

Escola Politécnica da Universidade de São Paulo

Departamento de Engenharia Mecatrônica



HandNavigator

Uma nova abordagem para a interação entre as mãos e o mundo virtual 3D

André de Carvalho Amaro

Orientação: Prof. Dr. Marcos Barretto

Escola Politécnica da Universidade de São Paulo

Departamento de Engenharia Mecatrônica



HandNavigator

Uma nova abordagem para a interação entre as mãos e o mundo virtual 3D

André de Carvalho Amaro

Orientação: Prof. Dr. Marcos Barretto

Agradeço a meus amigos, família e colegas de universidade que sempre me acompanharam em mais esta conquista.

FICHA CATALOGRÁFICA

Amaro, André de Carvalho

HandNavigator: uma nova abordagem para a interação entre as mãos e o mundo virtual 3D / A.C. Amaro. -- São Paulo, 2010. 89 p.

Trabalho de Formatura - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.

1. Interação homem-máquina 2. Realidade virtual 3. Visão computacional 4. Terceira dimensão (Interação) I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos II. t.

Sumário

1. Introdução	12
1.1. Intercâmbio, Início do projeto na equipe EVASION do INRIA e continuação do projeto no Brasil na Escola Politécnica da USP	12
1.2. Motivação	13
1.3. O SpaceNavigator e a idealização do HandNavigator	15
1.4. Revisão bibliográfica.....	17
1.4.1. Classificação dos dispositivos segundo o tipo de feedback	17
1.4.2. Classificação dos sensores segundo o tipo de controle	23
1.4.3. Movimentação no espaço virtual e interação com objetos	27
1.5. Soluções já existentes para captura do movimento das mãos	34
1.5.1. Soluções sem retorno háptico	35
1.5.2. Soluções com retorno háptico	35
1.5.3. Soluções da equipe EVASION do INRIA	36
1.6. Problemas das soluções existentes para captura do movimento das mãos	41
1.6.1. Problemas da captura de movimento sem retorno háptico	41
1.6.2. Problemas da captura háptica	42
1.6.3. Problemas das soluções da equipe EVASION do INRIA	42
1.7. Tabela das soluções existentes e suas características principais	44
2. Objetivos	46
3. Metodologia	48
4. Projeto	50
4.1. Forma do periférico e material de construção	50
4.1.1. Análises	50
4.1.2. Decisões	55
4.2. Sensor e tipo de feedback para os dedos	59
4.2.1. Sensores disponíveis para utilização na solução	59
4.2.2. Análise dos sensores disponíveis segundo dados da Tabela 1.	59
4.2.3. Decisão do sensor a ser utilizado	61
4.2.4. Definição do tipo de controle	62

4.2.5.	Definição do tipo de retorno a ser utilizado	63
4.2.6.	Definição do circuito para um sensor	64
4.3.	Programa a ser desenvolvido	64
4.3.1.	Descrição do programa	64
4.3.2.	Definição variáveis do programa	68
4.3.3.	Tratamento do tipo de controle do SpaceNavigator	71
4.3.4.	Tratamento do tipo de controle dos sensores dos dedos	71
4.3.5.	Estrutura Geral do Programa	72
4.3.6.	Diagrama das funções que serão utilizadas	72
5.	Desenvolvimento	74
5.1.	Desenvolvimento da solução	74
5.1.1.	Desenvolvimento modelo 3D (CATIA)	74
5.1.2.	Projeto de circuito eletrônico	74
5.1.3.	Projeto do programa	75
5.2.	Desenvolvimento do protótipo final.....	75
5.2.1.	Construção periférico	75
5.2.2.	Construção circuito eletrônico	77
5.2.3.	Programação interface computacional	78
5.2.4.	Integração com interface antiga	79
6.	Validação	81
7.	Conclusão	85
8.	Referências Bibliográficas	88

Índice de Figuras

Figura 1 – Logotipo INRIA.....	12		
Figura 2 - Minerva, o símbolo da Escola Politécnica da USP.....	13		
Figura 3 - Phantom 3D, antigamente utilizado para a manipulação 3D	14		
Figura 4 - SpaceNavigator.....	15		
Figura 5 - Graus de liberdade do SpaceNavigator	15		
Figura 6 - Idealização HandNavigator	15		
Figura 7 - Controle de abertura/fechamento de um dedo é feito a partir.....	16		
Figura 8 - Retorno semi-háptico	20		
Figura 9 - Touchpad	25		
Figura 10 - Mouse comum.....	25		
Figura 11 – Em vermelho, locais onde o arrasto é controlado por velocidade em alguns touchpads com esta funcionalidade.....	26		
Figura 12 - Localização do PointingStick	26		
Figura 13 – Captura sem contato	35		
Figura 14 – Captura mecânica	35		
Figura 15 – Captura mecânica na “luva de dados”.....	35		
Figura 16 - Rutgers Hand Master	36		
Figura 17 - CyberGrasp.....	36		
Figura 18 - Spidar.....	36		
Figura 19 - Protótipos V1 e V2	36		
Figura 20 - Detalhe lâmina deformável	36		
Figura 21 - HandNavigator V1 em utilização	37		
Figuras 22 , 23 e 24 - Forma de mouse, forma semelhante ao SpaceNavigator e forma de " flor", respectivamente	38		
Figura 26 - Protótipo V3a	Figura 27 - Protótipo V3b	Figura 28 - Protótipo V3c.....	39
Figura 25 - Falta de simetria da solução floral.....			39
Figura 29 - Sensor de Pressão.....			40
Figura 30 - Touchpad			40
Figura 31 - Trackball			40
Figura 32 - Scrollpad			40

Figura 33 - Alavanca mono-eixo.....	41
Figura 34 - Forma de mouse.....	50
Figura 35 - Forma do SpaceNavigator ou semelhante.....	51
Figura 36 - Não cancelamento de forças na força de "flor" mais esgonômica (reposicionamento dedão).....	52
Figura 37 - Cancelamento de forças na forma de "flor" não ergonômica (V2)	52
Figura 38 – Vetor $p(19)$ representa os vértices da mão	70
Figura 39 – Variáveis auxiliares para cálculo de molde(19)	71
Figura 40 - Protótipo construído	76
Figura 41 - Projeto 3D em CATIA do protótipo.....	76
Figura 42 – Protótipo final conectado ao circuito.....	77
Figura 43 – Tela final do programa simplificado.....	78
Figura 44 – Programa simplificado em utilização	79
Figura 45 – Protótipo sendo utilizado com programa antigo	80

Índice de Diagramas

Diagrama 1 - Feedbacks sensoriais	18
Diagrama 2 – Exemplo do primeiro tipo de retorno semi-háptico (com utilização de resistência mecânica).....	21
Diagrama 3 – Exemplo de segundo tipo de retorno semi-háptico (com utilização da vibração).....	22
Diagrama 4 - Esquema genérico de captação de um sensor.....	23
Diagrama 6 - Esquema de funcionamento do sensor de pressão.....	24
Diagrama 5 - Exemplo de diferença entre grandeza medida (tempo de retorno do laser) e inferida (distância), para o medidor de distâncias à laser DLE 50 - Bosch...	24
Diagrama 7 - Esquema de montagem de sensores em pares.	31
Diagrama 8 - Esquema de montagem de uma possível solução do HandNavigator.	54
Diagrama 9 - Desenho ilustrativo da idéia escolhida, com apoio de braço incluso. ..	57
Diagrama 10 - Os sensores disponibilizados pelo INRIA	59
Diagrama 11 - Funcionamento de circuito do ScrollPad	64
Diagrama 12 – Modelo plano da mão virtual na tela do computador	65
Diagrama 13 - Representação do dobramento parcial ou total de alguns dedos no modelo da mão virtual	66
Diagrama 14 – Representação das translações nos eixos x (positivo), y (positivo) e z (positivo e negativo)	66
Diagrama 15 - Representação das rotações nos eixos x (positivo), y (positivo) e z (positivo).....	66
Diagrama 16 - Representação da rotação no eixo x, exibindo a diferença entre as cores	67
Diagrama 17 - Diagramas que serão apresentados na tela do programa.....	67
Diagrama 18 – Diagrama de funções que serão utilizadas no programa (simplificado)	73
Diagrama 19 - Peças do protótipo feitas em CATIA v5r19 e a ordem de montagem do protótipo final.....	74
Diagrama 20 – Circuito final indicando portas da placa de aquisição NI-USB6009 a serem utilizadas	75

Índice de Gráficos

Gráfico 1 - Resultados Teste 1	81
Gráfico 2 – Resultados Teste 2	82
Gráfico 3 – Pesquisa opinião sobre o HandNavigator.....	83

Índice de Tabelas

Tabela 1 - Comparação das soluções existentes.....	45
Tabela 2 - Matriz de decisão para forma do periférico	56
Tabela 3 - Matriz de decisão para o material de construção do protótipo	58
Tabela 4 - Matriz de decisões para escolha do sensor	61

1. Introdução

1.1. Intercâmbio, Início do projeto na equipe EVASION do INRIA e continuação do projeto no Brasil na Escola Politécnica da USP

Em 2008 fui aprovado em um Intercâmbio entre a Escola Politécnica da USP e o Instituto Politécnico de Grenoble. Cursando o 3º ano da graduação de Engenharia de Automação nesta instituição, o plano seria de fazer seis meses de aula seguidos de seis meses de Trabalho de Fim de Estudos (TFE, *Travail de Fin d'Études*).

Executei meu TFE na equipe EVASION do Instituto Nacional de Pesquisas em Informática e Automação (INRIA, *Institute Nationale de Recherche en Informatique et Automatique*). A equipe EVASION



Figura 1 – Logotipo INRIA

foi criada em 1º de Janeiro de 2003. Os trabalhos de pesquisa desta equipe são relacionados a modelagem, animação e visualização de objetos e de fenômenos naturais. Para isto, dois grandes eixos de pesquisa são privilegiados: primeiramente, o desenvolvimento de ferramentas fundamentais destinadas a especificação de cenas e de objetos naturais complexos, o que inclui o ajuste de modelos alternativos para a forma, movimento e aparência, assim como a concepção de algoritmos que se baseiam no nível de detalhes adaptativos para melhor promover sua complexidade; de outra parte, a validação destas ferramentas sobre cenas naturais específicas, que vão desde o mundo mineral (e.g. oceano, lava, avalanches, nuvens) até o mundo animal (e.g. simulação de órgãos, partes do corpo e mesmo o cabelo de personagens, movimentos de animais), passando também por cenas vegetais (e.g. plantas, árvores).^[1]

Foi desenvolvido o projeto HandNavigator. O projeto já se encontrava iniciado pelos trabalhos de Paul Kry, Adeline Pihuit e Adrien Bernhardt^[2]. Portanto, já existiam dois protótipos finalizados e prontos para uso, assim como uma interface gráfica para manipulação de uma mão virtual que aparecia na tela de um computador. Durante meus seis meses de projeto, foram desenvolvidos mais três

protótipos. Após um convite para continuar o desenvolvimento deste trabalho na equipe EVASION, junto com a aprovação do Instituto Nacional de Pesquisa de Grenoble e da Escola Politécnica da Universidade de São Paulo, responsáveis pelo Intercâmbio, os trabalhos foram estendidos por mais seis meses, entre os meses de Julho e Dezembro de 2009, na França.

Após o término do período de extensão do Intercâmbio, foi proposta a continuação do tema do TFE francês como Projeto de Formatura da Escola Politécnica da Universidade de São Paulo.

