 3.17: Diagrama de classes do VR Proxy	45
Figura 3.18: Diagrama de sequência do carregamento do Environ	46
Figura 3.19: Diagrama de sequência do envio de um comando para o Environ	47
Figura 3.20: Diagrama de classes do Tactile Feedback Connector	48
Figura 3.21: Diagrama de sequência da operação de acionar o feedback tátil	48
Figura 4.1: Tela principal do Environ em modo desktop [Raposo et al 2009]	50
Figura 4.2: Interface de menus do Environ no padrão WIMP [Raposo et al 2009]	51

Figura 4.3: <i>Widgets</i> criados neste trabalho	53
Figura 4.4: Variação das cores do cursor	54
Figura 4.5: Mouse 3D com acelerômetro utilizado nos testes [Air Mouse 2011]	55
Figura 4.6: Menu principal da aplicação de testes	55
Figura 4.7: Opções da segunda e terceira tarefas	56
Figura 4.8: Opções da quarta e quinta tarefas	57
Figura 4.9: Instruções para o usuário durante a realização das tarefas	58
Figura 4.10: Sala utilizada nos testes com os usuários	60
Figura 4.11: Percentual de sucesso por tarefa para cada dispositivo	62
Figura 4.12: Média de tempo por grupo	63
Figura 4.13: Classificação das situações de usabilidade por frequência e prioridade	66
Figura 4.14: Média de notas por grupos por dispositivo (visão geral)	70
Figura 4.15: Notas por <i>widget</i> para cada dispositivo	70
Figura 4.16: Percentual de situações de usabilidade por dispositivo	75
Figura 4.17: Opinião dos usuários sobre o recurso de feedback tátil na v-Glove	76
Figura 4.18: Evolução do aprendizado das tarefas	77

Lista de Tabelas

Tabela 4.1: Classificação dos componentes de interface gráfica do Environ	52
Tabela 4.2: Média de idade dos participantes	59
Tabela 4.3: Situações de usabilidade observadas	65
Tabela 4.4: Sugestões dos participantes	71
Tabela B.1: Comandos de configuração do modem XBee [Digi 2009]	91
Tabela D.2: Dados de desempenho dos participantes nas tarefas	98
Tabela D.3: ANOVA gerado no Excel para o desempenho dos participantes	99
Tabela D.4: Notas dadas pelos participantes para os dispositivos	100
Tabela D.5: ANOVA gerado no Excel para as notas dos participantes	101

1 Introdução

Recentemente interfaces de toque e multi-toque baseadas em gestos têm se tornado bastante comuns no cotidiano de usuários de sistemas computacionais tradicionais e principalmente móveis. Telefones celulares do tipo *smartphone* e *tablets* se mostram bastante adequados para esse tipo de interação, tendo, assim, um papel significativo na disseminação de um alfabeto de gestos bastante diversificado por parte de seus usuários.

Baseado nessa experiência de sucesso, pode-se imaginar que este tipo de interação possa ser aplicado também a ambientes imersivos de realidade virtual. Nos últimos anos as pesquisas no campo de realidade virtual (RV) têm se concentrado no desenvolvimento de técnicas de interação para seleção e manipulação de objetos, navegação e *wayfinding* [Bacim, Bowman e Pinho 2009; Kulik, Hochstrate, Kunert e Froelich 2009 e Smith e Du'Mont 2009]. Técnicas para controle de sistemas e aplicações foram também introduzidas, permitindo a mudança de estados, ajuste de valores escalares e especialmente a escolha de alternativas dentre objetos ou menus de opções. Entretanto, interfaces para tarefas de controle de sistemas em ambientes virtuais (AV) não foram estudadas de forma mais aprofundada [Bowman e Wingrave 2001].

Uma vez que essas tarefas de controle são partes integrantes de interfaces desktop convencionais, técnicas de interação 2D bem conhecidas foram adaptadas para ambientes virtuais. Essa abordagem funciona satisfatoriamente para um número limitado de *widgets* 3D de controle (como botões, por exemplo), mas não atende adequadamente a situações mais complexas como seleção e navegação em menus e entrada de símbolos (digitação de caracteres alfanuméricos). Assim, essas soluções de menus que integram abordagens 2D adaptadas para ambientes imersivos encaram problemas como a necessidade de maior habilidade manual para se alcançar um item de menu na interação espacial, bem como a falta de feedback tátil [Hand 2003].

Dispositivos de interface tradicionais como mouse e teclado não se adaptam adequadamente ao contexto da imersividade, uma vez que sua utilização nesse tipo de ambiente não é ergonômica, já que o usuário pode estar em pé ou até mesmo em movimento. Alguns dispositivos foram criados e aprimorados especificamente para estes ambientes, tais como *wands*, mouses 3D e luvas, fornecendo uma resposta de uso mais adequada ao contexto do usuário de ambientes imersivos. Por outro lado, estes dispositivos possuem um custo elevado, necessitam de uma infraestrutura muitas vezes complexa (cabos, câmeras e sensores) e apresentam limitações quando confrontados com situações comuns em ambientes baseados no paradigma WIMP (*Windows, Icons, Menus and Pointers*), como a manipulação de controles com *widgets* 3D e a entrada de símbolos em campos numéricos e alfanuméricos.

Dessa forma, utilizando o modelo atual de interação para aplicações imersivas (baseado em *wands* e mouses 3D), o usuário se vê obrigado a realizar diversas mudanças de contexto a cada vez que necessita realizar uma tarefa não suportada no modo imersivo, especialmente a entrada de símbolos. Essas mudanças constantes de contexto da imersão para o WIMP introduzem uma ruptura no modo de interação do usuário com a aplicação. Além disso, essas mudanças forçam o usuário muitas vezes a desmontar o aparato necessário à imersão, por exemplo, HMD (*head mounted display*), *wand* e luvas, obrigando-o a sentar-se em uma mesa para realizar as tarefas de interação necessárias e tendo que retomar o modo imersivo a seguir (sem contar que em alguns casos isso pode demandar tarefas adicionais como desabilitar temporariamente a projeção estereoscópica e outros parâmetros da aplicação). Isso prejudica muito a utilização desse tipo de aplicação por períodos mais longos de tempo ou sem o suporte de uma segunda pessoa para as tarefas de mudança de contexto.

O objetivo deste trabalho é explorar as possibilidades de uso de uma interface de toque em um ambiente imersivo de realidade virtual. A pergunta a ser respondida é: “Como tentar usar algo similar a uma interface de toque (ou multi-toque) dentro de uma Cave?”. A abordagem escolhida foi o uso de uma tela de toque virtual mapeada no espaço onde o usuário da aplicação se encontra. Pretende-se investigar quais as vantagens e desvantagens desse tipo de abordagem e o que ainda ficaria faltando para uma solução satisfatória de entrada simbólica em ambientes imersivos.

Durante o desenvolvimento desta dissertação foi construída uma luva para interação com ambientes imersivos de realidade virtual 3D, chamada de v-Glove. Essa luva foi concebida através da utilização de componentes eletrônicos da família Lilypad Arduino [Buechley et al 2008] em conjunto com um sistema de rastreamento em três dimensões baseado em câmeras de vídeo ajustadas para captura de luzes infravermelhas. As câmeras utilizadas foram as disponíveis no controle remoto do videogame Wii, também conhecido como Wii Remote [Wii Remote 2011]. Os componentes Lilypad Arduino utilizados foram o micro controlador Arduino, atuadores de vibração e rádio transmissores para comunicação sem fio entre os outros componentes.

A luva construída tem duas funcionalidades principais. A primeira é o rastreamento da posição referente ao dedo indicador do usuário no espaço monitorado pelas câmeras e a sua conversão para coordenadas do tipo X, Y e Z. A segunda é a geração de uma vibração na extremidade do dedo indicador no momento em que essas coordenadas corresponderem à localização de uma área plana mapeada nesse mesmo espaço. Com base nessas duas funcionalidades pretende-se simular a utilização de uma tela de toque virtual em um ambiente imersivo.

Realizamos ainda uma avaliação de usabilidade da v-Glove, comparando duas versões da luva a um dispositivo de interação já consagrado em ambientes de realidade virtual. As versões da luva comparadas diferem apenas no recurso de feedback tátil, onde uma delas tem este recurso ativado e a outra não. Como dispositivo de interação alternativo foi utilizado um mouse 3D. Foram realizados testes quantitativos e qualitativos com três grupos de usuários com diferentes perfis de utilização de aplicações de realidade virtual 3D.

O restante deste trabalho está organizado da seguinte forma. O capítulo 2 descreve alguns trabalhos relacionados que servirão como referência e inspiração para essa dissertação. A solução proposta é apresentada em detalhes no capítulo 3. No capítulo 4 é apresentado um estudo de caso da solução proposta aplicada ao programa Environ [Raposo et al 2009] e no capítulo 5 são apresentadas as conclusões e algumas propostas de trabalhos futuros.

2 Trabalhos Relacionados

Levando-se em consideração que neste trabalho é proposta a montagem de uma solução composta por hardware e software, a seção de trabalhos relacionados foi dividida com base neste mesmo critério. Inicialmente são discutidos trabalhos referentes ao hardware construído e posteriormente serão tratados trabalhos relacionados ao software desenvolvido. Trabalhos que tenham servido de inspiração em ambas as categorias serão classificados na categoria onde entende-se que sua contribuição foi maior.

2.1. Hardware

Em termos de hardware foram pesquisados trabalhos que propusessem técnicas de interação com as mãos livres através do rastreamento com luvas ou técnicas de visão computacional baseadas em marcadores. Uma linha complementar de pesquisa diz respeito às vantagens da utilização do recurso de feedback tátil e sua influência na percepção da imersividade por parte do usuário. Alguns dos trabalhos pesquisados serão apresentados nas seções a seguir.

2.1.1. Sistema de rastreamento de dedos da A.R.T.

A empresa A.R.T. fabrica sistemas de rastreamento para ambientes imersivos de realidade virtual e possui uma luva para rastreamento de dedos [ART Fingertracking 2011], que é parte integrante de seu sistema completo de rastreamento (Figura 2.1 e Figura 2.2). Essa luva permite o rastreamento da orientação da mão do usuário e da posição de cada um dos dedos. Ela pode ser usada tanto na mão direita quanto na mão esquerda, está disponível em versões para 3 ou 5 dedos e provê comunicação sem fio entre a luva e a central de controle.

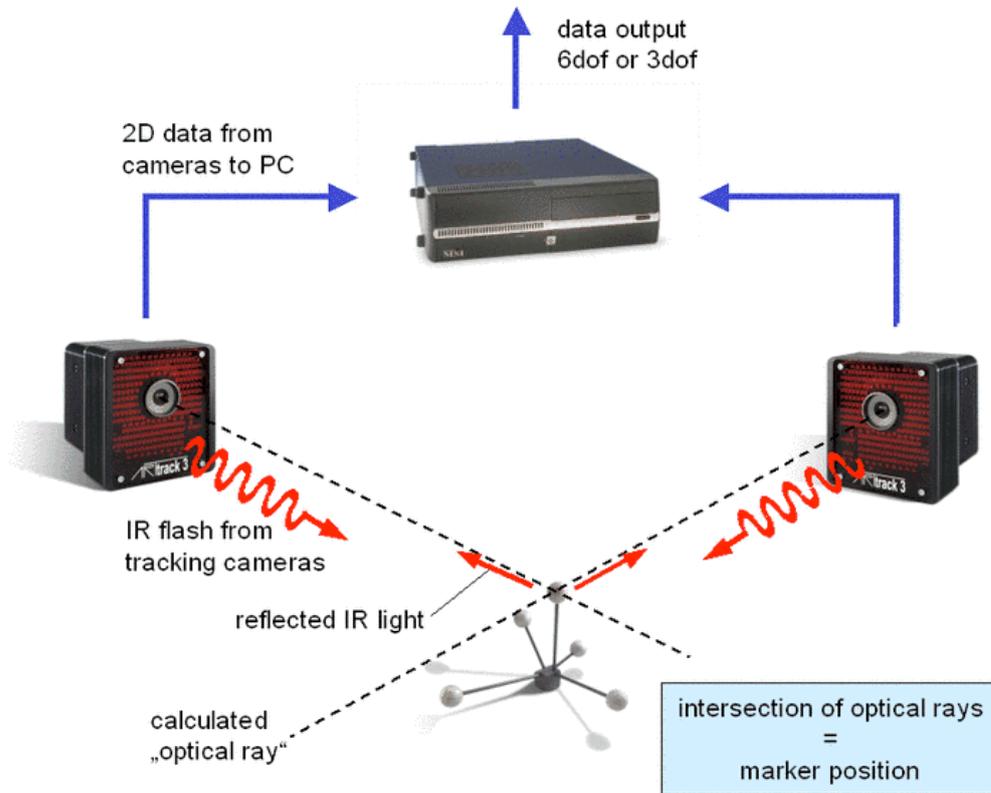


Figura 2.1: Sistema de rastreamento da A.R.T. [ART Tracking 2011].

Para identificar a posição dos dedos é utilizado um par de LEDs infravermelhos em cada dedo. Cada um desses conjuntos de LEDs emite luzes em uma frequência diferente, permitindo assim que as câmeras identifiquem os dedos de forma individual. A alimentação da luva é feita através de uma bateria recarregável.



Figura 2.2: Luva para rastreamento de dedos da A.R.T. com 3 e 5 dedos [ART Fingertracking 2011].

Uma limitação desse produto é a ausência de um módulo de feedback tátil na solução comercializada. O fabricante anunciou, entretanto, que já existe um protótipo com essa facilidade e que em breve será lançada uma nova versão comercial incluindo esse recurso. O conceito a ser aplicado neste caso é similar ao proposto por Scheibe, Moehring e Froehlich (2007), onde fios envolvem a ponta dos dedos proporcionando uma sensação de toque ao serem encurtados ao mesmo tempo em que uma vibração é aplicada sobre eles. Um micro-controlador é responsável pelo controle desse acionamento.

Esse produto serviu como inspiração principalmente por introduzir a ideia da comunicação sem fio entre a luva e a estação de controle. Além disso, o fato de ser uma solução comercial bastante utilizada por empresas e centros de pesquisa ao redor do mundo contribuiu por indicar uma solução com um elevado grau de maturidade de desenvolvimento e boa aceitação por parte de seus usuários [ART References 2011].

2.1.2.

Feedback tátil em aplicações imersivas de realidade virtual

Scheibe, Moehring e Froehlich (2007) propõem um sistema de feedback tátil para interações feitas com a ponta dos dedos em aplicações imersivas de realidade virtual. O sistema é constituído por dedais com fios que envolvem as pontas de cada um dos dedos da mão (Figura 2.3). Para simular o contato com uma superfície, o comprimento dos fios é encurtado de forma a fazer com que eles entrem em contato direto com a ponta dos dedos. Em acréscimo a isso, uma vibração é provocada nos fios causando no usuário a percepção de um estímulo tátil. Tanto o encurtamento e relaxamento dos fios quanto o processo de vibração é controlado através de um micro controlador que recebe comandos de uma aplicação de realidade virtual. O artigo exemplifica o uso do dispositivo em um protótipo que simula o interior de um veículo e avalia o alcance e a interação dos usuários com objetos nesse ambiente virtual. Os estudos revelaram que o feedback tátil auxiliou na execução de tarefas de manipulação direta com maior confiabilidade e naturalidade. Os usuários preferiram o sistema com o feedback tátil à versão sem esse recurso, pois sentiram um maior controle nas tarefas de manipulação no interior do veículo.

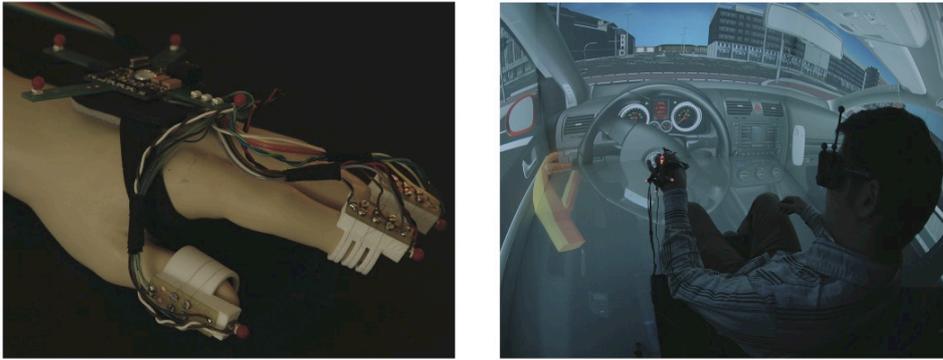


Figura 2.3: Tactile Feedback at the Finger Tips [Scheibe, Moehring e Froehlich 2007].

A principal inspiração que esse trabalho trouxe foi na aplicação do feedback tátil como forma de ampliar a sensação de imersão do usuário ao interagir com uma aplicação de realidade virtual. Uma vantagem da solução que estamos propondo se comparada a este trabalho é a ausência de fios entre a luva e o sistema de rastreamento e feedback tátil, o que melhora a mobilidade do usuário, principalmente considerando-se que em nosso caso de estudo esse usuário encontra-se em pé em uma Cave onde ele pode ter um elevado grau de mobilidade. No trabalho citado, por outro lado, o grau de mobilidade do usuário é limitado já que ele se encontra sentado no *cockpit* virtual do carro.

2.2. Software

Em relação ao software a ênfase da pesquisa foi na busca de iniciativas relacionadas ao conceito de uma tela de toque virtual, principalmente em relação a técnicas de interação com um sistema desse tipo. Uma vez que a literatura não é muito vasta neste aspecto, parte da pesquisa foi direcionada também para a área de realidade aumentada com foco em técnicas de interação diversas. Uma terceira linha de pesquisa foram *widgets* para aplicações 3D e suas vantagens e desvantagens dependendo do dispositivo de interação utilizado e do contexto do usuário. Alguns desses trabalhos serão apresentados nas seções a seguir.

2.2.1. Tela de toque virtual

Em Tosas e Li (2004) é proposta uma forma de interação para ambientes de realidade mista que permite a usuários interagir com uma tela de toque virtual

utilizando apenas as mãos. Recursos de visão computacional são utilizados para a detecção da mão e dos gestos referentes à digitação em um teclado numérico virtual. A ideia é que se possa interagir com elementos de interface gráfica de usuário apenas com as mãos através de cliques feitos com os dedos em janelas, ícones e menus de forma similar a sistemas gerenciadores de janelas como o MS Windows.

O sistema proposto é composto por um HMD (*Head Mounted Display*) que permite ao usuário a visualização da projeção do teclado virtual, uma câmera para filmar os movimentos da mão e o software de identificação e rastreamento da mão, que também gera uma resposta visual e audível quando teclas são pressionadas.

A detecção de um clique é feita através do encurtamento da imagem de um dedo da mão. O software monitora constantemente o comprimento dos dedos da mão, podendo assim identificar que uma variação nesse comprimento indica que o dedo foi dobrado com o intuito de se pressionar uma tecla.

O artigo propõe duas implementações de teclado virtual. A primeira é chamada de *teclado virtual estático*, ou seja, um teclado fixo em relação a uma localização no espaço. A interface virtual é inicializada no momento em que o usuário coloca sua mão sobre um desenho da forma da mão projetado no espaço pelo seu HMD. O sistema então grava uma imagem da mão do usuário e faz uma análise do modelo da mão daquele usuário, verificando parâmetros como o comprimento dos dedos, dos segmentos dos dedos, a tonalidade da pele etc. Uma vez feita essa identificação, o teclado virtual é apresentado e o usuário inicia sua interação (Figura 2.4).

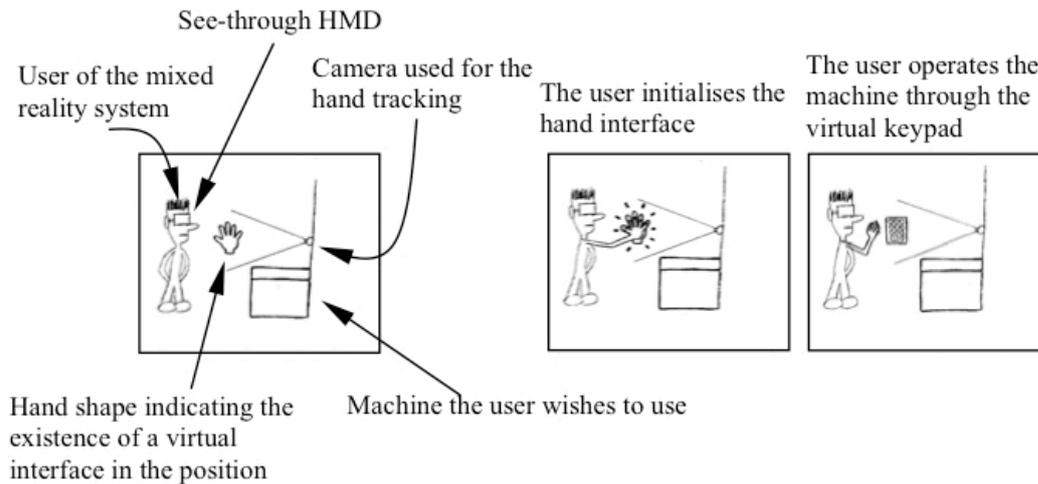


Figura 2.4: Inicialização do teclado virtual estático [Tosas e Li 2004].

A segunda implementação proposta por Tosas e Li (2004) é chamada de *teclado virtual flutuante*. Nesse caso a câmera para reconhecimento da mão é montada diretamente no HMD do usuário, o que faz com que o reconhecimento da mão tenha que ser feito pela parte de trás da mão, podendo ser mais complexo de se fazer do que o reconhecimento pela parte frontal. A vantagem dessa abordagem está na maior mobilidade do usuário, que pode se movimentar no ambiente virtual sempre “levando” o teclado consigo.

Uma situação observada nesse trabalho foi que a tela de toque virtual não é adequada para tarefas de entrada de dados muito longos, pois ao permanecer com a mão levantada por um período de tempo prolongado o usuário pode sentir cansaço e desconforto na mão e no braço. As situações de uso mais adequadas são aquelas em que é feita a seleção de itens de menu e de objetos presentes no ambiente virtual.

A prova de conceito construída no artigo limita-se a demonstrar o algoritmo de reconhecimento da mão e não chega a demonstrar todos os fundamentos teóricos discutidos no texto. Os autores criaram um teclado numérico onde o usuário digita números que são exibidos em um aplicativo de bloco de notas. A mão do usuário deve estar paralela ao plano da imagem da câmera e o fundo deve ser obrigatoriamente preto. O HMD não foi utilizado para projetar a imagem do teclado, sendo substituído por um painel de acrílico transparente com o desenho de uma grade representando os espaços das teclas do teclado virtual, como pode ser visto na Figura 2.5.

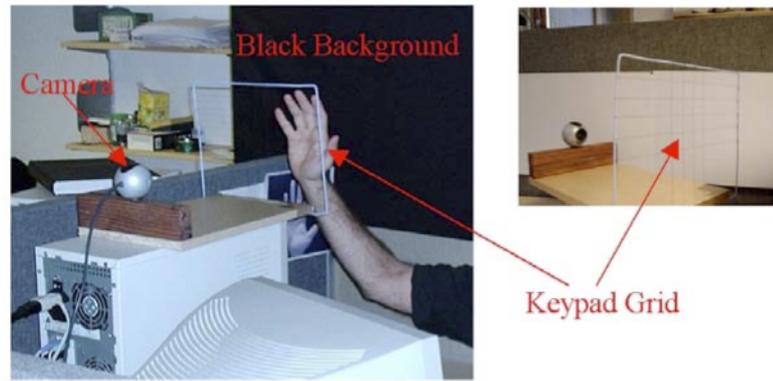


Figura 2.5: Virtual Touch Screen for Mixed Reality [Tosas e Li 2004].