A Escola Politécnica da Universidade de São Paulo foi criada em 1893 e foi incorporada a USP em 1934. Possui 15 departamentos de ensino e pesquisa e conta com mais de quatro mil alunos em graduação. O departamento de engenharia mecatrônica implantou o primeiro curso do tipo no país. As atividades do PMR estão relacionadas aos setores de infra-estrutura, industriais e de serviços, principalmente no que diz respeito aos desafios

tecnológicos da automação, da produtividade e da qualidade. O PMR tem cerca de 40 professores abrangendo as áreas de pesquisa em robótica, sensores e atuadores, além da análise de sistemas mecatrônicos, projeto de máquinas e processos de fabricação incluindo manufatura assistida por computador e mecânica de precisão. O PMR é resultado de uma ação pioneira da Escola Politécnica da USP e é uma referência não apenas nacional, mas reconhecido pelas suas realizações também nos meios técnicos e científicos internacionais. [3]



Figura 2 - Minerva, o símbolo da Escola Politécnica da USP

1.2. Motivação

O projeto de periféricos de computador é interessante devido ao fato de juntar três domínios específicos: a mecânica, quando se projeta o periférico em si; a eletrônica, no projeto de aquisição de dados e envio ao computador; e a computação, com a tarefa de interpretação de dados e representação no monitor de um computador. Dentro da mecatrônica, tais domínios se inter-relacionam, por

exemplo, nos sensores, onde há a necessidade de que uma interação mecânica altere grandezas elétricas para gerar medições precisas da interação mecânica; ou ainda na comunicação serial, onde circuitos eletrônicos especiais são necessários para promover a comunicação com o computador, para que este possa analisar e interpretar os dados da interação mecânica.

O fato de computadores atuais ainda não possuírem bom número de periféricos e interfaces que nos permita ter facilidade para lidar com ambientes 3D é como um desafio para este projeto. O SpaceNavigator é um periférico exemplar neste sentido: há muito tempo desejava-se ter a facilidade da manipulação 3D de objetos. A utilização dos antigos Phantom's (Figura 3), solução extensamente



Figura 3 - Phantom 3D, antigamente utilizado para a manipulação 3D

encontrada nos dias de hoje, não é exemplar uma vez que são de difícil utilização devido a seu controle de movimentos por posição, onde há a necessidade de mover mão real pelo espaço e sem apoio para se executar o controle. No SpaceNavigator, o controle é feito por forças, que impõe a forma de **variação de posição** da mão virtual, não havendo a necessidade, portanto, de deslocamentos da mão pelo espaço sem o devido apoio. Este fato gera conforto e facilidade de utilização para o SpaceNavigator.

É fato que seria um grande sonho da computação gráfica possuir a facilidade da utilização de nossas mãos em ambientes virtuais ^[2]. A motivação para o projeto é de que a utilização de nossas mãos em ambientes virtuais é algo desejado, porém soluções disponíveis não são comumente vendidas devido ao fato de possuírem alto custo, apresentarem falhas de projeto ou serem de difícil utilização ^[2]. Deseja-se desenvolver algo simples, útil, com manipulação fácil e baixo tempo para início de utilização (nulo, se possível). Pode-se notar que estas características são as principais nos dispositivos atualmente utilizados, e amplamente conhecidos, que interagem com a interface humana (i.e. teclado, mouse, joystick etc.).

1.3. O SpaceNavigator e a idealização do HandNavigator

O SpaceNavigator é fabricado pela empresa 3DConnexion. Ele pode ser visualizado na Figura 4. Ele permite o controle de 6 graus de liberdade (3 rotações e 3 translações) de um objeto virtual (Figura 5).

Após a constatação da facilidade de utilização do SpaceNavigator para o controle da posição e orientação uma mão no espaço, houve a idéia de se desenvolver um periférico que pudesse aumentar as capacidades de controle de uma mão virtual, estendendo tal controle também aos dedos. O projeto inicialmente idealizado é mostrado na Figura 6.



Figura 4 - SpaceNavigator

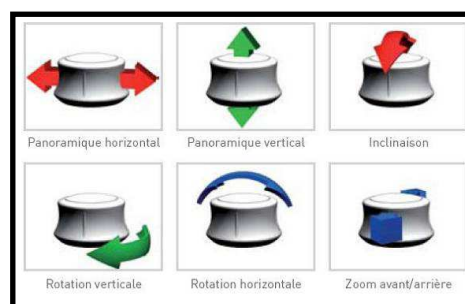


Figura 5 - Graus de liberdade do SpaceNavigator

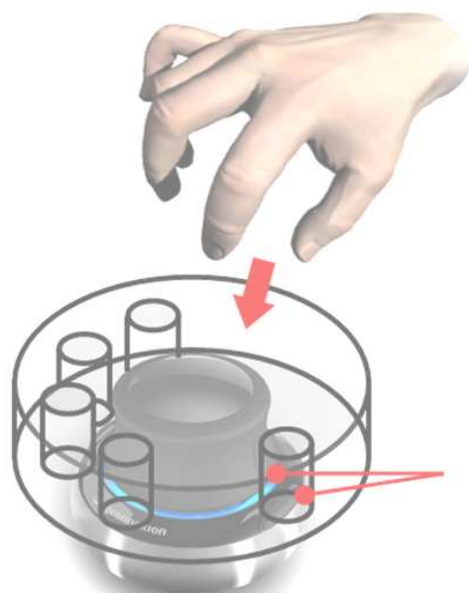


Figura 6 - Idealização HandNavigator

Apelidou-se tal periférico de “HandNavigator”. O nome HandNavigator é uma derivação de SpaceNavigator, um de seus componentes, em uma combinação com a palavra “hand” (mão, em inglês).

A idéia é, com a ajuda de sensores, de se tentar estender o controle da posição e da orientação da mão ao controle de fechamento e abertura dos dedos, mantendo-se a facilidade de para a manipulação de uma mão virtual. É

necessário considerar que a inserção do controle de novos graus de liberdade em um dispositivo extremamente sensível como o SpaceNavigator é complicado. É necessário um projeto que considere a separação do acionamento dos graus de liberdade.

O HandNavigator busca o controle simplificado, porém quase completo, de uma mão virtual na tela de um computador. A mão virtual é modelo virtual de uma mão real e que simula todos seus graus de liberdade. O HandNavigator possui controle quase completo dos graus de liberdade pois pode controlar uma boa parte deles. O HandNavigator pode ser considerado simplificado pelo fato de que o controle de alguns conjuntos de graus de liberdade diferentes são vinculados (e.g. no fechamento de um dedo, como mostra a Figura 7, um grau de liberdade de controle do HandNavigator controla o fechamento de 3 articulações, de maneira coordenada).

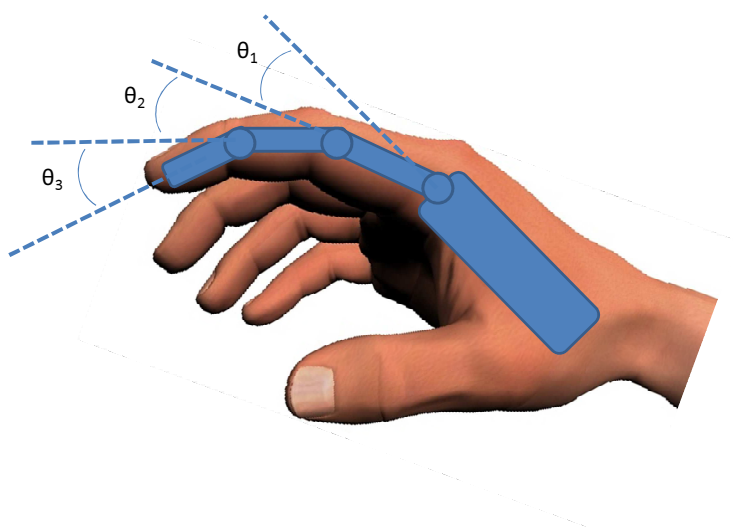


Figura 7 - Controle de abertura/fechamento de um dedo é feito a partir de um grau de liberdade que controla três articulações da mão virtual

É importante citar-se a opção de acoplamento dos controles de movimento dos dedos mínimo e anelar. Assim, o dedo real anelar controla dois os dedos virtuais (mínimo e anelar). Isso foi proposto devido a dificuldade observada na movimentação precisa do dedo mínimo real para controlar o dedo mínimo virtual e também após a verificação de que os movimentos dos dedos anelar e mínimo são geralmente correspondentes no mundo real [2]. Isso causa uma simplificação do dispositivo e de seu manuseio, e conseqüentemente uma diminuição de custos.

As simplificações são importantes para a controlabilidade do dispositivo. Isso justifica a opção de controle da abertura e fechamento dos dedos através de um grau de liberdade, além de justificar também a opção de acoplagem. Em 1.6 -

Problemas das soluções existentes para captura do movimento das mãos, se avaliarão os problemas dos dispositivos existentes, se constatando a dificuldade para o controle independente dos graus de liberdade propostos. Neste momento, limita-se a se justificar tais decisões a partir da verificação de que grande parte dos movimentos e gestos que fazemos com as mãos no dia a dia utilizam essencialmente o dedo polegar, o dedo indicador e o dedo médio, dedos com os quais se tem mais controle.

Na seção 1.4 - *Revisão bibliográfica* a seguir, será avaliada uma classificação para os dispositivos existentes segundo seus tipos de feedback ou de sensor, e depois será feita uma análise das soluções já existentes para o controle de uma mão virtual, incluindo versões anteriores do HandNavigator. Tal seção será útil para se entender o funcionamento do HandNavigator e, mais a frente neste trabalho, justificar as opções feitas para o novo dispositivo a ser desenvolvido.

1.4. Revisão bibliográfica

1.4.1. Classificação dos dispositivos segundo o tipo de feedback

Já existem soluções que permitem o controle de uma mão virtual. O termo “solução” engloba tanto um periférico (que controla) e quanto um programa de simulação virtual de um ambiente virtual que apresente uma mão virtual (que é controlada). Deseja-se analisar o tipo de feedback das soluções existentes. Inicia-se com uma breve introdução sobre tipos de feedback mais utilizados.

O retorno óptico via monitor do computador, para o projeto do HandNavigator, é essencial, pois a intenção é de controlar uma mão virtual na tela do computador. Existem também outros tipos de feedback óptico: na manipulação de uma mão robótica, por exemplo, onde o próprio utilizador vê o que faz, há um retorno óptico direto (sem monitor); no caso de manipulação de instrumentos em uma cirurgia, temos câmeras que proporcionam um retorno óptico que auxilia o controle por parte do médico. No caso do programa antigo que se utilizava com o HandNavigator, imagens 3D que simulam uma mão virtual, sua posição e orientação no meio são

mostradas no monitor de um computador, além da posição dos dedos (aberto, fechado e posições intermediárias). O retorno óptico é sempre utilizado, em uma ou mais de suas diversas formas.

Existe, porém, outros tipos de feedback que poderiam ser implementados em um projeto de controle de uma mão virtual. Como o controle é feito utilizando-se a mão para controlar objetos em sua interação virtual (de forças e toques), projetos que consideram o retorno háptico (i.e. via tato) são importantes. Sendo assim, temos o seguinte esquema de feedback para um utilizador de um possível dispositivo que controla uma mão virtual na tela do computador no Diagrama 1 abaixo:

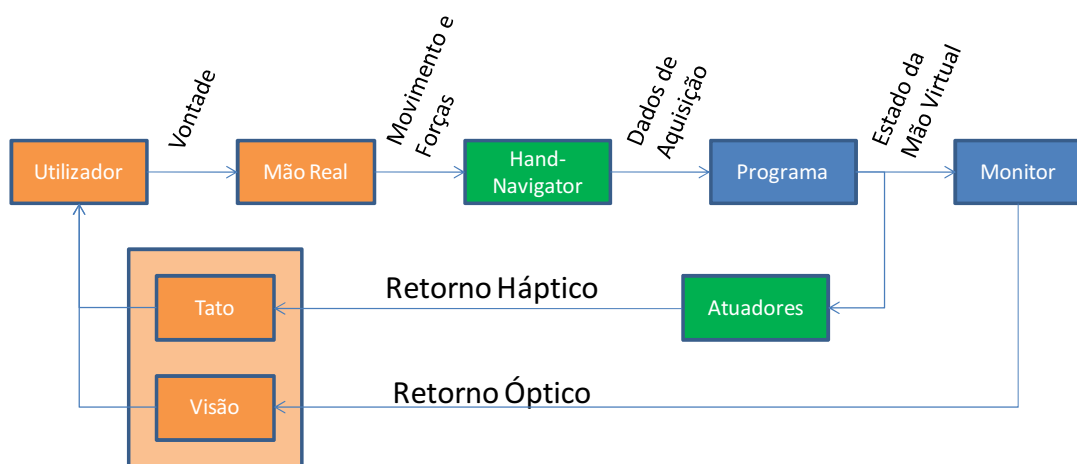


Diagrama 1 - Feedbacks sensoriais

No Diagrama 1, acima, tem-se três cores:

- **Laranja:** parte do corpo ou sentido humano, referentes ao utilizador;
- **Verde:** componentes pertencentes ao periférico;
- **Azul:** componente ou programa, pertencentes ao computador.

A seguir, se apresentará uma classificação dos dispositivos existentes de acordo com seu tipo de retorno. Vale a pena citar do objetivo inicial do HandNavigator e outras soluções análogas: um dispositivo periférico de computador que permita, primeiramente, controlar uma mão virtual apresentada em um ambiente virtual que é representado na tela de um computador para que se possa posteriormente interagir com objetos virtuais neste ambiente virtual. Em poucas palavras, o objetivo deste projeto é desenvolver uma solução que possa controlar

uma mão virtual apresentada na tela de um computador. Esse objetivo será melhor especificado na seção 2 - *Objetivos*.

O feedback háptico, no sentido que se quer avaliar os dispositivos, é portanto aquele que permite sentir quando se toca ou quando se segura um objeto, assim como avaliar a força exercida. O feedback háptico poderia ainda incluir as sensações de atrito, textura, rugosidade, liquidez, temperatura, entre outras, porém o objetivos macro deste projeto direcionam as sensações hápticas que se quer analisar neste momento, permitindo uma classificação e uma análise dos dispositivos existentes.

Um dispositivo que controla forças virtuais e toques virtuais na tela do computador deve, portanto, ter seu feedback háptico avaliado com relação as sensações de forças e de toque proporcionados no mundo real. Esta seção trata deste assunto.

Quanto ao retorno háptico, pode-se inicialmente dividir os dispositivos de controle de uma mão virtual em duas categorias ^[2]:

- **Dispositivos sem retorno háptico:** dispositivos que não possuem retorno das sensações hápticas virtuais, ou seja, o utilizador não dá conta de que há uma manipulação virtual apenas pelo uso de seu tato. O feedback é através da visão (óptico), apenas.
- **Dispositivos com retorno háptico:** dispositivos que possuem retorno de forças. Este retorno é feito através de **forças reais**, aplicadas aos dedos, que são **proporcionais** as forças virtuais da manipulação virtual. Geralmente a solução se torna cara devido à necessidade de atuadores mecânicos de força que tenham boa precisão e controlabilidade. O retorno háptico permite melhor aproveitar as capacidades do cérebro para tratar informações de contato e força exercida.

Existe, porém, uma terceira classe de dispositivos que procuram simular o retorno háptico. Esses são os chamados dispositivos semi-hápticos:

- **Dispositivos com retorno semi-háptico:** dispositivos com retorno através de **força análoga ou outras grandezas** (e.g vibração). Mesmo que tal retorno seja muitas vezes de excelente qualidade e tenha capacidade de aumentar o nível de interação do usuário com o mundo virtual, sua implementação pode se tornar cara ou complicada, dependendo do conceito utilizado. A intenção, porém, é baixar o custo em relação a uma solução háptica, porém obter um melhor desempenho do que uma solução sem retorno háptico [2].

O dispositivo semi-háptico pode ser considerado um dispositivo com uma melhora em relação ao dispositivo sem feedback háptico [5]. É fato que o dispositivo háptico costuma ter dispositivos complexos, com custos elevados, e mesmo assim ainda é difícil a simulação da percepção háptica [2]. O dispositivo semi-háptico com controle por velocidade surge como uma solução mais simples, porém possivelmente efetiva.

O dispositivo semi-háptico tenta enganar as sensações proprioceptivas do utilizador [6] ao utilizar um intermediário [7][8] (e.g. uma esponja, pequena bola, vibradores etc), como mostra a Figura 8.

O modo mais comum de se obter um feedback semi-háptico é utilizar um intermediário como um material que gere resistência elástica ao movimento da mão real (i.e. esponja, espuma etc), junto a um sensor de pressão. Nesse caso, a mão real (ou seus dedos), ao tentar acionar os sensores (pressionando-os), avaliará melhor a força que aplica devido ao fato de que o intermediário estará em compressão e exercendo forças resistivas sobre a mão real (ou o dedo real). Tal força de resistência não será necessariamente proporcional as forças que a mão virtual recebe virtualmente e existirá sempre, independente da mão virtual tocar ou não um objeto virtual (este seria o caso de um dispositivo háptico).



Figura 8 - Retorno semi-háptico

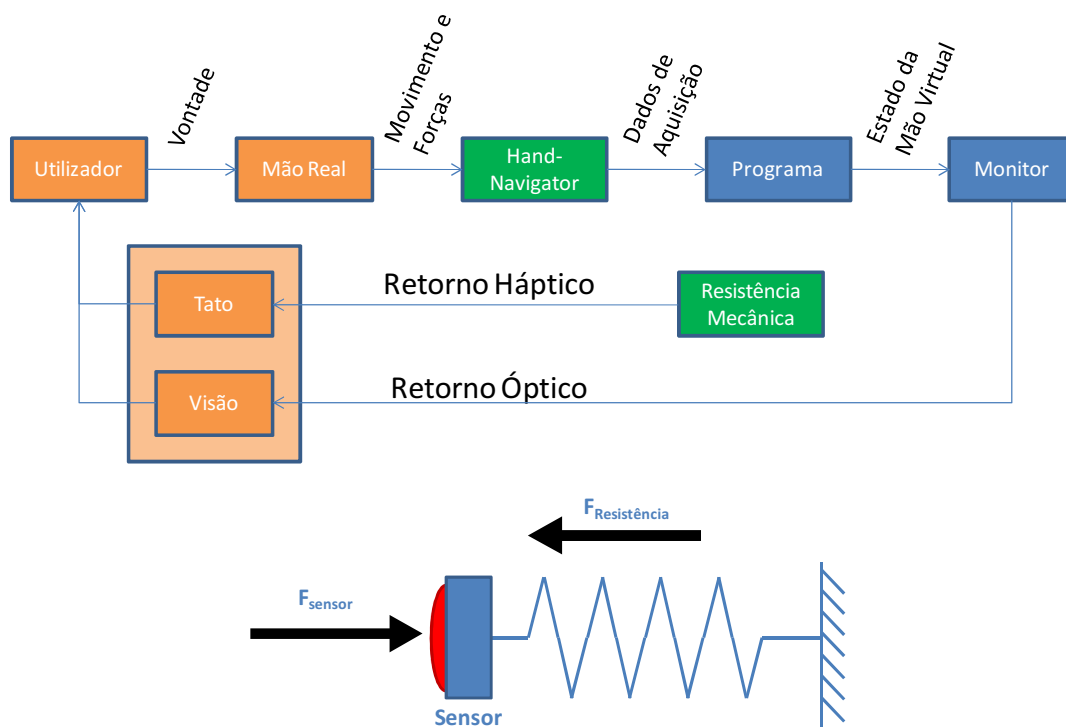


Diagrama 2 – Exemplo do primeiro tipo de retorno semi-háptico (com utilização de resistência mecânica)

Pode-se, portanto, definir o primeiro tipo de retorno semi-háptico (Diagrama 2) como aquele que tenta simular um tipo de sensação háptica virtual através do **mesmo tipo de sensações háptica** no mundo real. O mesmo tipo de sensação, no caso acima, seriam as forças. As forças virtuais de resistência ao movimento são simuladas pela força real de resistência à ação da força que é aplicada sobre o sensor. A força reativa é proporcionada pelo intermediário, representado como uma mola na figura inferior do Diagrama 2. Há uma não-proporcionalidade entre as forças reais e virtuais, e isto é devido ao fato de que o dispositivo não possui atuadores que recebem sinais do computador (i.e. da interação virtual), como se verifica na figura superior do Diagrama 2. Há, porém, um componente que gera resistência ao possuir um deslocamento imposto, servindo para a simulação das forças resistivas virtuais no mundo real.

Outro modo bastante utilizado em vídeo-games para implementar o retorno semi-háptico é a simulação das sensações hápticas com a utilização da vibração. A vibração poderia indicar diferentes situações: vibração sinalizando que houve algum tipo de toque no mundo virtual, ou vibração proporcional a força aplicada, entre

outros. Para o HandNavigator, este tipo de retorno tem a necessidade de um atuador que seja acionado pelo computador em uma situação específica, por exemplo quando há um toque em um objeto, ou quando se exerce uma força em outro objeto. É importante considerar tal dispositivo com sendo semi-háptico, uma vez que são vibrações simulando sensações hápticas de toque e de força. Analogamente, e como fonte de exemplo, pode-se citar o dispositivo de retorno vibratório considerado háptico devido ao fato de tentar simular sensações relativas ao atrito ou rugosidade, por exemplo, pois estes geram, entre outras sensações, vibração em uma mão ou dedo real.

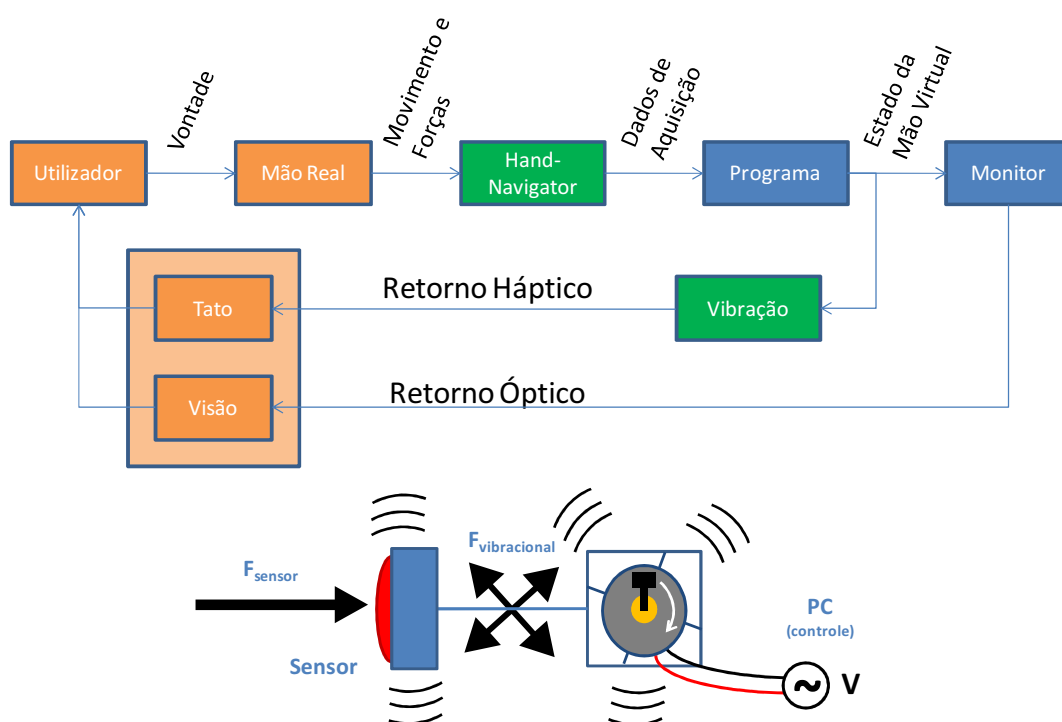


Diagrama 3 – Exemplo de segundo tipo de retorno semi-háptico (com utilização da vibração)

Pode-se definir, portanto, um segundo tipo de retorno semi-háptico como aquele que tenta simular um tipo de sensação háptica virtual através de **outro tipo de sensação háptica** no mundo real. No caso acima ilustrado no Diagrama 3, a sensação virtual seria a força ou o toque, enquanto a sensação real que simula tal força virtual é a vibração.