Este trabalho introduziu a ideia de tela de toque virtual, que é um dos aspectos fundamentais dessa dissertação. Mesmo não o tendo implementado integralmente, o cenário de uso completo que ele propõe foi decisivo na idealização e na fundamentação do nosso trabalho.

2.2.2. Teclado 3D de realidade aumentada

Em Lee e Woo (2003) é proposto um teclado vestível 3D para realidade aumentada, chamado de ARKB (*3D Augmented Reality Keyboard*), que permite a um usuário digitar textos ou controlar objetos sem a utilização de interfaces convencionais, como teclado e mouse. O ARKB utiliza-se de informações de profundidade obtidas a partir de uma câmera estereoscópica acoplada a um HMD, possuindo três módulos: (i) rastreamento 3D das mãos, (ii) interação natural com os dedos e (iii) feedback audiovisual.

O rastreamento é baseado em técnicas de visão computacional e funciona através do reconhecimento de marcadores fiduciais capturados pela câmera estereoscópica. Essa câmera fica acoplada a um HMD e calcula a posição 3D e a orientação desses marcadores utilizando o ARToolKit [ARToolKit 2011]. O sistema então projeta a imagem do teclado virtual em frente ao usuário, ao mesmo tempo em que o ARKB detecta as pontas dos dedos e faz seu rastreamento no espaço através de marcadores coloridos presentes em cada um dos dedos (Figura 2.6).

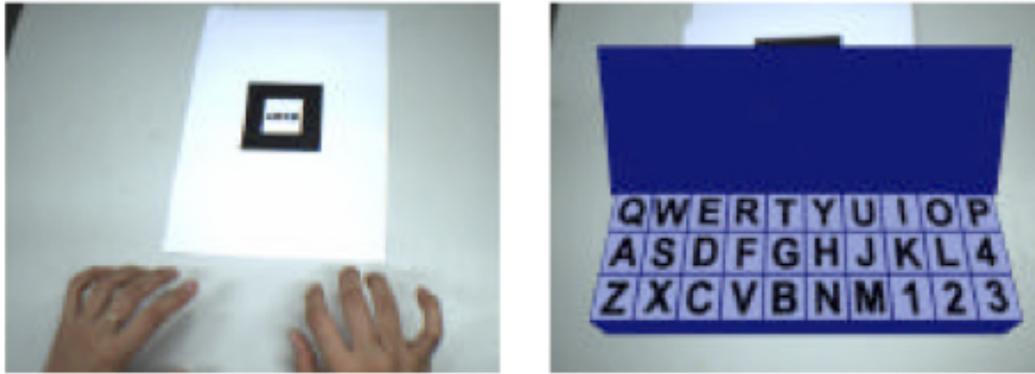


Figura 2.6: ARKB: 3D Augmented Reality Keyboard [Lee e Woo 2003].

A interação ocorre através da detecção de colisões entre as pontas dos dedos e as teclas virtuais utilizando-se informações de posicionamento 3D obtidas pelo módulo de rastreamento. Quando um dado número de pontos da nuvem de pontos gerada pela detecção das mãos colide com o volume de uma tecla virtual, assume-se que a digitação ocorreu. A mão do usuário é diferenciada do resto da imagem através da cor da pele e o feedback audiovisual ocorre sempre que uma colisão é detectada e tem como finalidade aumentar o realismo na experiência dos usuários.

A principal limitação dessa proposta é a baixa precisão no reconhecimento da digitação por conta de atrasos na geração e no processamento das imagens em tempo real. Além disso, esse formato do teclado virtual é próprio para utilização em superfícies planas como mesas, mas não é adequada a ambientes imersivos em RV onde os usuários estão na maior parte do tempo em pé. Por outro lado, o conceito de um teclado “móvel” e que pode ser ativado pelo usuário quando necessário serviu de inspiração para essa dissertação. O usuário de uma aplicação imersiva não precisa ter um teclado presente constantemente no seu ambiente de trabalho, mas precisa tê-lo à disposição no momento em que for utilizá-lo.

2.2.3. **Widgets 3D**

Em relação à construção de menus e *widgets* 3D, em Dachsel e Hubner (2007) é feita uma pesquisa detalhada acerca de diversas propostas de menus 3D, classificando-os de acordo com critérios como intenção de uso, aparência, localização, interação e usabilidade, dentre outros. Essa classificação assume

três contextos distintos de aplicação: ambientes virtuais imersivos e semi-imersivos, aplicações de realidade aumentada e realidade virtual para desktops. O resultado deste trabalho é uma taxonomia para menus 3D e sua aplicação a uma extensa variedade de menus 3D pesquisada na literatura.

Chen e Bowman (2009) propõem a adoção do projeto de técnicas de interação específicas por domínio de aplicação (*Domain-specific design*, ou DSD). Os autores comentam que existem duas abordagens mais comuns no projeto de técnicas de interação 3D para ambientes virtuais imersivos. Em muitos casos as técnicas de interação são projetadas para tarefas de uso genérico como a navegação, seleção, manipulação e controle do sistema, sem considerar seu domínio de uso. Por outro lado, outros tipos de técnicas de interação 3D são projetadas para aplicações específicas, o que gera bons resultados de usabilidade e utilidade para esta única aplicação, sendo sua reutilização bastante limitada. O principal objetivo do DSD é propor um meio termo entre essas duas abordagens, melhorando assim as atuais práticas e permitindo que as técnicas de interação 3D sejam projetadas de forma a resolver situações do mundo real, sem abrir mão do seu potencial de reutilização em um domínio em particular. Um framework de projeto em três camadas provê a fundamentação teórica do DSD, e os autores mostram como esse framework pode ser utilizado para ilustrar os múltiplos caminhos no projeto de técnicas de interação específicas para um domínio. O artigo apresenta um estudo de caso com uma aplicação real de realidade virtual para análise estrutural chamada Virtual-SAP, de forma a demonstrar como essa técnica pode ser aplicada. Os resultados experimentais indicam que o uso da abordagem DSD aumentou a utilidade dessa aplicação sem sacrificar sua usabilidade.

A partir desses trabalhos é possível perceber que a questão de formatos e padrões para menus e *widgets* 3D ainda demanda um aprofundamento nas pesquisas e uma maior discussão sobre padronização, semelhante ao que já existe em aplicações desktop 2D. Poucos trabalhos estudam *widgets* 3D desacoplados de um domínio específico de aplicação, o que talvez seja indicativo de que essa separação num contexto de aplicações de RV imersivas não faz muito sentido, ratificando iniciativas como o DSD proposto por Chen e Bowman (2009).

2.3. Considerações

Considerando-se que pesquisas sobre formas de interação para ambientes imersivos de RV ainda são muito concentradas na seleção, manipulação, navegação e *wayfinding* e que a literatura apresenta poucos trabalhos voltados para controle de aplicações, entrada simbólica e navegação em menus 3D, pretende-se neste trabalho buscar inspiração em técnicas já consagradas em outros tipos de ambientes para o ambiente da Cave. Propõe-se assim a construção de uma luva com feedback tátil para interação com uma tela de toque virtual mapeada dentro de um ambiente imersivo de RV. No Capítulo 3 discutiremos em detalhes cada um dos componentes dessa solução.

3

v-Glove: um dispositivo de interação para aplicações imersivas de realidade virtual

A proposta desse trabalho é a construção de uma luva para interação com uma tela de toque virtual mapeada em um sistema imersivo. Neste capítulo serão detalhados os componentes da solução proposta e, por isso, optou-se por uma abordagem narrativa na sequência em que o trabalho de prototipação foi desenvolvido. Assim os componentes criados serão apresentados na ordem em que foram desenvolvidos até que se chegasse à solução mais adequada. Situações avaliadas e que tenham sido descartadas posteriormente serão igualmente mencionadas com a finalidade de demonstrar o aprendizado adquirido durante cada uma das etapas do trabalho.

3.1.

Princípio de funcionamento

O princípio de funcionamento da v-Glove baseia-se no rastreamento da posição da ponta do dedo indicador da luva e do seu mapeamento para coordenadas em duas dimensões (X e Y), de forma similar a um mouse de mesa tradicional. Ao movimentar o dedo para o lado direito, por exemplo, o cursor do mouse desloca-se na tela seguindo esse mesmo padrão de movimento. A operação de clicar e arrastar objetos na superfície virtual funciona também baseada no rastreamento, sendo que neste caso é usado um terceiro eixo de referência (Z). Ao aproximar a ponta do dedo de um ponto pré-determinado em Z, o sistema detecta a intenção de clicar no objeto apontado pelo cursor naquele instante. Para reproduzir o movimento de arrastar basta manter o dedo nessa área de clique enquanto movimenta-se a mão nos eixos X e Y.

3.2.

Visão geral

A solução elaborada neste trabalho é composta pelos componentes de hardware mostrados na Figura 3.1. Todos os elementos construídos nesse

trabalho foram projetados pensando-se na sua utilização em uma Cave e, por isso, foram adaptados a características ergonômicas desse tipo de ambiente como a necessidade de mobilidade do usuário e a baixa luminosidade. Nas próximas seções serão apresentados em detalhes cada um desses componentes.

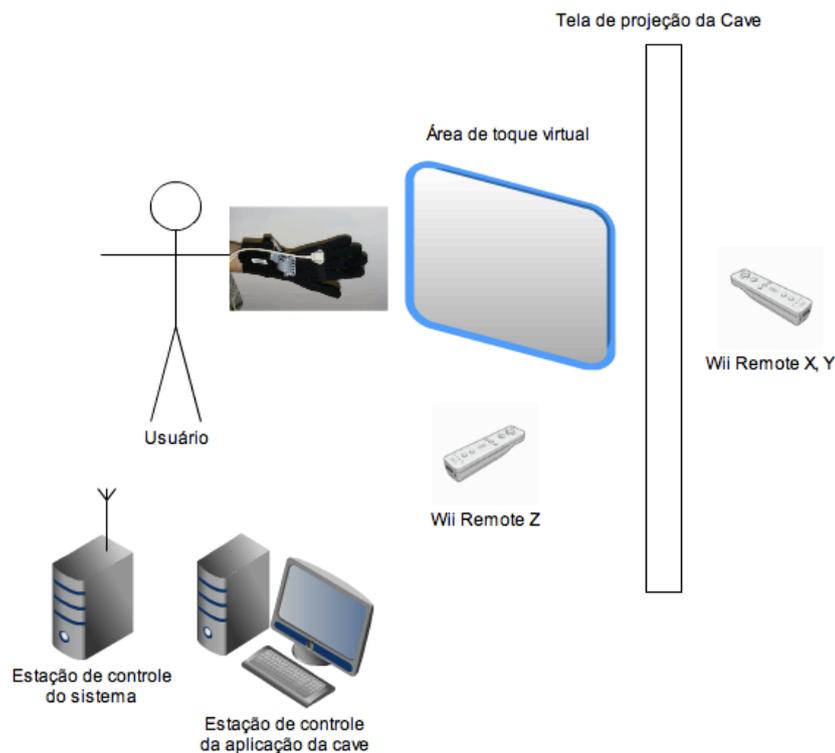


Figura 3.1: Visão geral da solução proposta

3.3. Luva

A primeira etapa deste trabalho foi a confecção de uma luva que permita a interação do usuário com a tela de toque virtual. A luva é feita de microfibras e foi costurada a partir de um molde de mão de um homem adulto, de forma a garantir um tamanho adequado para um maior número de usuários (Figura 3.2). Nada impede que futuramente sejam confeccionados modelos em tamanhos diferenciados conforme a necessidade. A escolha do tecido levou em consideração sua maleabilidade e espessura, pois é fundamental que a luva não restrinja a movimentação das mãos do usuário ou sua capacidade de manusear

uma *wand* sem ter que retirar a luva. Mesmo considerando-se que esses componentes serão utilizados em ambientes com controle de temperatura, é importante considerar que a luva não esquite demasiadamente a mão do usuário, por isso a preocupação com a espessura do tecido.



Figura 3.2: Primeiro protótipo da v-Glove

À luva foram inicialmente costurados os componentes eletrônicos responsáveis pelo sistema de feedback tátil. A costura dos componentes foi feita utilizando-se uma linha de costura com propriedades condutivas especialmente criada para essa finalidade [SparkFun 2011]. A linha de costura é banhada em prata e apresenta resistência elétrica de aproximadamente 45 Ohms por metro. A Figura 3.3 mostra um diagrama esquemático do circuito costurado na luva.

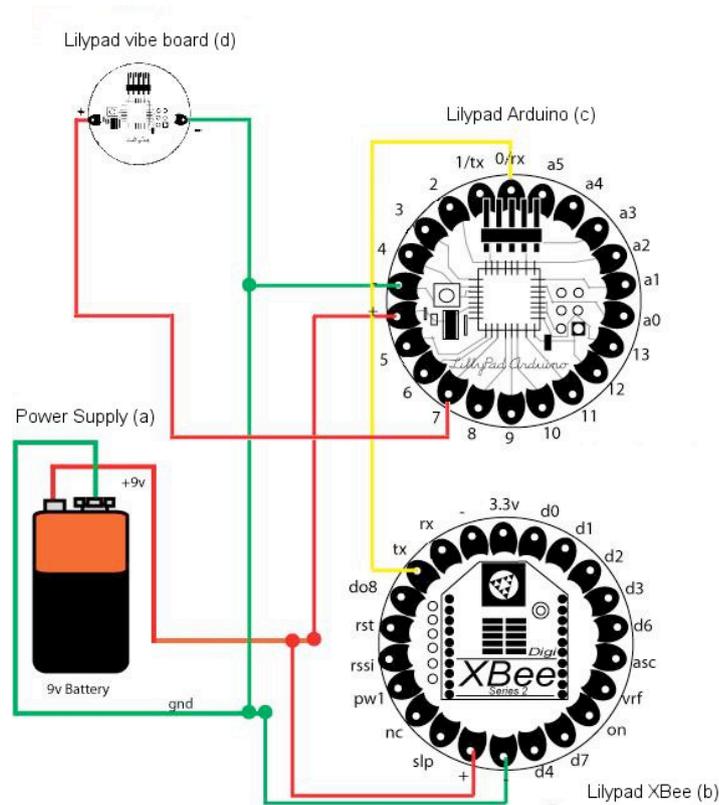


Figura 3.3: Diagrama esquemático do circuito da luva [Tune Glove 2011]

Uma bateria de 9V (a) provê a alimentação de todos os componentes do circuito. O módulo XBee acoplado ao adaptador Lilypad (b) recebe através de sua antena o comando proveniente da estação de controle. Os bytes são então enviados à porta de transmissão (tx) que está ligada diretamente à porta de recepção (0/rx) do micro-controlador Lilypad Arduino (c). O software armazenado na memória do micro-controlador recebe essa informação e envia sinal para a porta de número 7, que, por sua vez, está diretamente conectada ao polo positivo do atuador de vibração (d). Este, ao receber o sinal, faz com que o usuário sinta uma leve vibração na ponta dos dedos. A duração da vibração é controlada pelo software executado pelo micro-controlador Lilypad Arduino e dura 300 milissegundos. Maiores informações sobre este programa podem ser encontradas no Apêndice A.

3.3.1. Micro controlador Arduino

O micro-controlador Lilypad Arduino é uma versão do já consagrado processador Arduino [Arduino 2011] desenvolvida pela empresa norte-americana SparkFun Electronics [SparkFun 2011] com a finalidade de ser usado em roupas [Buechley et al 2008]. Este componente é composto por um microprocessador programável modelo Lilypad Arduino 328 (modelo ATmega328V) montado sobre uma placa de circuito impresso da linha de componentes Lilypad (Figura 3.4).

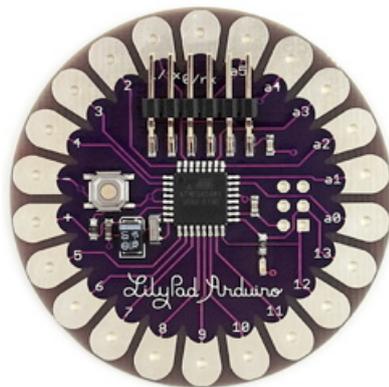


Figura 3.4: Micro-controlador Lilypad Arduino 328 [SparkFun 2011]

O Lilypad Arduino pode ser alimentado com tensões de 2.7 a 5.5 volts, possui quatorze portas de entrada e saída digital e seis portas de entrada analógica. Ele possui uma memória interna não volátil de 16 KB para armazenamentos dos programas e velocidade de *clock* é de 8 MHz. O módulo pode ser alimentado através de uma conexão USB utilizando um cabo FTDI/USB ou por uma fonte de alimentação externa.

3.3.2. Antena e modem XBee

O modem XBee é um componente fornecido pela empresa Digi International Inc. [Xbee 2011] e que pode ser acoplado a um módulo Lilypad XBee fornecido pela SparkFun Electronics para sua utilização em circuitos vestíveis de forma similar ao microprocessador Lilypad Arduino. O XBee implementa o protocolo de comunicação sem fio ZigBee [ZigBee 2004] e pode ser configurável através de uma conexão FTDI/USB de forma similar a modems

tradicionais. Mais detalhes sobre a configuração do XBee e os parâmetros utilizados nesse trabalho podem ser encontrados no Apêndice B. O modelo utilizado neste trabalho foi o XBee 1mW Chip Antenna, cuja antena é em formato de micro-chip, possui potência de 1mW, alcance de até 100 metros e taxa de transmissão de dados de 250 kbps.

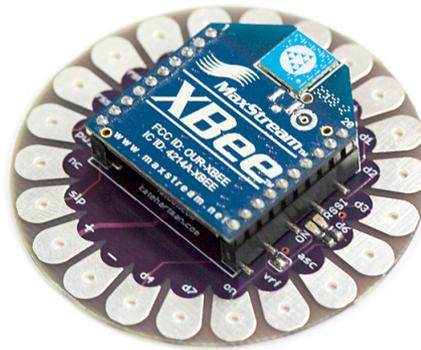


Figura 3.5: Modem XBee 1mW Chip Antenna [SparkFun 2011]

O módulo XBee montado na luva foi configurado como receptor. Um outro módulo idêntico foi conectado à estação de controle via porta USB, sendo a única diferença neste caso que o componente foi configurado como transmissor. Ao detectar um evento de toque, a aplicação executada na estação de controle envia um comando para o módulo XBee transmissor acoplado a ela. Este, por sua vez, envia o comando para o XBee receptor acoplado à luva que repassa a informação ao controlador Arduino, gerando o efeito de feedback tátil.

3.3.3. Feedback tátil

O componente Lilypad Vibe Board, também fornecido pela SparkFun Electronics, é um elemento atuador capaz de gerar uma vibração quando alimentado por um circuito de 5 Volts. Também construído em uma configuração vestível ele pode ser costurado de forma a funcionar com outros componentes similares como o Lilypad Arduino.



Figura 3.6: LilyPad Vibe Board [SparkFun 2011]

Para possibilitar o efeito desejado com o recurso de feedback tátil, o LilyPad Vibe Board foi costurado à ponta do dedo indicador da v-Glove.

3.3.4. Marcadores retroreflexivos

Na ponta do dedo indicador da luva é aplicada uma fita refletiva 3M modelo 8910 [3M 2011] (Figura 3.7(a)). Essa fita tem como finalidade refletir a luz infravermelha emitida pelo iluminador infravermelho posicionado próximo às câmeras e auxiliar no sistema de rastreamento dos dedos do usuário. A luz refletida pela fita é captada pela câmera IR e assim a posição do dedo do usuário pode ser estimada.



Figura 3.7: (a) Tecido refletivo 3M 8910 [3M 2011], (b) Tecido refletivo aplicado à v-Glove

Como sistema proposto utiliza duas câmeras IR, uma para captura das coordenadas X e Y e a outra para a coordenada Z, é importante que a fita retro-reflexiva seja visível pelas duas câmeras. Assim, a maneira mais adequada de se posicionar a fita para garantir uma boa visibilidade é envolvendo-a na ponta do dedo, conforme demonstrado na Figura 3.7(b).

Durante o desenvolvimento desse trabalho foram avaliadas algumas opções de materiais para serem usados como marcadores reflexivos. Uma primeira análise foi referente à utilização do dedo sem qualquer tipo de revestimento, mas viu-se que a pele humana reflete uma quantidade muito baixa de luz infravermelha. Uma segunda tentativa foi feita com papel alumínio convencional de uso doméstico. Este material possui uma alta capacidade de reflexão de luz convencional e também infravermelha, entretanto pela sua formação irregular a reflexão da luz infravermelha é captada com muito ruído pelas câmeras. A terceira opção avaliada foi a fita refletiva 3M 8910. Essa fita é bastante utilizada como elemento de sinalização urbana por sua alta capacidade refletiva à luz convencional e sua grande mobilidade (ela pode ser facilmente costurada a roupas, como por exemplo em uniformes de agentes de trânsito). O tecido refletivo 3M 8910 é composto por microesferas de vidro com grande angularidade que possibilitam alta visibilidade noturna e elevados índices de retro-refletividade. Nossos testes demonstraram que seu comportamento com luz infravermelha segue o mesmo padrão da luz convencional.

3.4. Câmeras de rastreamento

A câmera escolhida para o sistema de rastreamento foi a que acompanha o controle remoto do vídeo game Wii, conhecido como Wii Remote [Wii Remote 2011]. Essa câmera já possui um filtro para capturar luz apenas na faixa infravermelha, com maior sensibilidade para fontes de IR com comprimento de onda próximos a 940nm. O Wii Remote utiliza tecnologia Bluetooth de comunicação sem fio [Bluetooth 2011]. O modelo utilizado neste trabalho pode ser visto na Figura 3.8.



Figura 3.8: Wii Remote [Wii Remote 2011]

As principais vantagens da utilização do Wii Remote em relação a uma câmera convencional são: ele é sem fio, já possui filtro IR, possui bibliotecas que fornecem as coordenadas X e Y dos pontos de luz infravermelha capturados. Uma desvantagem do Wii Remote é que ele é capaz de identificar apenas quatro pontos de luz infravermelha simultaneamente, o que pode limitar uma aplicação que tenha a intenção de rastrear todos os dedos da mão. Isso não será um problema no nosso caso, pois a prova de conceito proposta nessa dissertação utiliza apenas o dedo indicador do usuário.

3.4.1. Posicionamento das câmeras

Como a câmera do Wii Remote fornece as coordenadas dos pontos em um plano de duas dimensões, acrescentamos uma segunda câmera posicionada lateralmente para a captura do eixo de profundidade. Dessa forma a solução final ficou sendo composta por duas câmeras independentes dispostas conforme o diagrama da Figura 3.9.



Figura 3.9: Posicionamento das câmeras IR em relação ao usuário

A primeira câmera é responsável pelo rastreamento nos eixos X e Y. Ela indica o posicionamento dos dedos do usuário nessas duas dimensões. A segunda câmera fica responsável pelo eixo Z. Ela fica posicionada na lateral do usuário e faz o mapeamento da superfície de toque virtual. Quando o dedo indicador do usuário se aproxima desse plano mapeado no espaço o software reconhece um toque na superfície e dispara um evento de feedback tátil para o software da luva. Assim é acionada a vibração no dedo do usuário.