Antes de finalizar a seção é importante dizer que os diferentes tipos de feedback háptico ou semi-háptico podem ser aplicados tanto para sensação dos

dedos quanto para sensação na mão, e podem ser utilizados mais de um ponto (e.g. diversos dedos, diversos pontos diferentes da mão) para melhorar intensificar ainda mais a imersão virtual. Além disso, diferentes tipos de feedback podem ser utilizados em um mesmo projeto (e.g. háptico para as mãos e semi-háptico para os dedos).

1.4.2. Classificação dos sensores segundo o tipo de controle

Um sensor pode ser utilizado para controlar um movimento virtual (e.g. de uma mão ou de um dedo) de diversas maneiras, entre as quais as mais comuns são controle por posição, velocidade, aceleração ou força. No HandNavigator, enquanto o SpaceNavigator será utilizado para controle da mão virtual, sensores serão utilizados para o controle dos dedos virtuais.

Esta seção analisa os diferentes tipos de funcionamento para o controle dos movimentos virtuais a partir dos sensores. Considere uma grandeza virtual sendo controlada por um sensor, o qual recebe a imposição de uma grandeza física no mundo real.

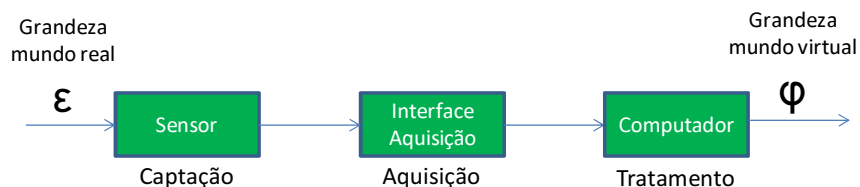


Diagrama 4 - Esquema genérico de captação de um sensor

Seja ε a grandeza real física que um sensor **infere**. Deve-se diferenciar grandeza medida de grandeza inferida: um sensor pode medir uma grandeza qualquer, porém com a intenção de inferir sobre outra. A idéia é poder calcular a grandeza que se quer inferir em função da grandeza medida. Para periféricos cujo controle se baseia na manipulação física, a grandeza ε é ou um **posição** ou uma **força**. No caso do HandNavigator, a grandeza ε é o deslocamento imposto no sensor ou a força exercida pelo dedo real no sensor.

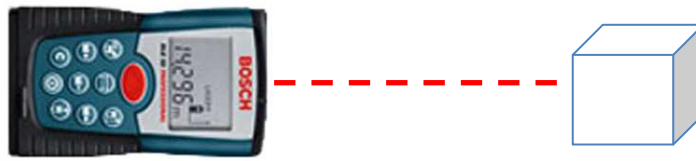
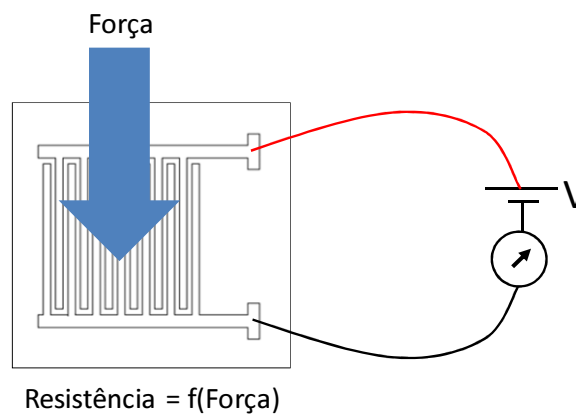


Diagrama 5 - Exemplo de diferença entre grandeza medida (tempo de retorno do laser) e inferida (distância), para o medidor de distâncias à laser DLE 50 - Bosch

Um exemplo da diferença entre grandezas medidas e inferidas pode ser visto no sensor de pressão (exemplo no Diagrama 6). Nele, o valor de uma resistência e a corrente que ela gera é responsável para inferir sobre a pressão exercida no sensor. Conseqüentemente, a partir da área onde a força que gera tal pressão é aplicada (área do sensor), pode-se inferir sobre a força exercida. Neste exemplo, inferiu-se sobre a força através da medida corrente gerada a partir da resistência. Uma montagem de circuito similar poderia permitir a medida da resistência a partir de um potencial elétrico (circuito bastante utilizado em circuitos para placas de aquisição). Como exemplo, outras grandezas medidas poderiam ser capacitância, corrente elétrica, campo elétrico, campo magnético etc.



**Diagrama 6 - Esquema de funcionamento do sensor de pressão
(ϵ = Força, inferida através da medição da corrente)**

Deseja-se controlar um movimento no mundo virtual. Portanto, a grandeza controlada ϕ vai controlar a movimentação. Esse movimento pode ser produzido a partir da imposição de uma aceleração, de uma velocidade ou de um deslocamento. Esses três casos definem, respectivamente, os controles por aceleração, velocidade

ou posição, que seguem abaixo descritos. O caso de controle por força é um caso especial do controle por aceleração e será especificado mais a frente.

O controle por posição é aquele no qual um $\Delta\epsilon$ imposto no sensor impõe um **deslocamento** virtual, $\Delta x_{\text{virtual}}$, em nosso caso, do dedo virtual. Assim, φ é tratado como uma posição virtual, x_{virtual} . Como exemplo, poderíamos citar o mouse de computador (Figura 10), onde $\Delta\epsilon$ é o deslocamento Δx_{real} do mouse, que causa um deslocamento $\Delta x_{\text{virtual}}$ no ponteiro do mouse na tela do computador. Analogamente, no touchpad de computadores portáteis (Figura 9), um Δx_{real} imposto por um dedo sobre o sensor gera um deslocamento $\Delta x_{\text{virtual}}$ do ponteiro do mouse sobre a tela.



Figura 10 - Mouse comum



Figura 9 - Touchpad

O controle por velocidade é aquele no qual a mudança de ϵ impõe uma nova **velocidade** virtual, v_{virtual} , em nosso caso, na movimentação do dedo virtual. Assim, φ é tratado como uma velocidade v_{virtual} .

Como exemplo, poderíamos citar as novas funções de controle de touchpads de alguns laptops mais atuais, quando utilizados para a tarefa de arrastar itens na tela do computador (e.g. um arquivo entre duas pastas). Estas funções de controle são utilizadas quando o dedo que se move sobre o touchpad físico chega ao limite do sensor (i.e. próximo a borda do sensor) durante uma operação de arraste (i.e. movimento com clique). Tais funções, neste caso, são ativadas, e uma velocidade é imposta ao mouse para evitar que a operação de arrasto do item se cancele. É importante notar que caso tais funções não estivessem implementadas, que é o caso da maioria dos laptops, seria necessária a retirada do dedo do sensor, seguida de reposição do dedo para clique e continuação da operação. Neste exemplo, a

velocidade v_{virtual} do ponteiro do mouse é controlada pela posição x_{real} de nosso dedo sobre o touchpad. Podemos ver o local dos touchpads com esta funcionalidade onde o controle do arrasto é feito por velocidade na Figura 11.

Outro exemplo de periférico que utiliza o controle por velocidade é o PointingStick (Figura 12). Comum em antigos computadores portáteis da marca IBM para controle do mouse, ele era utilizado como alternativo ao uso touchpad^[9]. Ele é um dispositivo cuja grandeza ε que o sensor capta é uma pressão real, P_{real} , exercida pelo dedo real sobre o PointingStick, na direção desejada. A P_{real} é a responsável pelo controle da velocidade v_{virtual} e da direção da movimentação do ponteiro do mouse na tela do computador.

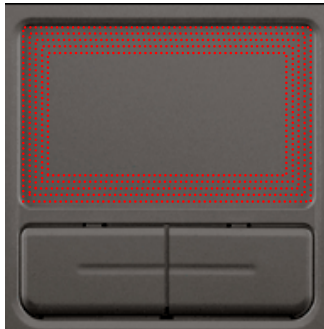


Figura 11 – Em vermelho, locais onde o arrasto é controlado por velocidade em alguns touchpads com esta funcionalidade



Figura 12 - Localização do PointingStick nos antigos PCs da IBM

O controle por aceleração é aquele no qual a mudança de ε impõe uma nova **aceleração** virtual α_{virtual} , em nosso caso, na movimentação do dedo virtual. Assim, φ é tratado como uma aceleração α_{virtual} .

Há um caso ainda a se considerar, do **controle por força**. O controle por força é um caso especial do controle por aceleração. Neste caso, a aceleração α_{virtual} imposta a um objeto virtual é inversamente proporcional a sua massa virtual. Assim, ao exercer uma mesma imposição ε , dois objetos virtuais diferentes poderiam obter uma aceleração virtual diferente, em função de seu tamanho e composição no mundo virtual.

Em resumo, pode-se dizer que se tem ε como uma grandeza que é responsável por controlar, que pode ser uma força ou uma posição. Tem-se φ como

grandeza controlada, que em nossa aplicação pode ser tratada como uma posição, velocidade ou aceleração.

1.4.3. Movimentação no espaço virtual e interação com objetos

(i) Movimentação no espaço virtual

Pode-se modelar um movimento de um corpo virtual (i.e. variação de sua posição) através da imposição de uma velocidade. A captação de uma grandeza $\varepsilon = \varepsilon^*$ pode fazer com que este deslocamento seja com velocidade constante ou variável de acordo com ε^* .

. Consideremos um corpo virtual **parado** em uma posição inicial $x_{0_virtual}$. Deseja-se movimentá-lo. No caso deste trabalho, tal corpo seria um dedo virtual, porém se omitirá a legenda $_{virtual}$ para simplificação da demonstração dos cálculos.

No caso de controle por velocidade, uma determinada grandeza ε^* impõe que a velocidade seja constante. No caso de controle por aceleração ou força, uma determinada grandeza ε^* impõe que a variação da velocidade seja de acordo com uma aceleração constante. Teoricamente, controles a partir de derivadas maiores da posição poderiam ser implementados. Seja $x^{(n)}$ a n-ésima derivada da posição x , e $x^{(n)}_0$ seu valor inicial. Como o corpo está inicialmente parado, todas as derivadas da posição o devem ser nulas, enquanto x_0 é fixado (imposição). Nestes casos, temos:

Controle por velocidade

- $x = x_0 + x^{(1)}t$
- $x^{(1)} = v_0 = \varepsilon^*$
- Posição inicial: x_0
- Imposição: não há $\Rightarrow x = x_0 + x^{(1)}t$

Controle por aceleração

- $x = x_0 + x^{(1)}_0t + x^{(2)}t^2/2$
- $x^{(2)} = a = \varepsilon^*$
- Posição inicial: x_0

- Imposição: $x^{(1)}_0 = 0 \Rightarrow x = x_0 + x^{(2)}t^2/2$

Controles por derivadas mais elevadas

- $x = x_0 + x^{(1)}_0 t + x^{(2)}_0 t^2/2 + x^{(3)}_0 t^3/3 + \dots + x^{(n-1)}_0 t^{(n-1)}/(n-1) + x^{(n)}_0 t^n/n$
- $x^{(n)} = \varepsilon^*$
- Posição inicial: x_0
- Imposição: $x^{(1)}_0 = x^{(2)}_0 = x^{(3)}_0 = \dots = x^{(n-1)}_0 = 0 \Rightarrow x = x_0 + x^{(n)}_0 t^n/n$

Comentou-se sobre o caso de como é feito o controle por aceleração e por velocidade devido ao fato que, uma vez que quando uma velocidade ou aceleração é imposta a um corpo, não há problemas de continuidade no movimento, bastando apenas que as imposições acima sejam satisfeitas.

Por outro lado, não se pode impor uma nova posição a um corpo sem levar em conta questões de continuidade do movimento (i.e. o corpo não pode sumir de um lugar e aparecer no outro), pois procura-se simular coisas verossímeis com o mundo real.

É devido a este motivo que no controle por posição a análise do movimento resultante a imposição de uma nova posição é mais complexa. Existem dois casos a se analisar:

- 1) **Caso a nova posição for imposta de maneira progressiva**, não haverá problemas quanto à continuidade do movimento. Esse é o caso onde o controlador impõe todas as posições intermediárias possíveis entre x_0 e x_{final} , que são captadas pelo sensor e passadas ao programa, que representa todos os estados na tela do computador.
- 2) **Caso a nova posição for imposta de maneira não progressiva**, existiriam diferentes maneiras de tratar o fato. Uma delas seria a imposição de uma velocidade (ou mesmo aceleração) para se transitar entre a posição inicial x_0 e a posição x_{final} , uma vez que assim se passaria por todas as posições intermediárias, gerando maior continuidade ao

movimento. Outra seria a implementação de um controle em posição relativa (como ocorre no controle tradicional dos MousePads).

A determinação sobre como será imposta a nova posição não é definida pelo usuário ou pela programação, mas sim pelo modo de funcionamento do sensor. Existem sensores nos quais é fisicamente impossível a imposição de Δx de maneira não progressiva, como no sensor de volume de um rádio por exemplo. Analogamente, existem sensores nos quais a imposição de uma nova posição absoluta é sempre independente da imediatamente anterior, como no MousePad, por exemplo, sem que isso signifique necessariamente que o controle do movimento do ponteiro do mouse seja absoluto.

(ii) Considerações sobre desacionamento de sensores que captam posição

No HandNavigator, define-se desacionamento de um sensor como a ação de retirar o dedo do sensor (não há toque e nem imposição de forças). Neste caso, por definição de objetivos, deseja-se o corpo controlado esteja **parado**. Pretende-se analisar o caso onde a grandeza ϵ é uma posição.

Uma posição deve ser mantida constante após o desacionamento (i.e. o corpo virtual controlado deve estar parado). Quando o ato de desacionamento do sensor não altera a posição captada por ele, tem-se o caso ideal, pois a condição de continuidade e a condição de desacionamento (corpo parado) estão impostos automaticamente pelos dados capturados pelo sensor. Ao se impor uma nova posição x_{imposto} no sensor, o deslocamento é $\Delta x_{\text{real}} = x_{\text{imposto}} - x_{\text{base}}$, onde x_{base} é sempre o mesmo e o movimento é contínuo. O movimento e seu controle podem ser ditos **absolutos**, pois a base é constante e independente do desacionamento.

Há, porém, o caso onde o desacionamento, em sensores de posição, é caracterizado pela aquisição de posição nula. Nestes casos, um algoritmo de computador pode se responsabilizar por inferir sobre a última posição $x_{\text{desacionamento}}$ antes do desacionamento. Sempre que se deseja voltar a acionar o sensor, como o mesmo não se encontra mais na sua posição antes de paragem, um programa de computador deverá ser responsável por avaliar que, após se sair da posição nula $x =$

0 (sensor não acionado), a primeira posição x^* adquirida no sensor será a nova base x_{base} para o movimento relativo do sensor, e portanto equivalente ao antigo $x_{desacionamento}$. Desta forma, tem-se que $\Delta x_{real} = x_{imposto} - x_{base}$, onde $x_{base} = x^*$. O movimento e seu controle podem ser ditos **relativos**, pois a base é redefinida a cada acionamento do sensor e é definida como sendo a última posição antes do desacionamento.

(iii) Considerações sobre desacionamento de por sensores de força

Lembrando, no caso do HandNavigator, define-se desacionamento de um sensor como quando se retira o dedo do sensor (não há toque e nem imposição de forças). Neste caso, o corpo controlado deve estar **parado**. No caso de ε ser uma força, após o desacionamento ε se anula. A possibilidade de se inferir sobre o valor da força imposta antes do desacionamento do sensor não nos serve a nada, pois realmente se deseja que a força captada e utilizada na simulação virtual seja nula e que a movimentação não ocorra quando o sensor não estiver acionado.

Na situação aqui analisada, supõe-se um sensor de forças que consiga inferir apenas sobre uma direção da grandeza ε do mundo real (i.e. não capta forças negativas, ou seja, aplicadas na direção inversa). Para analisar o controle de um corpo qualquer, tem-se que definir como o objeto virtual controlado irá reagir ao desacionamento do sensor, além de se definir como o corpo se movimentará em direção de sentido oposto. Há dois tipos de solução para este caso:

- 1) **Retorno automático:** Neste tipo de solução, abdica-se da intenção de que o corpo fique parado imediatamente após o desacionamento. Quando houver desacionamento, os dedos voltam a sua posição inicial de acordo com um movimento pré-definido de retorno. Há, portanto, movimento mesmo depois do desacionamento do sensor;
- 2) **Utilização de pares de sensores:** cada um dos sensores serviria ao acionamento para uma direção do movimento (Diagrama 7).

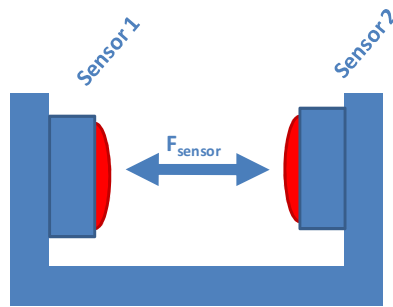


Diagrama 7 - Esquema de montagem de sensores em pares.

Em casos onde as condições descritas acima descritas se manifestam, a utilização de sensores em pares é uma oportunidade de se obter o controle da movimentação em ambas as direções e poder deixar o corpo controlado parado quando há desacionamento, enquanto o retorno automático pode ser uma solução mais barata e simples, onde abdica-se da intenção de que a mão pare imediatamente quando houver desacionamento do sensor, mas que pode porém já ser suficiente para uma aplicação específica.

(iv) Interação com objetos virtuais

É importante avaliar como corpos em contato se relacionam no ambiente virtual, assim como definir o controle. Quando há o contato entre o corpo que se controla e um objeto virtual, há a possibilidade de se continuar com o tipo de controle antigo ou mesmo ou de se impor uma nova maneira de controle (e.g. pode-se ter um objeto virtual cujo controle da movimentação é feito por velocidade e cuja interação com objetos é feita por controle por aceleração). Para variar entre dois tipos de controle, bastaria diferenciar o momento de ocorrência de cada um, definindo que um deles só ocorre quando o corpo controlado está em contato com outros objetos.

Avalia-se abaixo quatro casos principais de interação com objetos: objetos móveis/imóveis que sejam deformáveis/indeformáveis. Os tipos de controle discutido abaixo são os que ocorrem quando há contato entre o corpo controlado e o objeto. A mesclagem dos tipos de controle descritos abaixo durante as fases de contato e não contato é possível e dependeria de uma aplicação específica.

- 1) **Objetos indeformáveis imóveis (barreiras):** Caso o objeto não puder ser movimentado (i.e. uma barreira ao movimento), o programa, ao detectar o

toque, impõe que não é mais possível de se entender o movimento após a barreira.

- 2) **Objetos indeformáveis móveis:** Caso o objeto possa ser movimentado (i.e. um bloco virtual), o programa que simula o ambiente virtual, ao detectar a presença de contato, em geral, o programa tem duas opções:
- a. **Continuação do movimento:** nesse caso, supõe-se que não exista resistência. A movimentação do dedo continua normal, independente da ocorrência do contato, e o objeto se moverá.
 - b. **Imposição de resistência:**
 - i. Controle por posição: a resistência implica que, caso antes do toque, $\Delta\varepsilon = \gamma_1 \Delta x_{\text{virtual}}$, após o toque teremos $\Delta\varepsilon = \gamma_2 \Delta x_{\text{virtual}}$. No caso, γ_1 e γ_2 são constantes de proporcionalidade entre a grandeza imposta no mundo real e a grandeza controlada no mundo virtual, sendo que a resistência é imposta ao fazer-se $\gamma_2 < \gamma_1$.
 - ii. Controle por velocidade: analogamente ao caso anterior, faz-se que a relação a relação entre a grandeza imposta e a grandeza controlada (equivalente a velocidade do ponto de contato), que era $\Delta\varepsilon = \gamma_1 v_{\text{virtual}} t$, se torne $\Delta\varepsilon = \gamma_2 v_{\text{virtual}} t$, com $\gamma_2 < \gamma_1$.
 - iii. Controle por aceleração: analogamente ao casos anteriores, faz-se que a relação a relação entre a grandeza imposta e a grandeza controlada (equivalente a aceleração no ponto de contato), que era $\Delta\varepsilon = \gamma_1 \alpha_{\text{virtual}} t^2 / 2$, se torne $\Delta\varepsilon = \gamma_2 \alpha_{\text{virtual}} t^2 / 2$, onde $\gamma_2 < \gamma_1$. Caso as constantes γ forem inversamente proporcionais as massas virtuais dos objetos virtuais, esse controle pode ser chamado de **controle por força**, e é o tipo de controle mais semelhante a interação que tem-se no mundo real quando se entra em contato com um objeto.
- 3) **Objetos deformáveis imóveis:** Caso o objeto for deformável (e.g. massa de modelagem virtual fixa), assim como no item anterior, pode-se ou não impor resistência ao movimento. No caso de impô-lo, o controle por força é aqui também o tipo de controle mais semelhante a interação que tem-se

no mundo real quando se entra em contato com um objeto. Nesse caso, porém, a força não é mais definida pela massa do objeto em um controle em aceleração (afinal o objeto é imóvel), mas sim por uma constante de deformação virtual k_{virtual} e sua deformação virtual imposta $\Delta x_{\text{virtual}}$ (que pode ser imposta por controle de posição, velocidade ou aceleração). É importante dizer que k_{virtual} varia de acordo com o local de atuação e com a forma instantânea do objeto deformável.

- 4) **Objetos deformáveis móveis:** É importante uma boa definição de como uma força afeta o objeto móvel deformável (e.g. massa de modelagem virtual móvel). O programa possa deve ponderar a ordem grandeza das constantes proporcionais γ , que representam a massa virtual, e k_{virtual} , que representam a deformação virtual, para analisar quando uma força deforma o objeto, quando uma força move o objeto, ou quando ambas as interações coexistem, deformando e movendo o objeto.

Um modo interessante na implementação de diferentes tipos de controle (um para quando há contato e um para quando não há) seria a implementação de dois tipos de sensores: um que controla movimentação (por exemplo, com ε sendo um deslocamento ou velocidade) e outro que controla aplicação de forças em objetos (por exemplo, com ε sendo uma aceleração através de um controle de força).

(v) Implementação dos sensores com diferentes tipos de retorno háptico

Supondo um dispositivo no qual se deseja um retorno háptico de retorno de forças através de atuadores, por exemplo, com a utilização atuadores pneumáticos ou hidráulicos. Há que se observar que é natural a utilização de sensores de pressão, pelo fato de que este tipo de sensor avalia a força com que é pressionado, enquanto o retorno háptico exerce sobre a mão ou sobre os dedos um retorno de uma força também. Não existiria o caso de retorno háptico com retorno através de vibradores, pois isso seria considerado um retorno semi-háptico, como já foi considerado.

Analogamente, o caso de se implementar um retorno semi-háptico com um material deformável seria mais efetivo na utilização de sensores de pressão (maior

utilização da força), enquanto o caso de retorno semi-háptico a partir de vibradores seria igualmente efetivo tanto para sensores que captam força quanto para sensores que captam posição. Essa é a maior qualidade do retorno semi-háptico a partir de vibradores, pois ele possui um tipo de retorno equiparável para ambos os tipos de sensores.

A intenção desta seção é de exemplificar que um tipo específico de sensor pode ser mais apto a um tipo específico de retorno semi-háptico.

1.5. Soluções já existentes para captura do movimento das mãos

Deseja-se fazer uma análise das soluções já existentes de acordo com seu tipo de feedback e com os sensores utilizados. Como é possível a implementação de mais de um tipo de feedback em um dispositivo (i.e. um para a mão e outro para os dedos), é importante citar que, neste parágrafo, serão avaliadas as soluções divididas de acordo com seu **feedback para os dedos**, conforme havia sido feita a classificação na seção *1.4.1 - Classificação dos dispositivos segundo o tipo de feedback*, uma vez que temos o SpaceNavigator para o controle do movimento da mão. O SpaceNavigator possui feedback semi-háptico em seu controle (imposta através de molas internas ao dispositivo), e pode ser implementado com um controle de velocidade ou aceleração.