Durante a construção do protótipo avaliamos a possibilidade de utilizar as câmeras com posicionamento baseado nos conceitos de geometria epipolar, [Hartley e Zisserman 2004]. Esta técnica tem como vantagem permitir a reprodução do espaço virtual em três dimensões, facilitando o rastreamento de múltiplos dedos, o que é impossível na topologia de câmeras atual. Isso ocorre porque a posição lateral da câmera do eixo Z faz com que ocorra oclusão entre dedos próximos, impossibilitando a detecção de profundidade dos dedos. Entretanto, como inicialmente não há necessidade de se identificar mais de um dedo e a solução atual não requer calibração das câmeras, procedimento necessário ao utilizar-se geometria epipolar, esta solução acabou sendo descartada. Além disso, como nosso interesse resumia-se a identificar se a v-Glove atingiu um determinado ponto no plano de toque virtual, a solução com a

câmera lateral atendeu aos requisitos da prova de conceito sem impactar na complexidade do software construído. Ela não é afetada por variações no posicionamento das câmeras e reduz o tempo de preparação do ambiente para a realização dos testes uma vez que não demanda calibração prévia.

3.4.2. Iluminador infravermelho

A câmera infravermelha presente no Wii Remote é sensível a uma certa intensidade de luz infravermelha. Considerando-se que essa aplicação será utilizada em um ambiente imersivo com iluminação reduzida, o componente iluminador infravermelho é fundamental. Ele é formado por oito LEDs infravermelhos de 5mm e comprimento de onda de 940nm alimentados por uma fonte de alimentação de 12 volts (Figura 3.10).

Os LEDs utilizados no iluminador são do mesmo tipo utilizado por dispositivos de controle remoto domésticos e podem ser facilmente encontrados em lojas de componentes eletrônicos.

Esse componente iluminador tem como finalidade a emissão de luz infravermelha que será refletiva pela fita retro-reflexiva presente na luva sendo então capturada pela câmera do Wii Remote.

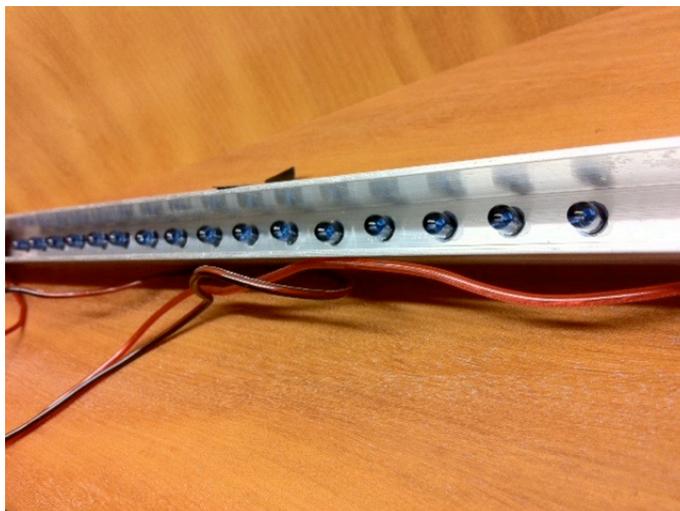


Figura 3.10: Iluminador infravermelho

Uma alternativa à utilização do iluminador infravermelho seria a colocação de um LED infravermelho diretamente na ponta do dedo indicador da luva, em

substituição à fita retro-reflexiva. Nesse caso a luz emitida pelo LED seria diretamente capturada pela câmera do Wii Remote, evitando-se o processo da reflexão. O problema dessa abordagem é que a presença de um LED na luva aumentaria o consumo de bateria da mesma, diminuindo-se assim sua autonomia. Pensando-se ainda na utilização de mais de um ponto de rastreamento esse problema aumentaria ainda mais.

3.5. Estações de controle

As estações de controle representam os microcomputadores utilizados no ambiente imersivo para controle do software da aplicação e da v-Glove. Neste trabalho definiremos conceitualmente duas estações de controle, que serão descritas nas subseções a seguir. Apesar de serem duas estações de controle, nada impede que ambas compartilhem a mesma plataforma de hardware.

3.5.1. Estação de controle de rastreamento e feedback tátil

A estação de controle do sistema de rastreamento e feedback tátil é o microcomputador onde o software desenvolvido é executado. Maiores detalhes sobre o software instalado nesta máquina serão discutidos na seção 3.7 (Arquitetura de Software). A Figura 3.11 mostra a estação de controle de rastreamento e feedback tátil com o modem XBee acoplado a uma das portas USB disponíveis na traseira do gabinete.



Figura 3.11: Estação de controle de rastreamento e feedback tátil com o modem XBee

Essa estação de controle é o componente do sistema responsável por gerenciar as coordenadas X, Y e Z enviadas pelas câmeras bem como o subsistema de feedback tátil. Além disso, o software presente nessa estação é responsável pela integração com as aplicações que são executadas na Cave de forma a fornecer eventos de entrada que possam permitir a interação do usuário com essas aplicações.

3.5.2. Estação de controle da aplicação

Este elemento do sistema é responsável pela execução da aplicação no sistema imersivo de realidade virtual. Em geral a estação de controle da aplicação é um microcomputador com uma elevada capacidade computacional e uma placa controladora gráfica de alto desempenho com capacidade para renderização de imagens em tempo real e suporte estereoscópico. No contexto deste trabalho a estação de controle da aplicação será responsável pela execução do programa Environ, que será melhor detalhado no capítulo 4 (Estudo de Caso: Environ).

3.6. Sistema da Cave

Cruz-Neira et al (1992) definem a Cave como uma interface de realidade virtual composta por uma sala cujas paredes, o teto e o piso possuem imagens projetadas. Um usuário presente neste ambiente tem seu entorno coberto por essas projeções. Ainda segundo os autores, o conceito da Cave se baseia em dois princípios: *suspension of disbelief*¹ e a perspectiva centrada no espectador. *Suspension of disbelief* indica a capacidade de um usuário de “se deixar levar” pela simulação, ou seja, de aceitar sua condição imersiva como real dentro do contexto da aplicação que está sendo utilizada. A perspectiva centrada no espectador propõe a visualização na perspectiva da localização do usuário, fazendo uso para isso de sensores que monitoram continuamente sua posição no espaço, ajustando a imagem projetada de acordo com essas informações.

O trabalho de Cruz-Neira et al (1992) foi o primeiro a propor essa idéia de telas ao redor do usuário. A partir dele, outras propostas surgiram com algumas variações e adaptações a contextos específicos de aplicação. Bowman et al (2004) citam alguns exemplos de trabalhos similares, como o CUBE (interface com 360 graus de visibilidade composta por 4 telas de projeção que ficam a cerca de 30 centímetros do rosto do usuário), a RAVE (dispositivo com tela flexível que pode ser configurado como uma tela plana de 9 metros ou como um ambiente imersivo com 4 telas), dentre outros.

¹ Suspensão da descrença seria uma tradução livre do termo *suspension of disbelief*. Entretanto, por não haver referência e este termo na literatura em português optamos por manter no texto o termo original em inglês.

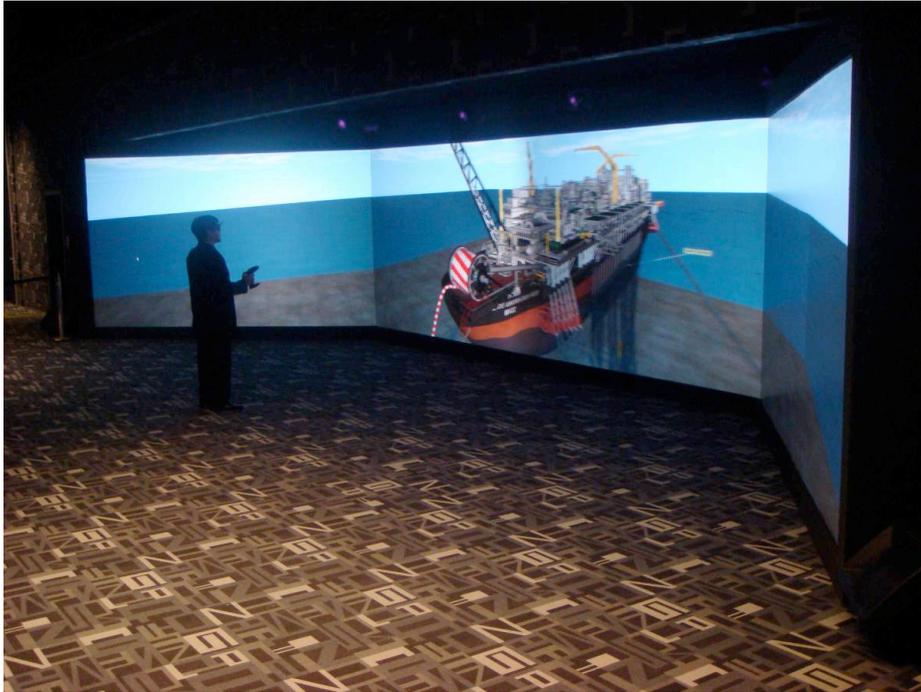


Figura 3.12: Cave montada no CENPES da Petrobras

A Figura 3.12 mostra a Cave construída no Centro de Pesquisas da Petrobrás (CENPES), no Rio de Janeiro. Como pode ser observado na imagem, esta Cave possui três telas com projeção estereoscópica de alta definição, não possuindo projeção no piso nem no teto. Um sistema de rastreamento permite ainda que a localização da cabeça do usuário seja rastreada por câmeras dispostas na parte superior das telas provendo suporte à perspectiva centrada no espectador.

3.7. Arquitetura de software

Todo o software desenvolvido neste trabalho usou como base o paradigma da orientação a objetos e a linguagem de programação escolhida foi Java. As bibliotecas de terceiros utilizadas seguem este mesmo princípio e possuem na sua maioria código fonte aberto ou possuem autorização de seus responsáveis para utilização em projetos sem fins lucrativos como este.

A arquitetura do software foi dividida em 5 módulos principais que interagem de forma a prover todas as funcionalidades necessárias para a aplicação da luva. Nas próximas seções esses módulos são apresentados em detalhes.

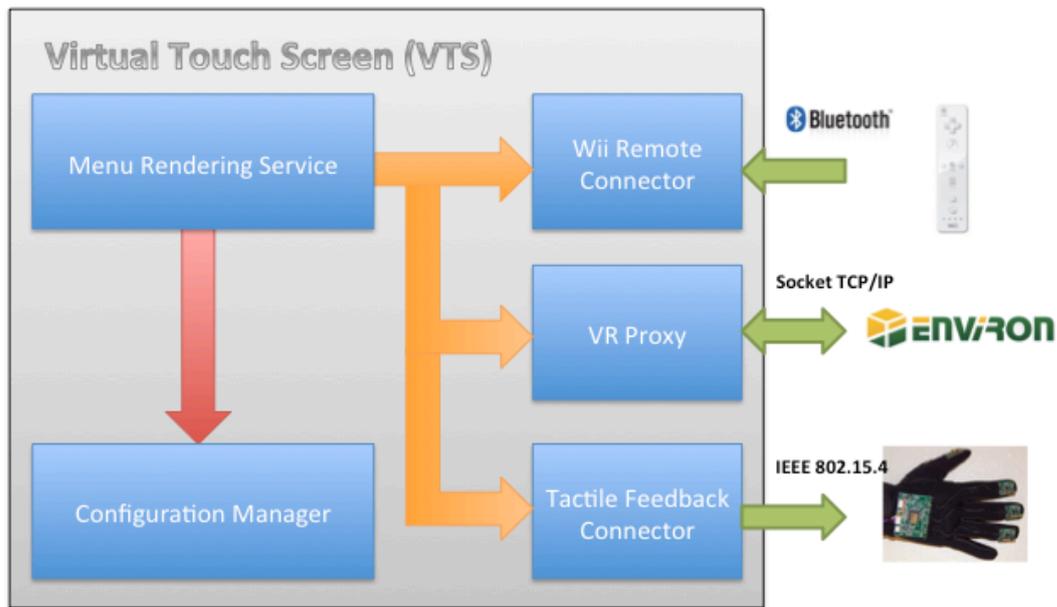


Figura 3.13: Diagrama de arquitetura do software

O diagrama de arquitetura da Figura 3.13 indica quais são esses módulos e mostra os elementos externos com os quais cada um deles interage, informando em cada caso a forma como é feita a integração. O componente VR Proxy interage com o programa Environ através de uma conexão socket TCP/IP. O conector de integração com o Wii Remote, por sua vez, realiza a comunicação através do protocolo Bluetooth, enquanto o módulo de controle do feedback tátil se comunica com a luva através do protocolo ZigBee (IEEE 802.15.4).

3.7.1. Menu Rendering Service

Este componente pode ser definido como o módulo central da aplicação. Ele é o primeiro a ser executado e é a partir dele que os módulos que interagem com os componentes externos do sistema são acionados. O serviço de renderização de menus é responsável também pela construção da interface gráfica de controle do software. A Figura 3.14 mostra o diagrama de classes desse componente do sistema.

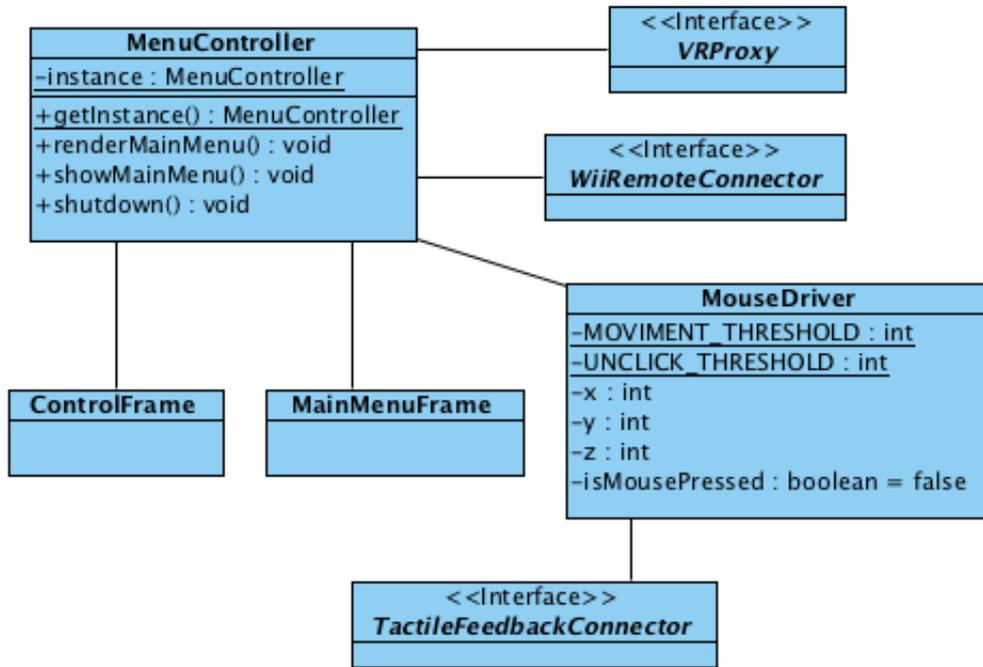


Figura 3.14: Diagrama de classes do Menu Rendering Service

A classe MenuController interage com os outros módulos do programa como VRProxy, WiiRemoteConnector e MouseDriver de forma a disparar ações de acordo com a forma de interação do usuário com os menus e controles da aplicação. A classe MouseDriver é responsável por gerenciar o movimento do cursor da v-Glove na tela e acionar o recurso de feedback tátil através da interface TactileFeedbackConnector. Existem ainda as classes ControlFrame e MainMenuFrame que representam painéis da interface gráfica com os quais o usuário interage.

3.7.2. Wii Remote Connector

Este módulo é o responsável pela comunicação do software com o Wii Remote para a captura das informações provenientes da câmera infravermelha. O software suporta a utilização de duas bibliotecas de conexão com o Wii Remote: Wiigee [Wiigee 2011] e WiiRemoteJ [WiiRemoteJ 2011]. As duas bibliotecas foram escritas em Java e utilizam o driver Bluetooth BlueCove versão 2.1.0 [BlueCove 2011]. Ambas foram testadas e possuem recursos similares, sendo que a WiiRemoteJ possui como vantagem o fato de permitir o ajuste de sensibilidade da câmera infravermelha do Wii Remote. Por esse motivo a

biblioteca WiiRemoteJ foi utilizada na versão final do software. A Figura 3.15 mostra o diagrama de classes do Wii Remote Connector.

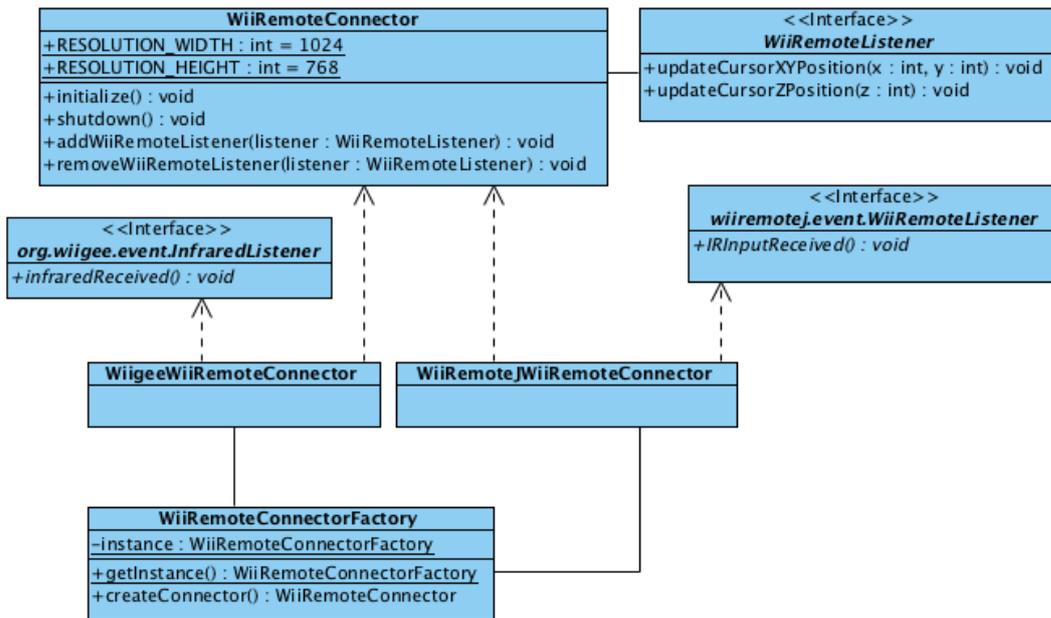


Figura 3.15: Diagrama de classes do Wii Remote Connector

A interface `WiiRemoteConnector` define as operações necessárias para uma implementação de conector para o Wii Remote. Como já foi mencionado, foram criadas duas implementações, uma utilizando a biblioteca `Wiigee` (e representada pela classe `WiigeeWiiRemoteConnector`) e outra utilizando a biblioteca `WiiRemoteJ` (representada pela classe `WiiRemoteJWiiRemoteConnector`). A classe `WiiRemoteConnectorFactory` é responsável pela criação de instâncias de conectores, enquanto as interfaces de *listener* são utilizadas para a recepção das coordenadas dos pontos de infravermelho a serem tratados pela aplicação.

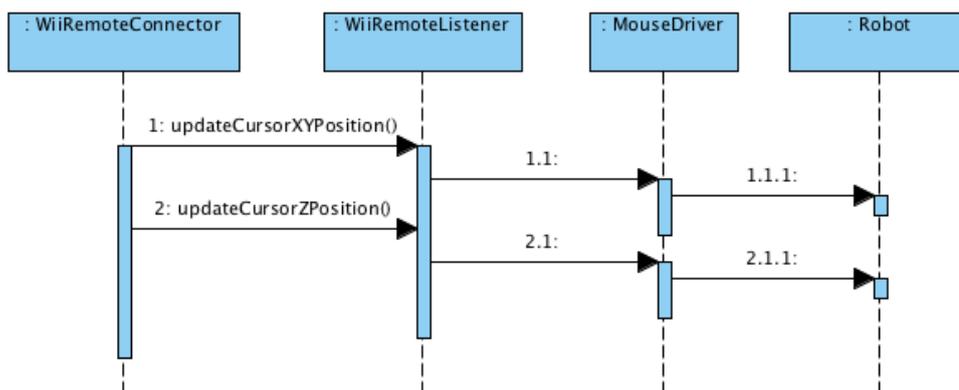


Figura 3.16: Diagrama de sequência da recepção de um evento de infravermelho

A Figura 3.16 mostra o diagrama de sequência referente ao fluxo de recepção de um evento de atualização de coordenadas desde o WiiRemoteConnector até seu tratamento pela aplicação. A classe MouseDriver implementa a interface WiiRemoteListener e recebe, através dela, eventos de atualização das coordenadas X e Y pelo método updateCursorXYPosition() e Z pelo método updateCursorZPosition(). Ela então aciona as chamadas de sistema operacional que controlam o posicionamento e o estado do cursor do mouse através da classe Robot, fornecida pela API padrão da linguagem Java.

3.7.3. VR Proxy

Este componente é responsável pela abstração da aplicação que está sendo utilizada na Cave em relação à luva. Ele define uma interface padronizada para a comunicação entre os módulos da aplicação e as aplicações imersivas de realidade virtual que serão utilizadas com a luva. A implementação de referência foi baseada no programa Environ que será alvo do estudo de caso apresentado no capítulo 4 (Estudo de caso: Environ). A Figura 3.17 mostra o diagrama de classes deste componente.

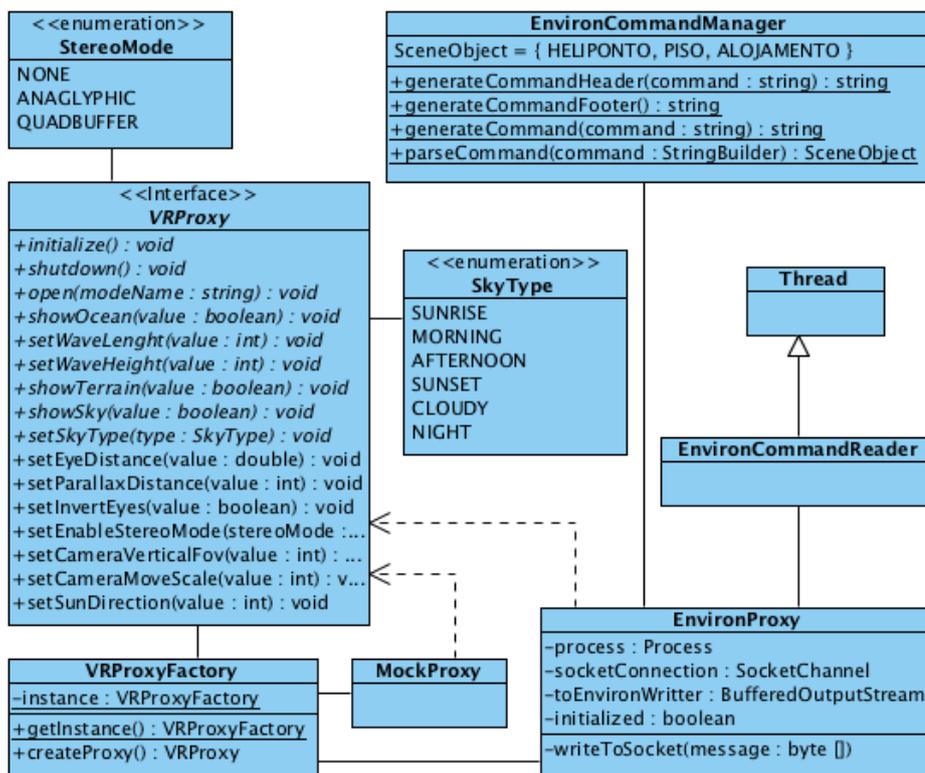


Figura 3.17: Diagrama de classes do VR Proxy

A interface VRProxy define as operações necessárias para uma implementação de proxy para aplicações de realidade virtual. A classe EnvironProxy representa a implementação de referência para o Environ e a classe MockProxy uma implementação para testes internos. A classe EnvironCommandManager é responsável pela conversão dos comandos de realidade virtual para funções no formato suportado pelo Environ. A classe EnvironCommandReader faz o tratamento dos comandos enviados pelo Environ em resposta aos comandos enviados pela aplicação. A classe VRProxyFactory é responsável pela criação de instâncias de proxy, enquanto o enumerador SkyType define os tipos de céu suportados pela aplicação e o enumerador StereoMode define os modos de operação estereoscópicos suportados.

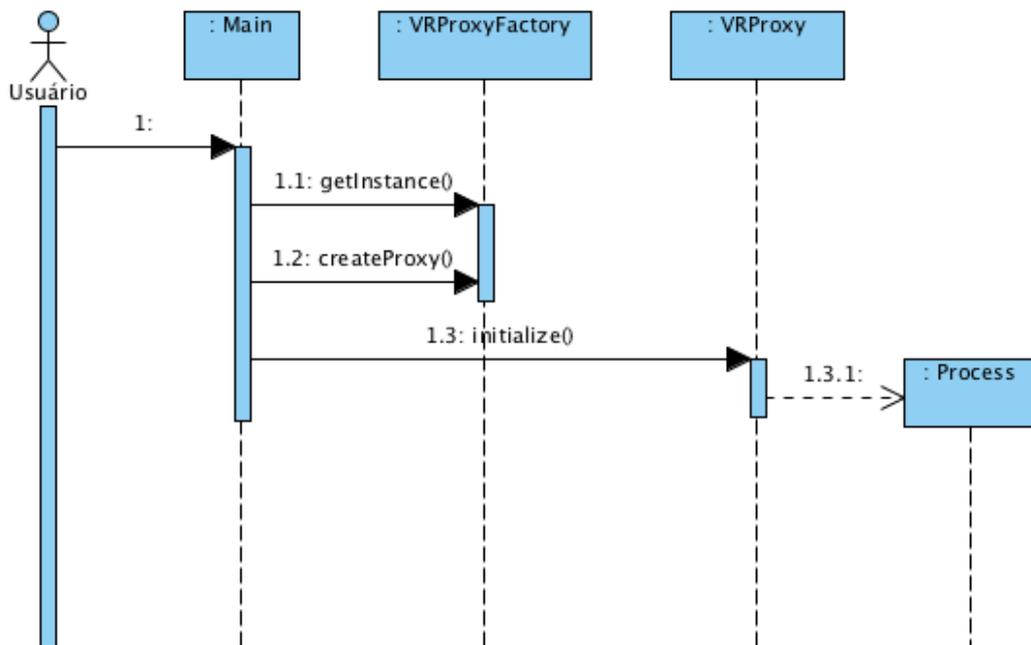


Figura 3.18: Diagrama de sequência do carregamento do Environ

A Figura 3.18 mostra o diagrama de sequência da operação de carregamento do Environ. Ao iniciar o software a classe MenuController é acionada e inicializa o componente VRProxy a partir da VRProxyFactory. Uma instância de EnvironProxy é devolvida e o proxy é iniciado, fazendo com que um novo processo seja criado no sistema operacional executando o arquivo binário do Environ.

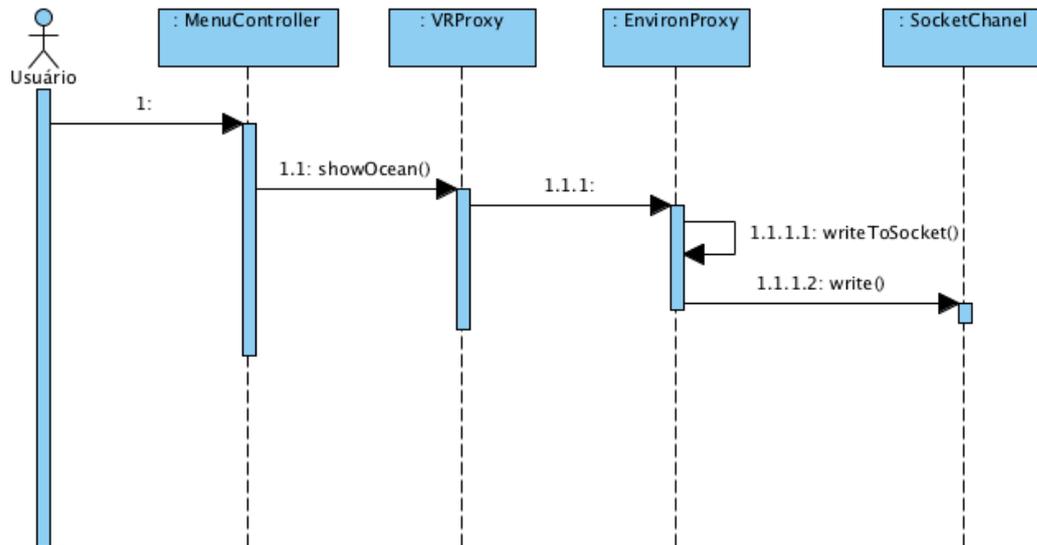


Figura 3.19: Diagrama de sequência do envio de um comando para o Environ

A Figura 3.19 mostra o diagrama de sequência da operação de envio de um comando da aplicação para o Environ. Ao clicar em uma opção no menu da aplicação é gerado um evento de interface gráfica de usuário. Esse evento é tratado pelo MenuController que, por sua vez, encaminha a execução do comando associado ao evento ao VRProxy. O EnvironProxy então envia um comando executável através da interface socket TCP/IP para o Environ, que finalmente executa o comando e responde informando se a execução foi concluída com sucesso ou não. Em caso de erro uma mensagem contendo o código e uma breve descrição do erro é retornada para o usuário.

3.7.4. Tactile Feedback Connector

Este componente é responsável pela geração do feedback tátil que provê ao usuário da luva a sensação de estar de fato tocando em uma superfície ao interagir com o sistema. Ele foi concebido de forma a suportar diversas estratégias de feedback tátil, tendo como implementação de referência a já mencionada com os componentes Lilypad Arduino (Figura 3.20).

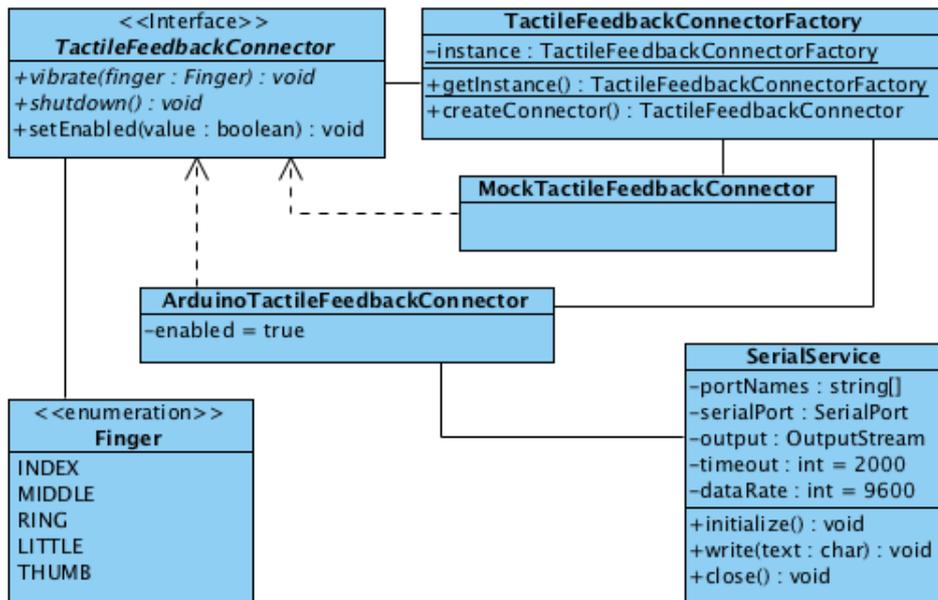


Figura 3.20: Diagrama de classes do Tactile Feedback Connector

A interface `TactileFeedbackConnector` define as operações que podem ser executadas por um conector de feedback tátil. A classe `ArduinoTactileFeedbackConnector` é a implementação de referência que suporta os componentes Lilypad Arduino, enquanto a classe `MockTactileFeedbackConnector` é utilizada para testes internos. A classe `TactileFeedbackConnectorFactory` é responsável pela criação de instâncias de conectores de feedback tátil. Já a classe `SerialService` representa um serviço para comunicação via porta serial que é utilizado pela classe `ArduinoTactileFeedbackConnector` na comunicação com a luva. A enumeração `Finger` representa os dedos da mão suportados pelo software.

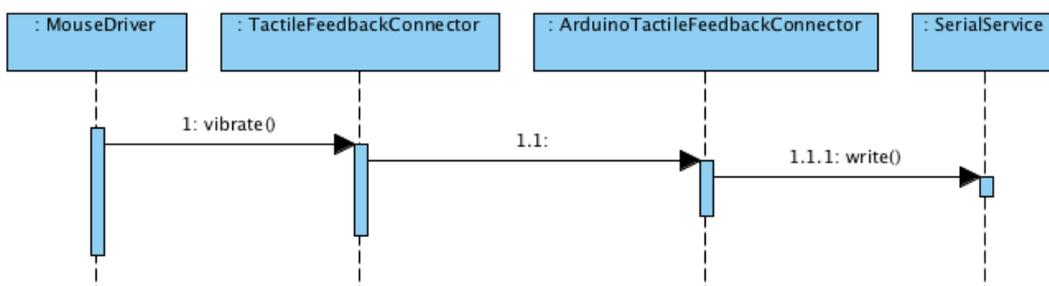


Figura 3.21: Diagrama de sequência da operação de acionar o feedback tátil

A Figura 3.21 mostra o diagrama de sequência da operação de acionar o feedback tátil na v-Glove. Ao identificar que o usuário se aproximou da interface de toque virtual, a classe `MouseDriver` dispara um evento de vibração através do

TactileFeedbackConnector, que por sua vez envia o comando de vibração para o SerialService, indicando qual dedo deve ser acionado. O SerialService então envia essa mensagem via conexão serial para o XBee presente na estação de controle, que repassa o comando para a v-Glove.

3.7.5. Configuration Manager

Este módulo é responsável por gerenciar os parâmetros de configuração necessários para a execução do software. O Configuration Manager pode ser acessado por qualquer classe do programa que precise fazer a leitura de um parâmetro de configuração.

4 Estudo de Caso: Environ

De forma a avaliar a usabilidade da v-Glove elaboramos um estudo de caso exploratório baseado no programa Environ [Raposo et al 2009]. O Environ é uma ferramenta de visualização 3D de modelos CAD voltada para atividades de engenharia na indústria de petróleo e gás natural e foi desenvolvido pelo Grupo de Tecnologia em Computação Gráfica (Tecgraf) da PUC-Rio. O Environ foi concebido para ser utilizado tanto como aplicação desktop para um único usuário (Figura 4.1) como em modo imersivo para ambientes de realidade virtual 3D (conforme pode ser visto na Figura 3.12, que mostra a Cave da Petrobras).

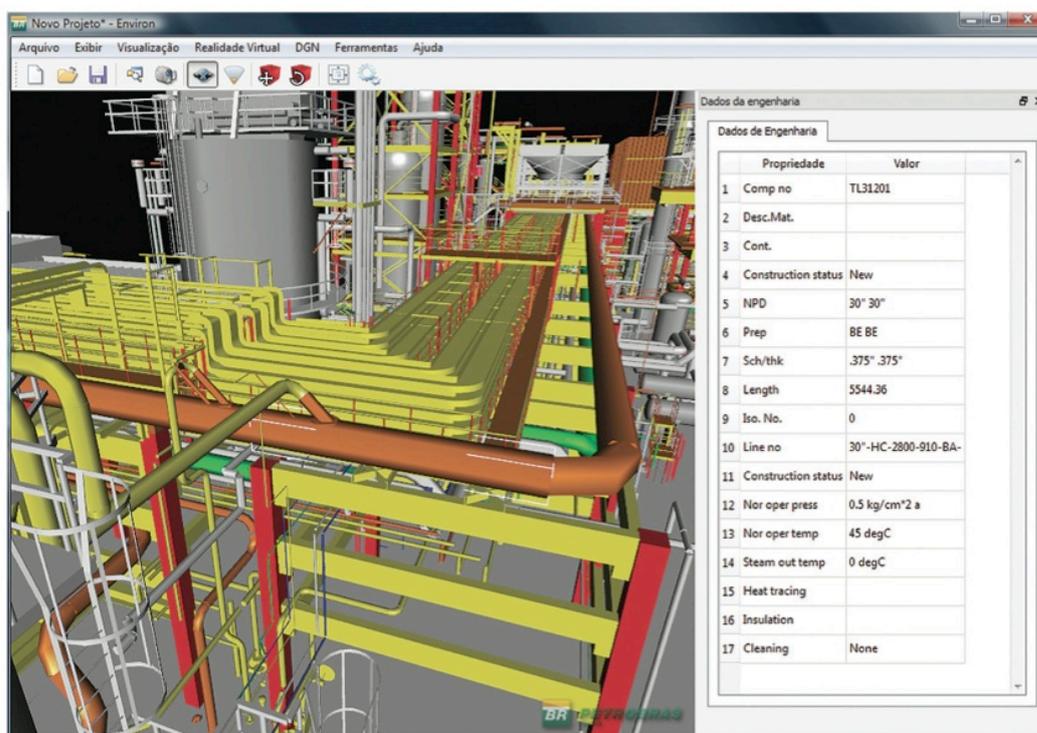


Figura 4.1: Tela principal do Environ em modo desktop [Raposo et al 2009]

E foi justamente por já prover um suporte ao modo imersivo, juntamente com o fato de já ser uma aplicação estável e com um número elevado de usuários que a usam diariamente, que o Environ foi escolhido para o estudo de caso deste trabalho. Espera-se com a utilização de uma aplicação já

consolidada a obtenção de resultados significativos que possam trazer contribuições reais a esse grupo de usuários no futuro.

Por ter sido uma aplicação concebida inicialmente para uso em desktops, o Environ ainda não foi totalmente adaptado para ambientes imersivos. Apesar de já contemplar suporte a visualização estereoscópica que permite sua utilização em ambientes de realidade virtual, a interação do usuário com o software é feita através de menus e controles baseados no padrão WIMP, como pode ser visto na imagem da Figura 4.2.

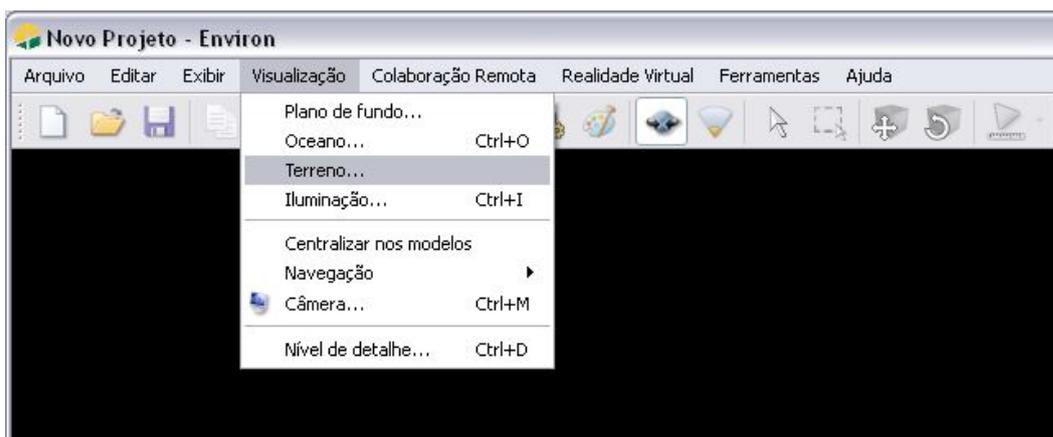


Figura 4.2: Interface de menus do Environ no padrão WIMP [Raposo et al 2009]

Por isso foi necessária a adaptação do sistema através da criação de novos menus e *widgets* com um formato mais adequado para a interação com a v-Glove. Essa adaptação ocorreu apenas nos componentes relevantes para os testes realizados durante este trabalho, uma vez que não havia tempo suficiente para um reprojeto mais abrangente de toda a interface do Environ. Mais detalhes sobre as adaptações feitas serão discutidos na seção 4.1.

Limitaremos nessa prova de conceito o escopo do Environ às seguintes tarefas:

- Navegação em modelos CAD
- Seleção de objetos em modelos CAD
- Navegação e seleção de opções em menus da aplicação
- Configurações e ajustes do programa

4.1. Adaptações no Environ

Com a finalidade de adaptar a utilização do Environ para a v-Glove foi criada uma nova interface de menus e *widgets* exclusivos para os testes. Além disso, foi feita uma adaptação no formato do cursor do mouse exibido durante a interação com os menus da aplicação.

4.1.1. Interface de menus e *widgets*

Foi feito inicialmente um levantamento dos componentes de interface gráfica utilizados no Environ. Esta listagem dos componentes foi então classificada de acordo com sua utilização na aplicação, sendo assim priorizados os componentes mais utilizados (Tabela 4.1). A partir destes componentes foram selecionadas as tarefas que fariam parte dos testes com os usuários da v-Glove e relacionados os componentes que deveriam sofrer algum tipo de adaptação.

Prioridade	Componente	Tipo
1	Botão	Simple
2	Checkbox	Simple
3	Slider	Simple
4	Combobox	Simple
5	Textbox	Simple
6	Menu	Composto
7	Integer Textbox	Simple
8	Toolbar	Composto
9	File Open	Composto
10	File Save	Composto
11	Radio Button	Simple
12	Color selection box	Composto

Tabela 4.1: Classificação dos componentes de interface gráfica do Environ

Com base nessa análise foram selecionados os cinco componentes com maior prioridade para terem suas interfaces adaptadas à utilização com a v-

Glove. A Figura 4.3 mostra como ficaram os componentes após a adaptação: em (a) pode ser observada a *checkbox*, em (b) o *slider*, (c) mostra a *combobox* e (d) a *textbox* utilizada em conjunto com o botão para prover a funcionalidade de entrada numérica.

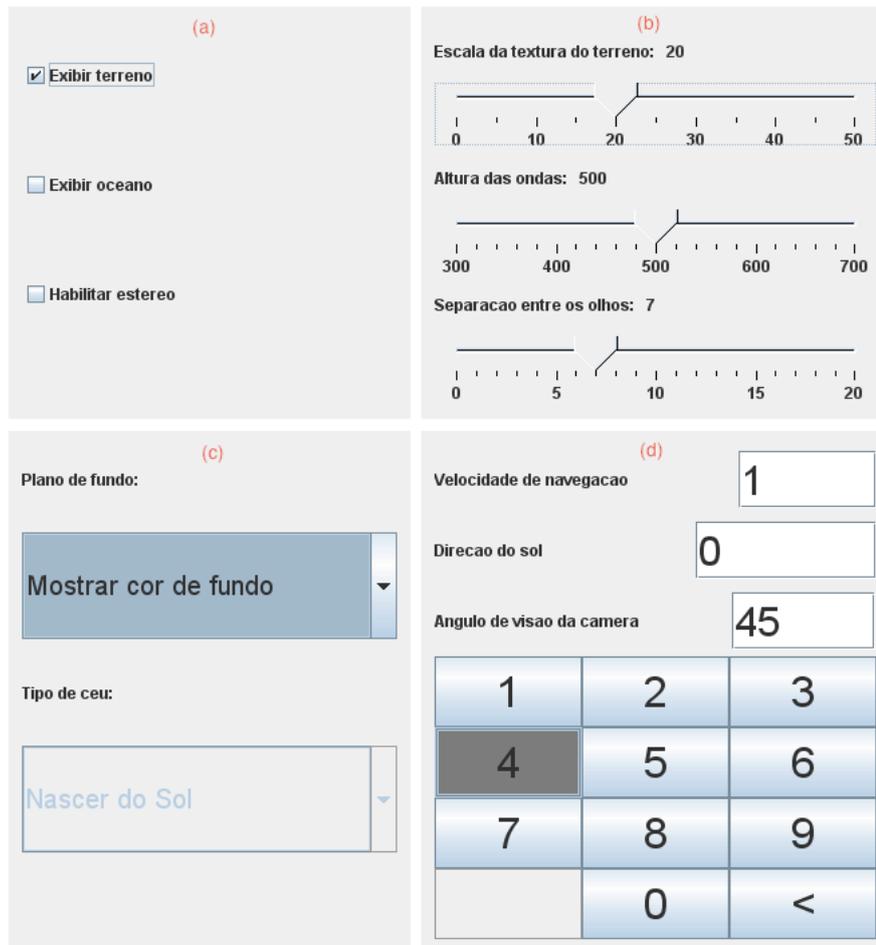


Figura 4.3: *Widgets* criados neste trabalho

4.1.2. Cursor do mouse

Como complemento ao recurso de feedback tátil, foi realizada uma alteração no comportamento do cursor do mouse quando o usuário interage com o sistema. Dependendo da localização da v-Glove em relação à interface de toque, a cor do cursor na tela é alterada. A Figura 4.4 mostra as cores possíveis do cursor.

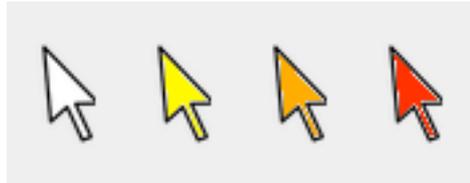


Figura 4.4: Variação das cores do cursor

O cursor permanece branco enquanto a luva encontra-se fora da área de cobertura da câmera responsável pelo eixo Z ou quando estiver a uma distância de mais de 400 pixels da área de toque. A cor muda para amarelo quando a distância entre a luva e a superfície de toque for maior que 200 e menor que 400 pixels, e muda para laranja quando essa distância for inferior a 200 pixels. Finalmente o cursor ficará vermelho quando for atingida ou ultrapassada a área de toque.

4.2. Experimento

Realizamos testes quantitativos e qualitativos com usuários divididos em grupos de acordo com seu perfil de utilização de aplicações 3D. No primeiro grupo foram selecionados usuários que fazem parte da equipe de desenvolvimento do Environ. No segundo grupo, usuários que tenham experiência no desenvolvimento de aplicações 3D, mas que não façam parte da equipe do Environ e nunca utilizaram este software. O terceiro grupo foi composto por usuários sem experiência no desenvolvimento ou uso de aplicações 3D. O objetivo dessa divisão em grupos foi permitir uma comparação no desempenho dos grupos dependendo do seu grau de familiaridade com o desenvolvimento de software 3D e especialmente do Environ.