Apesar dos diferentes tipos de feedback existentes, todas as soluções avaliadas possuíam um mesmo feedback utilizado para todos os dedos. Além disso, também em todas as soluções os sensores utilizados faziam a medição de força ou de posição, sempre através de captura mecânica ou óptica.

O item *1.5.1* falará sobre soluções que não contam com retorno háptico. Já o item *1.5.2* falará das soluções que contam com esse tipo de retorno. A seção *1.5.3* falará dos dispositivos HandNavigator já desenvolvidos na França. Após esta seção sobre dispositivos já existentes, a seção *1.6* falará dos problemas que todas as soluções analisadas.

1.5.1. Soluções sem retorno háptico

Existem dois tipos de soluções existentes para capturar um movimento: soluções baseadas na captura óptica (captura sem contato, Figura 13) e soluções baseadas na captura mecânica (os sensores são associados a um exoesqueleto em contato com a mão, Figura 14 e Figura 15). Na Figura 15, é mostrada a solução chamada de “luva de dados” [10].



Figura 13 – Captura sem contato

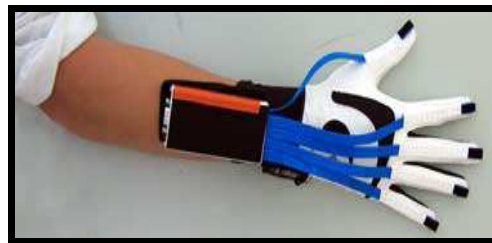


Figura 14 – Captura mecânica

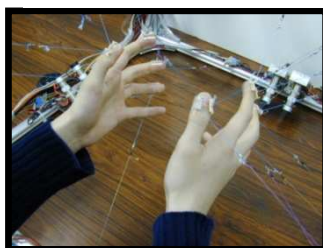


Figura 15 – Captura mecânica na “luva de dados”

1.5.2. Soluções com retorno háptico

Entre as soluções existentes, algumas incluem retorno háptico. Algumas só incluem retorno háptico para os dedos, enquanto outras incluem também para a mão. O retorno háptico, neste caso, permite um retorno de força de intensidade proporcional a aquela que se deveria ter caso se utilizasse nossa mão real para realizar a mesma tarefa que se deseja executar virtualmente. Simula também, portanto, o toque, que pode ser considerado como uma força real de intensidade máxima e que impede qualquer deslocamento extra. Soluções existentes com retorno háptico são mostradas nas soluções da Figura 18, e .

Respectivamente, as figuras representam as soluções “Spidar” [11], “Rutgers Hand Master” [12] e “CyberGrasp” (produzida pela empresa VRSoft).



Figuras 16, 17 e 18 – Spidar, Rutgers Hand Master e CyberGrasp, da esquerda para a direita

1.5.3. Soluções da equipe EVASION do INRIA

Antes do início do estágio, já existiam 2 protótipos construídos. Esses protótipos são chamados V1 e V2 e são apresentados na Figura 19 abaixo:

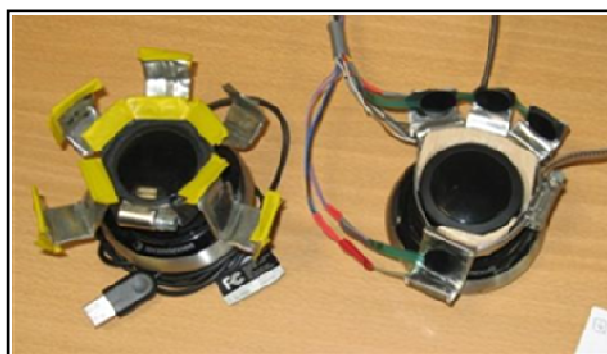


Figura 19 - Protótipos V1 e V2

O protótipo V1 é representado a esquerda da Figura 19. Ele utiliza retorno semi-háptico, que é assegurado através de uma lâmina, que se deforma e gera força de resistência quando é pressionada. Esse tipo de retorno háptico é aliado a dois sensores de pressão para captação de forças em ambos os sentido (Figura 20). Durante a utilização, o braço pode ficar apoiado pelo cotovelo. Além disso, não há necessidade de calibração, pois o SpaceNavigator não necessita disto. A figura Figura 21 mostra o dispositivo V1 em utilização.



Figura 20 - Detalhe lâmina deformável

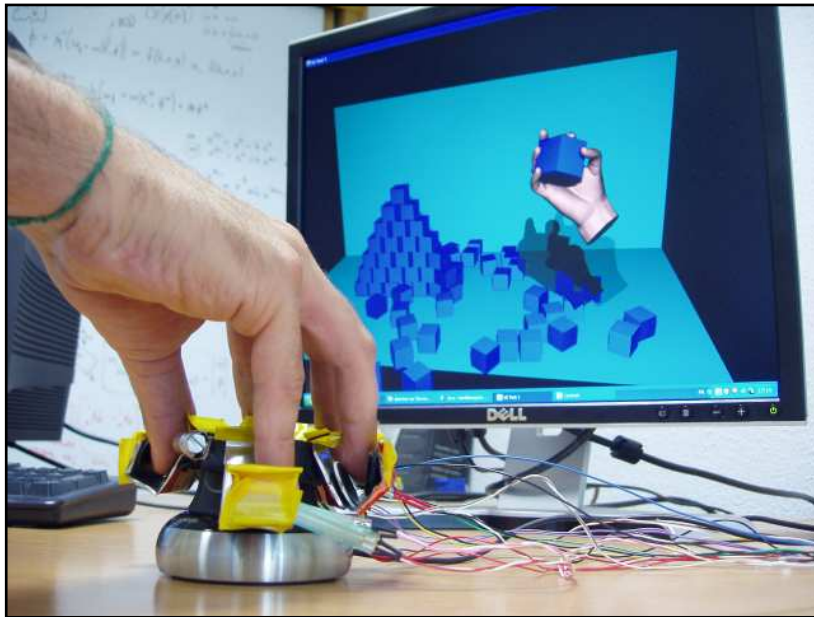


Figura 21 - HandNavigator V1 em utilização

O protótipo V2 (a direita na Figura 19) possui essencialmente as mesmas características de seu antecessor V1: sensores de pressão em lâminas de ferro que garantem retorno semi-háptico. Sua diferença principal é relacionada a seu design, mais confortável. O maior conforto é devido seu novo design, que procura deixar o sensor do dedão em oposição aos outros três dedos ao invés de se ter os 5 dedos igualmente espaçados ao redor do periférico.

O design se tornou um fator importante para a usabilidade dos dispositivos. Esta conclusão foi obtida a partir do momento onde se iniciaram as análises de ergonomia, iniciadas pela empresa Idénéa (<http://www.idenea.fr/>), uma empresa francesa de consultoria em ergonomia contratada para avaliação da ergonomia dos protótipos V1 e V2. Posteriormente a empresa também analisou as outras soluções que serão discutidas mais a frente. Os resultados das análises serão apresentados no decorrer da apresentação dos protótipos (discutidos nesta seção) e de seus problemas (discutidos na seção 1.6 - *Problemas das soluções existentes para captura do movimento das mãos*). Estes relatórios, em geral, apontavam design e o conforto como essenciais para o sucesso de uma solução, e as referências bibliográficas [13] e [14] dão mais detalhes sobre o assunto.

Após o término do desenvolvimento do dispositivo V2, inicia-se meu trabalho de fim de estudos francês (TFE) com o mesmo projeto. Em meu TFE, três novos periféricos foram desenvolvidos. As soluções foram baseadas em modelos que tivessem forma de um mouse, com sensores devidamente distribuídos nas posições naturais onde se instalavam os dedos ao redor da forma. A forma de mouse foi escolhida entre as três propostas que se tinha na época do desenvolvimento, a citar: forma de mouse (esquerda), forma de HandNavigator (meio), e forma similar de “flor” (similar ao V1 e V2, porém com melhoras em relação ao design, a direita).



Figuras 22 , 23 e 24 - Forma de mouse, forma semelhante ao SpaceNavigator e forma de " flor", respectivamente

A decisão pela forma de mouse para o V3 foi devido aos seguintes fatos ^[14]:

- Fácil entendimento de como deve ser posicionada a mão sobre o dispositivo já na primeira utilização, devido a forma já conhecida;
- Há espaço para os sensores serem colocados ;
- Compensação de esforços pelas « orelhas », pequenas extensões nas bordas da forma de mouse, para apoiar superiormente o dedo e o dedo mínimo, afim de compensar esforços aplicados pelos outros dedos nos sensores frontais de pressão;
- Possibilidade de troca de sensores ;
- Forma de um mouse: facilmente retida;
- Melhor ergonomia ;
- Sem pressionar sensores com as unhas (como ocorria no V1 e no V2).

Além disso, as outras soluções apresentavam problemas descritos a seguir [14].

- **Forma SpaceNavigator:** Falta de espaço para grandes sensores ao redor do botão era o maior dos problemas. Mesmo que uma forma maior e mais ergonômica fosse feita (conforme mostra a Figuras 22, com uma nova forma elaborada a partir de massa de modelagem), é impossível colocar sensores ao redor da forma devido ao seu pequeno tamanho.
- **Forma floral:** Foi avaliado pela análise ergonômica que um melhor conforto seria garantido com dedão ao lado ^[13]. Isso causava, porém, uma resultante na tentativa do cancelamento de forças caso o dedão estivesse em posição ideal (falta de simetria, Figura 25). Outros problemas incluíam pressionamento do sensor com a unha, postura mão-pulso que evoca movimentos difíceis e retorno semi-háptico difícil de ser sentido ^[13].

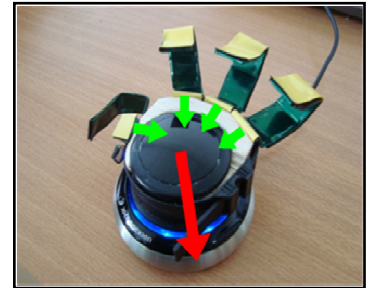


Figura 25 - Falta de simetria da solução floral

Após seleção da forma de mouse, três protótipos foram desenvolvidos e podem ser observados abaixo:



Figura 26 - Protótipo V3a



Figura 27 - Protótipo V3b



Figura 28 - Protótipo V3c

A diferença essencial entre os três protótipos eram seus sensores, a citar: Pressão (V3a), Touchpad (V3b) e Trackball (V3c). Todos os protótipos construídos possuíam a forma de um mouse, devido às razões já explicitadas. Os três sensores foram escolhidos devido a serem os únicos entre os analisados que poderiam ser implantados, depois de uma análise dos objetivos do projeto e tamanho e funcionalidade dos sensores ^[14]. Os sensores analisados e a característica principal para sua escolha, ou não, foram:

- **Sensor de pressão:** O sensor de pressão (Figura 29) foi utilizado no protótipo A. Ele foi escolhido, pois, já havia sido testado nos protótipos V1 e V2, assim como porque entre os sensores disponíveis ele era o único que captava forças, portanto seria o único que possibilitaria uma futura implementação com feedback semi-háptico através de resistência mecânica, futuramente, caso o protótipo V3a mostrasse ser uma boa solução.

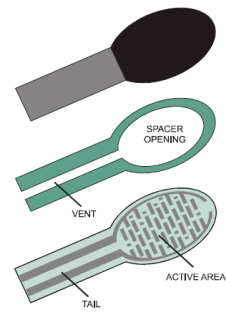


Figura 29 - Sensor de Pressão

- **Touchpad:** O touchpad (Figura 30) foi escolhido devido a idéia de se fazer um controle multi-touch em posição. O feedback via vibração (para todos os dedos juntos, porém, devido ao fato de só se ter um sensor) era visado futuramente caso a solução tivesse boa controlabilidade.



Figura 30 - Touchpad

- **TrackBall:** O TrackBall (Figura 31) também foi escolhido devido a idéia de se fazer um controle em posição. O feedback via vibração poderia ser feito independentemente para cada um dos dedos, mas era um plano futuro apenas caso a solução apresentasse boa controlabilidade.