4.2.1. Formato do Estudo

O experimento foi composto por três sequências de tarefas a serem realizadas por cada um dos participantes. Em cada uma das sequências foi utilizado um dispositivo de interação diferente, dentre os seguintes: v-Glove sem o recurso de feedback tátil, v-Glove com o recurso de feedback tátil e mouse com acelerômetro [Air Mouse 2011] (Figura 4.5). O objetivo de testes com diferentes dispositivos foi comparar inicialmente a luva com e sem o recurso de

feedback tátil e posteriormente compará-la a um dispositivo utilizado atualmente por usuários de aplicações imersivas de realidade virtual.



Figura 4.5: Mouse 3D com acelerômetro utilizado nos testes [Air Mouse 2011]

Em cada uma das sequências foram executadas 5 tarefas de interação. Inicialmente o usuário tinha um tempo de dois minutos para se familiarizar com o dispositivo em tarefas de seleção de botões e navegação no Environ. Essas tarefas iniciais eram diferentes da tarefa a ser executada nos testes para evitar efeitos de aprendizado antes da efetiva realização da sequência de tarefas. Ao final do tempo livre, o usuário deveria clicar em um botão que dava início aos testes. A partir desse momento iniciava-se a tomada de tempo e a filmagem das ações do participante.

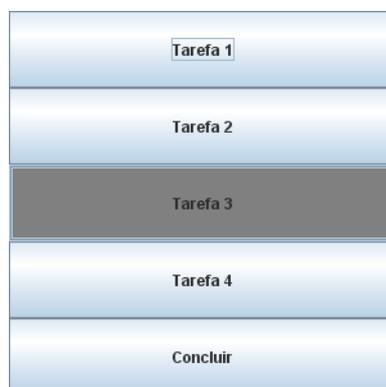


Figura 4.6: Menu principal da aplicação de testes

Na primeira tarefa o usuário era solicitado a navegar por uma cena do Environ e procurar por um determinado objeto, tendo que selecioná-lo. Ao selecionar o objeto correto, o sistema exibia um menu com as tarefas seguintes

(Figura 4.6), que deveriam ser executadas clicando-se no botão apropriado nesse menu de opções.

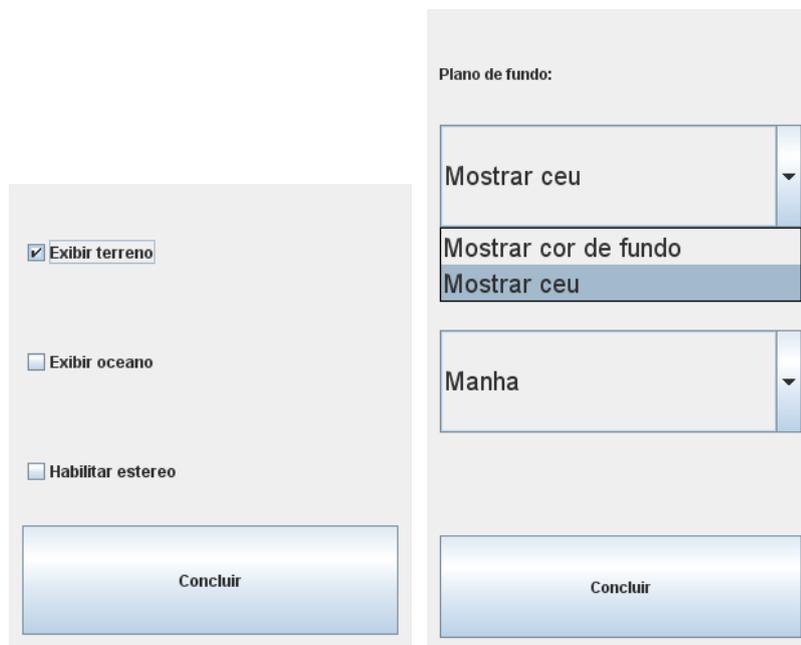


Figura 4.7: Opções da segunda e terceira tarefas

Na segunda tarefa o usuário deveria selecionar um dos componentes do tipo *checkbox* apresentados na tela e, em seguida clicar em um botão para concluir a tarefa (Figura 4.7 (a)). A terceira tarefa envolvia a seleção de opções em dois componentes do tipo *combobox*, seguidas da seleção do botão de concluir (Figura 4.7 (b)). Na quarta tarefa o objetivo era selecionar o valor solicitado em um controle do tipo *slider*, em seguida clicando no botão de concluir (Figura 4.8 (a)). Na quinta e última tarefa o usuário deveria corrigir o valor digitado em uma *textbox*, digitando a seguir um novo valor numérico e selecionando o botão de concluir (Figura 4.8 (b)). Ao final das cinco tarefas o participante deveria selecionar um botão para concluir aquele conjunto de tarefas.

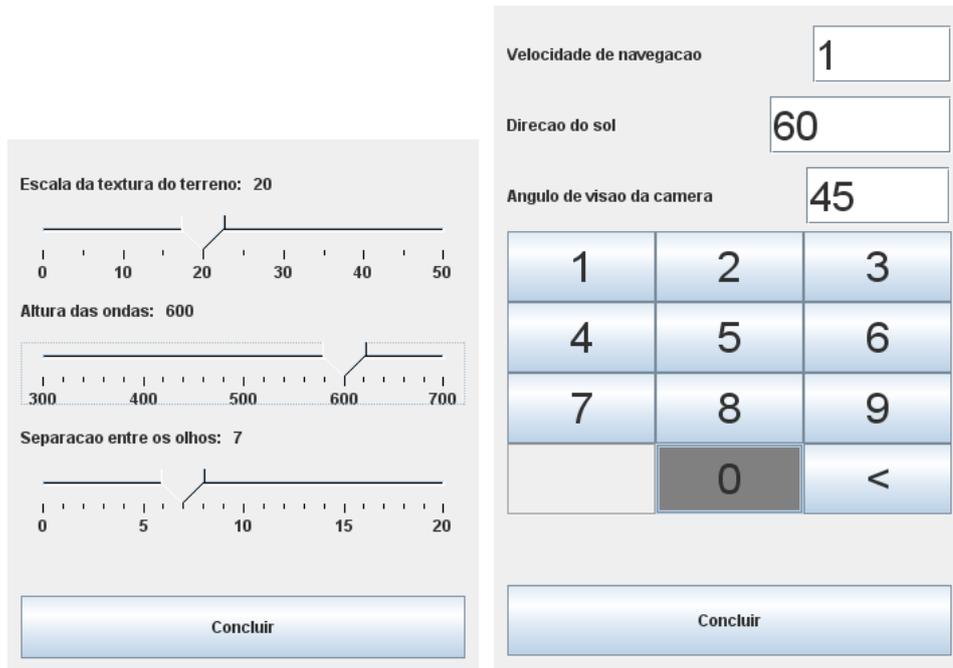


Figura 4.8: Opções da quarta e quinta tarefas

Terminada a primeira sequência de tarefas o usuário tinha seu dispositivo trocado e a sequência era repetida. A única diferença entre duas sequências de tarefas eram as opções que deveriam ser selecionadas. Isso foi feito com a finalidade de evitar cansar o usuário tendo que executar exatamente a mesma sequência de tarefas por três vezes.

Para reduzir os efeitos de aprendizado entre uma tarefa e a seguinte, a ordem de utilização dos dispositivos não se repetiu entre usuários de um mesmo grupo. Considerando-se que havia três dispositivos e seis usuários em cada grupo, foi possível determinar uma ordem de utilização exclusiva para cada usuário dentro de seu grupo, já que a permutação de um universo de três elementos em grupos de três resulta em um total de 6 combinações possíveis de sequências.

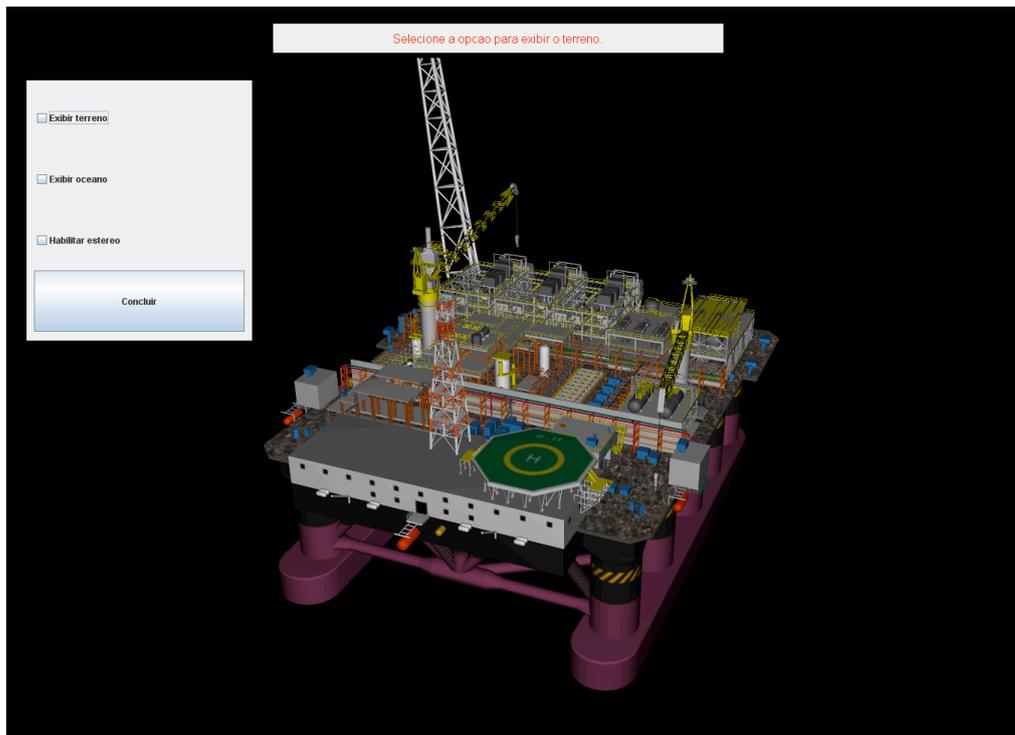


Figura 4.9: Instruções para o usuário durante a realização das tarefas

Para cada uma das tarefas o próprio sistema exibia as instruções do que deveria ser feito pelo usuário (Figura 4.9), evitando que este fosse obrigado a ler as instruções a partir de uma folha de papel ou mesmo que dependesse do avaliador para receber as instruções. Assim o avaliador pode dedicar-se integralmente à observação dos testes.

Fez parte do experimento o preenchimento de dois questionários, um antes e outro ao final dos testes, além de uma ficha de acompanhamento que ficava de posse do avaliador e onde deveriam ser anotados todos os problemas (*issues*) ocorridos durante os testes. O questionário inicial teve como objetivo verificar o perfil dos usuários e foi baseado em exemplos disponíveis em Mitchell (2007). O questionário pós-testes tinha como finalidade observar a opinião dos usuários em relação aos dispositivos utilizados. Para facilitar a análise estatística seguiu-se nesse questionário o padrão de respostas em uma escala de Likert com 7 pontos [Tullis e Albert 2008]. Todos os testes foram gravados em vídeo e as entrevistas de preenchimento dos questionários após os testes foram gravadas em áudio para permitir uma análise posterior mais detalhada.

Todos os participantes preencheram e assinaram um termo de consentimento livre e esclarecido antes do início dos testes. Este termo

explicava a finalidade dos testes, o papel do participante no processo e fornecia a opção de auto-exclusão em qualquer etapa do processo. Um exemplo do termo de consentimento pode ser encontrado no Apêndice C.

4.2.2. Grupos de Usuários

Um total de dezoito pessoas participou dos testes. Os usuários foram divididos em três grupos de mesmo tamanho (6 participantes) de acordo com sua experiência prévia no uso e no desenvolvimento de aplicações 3D. O primeiro grupo foi chamado de “**Grupo E**” e foi composto por analistas de sistemas que trabalham há pelo menos 6 meses no desenvolvimento do software Environ. Do segundo grupo, denominado “**Grupo O**”, faziam parte analistas de sistemas com experiência no desenvolvimento de software para computação gráfica, mas que não tenham participado do desenvolvimento do Environ. O terceiro grupo, denominado “**Grupo N**”, foi composto por indivíduos sem experiência no uso de software 3D.

4.2.3. Perfil dos Participantes

A faixa etária dos participantes selecionados para os testes variou de 21 a 60 anos, resultando em uma média de idade de 29 anos (a Tabela 4.2 mostra a média de idade dos participantes por grupos). Dezesesseis indivíduos eram do sexo masculino (89%) e dois do sexo feminino (11%). Todos os participantes eram destros e tinham por hábito utilizar mouse com a mão direita. Este era um ponto importante a ser observado, uma vez que o protótipo da v-Glove confeccionado para os testes funciona apenas na mão direita, de forma que os usuários canhotos poderiam ter seus resultados comprometidos pela utilização do dispositivo na mão não-dominante.

	<i>Média de idade</i>	<i>Menor idade</i>	<i>Maior idade</i>
Geral	29	21	60
Grupo E	25	22	28
Grupo O	31	26	37
Grupo N	31	21	60

Tabela 4.2: Média de idade dos participantes

Os participantes tinham em sua totalidade uma média semanal de mais de quinze horas de utilização de computador, sendo que 94% deles indicaram ter algum tipo de experiência com aplicações 3D. Exemplos de aplicações mencionadas foram jogos, ferramentas de modelagem 3D e visualizadores CAD. Todos os usuários do Grupo O indicaram conhecer o Environ e já ter tido algum tipo de contato com a aplicação, mas foi um contato superficial, não fazendo parte do grupo de desenvolvedores nem de usuários do sistema.

4.2.4. Equipamentos

Para os testes foi utilizada uma sala de visualização disponível em um dos prédios do Tecgraf (Figura 4.10). Na sala encontram-se dois projetores ProjectionDesign evo22sx+ [Projection Design 2011] configurados em modo estereoscópico passivo através da utilização de um filtro de polarização circular. A imagem é projetada em uma tela anti-depolarizadora (composta por um material que retém a polarização) de 120 polegadas. O microcomputador utilizado como estação de controle da aplicação possui uma placa gráfica NVIDIA Quadro FX 4600 [Nvidia Quadro 2011] com duas saídas de vídeo, processador Intel core i7 com 3.20 GHz e 12 Gigabytes de memória RAM.

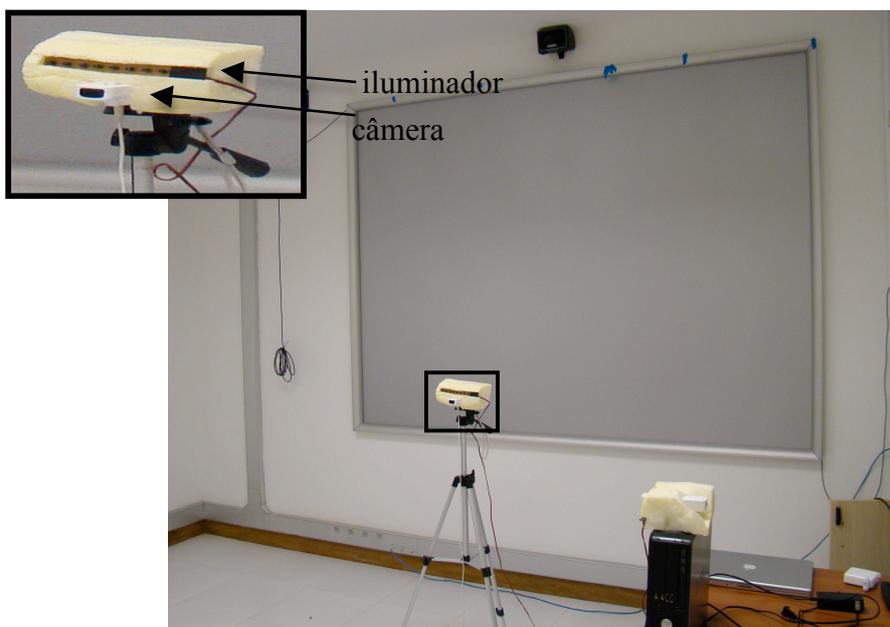


Figura 4.10: Sala utilizada nos testes com os usuários

Nesta sala foram instaladas as duas câmeras utilizadas no rastreamento da v-Glove e foi definida uma área no espaço a ser mapeada pelas câmeras como o plano de toque para interação com a luva.

4.2.5. Teste Piloto

Antes do início dos testes com os usuários foram efetuados dois testes piloto com o objetivo de validar o roteiro do experimento e corrigir eventuais problemas. O teste piloto reproduziu o roteiro completo de um teste normal, inclusive com o preenchimento dos questionários e a gravação em áudio e vídeo. O perfil dos participantes utilizados no teste piloto foi compatível com o dos usuários do Grupo N, uma vez que se espera desse grupo um maior grau de dificuldade na realização das tarefas.

Essa etapa foi muito importante, pois nela foi possível observar diversos pontos de melhoria no modelo inicial de experimento que havia sido desenhado. Durante o piloto verificou-se a necessidade de correção de falhas na aplicação construída para os testes, aperfeiçoamento das instruções apresentadas para os participantes, aperfeiçoamentos no formato dos *widgets* e até mesmo melhorias no processo de coleta dos dados.

4.3. Resultados dos Testes

Como já mencionado, os dados resultantes dos testes foram coletados através do preenchimento de questionários e formulários de acompanhamento, gravação em vídeo da realização das tarefas e gravação do áudio das entrevistas. Todas as informações coletadas foram concentradas em uma planilha eletrônica para facilitar o processo de análise e comparação dos dados. Os resultados foram então separados em quatro categorias de acordo com as métricas de usabilidade a serem observadas.

Em Tullis e Albert (2008) são apresentados dez cenários de estudo de usabilidade e os mesmos são relacionados com as métricas de usabilidade que se aplicam para cada tipo de estudo. De acordo com a classificação dos autores, nosso trabalho se encaixa no cenário de “comparação de designs alternativos” e a ele se aplicam cinco métricas: sucesso da tarefa (*task success*), duração da

tarefa (*task time*), métricas de situações de usabilidade (*issue-based metrics*), métricas auto-relatadas (*self-reported metrics*) e métricas combinadas e comparativas (*combined and comparative metrics*). As quatro primeiras métricas de usabilidade serão discutidas em mais detalhes nas seções a seguir. A quinta métrica será apresentada como parte da seção 4.4 (Discussões).

4.3.1. Sucesso da Tarefa (*task success*)

A primeira análise que se fez nos dados capturados durante os testes foi em relação ao sucesso ou falha na realização das tarefas solicitadas. A maioria dos usuários não teve dificuldades em completar 100% das tarefas. A Figura 4.11 mostra o percentual de sucesso para cada uma das tarefas realizadas por dispositivo.

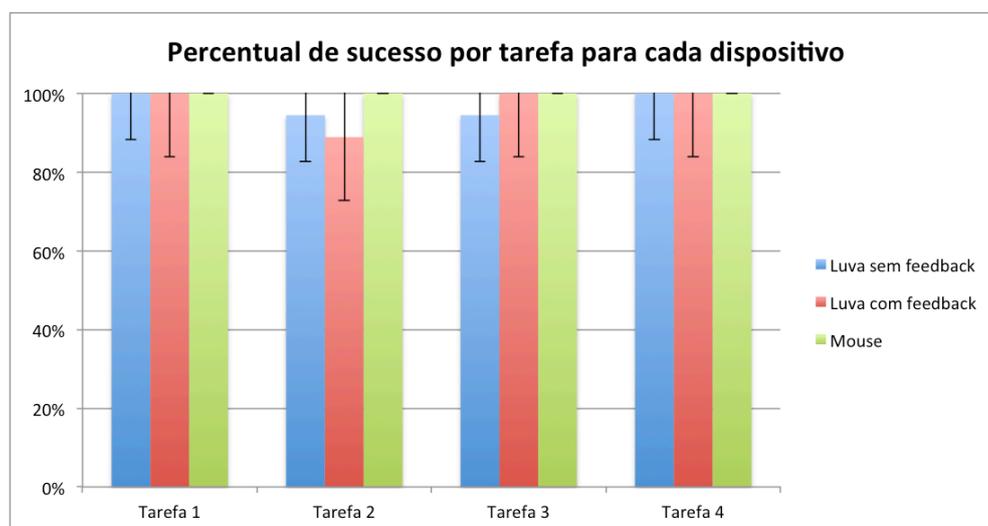


Figura 4.11: Percentual de sucesso por tarefa para cada dispositivo

Pode-se observar que 100% dos usuários completaram as tarefas realizadas com o mouse, enquanto na luva com feedback tátil o percentual de sucesso da tarefa 2 foi de 89%. Na luva sem feedback tátil verificou-se um percentual de sucesso de 94% nas tarefas 2 e 3.

4.3.2. Duração da tarefa (*task time*)

A seguir realizou-se uma análise do desempenho dos usuários de cada um dos grupos na realização das tarefas. Para isso foram usadas as medidas de duração em segundos das sequências de tarefas. A Figura 4.12 mostra o gráfico

com a média de tempo de cada um dos grupos de usuários por dispositivo (os intervalos de confiança utilizados foram de 90%).

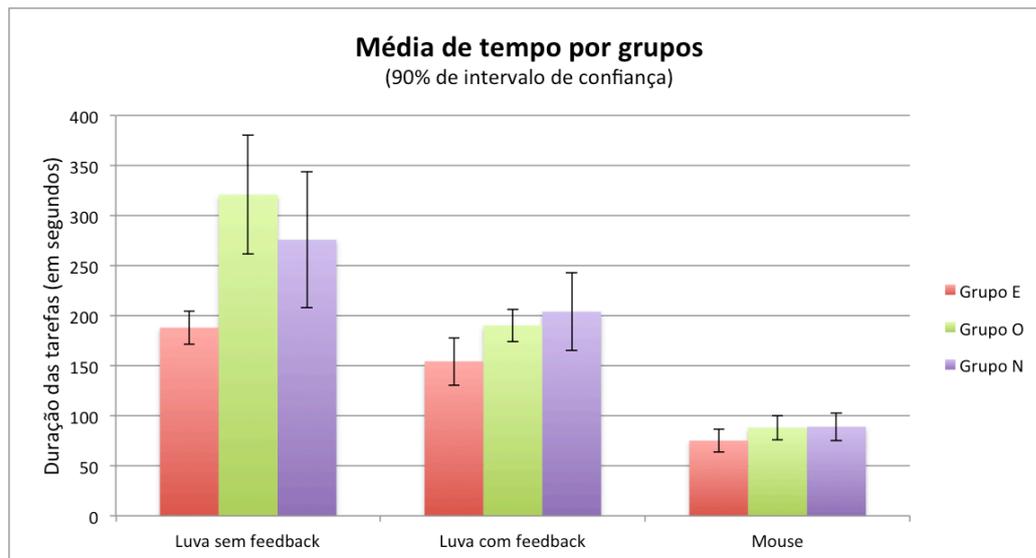


Figura 4.12: Média de tempo por grupo

Visualmente podem-se observar duas situações no gráfico. Primeiramente que a luva sem feedback tátil possui a pior média de desempenho, seguida pela luva com feedback e pelo mouse. Além disso, pode-se observar que o tempo de realização das tarefas parece diminuir de acordo com o nível de experiência do usuário com aplicações 3D. Mais à frente discutiremos se estas diferenças possuem de fato alguma relevância estatística.

4.3.3.

Métricas de situações de usabilidade (*issue-based metrics*)

A terceira análise efetuada nos dados dos testes foi sobre as métricas de problemas de usabilidade. Durante a realização dos testes os avaliadores coletaram informações de situações ocorridas durante a execução das tarefas. Posteriormente foi feita uma tabulação classificando os problemas encontrados por frequência, prioridade, *rating* e categoria.

4.3.3.1.

Frequência

Para o cálculo de frequência foi feita a contagem do número de usuários onde foi observada uma determinada situação.

4.3.3.2. Prioridade

A priorização das situações foi realizada de acordo com o modelo de três níveis proposto por Tullis e Albert (2008). A prioridade alta indica situações que levem à não conclusão de uma tarefa. Situações de prioridade média são aquelas que contribuem para o insucesso na realização de uma tarefa, mas que não são a causa direta desse insucesso. E a prioridade baixa indica situações que causam algum tipo de incômodo ou frustração no participante, mas que não estão relacionados a falhas na realização de alguma tarefa.

4.3.3.3. Rating

Para uma melhor classificação das situações Tullis e Albert (2008) indicam a adoção de um fator de classificação composto ao invés de considerar-se apenas a prioridade ou a frequência. Pode-se afirmar que uma situação com prioridade alta que ocorre apenas com um participante é mais importante do que uma de prioridade média, mas que ocorre com metade dos participantes?

Com a finalidade de se levar em consideração na classificação essas duas grandezas de forma equivalente é que o *rating* foi introduzido como grandeza. O cálculo do *rating* é feito somando-se o valor da frequência e da prioridade para uma determinada situação.

4.3.3.4. Categoria

Como forma de organizar as situações verificadas foram utilizadas quatro categorias: processo de testes (T), interface gráfica (G), operação de dispositivo (D) e infraestrutura (I).

Problema	Frequência	Prioridade	Rating	Categoria
Falta de precisão por deslocamento da mão em Z	8	3	11	D
Falta de precisão do <i>slider</i> na luva com feedback	6	3	9	G
Cansaço no braço ou na mão	5	3	8	D
Dificuldade em diferenciar cursor laranja do vermelho	4	3	7	G
Falta de precisão do <i>slider</i> na luva sem feedback	4	3	7	G

Dificuldades para selecionar a combo box em dois cliques	4	3	7	G
Dificuldades com a precisão dos botões na luva sem feedback	5	2	7	D
Dificuldades de compreensão das instruções	3	3	6	T
Selecionou opção errada em alguma tarefa	2	3	5	D
Problema de travamento do software (teve que ser reiniciado)	2	3	5	I
Dificuldades com a postura para utilizar o sistema (não conseguiu achar um posicionamento ideal)	2	3	5	D
Não executou alguma tarefa por clicar acidentalmente em 'Concluir' antes de executá-la	2	3	5	D
Dificuldades com a precisão da <i>checkbox</i> na luva sem feedback	3	2	5	D
Clicou direto no valor final do <i>slider</i> , não utilizou o conceito de arrastar	4	1	5	G
Usou a técnica do balanço do corpo para melhorar a precisão na luva sem feedback	2	2	4	D
Dificuldades com a precisão do <i>slider</i> no mouse	1	3	4	G
Dificuldades ao clicar nos botões na luva com feedback	1	3	4	D
Teve dificuldades inicialmente com a navegação do mouse por ter que manter o botão pressionado ao movimentar	2	1	3	D
Usou o scroll (zoom) do mouse mesmo sem ter sido pedido	2	1	3	T
Usou a mão esquerda para melhorar a precisão	2	1	3	D
Tentou orientar o movimento da mão pela tela e não pela câmera	1	2	3	D
Executou as tarefas fora da ordem porque teve dificuldades em clicar nos botões do menu	1	1	2	D
Tentou levar o cursor da luva para o canto para "resetar" como se faz com o mouse	1	1	2	D

Tabela 4.3: Situações de usabilidade observadas

A Tabela 4.4 mostra a listagem de situações observadas durante os testes e o valor observado para cada uma das grandezas mencionadas. Partindo-se dos dados da tabela foram selecionadas as oito situações com o maior *rating* para uma análise mais detalhada. Essas situações já permitem que se tenha um indicativo de pontos a serem melhorados na v-Glove.

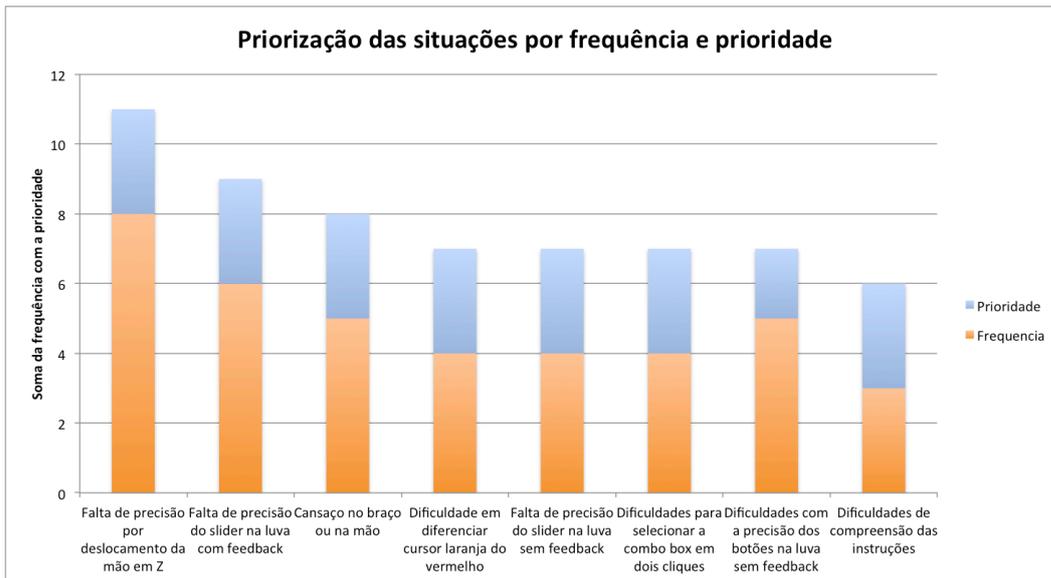


Figura 4.13: Classificação das situações de usabilidade por frequência e prioridade

O gráfico da Figura 4.13 mostra o *rating* das oito situações selecionadas para análise. É possível observar para cada uma delas a contribuição referente às grandezas de prioridade (em azul) e frequência (em laranja) na formação do *rating*. A seguir descreveremos cada uma dessas oito situações individualmente:

- Falta de precisão por deslocamento da mão em Z** – esta foi a situação que ocorreu com maior frequência durante os testes. Ao deslocar a mão de sua posição original até a superfície virtual de toque é muito difícil para o usuário manter a posição de forma a não variar a localização do cursor nos eixos X e Y. Isso faz com que ao atingir a área de clique a posição do cursor não seja mais a mesma de quando o movimento foi iniciado, fazendo com que o clique ocorra em uma região que não seja a desejada pelo usuário. Esta situação pode acarretar dois efeitos indesejados. O primeiro é a falha no clique e o insucesso na seleção da opção desejada ou no clique de um botão, o que faz com que a operação tenha que ser repetida até que se obtenha sucesso. O segundo efeito é a possibilidade do clique ocorrer em uma área onde exista outro componente de interface, ocasionando a seleção de uma opção incorreta. A consequência deste segundo efeito pode ser a não realização de uma tarefa ou a necessidade de repeti-la, caso isso seja possível.

- **Falta de precisão do *slider* na luva com feedback e falta de precisão do *slider* na luva sem feedback** – o *slider* foi o componente de interface gráfica mais afetado pelo problema da falta de precisão relatado no item anterior. Uma vez que o usuário selecionava o indicador do *slider* ele tinha dificuldades em colocá-lo na posição referente ao valor desejado na régua. Isso ocorria porque, dependendo do tamanho da escala, a distância entre dois marcadores na régua do *slider* era pequena a ponto de fazer com que uma oscilação mínima na mão do usuário impedisse seu posicionamento na localização adequada. Esse problema foi observado tanto na luva com o recurso de feedback tátil quanto na luva sem este recurso. Por terem nos dois casos a mesma causa optou-se por analisá-los em conjunto.
- **Cansaço no braço ou na mão** – o posicionamento da mão com a v-Glove para a interação exige que se mantenha o braço esticado e elevado um pouco à frente do corpo. Nos casos onde se realizou operações com a luva por um período de tempo mais elevado, alguns usuários relataram cansaço no braço e dores similares às de tendinite no pulso ou no antebraço. Relatos referentes a cansaço e fadiga já eram esperados levando-se em consideração resultados já verificados em trabalhos similares. Entretanto, entende-se que houve um agravamento nessas situações causado pelo problema da falta de precisão no clique com a v-Glove, que acarretava um número maior do que o esperado de repetições na execução das tarefas resultando em uma maior duração das tarefas e ocorrências de cansaço em alguns usuários.
- **Dificuldade em diferenciar cursor laranja do vermelho** – O feedback visual criado para auxiliar na percepção da profundidade da mão no eixo Z tinha como últimas cores o laranja (que indicava que a v-Glove estava próxima à área de clique) e o vermelho (indicando que a área de clique havia sido atingida). Alguns usuários tiveram dificuldades em perceber a diferença entre as tonalidades de laranja e vermelho utilizadas, causando uma confusão na percepção do clique especialmente no caso da luva sem o feedback tátil. Na v-Glove com feedback tátil esse problema ocorreu com menor intensidade, pois mesmo se não conseguisse diferenciar as cores do cursor o usuário tinha o auxílio do recurso de vibração para saber se havia atingido a área de clique. O problema da

pouca diferença na tonalidade também foi agravado por limitações no espectro de cores dos projetores utilizados, situação verificada ao comparar-se a imagem do cursor no projetor com a mesma imagem gerada em um monitor LCD convencional.

- **Dificuldades para selecionar a combo box em dois cliques** – alguns usuários tiveram dificuldades na seleção do componente *combobox*, tanto na luva com o recurso de feedback tátil quando na versão sem este recurso. Para selecionar uma opção na *combobox* o usuário deveria clicar inicialmente sobre o componente, fazendo com que fossem listadas as opções disponíveis, e depois clicar uma segunda vez na opção desejada, concluindo a operação. Esta operação em duas etapas ocasionou dificuldades porque ao clicar a primeira vez sobre o componente o usuário ficava em dúvida se deveria manter a mão posicionada na área de clique ou se deveria retornar a mão à posição inicial e fazer um novo clique.
- **Dificuldades com a precisão dos botões na luva sem feedback** – este problema está relacionado a dois itens já mencionados: a falta de precisão no deslocamento da mão em Z e a dificuldade de diferenciar cursor laranja do vermelho. Sem poder contar com o auxílio do recurso de feedback tátil, o usuário dependeria da indicação de cor do cursor para saber se atingiu a superfície de clique. Com o problema da tonalidade este recurso ficou comprometido, sendo ainda agravado pela falta de precisão no movimento, gerando assim consequências similares às já observadas no primeiro item, como o clique em áreas diferentes da desejada.
- **Dificuldades de compreensão das instruções** – observou-se ainda que alguns usuários tiveram dificuldades na compreensão de algumas instruções presentes no software. Tarefas não foram compreendidas adequadamente ocasionando a escolha de opções incorretas ou uma demora maior na realização da tarefa até que se tivesse o entendimento adequado. Alguns usuários solicitaram ajuda ao avaliador para compreender o que deveria ser feito na tarefa e um deles não entendeu que deveria executar apenas o que era solicitado nas instruções, acabando por interagir com todos os componentes de interface gráfica presentes na tela e não apenas com o que era indicado nas instruções.

4.3.4. Métricas auto-relatadas (*self-reported metrics*)

A partir dos dados coletados nas entrevistas pós-testes, tivemos condições de analisar métricas auto-relatadas de forma quantitativa e qualitativa. O preenchimento do questionários com perguntas na escala Likert é a base da análise quantitativa apresentada a seguir. A gravação do áudio das entrevistas em formato de bate-papo com os participantes permitiu o levantamento de uma grande quantidade de informações sobre problemas e sugestões de melhorias por parte dos usuários. Essas informações serão apresentadas na seção 4.3.4.2 (Análise qualitativa).

4.3.4.1. Análise quantitativa

A Figura 4.14 mostra a média das notas em resposta à primeira pergunta do questionário pós-testes, que pedia para o usuário indicar sua opinião sobre a utilização de cada um dos dispositivos com uma nota variando de 1 a 7, onde o primeiro valor significa “muito ruim” e o último “muito bom”. A seguir foram feitas perguntas solicitando a opinião dos participantes sobre cada uma das tarefas individualmente. Os resultados observados indicam que a opinião referente a cada questão individualmente segue a mesma tendência observada na questão inicial, que cobre a visão geral. Por isso optou-se por não analisar as questões de forma individual nesta seção, discutindo-se apenas os feitos da visão geral dos usuários sobre a realização das tarefas em cada um dos dispositivos.

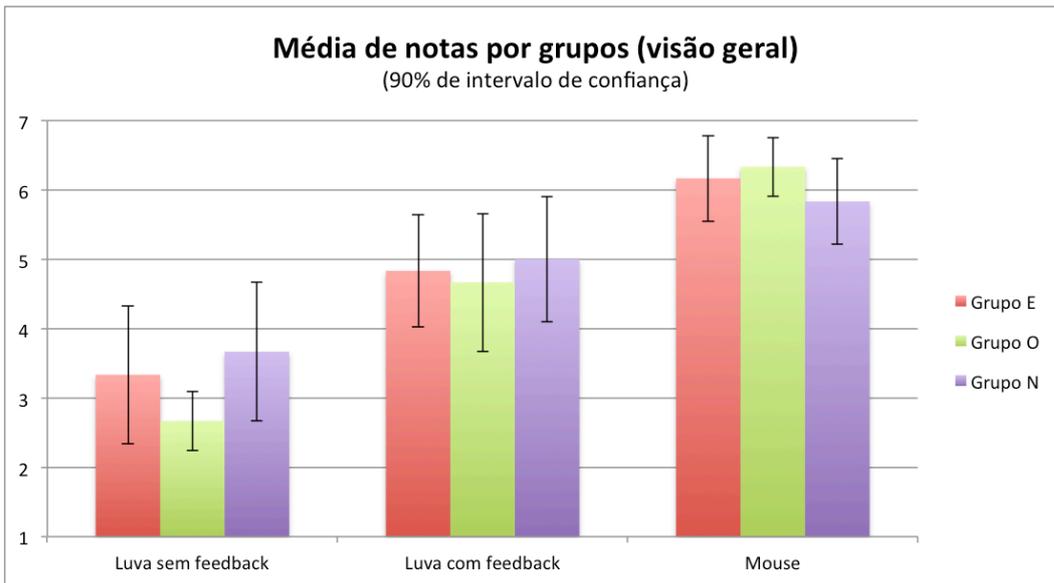


Figura 4.14: Média de notas por grupos por dispositivo (visão geral)

Os resultados das avaliações foram compatíveis com os resultados de desempenho observados nas métricas de duração das tarefas. A v-Glove sem o recurso de feedback tátil teve a pior avaliação, seguida pela luva com o recurso de feedback tátil e pelo mouse. Veremos mais à frente se essas diferenças representam alguma relevância em termos estatísticos.



Figura 4.15: Notas por *widget* para cada dispositivo

A Figura 4.15 mostra um gráfico de radar comparando a avaliação dos usuários por componente de interface gráfica utilizado em cada um dos dispositivos. Nesse caso a divisão dos grupos foi ignorada e o domínio de participantes considerado refere-se aos dezoito indivíduos que colaboraram com os testes. Pode-se observar neste gráfico que os componentes *checkbox* e de entrada numérica seguem um mesmo padrão ao se comparar os três dispositivos. Já no caso dos componentes *slider* e *combobox* observa-se que a distância entre as duas versões da luva e o mouse é maior, sendo perceptível pela deformação no gráfico tanto no setor sul (referente ao *slider*) quanto no setor leste (referente à *combobox*).

4.3.4.2. Análise qualitativa - sugestões de melhorias

Por tratarem-se de usuários que, em sua maioria, já possuíam experiência no desenvolvimento de aplicações 3D, buscamos durante as entrevistas após os testes capturar críticas e contribuições dos participantes entrevistados. A Tabela 4.5 lista as sugestões dadas pelos usuários durante as entrevistas. As sugestões comuns a mais de um usuário são indicadas com o valor da frequência entre parênteses à direita da descrição.

Sugestões dos participantes
Aumentar a escala de movimento da mão, pois ela é pequena em relação ao cursor na tela
Considerar o clique no local do primeiro contato com a área de clique e não ao sair dela
Detectar o clique a partir de uma variação mínima no eixo Z após atingir a área de toque
Diminuir a intensidade da vibração
Fazer o clique de outra forma para ser menos cansativo para a mão
Fazer o clique em um movimento de pinçar entre o dedo indicador e o polegar (2)
Fazer o reconhecimento do clique em uma região relativa à localização do usuário e não fixa
Fazer o <i>slider</i> de forma vertical e não mais horizontal
Implementar o feedback tátil quando o cursor atingir algum <i>widget</i> que permita interação
Implementar um feedback tátil progressivo (2)
Implementar um filtro de ruídos de movimento similar ao que existe no mouse
Incluir um botão para clique na luva (2)
Manter a vibração (numa intensidade menor) enquanto a v-Glove estiver na área de clique
Melhorar o feedback visual sincronizando a mudança de cor do cursor com mudanças nos <i>widgets</i>

Tabela 4.4: Sugestões dos participantes

Algumas das sugestões mencionadas pelos usuários chegaram a ser avaliadas durante a construção do protótipo da v-Glove. Para cada uma delas apresentaremos a seguir os motivos que nos levaram a descartá-las durante o desenvolvimento deste trabalho.

- **Considerar o clique no local do primeiro contato com a área de clique e não ao sair dela** – essa abordagem foi testada durante a construção do driver de controle do cursor da v-Glove, mas acabou sendo descartada após a realização do primeiro teste piloto. O princípio de funcionamento da operação de clicar em um objetivo com a luva é o seguinte: ao atingir a área de clique da câmera o driver da v-Glove gera um evento do tipo *MOUSE_PRESS*, que corresponde ao botão do mouse pressionado. Esse estado é mantido até que o dedo do usuário saia da área de clique, quando então é gerado um evento de *MOUSE_RELEASE*. Apenas nesse momento é que o clique é efetuado, seguindo o mesmo princípio de um mouse convencional. O problema neste caso é que entre estes dois estados ocorrem variações involuntárias (nos eixos X, Y e Z) do posicionamento da mão do usuário. Por isso, no instante em que o clique é considerado é possível que o posicionamento da mão do usuário tenha sido alterado para outra área que não a desejada por ele para interagir. Por limitações do driver, para considerar o clique no instante em que o mouse é pressionado seria necessário reposicionar o cursor de volta para as coordenadas do momento em que ocorreu o evento de *MOUSE_PRESS*. Isso, entretanto, acaba por gerar uma mudança brusca na posição atual do cursor que prejudica a utilização da v-Glove, conforme pode ser observado no teste piloto. Além disso, essa mudança causa um problema quando o usuário ao invés de selecionar deseja arrastar o objeto, uma vez que ao iniciar a operação de arrastar o cursor retorna bruscamente à sua posição anterior causando um salto de posicionamento do objeto na tela.
- **Detectar o clique a partir de uma variação mínima no eixo Z após atingir a área de toque** – variações mínimas ocorrem naturalmente, uma vez que é muito difícil para o usuário manter sua mão completamente imóvel no ar. Por isso o algoritmo da aplicação filtra movimentos com intensidades menores do que um valor de *threshold* configurável, fazendo com que os movimentos sejam considerados apenas se forem maiores do que esse *threshold*. Para considerar essa sugestão, a variação no eixo Z deveria ser maior do que este valor de descarte, causando um impacto na precisão do movimento de arrastar do cursor.

- **Diminuir a intensidade da vibração** – o componente de vibração utilizado no protótipo (Lilypad Vibe Board) não suporta o ajuste de intensidade da vibração, o que impede a implementação dessa funcionalidade. Para obtermos este resultado teríamos que substituir este componente por outro com suporte a diferentes intensidades de vibração.
- **Implementar o feedback tátil progressivo** – esta funcionalidade não pode ser implementada pelo mesmo problema do tópico anterior. Sem o suporte a diferentes níveis de vibração no hardware utilizado não é possível a implementação do conceito de feedback tátil progressivo.
- **Incluir um botão para clique na luva** – a ideia de se acrescentar um botão na luva para fazer o clique foi considerada nos primeiros estudos sobre a operação de clique na v-Glove. Entretanto, ela acabou sendo descartada porque, no nosso entendimento, o fato do usuário ter que pressionar um botão reduz a naturalidade da interação com a v-Glove, deixando-a muito similar a um mouse como o que foi utilizado nos testes. O conceito de pressionar um botão para clicar acaba por ser intuitivo pela analogia que atualmente se tem com o mouse tradicional, mas distancia-se da proposta desse trabalho onde a intenção era propor formas de interação mais próximas do que se faz com objetos no mundo real.
- **Manter a vibração (numa intensidade menor) enquanto a v-Glove estiver na área de clique** – a ideia de manter a vibração enquanto o dedo do usuário estiver na área de clique chegou a ser testada durante a construção do protótipo, mas acabou sendo descartada porque a manutenção da vibração por um período prolongado causa desconforto no usuário. Como já foi comentado, a falta de suporte ao ajuste de intensidade da vibração no hardware utilizado não permitiu também que se testasse a ideia de manter a vibração com menor intensidade.

As sugestões que não haviam sido consideradas durante a construção da v-Glove serão relacionadas como alternativas para trabalhos futuros e serão discutidas em maiores detalhes no capítulo 5.

4.4. Discussões

Nesta seção faremos uma comparação dos resultados obtidos nas quatro métricas de usabilidade aplicadas. A partir dessa análise espera-se obter uma visão mais clara do potencial da v-Glove como dispositivo de interação para ambientes imersivos de realidade virtual e um direcionamento sobre os pontos a serem aprimorados em trabalhos futuros.

4.4.1. Comparação dos resultados das métricas de usabilidade e impacto do perfil dos usuários

A análise de sucesso das tarefas não permitiu descobertas muito relevantes, já que os testes foram desenhados de forma a que os usuários não tivessem dificuldades em completar as tarefas. Os poucos casos em que os participantes tiveram dificuldades em executar suas tarefas ocorreram porque o usuário selecionou uma opção errada por engano ou acidentalmente. Estes problemas foram identificados em algumas situações de usabilidade observadas e podem ser resolvidos com melhorias no software criado para os testes dos usuários.

Nas métricas de desempenho na realização das tarefas, por outro lado, foi possível observar uma diferença entre o tempo médio gasto em cada um dos dispositivos e no tempo médio de realização das tarefas em cada um dos grupos.

Os dados visualmente indicam uma leve tendência de um melhor desempenho de acordo com o nível de experiência do usuário com o desenvolvimento de aplicações 3D. Entretanto, realizando-se uma análise da variância (ANOVA) de dois fatores com replicação foi possível constatar que essa diferença não possui relevância estatística quando comparamos o desempenho de um dos grupos em relação aos outros ($F(3,20) = 2,56, p < 0,09$). A mesma técnica confirmou também que as diferenças de tempo entre os dispositivos são de fato relevantes ($F(3,20) = 19,19, p < 0,01$), o que comprova os resultados obtidos nas primeiras observações, onde projetamos que o mouse teria o melhor desempenho, seguido pela luva com feedback tátil e, por último, a

versão da luva sem o feedback tátil. Maiores detalhes sobre análise da variância (ANOVA) realizada neste trabalho podem ser encontrados no Apêndice D.