Figura 31 - Trackball

- **ScrollPad:** O ScrollPad (Figura 32) não foi escolhido para incorporar a geração de protótipos V3, em uma

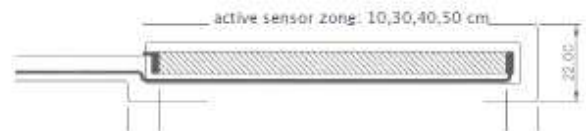


Figura 32 - Scrollpad

possível implementação com controle de velocidade. Como se visava a implementação de sensores sobre a forma de um mouse e só foram encontrados tamanhos mínimos de 10cm para os ScrollPads, sua utilização foi descartada. Além disso, ele era caro, e a possível solução de implementação de um TouchPad multitouch já seria quase equivalente.

- **Alavanca mono-eixo:** A alavanca mono-eixo (Figura 33 - Alavanca mono-eixo) também não foi escolhida para incorporar a geração de protótipos V3. Seu ponto mais interessante era a possibilidade da utilização do controle por velocidade, o que só havia



sido feito com o sensor de força até o momento. A Figura 33 - Alavanca mono-eixo alavanca encontrada possuía apenas 5 posições de estabilidade, ou seja, só seria possível o controle de duas velocidades (rápido e devagar) para cada um dos sentidos (abertura / fechamento dos dedos), além da última posição para estado de paragem. Procurava-se por uma alavanca mono-eixo que pudesse ser de controle contínuo, o que não foi encontrado, e, portanto, a opção de sua utilização foi descartada.

1.6. Problemas das soluções existentes para captura do movimento das mãos

1.6.1. Problemas da captura de movimento sem retorno háptico

As soluções da captura de movimento, seja ela mecânica seja ela óptica, são, em geral, problemáticas para o utilizador. Às vezes, é necessária uma calibração antes do uso. Este é um ponto negativo, pois nunca se pode utilizar o dispositivo imediatamente, devido à “preparação” necessária. Às vezes, também não é possível de se parar a manipulação da virtual e continuá-la mais tarde sem a necessidade de uma nova calibração: em geral, uma calibração deve ser refeita toda vez que se for utilizar o dispositivo, o que demanda mais tempo ^[14].

A questão das oclusões, para os sistemas baseados em visão computacional ou captura óptica, também é crítico ^[2]. Sempre haverá um ponto que não é visível pela câmera, por exemplo, quando um dedo recobre outro. Uma solução poderia ser a implementação de um número maior de câmeras, porém o custo aumentaria consideravelmente ^[2], além do fato que o problema estará sempre presente quando duas partes se tocam ou quando uma parte recobre outra.

Feedback háptico ou semi-háptico não estão presentes. A realização de gestos no ar e sem contato não contribui para o retorno háptico do utilizador, que é importante para o controle de qualquer tarefa ^[2], seja ela no mundo real ou no mundo virtual. É importante observar que, na vida real, se coordena os próprios gestos em um objeto de acordo com o que sentimos devido às sensações proprioceptivas também ^[2].

1.6.2. Problemas da captura háptica

Os sistemas hápticos existentes atualmente não são constantemente ou comumente utilizados pelo grande público, profissionais ou empresas. As principais razões são:

- **Complexidade dos sistemas:** sistemas hápticos são, em geral, complexos e difíceis de instalar, o que requer, em geral, uma pessoa que conheça bem o sistema, assim como sua manipulação, instalação e calibração (que geralmente também é necessária) ^{[2][14]};
- **Custo elevado:** devido à complexidade deste tipo de sistema, principalmente devido a seus atuadores, seu preço é consideravelmente elevado ^{[2][14]};
- **Dificuldade de simular perfeitamente a percepção háptica:** mesmo que esta solução tente resolver os problemas relativos a percepção háptica, é importante mencionar que as soluções existentes ainda não estão próximas de uma solução ideal, onde uma boa correspondência entre as forças virtuais e reais ocorreria ^{[2][14]}.

1.6.3. Problemas das soluções da equipe EVASION do INRIA

As soluções V1 e V2 inovaram ao implementar técnicas semi-hápticas que já tinham boa implementação (SpaceNavigator) para o controle de uma mão virtual, porém pecaram na escolha dos sensores para controle dos dedos. Sensores de pressão para captação do movimento dos dedos serviam para a implementação do retorno semi-háptico junto a materiais e estruturas que promoviam com resistência

elástica, porém eram um problema devido ao fato de somente captarem esforços de magnitude considerável, suficientes para interferir no movimento do SpaceNavigator. Isso gerava uma alteração do movimento da mão virtual quando se tentava, na verdade, a movimentação dos dedos virtuais ^[13]. Além disso, era muito difícil perceber o retorno semi-háptico nos dedos devido a baixa deformação das lâminas ^[13]. No movimento de abertura dos dedos, unhas deviam ser utilizadas para acionar os sensores de pressão, o que agravava mais ainda os problemas apresentados acima ^[13]. Finalmente, no quesito ergonomia, a falta de apoio do pulso e a postura dobrada que ele ficava durante a utilização eram fatores críticos de ergonomia ^[13].

O dispositivo V3a contava com o que se chamava de “orelhas” para tentar compensar forças aplicadas nos sensores de pressão ^[14]. Imaginava-se, futuramente, caso a solução funcionasse, de se implementar esta solução a um retorno semi-háptico com a utilização de materiais de resistência linear. Percebeu-se, porém, que as orelhas não eram efetivas para promover o cancelamento das forças, pois os sensores de pressão necessitavam de pressão imposta elevada para funcionar. Assim, a tentativa de acionamento dos sensores de pressão inferia ainda no movimento do SpaceNavigator, e conseqüentemente no movimento da mão virtual. A falta de apoio para o pulso também foi herdada dos protótipos V1 e V2, apesar de que a péssima postura mão-pulso que estes continham foi totalmente solucionada com a forma de mouse escolhida.

O dispositivo V3b contava com um TrackBall para controle do dedão e um MousePad para controle dos outros dedos. Pensava-se em futuramente implementar o feedback semi-háptico para o dispositivo, através de vibração conjunta para todos os dedos. O MousePad não era multi-touch, ou seja, não podia captar o movimento de mais de um dedo. Dispositivos multi-touch são de alto valor no mercado, o que já afasta imediatamente este tipo de solução para o nosso caso. Mesmo assim, testes foram executados considerando-se o controle dos quatro dedos virtuais (sem ser o dedão) através de um único dedo real no TouchPad, mais o controle do dedão via TrackBall. O problema de interferência de graus de liberdade se repete, principalmente devido ao fato do grande braço, em relação ao SpaceNavigator, para as relativamente pequenas forças que eram aplicadas sobre o TouchPad.

O dispositivo V3c contava com TrackBalls separados para o controle dos dedos virtuais. Pensava-se em futuramente implementar o feedback semi-háptico para o dispositivo, através de vibração independente por dedo. O retorno através de vibradores seria o único disponível, uma vez que não é aplicada força e, portanto, a utilização de materiais de resistência elástica para promover retorno semi-háptica não são sempre adequados em sensores que não são acionados através de forças. Porém havia um problema devido ao funcionamento dos sensores tipo TrackBall, que possuem uma discretização com grandes gaps físicos, ou seja, às vezes eram necessárias três ou mais reposições dos dedos sobre o sensor, rodando o trackball, para concluir o fechamento ou a abertura de um dedo virtual. Isso era necessário para evitar que o fechamento do dedo virtual tivesse poucas posições intermediárias entre o aberto e o fechado. Apesar disto, esse foi o protótipo cujo acionamento dos sensores de movimentação dos dedos virtuais menos interferia na movimentação da mão virtual. Isso se devia ao fato do protótipo ser reduzido, e, portanto, havia um pequeno braço para o momento das forças que eram aplicadas no TrackBall. Mesmo assim, apesar das interferências serem menores, elas ainda persistiam. Persistia também a falta de apoio para o pulso.

Após análise e utilização de todas as soluções, pode-se avaliar que o principal problema das soluções desenvolvidas no INRIA são relacionadas a falta de apoio para o pulso e dificuldade na independência de acionamento dos graus de liberdade da mão e dos dedos. Com respeito ao segundo ponto, o problema é que o SpaceNavigator é muito sensível, e, portanto, o acoplamento de sensores ao seu corpo, se tiver que ser feito, deve ser feito de maneira bem planejada e com sensores bem escolhidos e estrategicamente posicionados.

1.7. Tabela das soluções existentes e suas características principais

Na tabela da página seguinte, mostra-se as características principais das soluções analisadas. Tais características incluem dados sobre sensor utilizado e tipo de retorno háptico (tanto para a mão quanto para os dedos), além de principais pontos positivos e negativos.

Tabela 1 - Comparação das soluções existentes

Protótipos	Controle dos dedos				Controle da mão				PRÓS	CONTRAS
	Tipo de Sensor		Feedback		Tipo de Sensor		Feedback			
	Controle	Através de...	Controle	Através de...	Controle	Através de...	Controle	Através de...		
Soluções sem feedback nos dedos	Diversos (com captura óptica ou mecânica)	Não há	Em geral, por posição	Diversos (com captura óptica ou mecânica)	Em geral, por posição	Em geral, por posição	Geralmente não há	Alguns dispositivos não há	- Falta de feedback - Existência de oclussões (para captura óptica). - Necessidade de calibração - Falta total de apoio para manuseio (movimentação aérea)	
	Mecânica	Háptico	Em geral, por posição	A atuadores elétricos ou pneumáticos de força	Mecânica	Háptico ou não existente	Háptico ou não existente	A atuadores elétricos ou pneumáticos de força	- Necessidade de calibração - Falta total de apoio para manuseio (movimentação aérea) - Sistemas são muito complexos - Há feedback háptico, mas geralmente é difícil simular perfeitamente o mundo virtual com perfeição - Alto preço	
Soluções com feedback háptico nos dedos	Sensor de pressão	Semi-háptico	Velocidade	Resistência elástica (Metal deformado)	Space- Navigator	Semi-háptico	Velocidade	Resistência elástica (Hand-Navigator)	- Baixa qualidade do feedback semi-háptico dos dedos - Sensor de pressão só capta pressões elevadas - Unhas para acionar sensores - Postura mão-pulso não usual e falta de apoio para o pulso - Alta interferência entre movimentos dos dedos e da mão - Dedão em posição altamente desconfortável	
	Sensor de pressão	Semi-háptico	Velocidade	Resistência elástica (Metal deformado)	Space- Navigator	Semi-háptico	Velocidade	Resistência elástica (Hand-Navigator)	- Baixa qualidade do feedback semi-háptico dos dedos - Sensor de pressão só capta pressões elevadas - Unhas para acionar sensores - Postura mão-pulso continua não usual - Falta de apoio para o pulso - Alta interferência entre movimentos dos dedos e da mão	
Soluções com feedback semi-háptico nos dedos (Elaboradas durante TFE na França)	Sensor de pressão	(semi-háptico, futuramente)	Velocidade ou Posição	(vibradores ou resistência elásticos, futuramente)	Space- Navigator	Semi-háptico	Velocidade	Resistência elástica (Hand-Navigator)	- Falta de apoio para o pulso - Sensor de pressão só capta pressões elevadas - Alta interferência entre movimentos dos dedos e da mão - Cancelamento de forças pelas 'orelhas' era ainda fraco	
	MousePad (e TrackBall para dedão)	(semi-háptico, futuramente)	Posição	(vibradores, futuramente)	Space- Navigator	Semi-háptico	Velocidade	Resistência elástica (Hand-Navigator)	- Falta de apoio para o pulso - Alta interferência entre movimentos dos dedos e da mão - Touchpad multitouch é caro e não foi encontrado	
c	TrackBall	(semi-háptico, futuramente)	Posição	(vibradores, futuramente)	Space- Navigator	Semi-háptico	Velocidade	Resistência elástica (Hand-Navigator)	- Falta de apoio para o pulso - Trackballs com pequena discretização mecânica (gaps) - Média interferência entre movimentos dos dedos e da mão	

2. Objetivos

A intenção do protótipo é de que um equipamento periférico, controlado por uma mão (i.e. mão real), possa controlar uma mão que aparece na tela do computador (i.e. mão virtual), através de um programa específico. Solução, aqui, engloba periférico, circuito e programa a serem desenvolvidos. Mão real/virtual engloba o conjunto da mão e seus dedos em ambos os ambientes.