Os resultados da métrica de situações de usabilidade indicam que 58% das situações observadas durante os testes estão relacionadas a problemas no funcionamento dos dispositivos de interação. O gráfico da Figura 4.16 mostra a percentual dessas situações de usabilidade relacionadas a todos os dispositivos, apenas à luva e apenas ao mouse. No caso da luva consideramos em um mesmo grupo situações relacionadas à luva com e sem o recurso de feedback tátil.

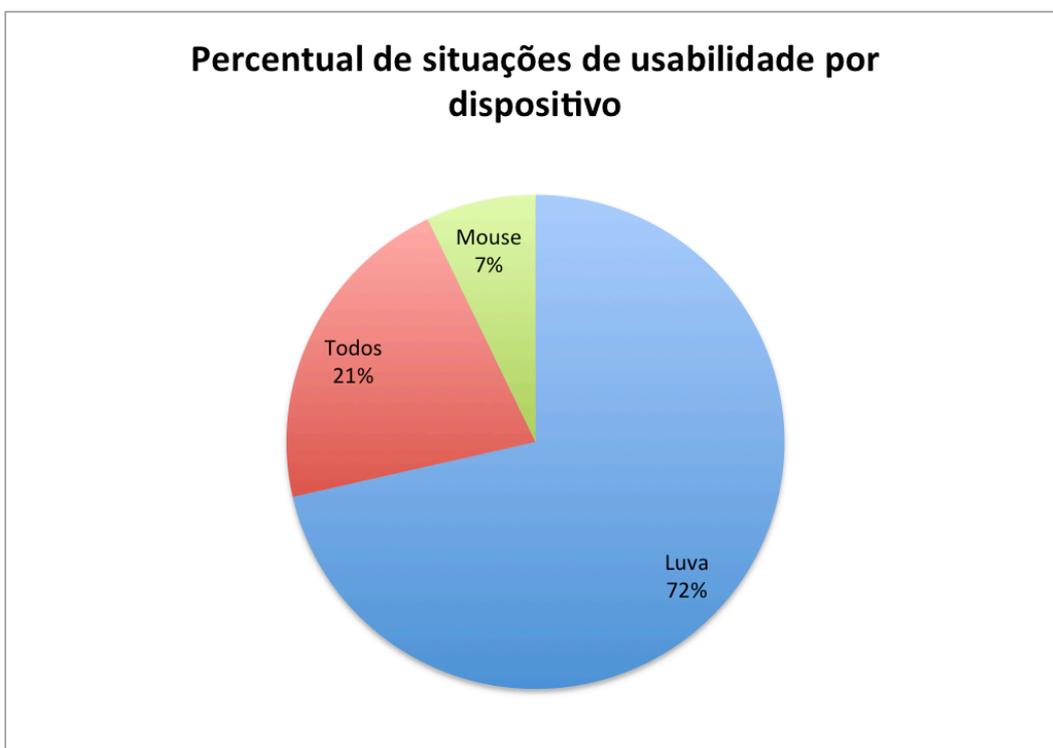


Figura 4.16: Percentual de situações de usabilidade por dispositivo

Pode-se observar no gráfico que apenas 7% das situações observadas são exclusivas do mouse, enquanto 72% estão relacionadas à operação da luva com e sem feedback tátil. Os 21% restantes referem-se a situações observadas tanto na luva quando no mouse, como problemas de postura de uso dos dispositivos, dores ou cansaço em sua operação.

A partir desses resultados é possível projetar que um trabalho direcionado à correção e redução dessas situações e problemas de usabilidade na v-Glove

poderá resultar em uma melhora nos indicadores da luva em comparação ao mouse.

No caso das métricas auto-reportadas baseadas nas notas dadas pelos usuários tivemos um resultado similar ao obtido na análise de desempenho. As avaliações não seguem um padrão de acordo com os grupos, mas a análise da variância (ANOVA) de dois fatores com replicação indicou uma diferença com relevância estatística entre os dispositivos ($F(3,2) = 41,23$, $p < 0,01$).

4.4.2. Impacto do feedback tátil na v-Glove

O objetivo ao incluirmos a luva sem o recurso de feedback tátil na lista de dispositivos avaliados foi verificar se este recurso melhora a usabilidade da luva se comparado com uma versão da mesma luva sem ele. O resultado do ANOVA de dois fatores com replicação para a comparação entre o desempenho dos usuários indica que a diferença entre a luva com feedback tátil e sem feedback tátil possui relevância estatística ($F(4,17) = 5,23$, $p < 0,03$). As opiniões colhidas dos usuários reforçam ainda mais essa constatação, especialmente quando lhes foi solicitado que classificassem com uma nota de 1 a 7 a importância do recurso de feedback tátil na v-Glove.

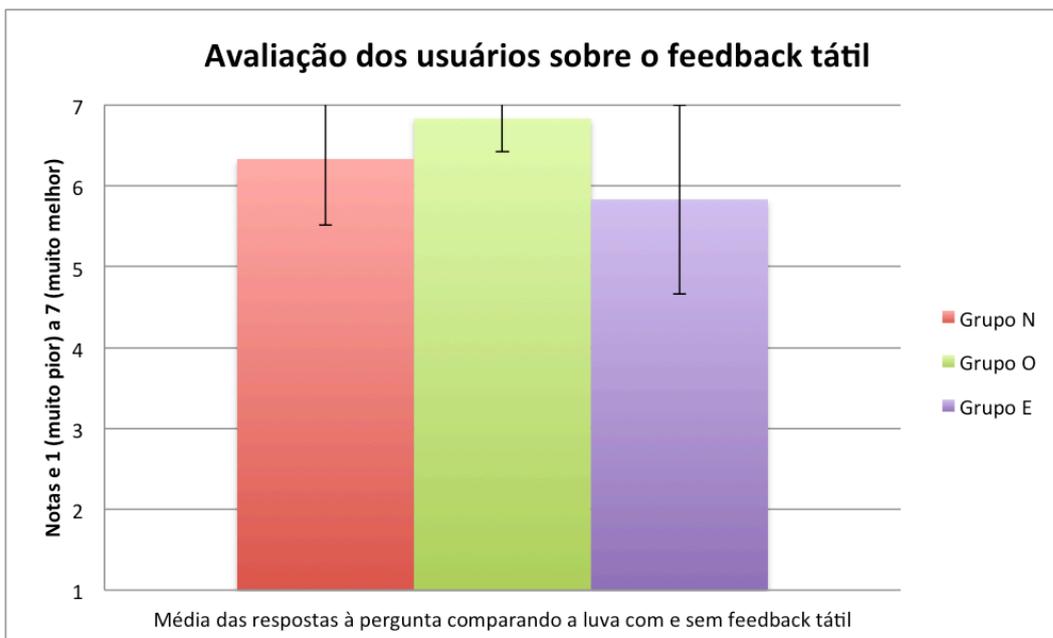


Figura 4.17: Opinião dos usuários sobre o recurso de feedback tátil na v-Glove

A Figura 4.17 mostra o gráfico com a média das respostas dos usuários à pergunta: “Comparando-se a luva com e sem o recurso de feedback tátil, qual nota na escala de 1 a 7 você atribuiria a este recurso, sendo 1 equivalente a muito pior e 7 indicando que o feedback tátil é muito melhor?”. A média geral das notas foi 6,3, variando de 5,8 no Grupo E a 6,8 no Grupo O.

4.4.3. Impacto do aprendizado na avaliação

Um efeito já esperado era uma melhora de desempenho natural na segunda e na terceira sequências de tarefas, resultado do efeito do aprendizado de uso da interface. No caso da luva havia ainda o efeito de aprendizado do funcionamento da própria luva, uma vez que entre a versão com e sem o recurso de feedback tátil não havia nenhuma outra diferença além da presença ou não da vibração. O gráfico da Figura 4.18 mostra a evolução dos tempos de duração das tarefas para as duas versões da luva de acordo com a ordem em que cada dispositivo era utilizado na sequência de tarefas. Neste caso não estamos considerando o agrupamento dos usuários por perfil, mas sim comparando todos os usuários que realizaram tarefas com cada uma das luvas antes ou depois de ter realizado a tarefa com a outra versão.

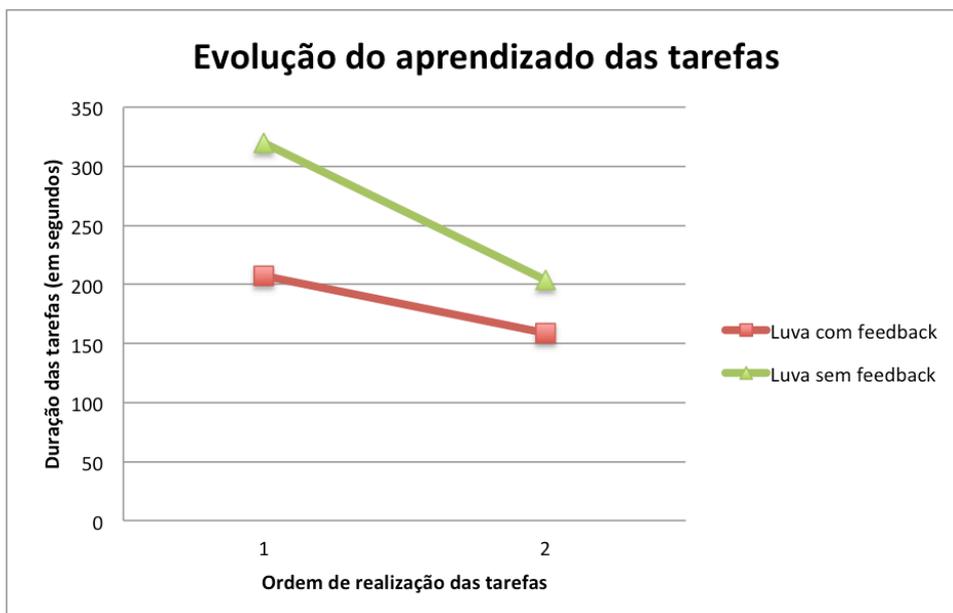


Figura 4.18: Evolução do aprendizado das tarefas

O gráfico confirma que o efeito de aprendizado influencia diretamente no desempenho dos usuários na realização das tarefas. A média de tempo de

duração dos usuários que utilizaram a luva sem feedback tátil antes da versão com feedback foi de 319 segundos, enquanto os usuários que utilizaram a versão sem feedback depois de já ter usado a luva com feedback tiveram média de 204 segundos. Isso indica uma melhora de 36% no desempenho dos usuários como efeito do aprendizado do dispositivo.

No caso da v-Glove com feedback tátil, os usuários que a utilizaram antes da versão sem este recurso tiveram um tempo médio de 207 segundos, enquanto a média de quem usou a luva com feedback após ter usado a versão sem este recurso foi de 159 segundos. Neste caso o efeito do aprendizado ocasionou uma melhora de 23% em média.

5 Conclusão

Neste trabalho propusemos a criação da v-Glove, um dispositivo de interação para ambientes imersivos de realidade virtual. A v-Glove é uma luva que permite ao usuário interagir com aplicações de realidade virtual de forma natural através do movimento da mão no espaço e a seleção e manipulação de objetos através da aproximação da mão a uma área de toque virtual mapeada neste mesmo espaço. Como prova de conceito criamos uma aplicação capaz de interagir com o software Environ em tarefas de manipulação e configuração de parâmetros em modelos de plataformas de petróleo 3D visualizados em um ambiente imersivo. Realizamos um estudo de usabilidade da luva com avaliações quantitativas e qualitativas com três grupos de usuários com diferentes perfis de utilização de aplicações 3D.

Identificamos que a luva ainda não possui condições de substituir integralmente um dispositivo como o mouse 3D e que, por ser um produto comercial produzir em escala industrial, o mouse leva uma grande vantagem por questões como acabamento, resistência e ergonomia, enquanto o protótipo da luva desenvolvido apresentar uma certa fragilidade ao ser manuseada. Entretanto observamos um grande potencial de uso caso algumas situações de usabilidade identificadas nas avaliações sejam aprimoradas. O uso da luva como elemento de controle de cursor na tela teve uma resposta bastante positiva na opinião dos usuários quando comparado ao mouse, sendo sua principal deficiência o movimento necessário para a seleção de objetos e *widgets* de interface gráfica. Entendemos que a luva também foi prejudicada em sua comparação com o mouse 3D porque as tarefas de interação suportadas pelo Environ já foram mapeadas para um modelo 2D de forma a facilitar sua utilização com o mouse em ambientes desktop.

Uma parte significativa deste trabalho foi dedicada à adaptação dos componentes de interface gráfica (*widgets*) ao contexto de aplicações imersivas. Assim como no caso de aplicações de toque e multi-toque, os controles das

aplicações necessitam prever seu uso dentro das peculiaridades de um ambiente imersivo. Aplicações projetadas para o padrão WIMP dificilmente poderão ser utilizadas nesse tipo de ambiente sem algum tipo de adaptação.

Em relação aos componentes de interface gráfica estudados, identificamos que o *slider* é o *widget* mais problemático, independente do dispositivo de interação utilizado. Mesmo com o mouse foi possível observar situações de usabilidade neste componente, o que nos leva a concluir que é necessário que se pense em um redesenho deste elemento de interface de forma a melhor adequá-lo ao contexto da imersividade. Sugestões de alguns usuários recomendam que se altere o formato deste componente para que a régua de valores seja vertical e não mais horizontal, como usado atualmente.

No componente *combobox* também identificamos alguns problemas de interação, já que para que o usuário possa selecionar uma opção são necessários pelo menos dois cliques, um para visualizar as opções disponíveis e outro para escolher a opção desejada. Este *widget* necessita também ser redesenhado de forma melhorar seu uso. Dentre as possíveis alternativas de formatos estão as propostas por Gerber e Bechmann (2004) e Dachsel e Ebert (2001). Estes autores propõem respectivamente a adoção de menus de opções nos formatos de anel (*ring menu*) e cilíndrico (*collapsible cylindrical trees*).

Sobre o recurso de feedback tátil observamos que este é de fato essencial para a operação da luva, resultado confirmado nos testes comparativos da luva com e sem este recurso. Alguns aprimoramentos ainda podem ser feitos a esta funcionalidade, como o suporte à variação de intensidade da vibração e a implementação do feedback tátil progressivo. Essas modificações levam em consideração as ponderações de Wright (2011) visando uma melhor adequação às expectativas dos usuários.

5.1. Trabalhos futuros

Existem muitas possibilidades de trabalhos futuros a partir das conclusões que chegamos neste trabalho. Muitos direcionamentos surgiram a partir de observações feitas durante os testes e principalmente a partir das entrevistas com os usuários. Algumas iniciativas são direcionadas a melhorias no princípio

de funcionamento da luva, enquanto outras envolvem a expansão de sua utilização para domínios de tarefas de interação mais complexos. As seções a seguir apresentarão nossas principais ideias nesse sentido.

5.1.1. Teste da v-Glove com tarefas de interação 3D

As tarefas de interação selecionadas para o estudo de caso deste trabalho foram essencialmente tarefas 2D para interação com objetos presentes em uma cena de visualização 3D, uma vez que o Environ faz um mapeamento das tarefas de interação para um modelo de duas dimensões voltado para sua utilização com o mouse em ambientes desktop. Sugerimos que a v-Glove seja testada num contexto de tarefas de interação 3D como a rotação e movimentação de objetos, tarefas essas onde se espera que o mouse enfrente maiores dificuldades.

5.1.2. Ajuste na escala de cores do feedback visual

De forma a melhorar a percepção do usuário em relação ao feedback visual do cursor, a escala de cores deve ser modificada, substituindo-se a tonalidade de laranja por uma cor que tenha menor chance de ser confundida com o vermelho dependendo das características da tela ou do projeto utilizado. Modelos de escala de cor como o proposto por MacAdam (1974) devem ser estudados.

5.1.3. Suporte multi-toque e de gestos

As tarefas de interação realizadas durante o estudo de caso deste trabalho utilizaram o conceito de toque, mas não chegaram a explorar a possibilidade de multi-toque e do suporte a gestos. Sugerimos portanto que a v-Glove seja testada no contexto de uma aplicação que utilize este tipo de interação, com tarefas como a rotação e manipulação de objetos.

5.1.4. Interação com as duas mãos

De forma a expandir o uso da v-Glove sugerimos a construção de uma segunda luva para ser usada na mão esquerda permitindo assim a interação

com as duas mãos simultaneamente. Neste modelo entendemos que a luva tem um grande potencial para tarefas de interação baseadas em gestos, especialmente para tarefas como a manipulação de objetos 3D na cena.

5.1.5. Alteração no Posicionamento das câmeras

Configurar o posicionamento das câmeras baseado nos conceitos de geometria epipolar. Isso resolveria o problema da oclusão no eixo Z permitindo a utilização da luva com o suporte ao rastreamento da profundidade de mais de um dedo simultaneamente.

5.1.6. Clicar através do movimento de pinçar entre os dedos

A partir do que se observou na análise de situações de usabilidade e com base em algumas sugestões de participantes sugerimos que sejam feitos testes com a v-Glove alterando-se a forma de efetuar o clique. Ao invés do conceito de aproximar o dedo da superfície de toque virtual, sugerimos que seja testada a ideia de clicar através do gesto de “pinçar” entre os dedos indicador e o polegar. Para selecionar um arrastar um objeto o usuário deveria efetuar o gesto de juntar as pontas desses dois dedos, liberando o clique ao separá-los novamente.

Tal funcionalidade é característica das luvas do tipo *pinch glove* existentes no mercado [Bowman et al 2001], cuja integração com a proposta da v-Glove poderá ser estudada no futuro.

5.1.7. Plano de toque virtual relativo à localização do usuário

Também com base na análise de situações de usabilidade e nas sugestões dos usuários sugerimos que se faça uma outra alteração na forma de clicar da v-Glove. Ao invés de identificar o clique pela aproximação da luva a uma área fixa mapeada no espaço virtual da sala onde o usuário se encontra, propõe-se que a identificação do clique seja feita através do mapeamento do movimento da luva a partir da posição atual do usuário. A superfície de toque virtual não teria mais uma localização absoluta, mas sim relativa ao ponto onde o usuário se encontra em um determinado momento dentro da sala.

6

Referências Bibliográficas

- 1 3M - Disponível em http://solutions.3m.com.br/wps/portal/3M/pt_BR/SegPessoal/Home/ProdutosServicos/8910_9910/. Último acesso em Janeiro de 2011.
- 2 Air Mouse - Disponível em <http://www.gyration.com/index.php/us/products/in-air-micekeyboards/go-air-mouse.html#specs>. Último acesso em Fevereiro de 2011.
- 3 AGRESTI, A; Comparing Groups: Analysis of Variance (ANOVA) Methods. Disponível em <http://www.stat.ufl.edu/~aa/sta6127/ch12.pdf>. Último acesso em Março de 2011.
- 4 Anova - Disponível em <http://www.laits.utexas.edu/orkelm/excelpor/anova2manypor.htm>. Último acesso em Março de 2011.
- 5 Arduino - Disponível em <http://www.arduino.cc/>. Último acesso em Janeiro de 2011.
- 6 ART Fingertracking - Disponível em <http://www.ar-tracking.de/Fingertracking.54+B6Jkw9.0.html>. Último acesso em Janeiro de 2011.
- 7 ART References - Disponível em http://www.ar-tracking.de/References.64+B6Jkw9MA__.0.html. Último acesso em Janeiro de 2011.
- 8 ART Tracking - Disponível em <http://www.ar-tracking.de/System-overview.20.0.html>. Último acesso em Janeiro de 2011.
- 9 BACIM, F.; BOWMAN, D.; PINHO, M.; Wayfinding Techniques for MultiScale Virtual Environments. IEEE Symposium on 3D User Interfaces, Louisiana, EUA, 2009.
- 10 BlueCove - Disponível em <http://bluecove.org/>. Último acesso em Janeiro de 2011.
- 11 Bluetooth - Disponível em <http://www.bluetooth.com/English/Pages/default.aspx>. Último acesso em Janeiro de 2011.
- 12 BOWMAN, D.; WINGRAVE, C.; CAMPBELL, J.; LY, V.; Using Pinch Gloves for both Natural and Abstract Interaction Techniques in Virtual Environments. Proceedings of HCI International, EUA, 2001.
- 13 BOWMAN, D.; KRUIJFF, E.; LAVIOLA, J.; POUPYREV, I.; 3D User Interfaces Theory and Practice. Editora Addison-Wesley (ISBN 0-201-75867-9). Boston, MA, EUA, Julho de 2004.

- 14 BOWMAN, D.; WINGRAVE, C.; Design and Evaluation of Menu Systems for Immersive Virtual Environments. IEEE Virtual Reality Conference, Yokohama, Japão, 2001.
- 15 BUECHLEY, L.; EISENBERG, M.; CATCHEN, J.; CROCKETT, A.; The LilyPad Arduino: Using Computational Textiles to Investigate Engagement, Aesthetics, and Diversity in Computer Science Education. CHI 2008, Florence, Italy.
- 16 CHEN, J.; BOWMAN, D.; Domain-Specific Design of 3D Interaction Techniques: An Approach for Designing Useful Virtual Environments Applications. MIT Presence, Vol. 18, No. 5, páginas 370-386, Outubro de 2009.
- 17 CRUZ-NEIRA, C.; SANDIN, D.; DEFANTI, T.; KENYON, R.; HART, J.; The CAVE: Audio Visual Experience Automatic Virtual Environment. Communications of the ACM, Vol. 35, No. 6, páginas 64-72, Junho de 1992.
- 18 DACHSELT, R.; EBERT, J.; Collapsible cylindrical trees: a fast hierarchical navigation technique. Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01). Washington, DC, EUA: IEEE computer society; 2001. p. 79–86.
- 19 DACHSELT, R.; HUBNER, A.; Three-dimensional menus: A survey and taxonomy. Computers & Graphics, Volume 31, Issue 1, Janeiro de 2007.
- 20 DARKEN, R.; DUROST, R.; Mixed-Dimension Interaction in Virtual Environments. Proceedings of the ACM symposium on Virtual reality software and technology, Monterey, CA, EUA, 2005.
- 21 Digi - XBee / XBee-PRO RF Modules – 802.15.4 - Product Manual v1.xEx. Digi International Inc, Minnetonka, MN, EUA, Setembro de 2009.
- 22 GERBER, D.; BECHMANN, D.; Design and evaluation of the ring menu in virtual environments. IPT 2004: Eighth immersive projection technology workshop, Ames, IA, EUA, 2004.
- 23 HAND, C.; A Survey of 3D Interaction Techniques. Computer Graphics Forum, Volume 16, Issue 5, páginas 269-281, Dezembro de 1997.
- 24 HARTLEY, R.; ZISSERMAN, A.; Multiple View Geometry in Computer Vision, Second Edition. Cambridge University Press, Cambridge, UK, Março de 2004.
- 25 KULIK, A.; HOCHSTRATE, J.; KUNERT, A.; FROELICH, B.; The Influence of Input Device Characteristics on Spatial Perception in Desktop-Based 3D Applications. IEEE Symposium on 3D User Interfaces, Louisiana, EUA, 2009.
- 26 MACADAM, D.; Uniform color scales. Journal of the Optical Society of America, volume 64, número 12, EUA, Dezembro, 1974.
- 27 MITCHELL, P.; A Step-by-step Guide to Usability Testing. Editora iUniverse, Miami, FL, EUA, 2007.

- 28 Nvidia Quadro - Disponível em http://www.nvidia.com/object/quadro_fx_5600_4600.html. Último acesso em Fevereiro de 2011.
- 29 LEE, M.; WOO, W.; ARKB: 3D vision-based Augmented Reality Keyboard. International Conference on Artificial Reality and Telexistence, Toquio, Japão, 2003.
- 30 Projection Design - Disponível em <http://www.projectiondesign.com/Default.asp?CatID=1764>. Último acesso em Fevereiro de 2011.
- 31 RAPOSO, A., SANTOS, I., SOARES, L., WAGNER, G., CORSEUIL, E., GATTASS, M. Environ: Integrating VR and CAD in Engineering Projects. IEEE Computer Graphics & Applications, v.29, n.6, p.91-95, 2009. (ISSN 0272-1716). DOI: 10.1109/MCG.2009.118.
- 32 SCHEIBE, R.; MOEHRING, M.; FROEHLICH, B.; Tactile Feedback at the Finger Tips for Improved Direct Interaction in Immersive Environments. Virtual Reality Conference, Charlotte, NC, Março de 2007.
- 33 SCHMULLER, J.; Statistical Analysis with Excel for Dummies, 2nd Edition. Wiley Publishing, Indianapolis, Indiana, EUA, 2009.
- 34 SMITH, S.; DU'MONT, S.; Measuring the Effect of Gaming Experience on Virtual Environment Navigation Tasks. IEEE Symposium on 3D User Interfaces, Louisiana, EUA, 2009.
- 35 SparkFun - Disponível em <http://www.sparkfun.com>. Último acesso em Janeiro de 2011.
- 36 TOSAS, M.; LI, B.; Virtual Touch Screen for Mixed Reality. ECCV 2004 Workshop on HCI, Prague, Czech Republic, Maio de 2004.
- 37 TULLIS, T.; ALBERT, B.; Measuring the User Experience. Editora Morgan Kaufmann (ISBN-13 978-0-12-373558-4), EUA, 2008.
- 38 Tune Glove - Disponível em <http://www.flickr.com/photos/jtanenbaum/4156417140/>. Último acesso em Janeiro de 2011.
- 39 Wii Remote - Disponível em <http://www.nintendo.com/wii/console/controllers>. Último acesso em Janeiro de 2011.
- 40 Wiigee - Disponível em <http://www.wiigee.org/>. Último acesso em Janeiro de 2011.
- 41 WiiRemoteJ - Disponível em <http://www.qj.net/qjnet/wii/wiiremotej-v14-java-library-wiimote-and-balance-board-interaction-via-bluetooth.html>. Último acesso em Janeiro de 2011.
- 42 WRIGHT, A.; The Touchy Subject of Haptics. Communications of the ACM, volume 54, número 1, Janeiro de 2011.
- 43 Xbee - Disponível em <http://www.digi.com/products/wireless/point-multipoint/xbee-series1-module.jsp#docs>. Último acesso em Janeiro de 2011.

- 44 ZigBee - ERGEN, S.; ZigBee/IEEE 802.15.4 Summary. Disponível em <http://www.zigbee.org/Specifications.aspx>. Último acesso em Janeiro de 2011.

Apêndice A – Estrutura de um programa Arduino e código fonte comentado

Um programa Arduino é chamado de *sketch* e é escrito em uma linguagem de programação similar à linguagem C. A estrutura de um *sketch* pode ser dividida em três partes: uma área para declaração de variáveis e funções criadas pelo usuário, a função `setup()` e a função `loop()`. Estas duas últimas serão descritas em maiores detalhes a seguir.

Função `setup()`: esta função é a primeira a ser chamada quando o programa inicia. Ela pode ser utilizada para a inicialização de variáveis e portas da placa controladora ou para o carregamento de bibliotecas de terceiros. A função `setup` é executada apenas uma vez, na inicialização da placa ou quando é feito um reset na mesma.

Função `loop()`: função chamada continuamente durante a execução do programa, permitindo que o código escrito pelo programador interaja com o micro controlador Arduino. A primeira execução dessa função ocorre somente após o termino da execução da função `setup`.

A seguir segue a transcrição do código fonte do *sketch* criado para este trabalho com comentários explicando sua funcionalidade linha-à-linha:

```
1   int finger1Pin = 7;
2   int finger2Pin = 8;
3   int finger3Pin = 9;
4   int finger4Pin = 10;
5   int finger5Pin = 11;
6   int incomingByte = -1;
7
8   void setup()
9   {
10      Serial.begin(9600);
11      pinMode(finger1Pin, OUTPUT);
12      pinMode(finger2Pin, OUTPUT);
13      pinMode(finger3Pin, OUTPUT);
14      pinMode(finger4Pin, OUTPUT);
15      pinMode(finger5Pin, OUTPUT);
16      Serial.flush();
17  }
18
19  void loop()
20  {
```

```
21     if (Serial.available() > 0)
22     {
23         incomingByte = Serial.read();
24
25         if (incomingByte == '1')
26         {
27             digitalWrite(finger1Pin, HIGH);
28             delay(300);
29             digitalWrite(finger1Pin, LOW);
30         }
31     else if (incomingByte == '2')
32     {
33         digitalWrite(finger2Pin, HIGH);
34         delay(300);
35         digitalWrite(finger2Pin, LOW);
36     }
37     else if (incomingByte == '3')
38     {
39         digitalWrite(finger3Pin, HIGH);
40         delay(300);
41         digitalWrite(finger3Pin, LOW);
42     }
43     else if (incomingByte == '4')
44     {
45         digitalWrite(finger4Pin, HIGH);
46         delay(300);
47         digitalWrite(finger4Pin, LOW);
48     }
49     else if (incomingByte == '5')
50     {
51         digitalWrite(finger5Pin, HIGH);
52         delay(300);
53         digitalWrite(finger5Pin, LOW);
54     }
55     }
56 }
```

Linhas 1 a 5: declaração das variáveis referentes às portas do micro controlador Arduino utilizadas (as portas usadas foram as de números 5, 6, 7, 8 e 9).

Linha 6: declaração de variável utilizada para leitura de bytes da porta serial.

Linha 10: define a taxa de transmissão da porta serial em bits por segundo.

Linhas 11 a 15: define o modo de operação das portas utilizadas da placa como saída (*OUTPUT*). Isso significa que essas portas serão utilizadas pelo micro controlador para a escrita de dados e não para a leitura de informações.

Linha 16: limpa os dados presentes na porta serial.

Linha 21: executa o bloco de código a seguir caso haja informação disponível para ser lida na porta serial.

Linha 23: lê um byte da porta serial para uma variável local.

Linhas 25, 31, 37, 43 e 49: executa os respectivos blocos de código caso o byte lido seja 1, 2, 3, 4 ou 5.

Linhas 27 a 29, 33 a 35, 39 a 41, 45 a 47 e 51 a 53: escreve na respectiva porta de saída um sinal de alta voltagem (5V), aguarda por 300 milissegundos e retira o sinal da porta após esse período (escrevendo um sinal de baixa voltagem, equivalente a 0V).

Apêndice B

Os modems XBee pode ser programados para operar com diversas configurações e topologias de rede. Neste trabalho utilizamos uma configuração de rede ponto-a-ponto com dois modems, onde um deles atua como transmissor e o outro como receptor. O modem transmissor é o que se encontra acoplado à estação de controle, enquanto o receptor é o módulo que fica na v-Glove.

Ao identificar a necessidade de emitir um sinal de vibração, a estação de controle envia essa informação através do módulo XBee transmissor que, por sua vez, retransmite o sinal para o XBee acoplado à v-Glove. Este encaminha as informações recebidas para o micro controlador LilyPad Arduino presente na luva que efetua a vibração de acordo com o comando recebido.

O XBee é configurado através de uma sequência de comandos enviados via um aplicativo de terminal que se conecta ao modem por uma porta serial. A Tabela B.1 apresenta a descrição dos comandos utilizados neste trabalho.

Comando AT	Nome e Descrição	Faixa de valores	Valor padrão
RE	Restore Defaults: restaura os valores padrão de fábrica.	-	-
ID	PAN ID: define o endereço da PAN (<i>Personal Area Network</i>). Utilize 0xFFFF para fazer broadcast para todas as redes.	0 - 0xFFFF	0x3332
MY	16-bit Source Address: define o endereço de 16 bits que identificará o modem na rede.	0 - 0xFFFF	0
DL	Destination Address Low: define os 32 bits mais baixos do endereço do módulo de destino.	0 - 0xFFFFFFFF	0
IR	Sample Rate: define a taxa de amostragem do módulo.	0 - 0xFFFF	0
IT	Samples before TX: define o número de amostras a serem coletadas antes de se iniciar a transmissão dos dados.	1 - 0xFF	1
IA	I/O Input Address: define o endereço do módulo para onde serão enviados os dados de saída.	0 - 0xFFFFFFFFFFFFFFFF	0xFFFFFFFFFFFFFFFF
WR	Write: escreve os valores dos parâmetros na memória não volátil, fazendo com que estes permaneçam mesmo quando o módulo não está	-	-

	alimentado por uma fonte de energia.		
CN	Exit Command Mode: faz com que o módulo saia do modo de comandos AT.	-	-

Tabela B.1: Comandos de configuração do modem XBee [Digi 2009]

A sequência de comandos utilizada para configurar o modem XBee presente na estação de controle (transmissor) foi a seguinte:

```
ATRE, ID4567, MY1, DL2, IR64, IT1, IA2, WR, CN
```

O primeiro comando restaura as configurações originais de fábrica e o segundo define o identificador da rede PAN como 4567. Os comandos seguintes definem respectivamente o endereço do módulo transmissor (1) e o endereço de destino das mensagens enviadas por ele (2). Os próximos comandos definem a taxa de amostragem do módulo XBee como 64 e o número de amostras coletadas antes de enviar como 1. A seguir é definido o endereço do módulo para onde são enviadas as informações de saída do XBee como 2. Finalmente o comando de gravação faz com que as informações sejam persistidas na memória não volátil e o último comando sai do módulo de entrada de comandos AT.

A sequência de comandos utilizada para configurar o modem XBee presente na luva (receptor) foi a seguinte:

```
ATRE, ID4567, MY2, DL1, IR64, IT1, IA1, WR, CN
```

As únicas diferenças na configuração do módulo receptor são os endereços do módulo, que neste caso será 2, e os endereços de destino e do módulo de saída dos dados que passam a ter valor 1 em ambos. Os outros comandos são exatamente iguais.

Apêndice C – Material para Observação de Uso

C.1 Termo de Consentimento

Você foi convidado(a) pelo Tecgraf – *Grupo de Tecnologia em Computação Gráfica* – um laboratório de pesquisas em Computação Gráfica do Departamento de Informática da PUC-Rio, para participar de um teste de avaliação da Interação Humano-Computador de uma luva projetada para a execução de tarefas em ambientes imersivos de realidade virtual.

Neste teste buscamos verificar a usabilidade de duas versões da luva em comparação à utilização de um mouse tridimensional, durante a realização de quatro tarefas de configuração de parâmetros em um ambiente virtual imersivo.

Por esta razão, solicitamos seu consentimento para a realização deste teste e, em seguida, uma breve entrevista. Para tanto, é importante que você tenha algumas informações:

1. Os dados coletados durante o teste destinam-se **estritamente** a atividades de pesquisa e desenvolvimento.
2. A equipe desta pesquisa tem o compromisso de divulgar os resultados de suas pesquisas para fins acadêmicos. A divulgação destes resultados pauta-se no **respeito a sua privacidade** e o **anonimato** dos mesmos é preservado em quaisquer documentos que elaboramos.
3. O consentimento para o teste é uma escolha livre, feita mediante a prestação de todos os esclarecimentos necessários sobre a pesquisa.
4. A realização do teste pode ser interrompida a qualquer momento, segundo a disponibilidade do participante. Neste caso, a equipe se compromete a descartar o teste para fins da avaliação a que se destinaria.
5. Nossa equipe encontra-se disponível para contato através do telefone (21) 9358-9823, ou pelo email gallotti@tecgraf.puc-rio.br. Caso queira tirar alguma dúvida, procure por Paulo Gallotti.

De posse das informações acima, gostaríamos que você se pronunciasse acerca do teste.

() Dou meu consentimento para sua realização.

() Não autorizo sua realização.

Rio de Janeiro, _____ de Fevereiro de 2011.

<i>Participante</i>	<i>Avaliador</i>
Nome: _____	Nome: _____
Assinatura: _____	Assinatura: _____

C.2 Questionário inicial – Levantamento de perfil do participante

ID: _____

Nome: _____

Sexo: _____

Idade: _____

Destro () Canhoto () Usa o mouse em qual mão? _____

Ocupação (e formação): _____

1. Quantas horas em média você passa no computador por semana?

() até 5 horas

() até 10 horas

() até 15 horas

() mais de 15 horas

2. Tem experiência de uso de aplicações 3D? Que tipo de aplicações? Quanto tempo de experiência?

3. Tem experiência de uso com o Environ? Quanto tempo de experiência?

4. Tem experiência com desenvolvimento de software 3D? Quanto tempo?

5. Já participou do desenvolvimento do Environ? Por quanto tempo?

C.3 Formulário de acompanhamento dos testes

ID: _____

Hora de chegada: _____

Hora de saída: _____

Listar erros do usuário ocorridos durante a execução das tarefas (Modos: M - mouse, L - luva, LF – luva com feedback tátil).

Sequência	1	2	3
Modo			
Tarefa 1			
Tarefa 2			
Tarefa 3			
Tarefa 4			

C.4 Questionários pós-testes

ID: _____

1. Qual a sua avaliação em relação à utilização da luva SEM o recurso de feedback tátil?

Visão geral:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

Tarefa 1:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

Tarefa 2:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

Tarefa 3:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

Tarefa 4:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

2. Qual a sua avaliação em relação à utilização da luva COM o recurso de feedback tátil?

Visão Geral:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

Tarefa 1:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

Tarefa 2:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

Tarefa 3:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

Tarefa 4:

Muito ruim

--	--	--	--	--	--	--	--	--

 Muito bom
1 2 3 4 5 6 7 N/A

3. Qual a sua avaliação em relação à utilização do mouse 3D?

Visão Geral:

Muito ruim Muito bom

1 2 3 4 5 6 7 N/A

Tarefa 1:

Muito ruim Muito bom

1 2 3 4 5 6 7 N/A

Tarefa 2:

Muito ruim Muito bom

1 2 3 4 5 6 7 N/A

Tarefa 3:

Muito ruim Muito bom

1 2 3 4 5 6 7 N/A

Tarefa 4:

Muito ruim Muito bom

1 2 3 4 5 6 7 N/A

4. Em relação ao recurso de feedback tátil, como você o classificaria em comparação à versão da luva sem este recurso?

Muito pior Muito melhor

1 2 3 4 5 6 7 N/A

5. Classifique os modos de interação utilizados em ordem de preferência (de 1 a 3):

- () Luva sem feedback tátil
- () Luva com feedback tátil
- () Mouse 3D

6. Você sente falta de uma referência física ao manipular a interface no espaço?

7. Em relação a solução com a luva e o recurso de feedback tátil, você considera que a interface proposta é intuitiva?

Apêndice D – Análise da variância (ANOVA)

Este apêndice descreve o processo de geração de uma análise da variância (ANOVA) realizada para comparar a relevância estatística dos dados observados no estudo de usabilidade da v-Glove. Para a geração das análises utilizamos como ferramenta o software Microsoft Excel. Existem diversos tipos de ANOVA, mas apenas três deles são suportados no Excel: ANOVA de fator único, ANOVA de dois fatores com replicação e ANOVA de dois fatores sem replicação. Como no nosso trabalho temos dois fatores (o dispositivo utilizado e o grupo que utilizou) e todos os usuários testaram todos os dispositivos, usaremos o ANOVA de dois fatores com várias mostras por grupo, chamado pelo Excel de ANOVA de dois fatores com replicação. Exemplificaremos nas seções a seguir o processo de geração do ANOVA para a análise do desempenho dos participantes na realização das tarefas e para a análise das notas dadas pelos participantes para cada um dos dispositivos.

D.1 ANOVA no desempenho das tarefas

Para comparar o desempenho de três dispositivos distintos em três grupos de usuários utiliza-se o ANOVA de dois fatores com várias mostras por grupo. Como mencionado anteriormente, os dois fatores nesse caso são **dispositivo** e **grupo de usuários**. A Tabela D.1 mostra os dados que foram utilizados para geração da análise. A quantidade de dispositivos é representada pelas colunas da tabela e os grupos são representados pelas linhas. É importante ressaltar que o número de grupos e de dispositivos pode variar dependendo do estudo a ser feito.

Desempenho			
	L	LF	M
Grupo E	162	125	73
	154	94	56
	261	132	97
	201	203	120
	160	247	54
	190	125	51
Grupo N	162	271	81
	233	109	63
	604	351	156
	165	178	75
	254	204	86
	237	111	73
Grupo O	538	196	103
	341	195	141
	204	212	62
	215	198	74
	442	113	83
	185	227	66

Tabela D.2: Dados de desempenho dos participantes nas tarefas

O ANOVA de dois fatores com várias mostras por grupo pode ser gerado no Microsoft Excel clicando-se no menu **Ferramentas (Tools) > Análise de Dados (Data Analysis)** e selecionando a opção **Anova: Dois Fatores com Replicação (Anova: Two-Factor With Replication)** [Schmuller 2009]. Como dados de entrada foram selecionadas as linhas e colunas da Tabela D.1, definindo-se o valor 6 como número de amostras por grupo. O resultado da análise gerada pelo Microsoft Excel pode ser visto na Tabela D.2.

Anova: Two-Factor With Replication

SUMMARY	Luva sem feedback	Luva com feedback	Mouse	Total		
<i>Grupo E</i>						
Count	6	6	6	18		
Sum	1128	926	451	2505		
Average	188	154.33	75.17	139.17		
Variance	1623.60	3363.07	778.17	4064.03		
<i>Grupo N</i>						
Count	6	6	6	18		
Sum	1655	1224	534	3413		
Average	275.83	204	89	189.61		
Variance	27350.97	8889.6	1138	17263.31		
<i>Grupo O</i>						
Count	6	6	6	18		
Sum	1925	1141	529	3595		
Average	320.83	190.17	88.17	199.72		
Variance	21110.17	1581.37	882.97	16535.04		
<i>Total</i>						
Count	18	18	18			
Sum	4708	3291	1514			
Average	261.56	182.83	84.11			
Variance	17952.50	4532.62	865.75			
ANOVA						
Source of Variation	SS	df	MS	F	P-value	F crit
Sample	37883.11	2	18941.6	2.56	0.09	3.20
Columns	284578.78	2	142289.39	19.19	0.00000	3.20
Interaction	25492.11	4	6373.0	0.86	0.50	2.58
Within	333589.50	45	7413.1			
Total	681543.50	53				

Tabela D.3: ANOVA gerado no Excel para o desempenho dos participantes

O resultado de uma ANOVA é expressado através do valor estatístico F. Nesse caso, o valor de F da mostra (*Sample*) dos Grupos E, N e O é 2,56. Para saber se este resultado é significativo (ou seja, se a probabilidade P tem um valor inferior a 0.05), o valor de F precisa chegar pelo menos a 3,20 (que é o valor crítico de F). Como o valor de F é 2,56 e ele não é maior que o valor crítico de F, não podemos dizer que exista uma diferença significativa na comparação entre o desempenho dos grupos [Anova 2011].

No caso do valor de F para as colunas (*Columns*), o resultado foi significativo, pois o valor de 19,19 é superior ao valor crítico de F que é 3,20. Em outras palavras, pode-se afirmar que existe relevância estatística na diferença entre o desempenho dos participantes nos três dispositivos testados.

Ao mesmo tempo, porém, é preciso interpretar o valor de F que está relacionado ao efeito de uma interação. Dependendo da situação, o efeito de uma interação pode negar o efeito principal. Neste exemplo não existe efeito de interação já que o valor de F é 0,86, menor que o valor crítico de 2,58. Na prática isto quer dizer que não existe uma diferença significativa entre os grupos no desempenho das tarefas para cada um dos dispositivos [Agresti 2007].

D.2 ANOVA no questionário pós-testes

Para comparar as notas dadas pelos participantes dos grupos para cada um dos dispositivos utiliza-se novamente o ANOVA de dois fatores com várias mostras por grupo. Os dois fatores são os mesmos da seção anterior: dispositivo e grupo de usuários. A Tabela D.3 mostra os dados que foram utilizados para geração da análise.

Visão geral			
	L	LF	M
Grupo E	2	5	6
	4	5	6
	2	3	5
	3	5	7
	5	5	6
Grupo O	4	6	7
	2	4	6
	3	5	7
	2	3	6
	3	6	6
Grupo N	3	6	6
	3	4	7
	2	3	5
	4	5	6
	5	6	5
	5	5	6
	3	6	6
	3	5	7

Tabela D.4: Notas dadas pelos participantes para os dispositivos

O processo de geração da análise no Excel é exatamente igual ao feito na seção anterior. A diferença é que utilizaremos agora como dados de entrada a

Tabela D.3, que mostra as notas dos participantes para cada um dos dispositivos. O resultado da análise gerada pelo Microsoft Excel pode ser visto na Tabela D.4.

Anova: Two-Factor With Replication

SUMMARY	Luva sem feedback	Luva com feedback	Mouse	Total		
<i>Grupo E</i>						
Count	6	6	6	18		
Sum	20	29	37	86		
Average	3.33	4.83	6.17	4.78		
Variance	1.47	0.97	0.57	2.30		
<i>Grupo O</i>						
Count	6	6	6	18		
Sum	16	28	38	82		
Average	2.67	4.67	6.33	4.56		
Variance	0.27	1.47	0.27	2.97		
<i>Grupo N</i>						
Count	6	6	6	18		
Sum	22	30	35	87		
Average	3.67	5	5.83	4.83		
Variance	1.47	1.2	0.57	1.794		
<i>Total</i>						
Count	18	18	18			
Sum	58	87	110			
Average	3.22	4.833	6.11			
Variance	1.12	1.09	0.46			
ANOVA						
Source of Variation	SS	df	MS	F	P-value	F crit
Sample	0.78	2	0.39	0.43	0.66	3.20
Columns	75.44	2	37.72	41.23	0.000000000067	3.20
Interaction	3.44	4	0.86	0.94	0.45	2.58
Within	41.17	45	0.91			
Total	120.833	53				

Tabela D.5: ANOVA gerado no Excel para as notas dos participantes

O valor de F da mostra neste caso é 0,43. Como o valor é menor que o valor crítico de F, que é 3,20, podemos mais uma vez afirmar que não existe uma diferença significativa na comparação entre o desempenho dos grupos.

No caso do valor de F para as colunas, o resultado foi significativo, pois o valor de F = 41,23 é superior ao valor crítico de 3,20. Em outras palavras, pode-se mais uma vez afirmar que existe relevância estatística na diferença entre o desempenho dos participantes nos três dispositivos testados.

Em relação ao efeito de interação temos um valor de F de 0,94, inferior ao valor crítico de 2,58, confirmando o que não existe uma diferença significativa entre os grupos nas notas dadas para cada um dos dispositivos.