Conforme ideais desenvolvidos no início do projeto e explicitados em 1.3 - *O SpaceNavigator e a idealização do HandNavigator*, deseja-se o SpaceNavigator para garantir o feedback semi-háptico para as mãos (característica já presente no periférico).

A análise do tipo de sensor e o feedback a ser utilizado para os dedos faz parte deste trabalho de desenvolvimento. Deseja-se que a mão e os dedos virtuais se mantenham imóveis (estáveis) quando ocorre desacionamento, pois este é um princípio do projeto desde o início de seu desenvolvimento.

O protótipo deve garantir boa manipulação da mão virtual, a partir de movimentos suaves. Procura-se um periférico que tenha bom controle, com alta independência entre os graus de liberdade, boa ergonomia e design para se tornar confortável. A questão do feedback tátil para o dedo (háptico, semi-háptico ou inexistente) será considerada após avaliar-se o tipo de sensor a ser utilizado, e não há imposições quanto a isto. Deseja-se a melhor interação entre, controlabilidade, retorno, custo e independência no acionamento dos graus de liberdade.

A controlabilidade dos dedos anular e mínimo juntos será mantida. As simplificações que esta junção de controle implica são bem favoráveis ao projeto em termos de menos necessidades físicas, enquanto em nenhuma análise isso foi considerado um ponto negativo (vide *Tabela 1 - Comparação das soluções existentes*).

Outro objetivo é a portabilidade do periférico. Para que o dispositivo seja portátil em seu transporte, objetiva-se a utilização de materiais leves em sua construção, de preferência em um periférico pequeno. Quanto à instalação, procura-se obter um dispositivo que possua fios e equipamentos eletrônicos simples e fáceis de se instalar, para evitar alto tempo de instalação antes do uso. Quanto ao tempo para início de utilização, deseja-se tempo nulo (i.e. não há necessidade de calibragem) ou pequeno tempo para calibração.

A intenção é a utilização do novo protótipo com o antigo programa de manipulação virtual. Isso garantirá a interação em um ambiente virtual 3D. Como é desejável, porém, será feito o desenvolvimento de um programa próprio para este projeto. O programa a ser desenvolvido servirá para validação do periférico proposto no que diz respeito a sua interface com o computador. É fato que imagens 3D seriam de grande auxílio para a manipulação, porém apenas imagens 2D serão utilizadas para esta validação. Esta simplificação no desenvolvimento é devida a fatores de limite de tempo de execução para o projeto, além do fato que se deseja focar na parte mecânica, como continuação dos estudos que foram executados na França. O programa a ser desenvolvido não incluirá contato com objetos.

3. Metodologia

Durante o projeto na França, em geral avaliavam-se os resultados de análises dos protótipos mais recentemente construídos para o desenvolvimento dos próximos. Neste momento, porém, o objetivo do projeto é a construção de um periférico totalmente novo, no qual não se levará em conta apenas os pontos positivos e negativos somente dos últimos protótipos construídos, mas de todos os já desenvolvidos e existentes. Procurar-se-á fazer uma análise extensa de todos os protótipos e análises até hoje construídos, além da análise dos estudos específicos já feitos para todos eles, em todas as fases de seus projetos. A metodologia, portanto, não é de evolução direta, mas sim de reavaliação dos princípios de funcionamento e características gerais dos projetos existentes para o desenvolvimento de um novo protótipo. A *Tabela 1 - Comparação das soluções existentes* será, portanto, extensivamente analisada, com foco nos problemas que persistiram na evolução dos protótipos e também nos problemas muito críticos.

A metodologia para obtenção da solução final construída será executada de acordo com os três itens descritos abaixo:

- **Estudo de projeto:** a análise de todos os dispositivos estudados a partir da tabela das soluções existentes e suas características, que já inclui dados dos estudos de ergonomia, de antigos estudos e também das tecnologias avaliadas durante a Revisão Bibliográfica. Esta análise servirá primeiramente para a escolha da forma do futuro periférico, e posteriormente para a escolha de seu sensor. Após isto se fará a escolha de materiais para construção do periférico, especificação do circuito para o sensor selecionado e descrição do programa a ser desenvolvido.
- **Desenvolvimento da solução:** projeto 3D de periférico, projeto do circuito eletrônico e projeto do funcionamento do programa.
- **Desenvolvimento protótipo:** parte de desenvolvimento final, no que diz respeito à construção e programação, além da integração com dispositivo antigo.

Dessa forma, o estudo de projeto será apresentado em *4 - Projeto*. O desenvolvimento da solução final, assim como sua implementação (desenvolvimento do protótipo) estarão detalhados na seção *5 - Desenvolvimento*.

Finalmente, uma validação para o protótipo será feita em *6 - Validação*, onde testes para verificação serão executados para validação da solução proposta.

4. Projeto

Nos próximos itens serão escolhidas a forma e o sensor a se utilizar no projeto, seguido de um processo de tomada de decisão através de tabelas de decisão.

4.1. Forma do periférico e material de construção

4.1.1. Análises

Após análise do item 1.5.3 - *Soluções da equipe EVASION do INRIA*, foram identificadas algumas formas que tinham características favoráveis ao desenvolvimento e cuja forma de utilização poderia ser revisada para melhor corrigir os problemas existentes:

- **Forma de mouse:** A forma de mouse (Figura 34) é interessante por ser reconhecida para seu tipo de utilização (manipulação virtual). Sua utilização nos protótipos V3a, V3b e V3c mostrou que a maioria das pessoas sabia a maneira correta de colocar a mão no protótipo e os dedos sobre



Figura 34 - Forma de mouse

sensores. Nem sempre, porém, estas mesmas pessoas sabiam manipular o protótipo de maneira apropriada já na primeira tentativa. O maior problema desta forma reside na dificuldade da separação das forças, o que pode ser constatado em todos os protótipos construídos com esta forma, porém de forma mais amena no último, o V3a, devido a suas dimensões reduzidas. A forma de mouse sugere a inserção de sensores nos locais destinados ao posicionamento das pontas dos dedos, porém é exatamente o fato de inserir

sensores nesta forma que tem gerado problemas de interferência no controle dos diferentes graus de liberdade dos dedos e da mão.

- **Forma do SpaceNavigator:** a forma do botão do SpaceNavigator (Figura 35) é ideal para sua manipulação. A idéia de utilização da mesma forma para o HandNavigator foi com inserção de sensores sobre o próprio botão já foi avaliado em [14] e descartado devido ao



Figura 35 - Forma do SpaceNavigator ou semelhante

pequeno tamanho do botão do SN para ainda incorporar novos sensores. A nova idéia seria uma implementação de sensores fora do botão do SpaceNavigator, o que renderia, no mínimo, uma boa separação dos graus de liberdade, um dos maiores problemas enfrentados até o momento, segundo a Tabela 1. Pequenas alterações da forma do botão poderiam também ser feitas, como mostra a Figura 35, feita com a utilização de massa de modelar sobre o botão do SpaceNavigator.

Durante a análise das duas formas citadas acima, uma observação importante pode ser feita: em ambos os casos, um grande problema é a localização dos sensores. **Em todos os periféricos mecânicos que possuem sensor com posição fixa em relação ao botão do SpaceNavigator** (i.e. que tenham seus movimentos físicos acompanhando os movimentos do botão de acionamento do SpaceNavigator), **houve problemas para separação do controle dos graus de liberdade dos dedos e da mão.**

Há, portanto, uma boa justificativa para a não consideração da forma floral entre as formas analisadas para este projeto. Ela não foi considerada uma vez que o cancelamento de forças no periférico é uma tarefa difícil de ser conseguida, conforme pode ser observado nas análises das características dos protótipos anteriores da revisão bibliográfica e também nos estudos de ergonomia dos primeiros protótipos, além da evidência na Tabela 1 de itens como falta de

ergonomia, devido à falta de apoio para o pulso, que não pode ser colocado devido à péssima postura mão-pulso imposta pela forma analisada.

O formato floral já foi testado em uma prévia de protótipos para os futuros V3, e não apresentou bons resultados. Sua ergonomia é ruim e, quando tentou-se melhorá-la a partir melhor disposição dos dedos ao redor do Space-Navigator, percebeu-se que isto afetava o cancelamento das forças exercidas pelos dedos, como mostram a Figura 37 e a Figura 36.

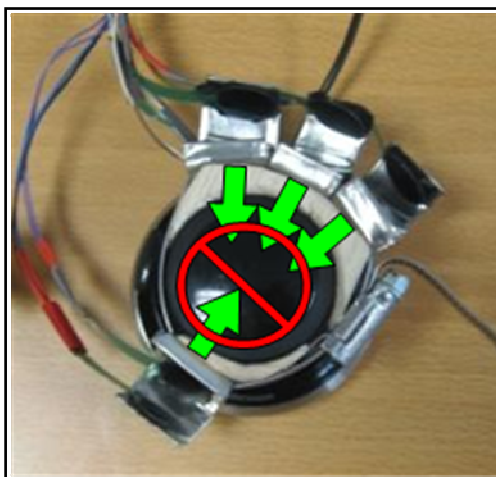


Figura 37 - Cancelamento de forças na forma de "flor" não ergonômica (V2)

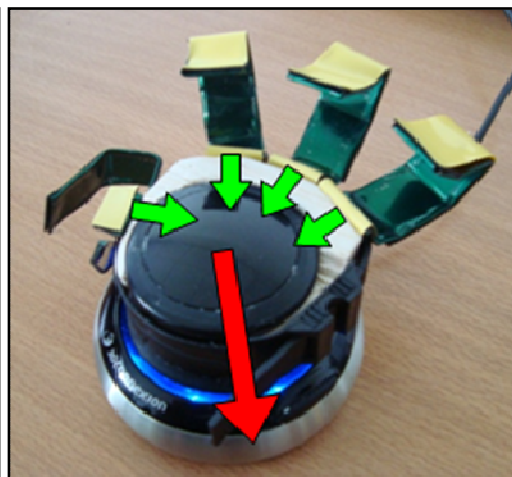


Figura 36 - Não cancelamento de forças na forma de "flor" mais ergonômica (reposicionamento dedão)

Voltando para a escolha entre forma de mouse e forma do SpaceNavigator, pode-se notar que a forma de mouse, conforme mostra a Tabela 1 de análise das soluções, se mostrou amplamente problemática com o cancelamento de forças. Caso esta forma for utilizada, o projeto deverá considerar uma boa análise de design e de ergonomia para garantir um bom cancelamento das forças, além de um bom design 3D com boa construção (a idéia do cancelamento de forças esta intimamente ligada a geometria do periférico, e, portanto, ao design 3D). Busca-se uma solução onde os sensores estejam sobre a forma de mouse do periférico (portanto, fixos ao botão do SpaceNavigator), uma vez que a idéia de forma de mouse vem da idéia de se obter uma melhor ergonomia para utilização do SpaceNavigator, aliado a uma forma especial para fácil acionamento dos sensores, sem interferência entre os graus de liberdade controlados.

A análise que leva em conta a retirada dos sensores de cima do mouse não faz sentido aqui. Caso se abdicar de sensores sobre a forma de mouse, haveria a necessidade de um mouse pequeno, para que a mão real do utilizador não tenha necessidade ser retirada do HandNavigator para poder acionar os sensores de controle do movimento dos dedos. Esta opção seria equivalente a utilização de uma forma de Space-Navigator, pois, sem sensores e de formato menor, a forma reduzida de mouse tende ao formato do botão do SpaceNavigator.

A idéia de posicionamento dos sensores em um local sem contato fixo com o botão do SpaceNavigator (i.e. movimentação física não acoplada entre SpaceNavigator e sensores) será a única a ser considerada caso de optar por utilizar a forma de SpaceNavigator no periférico. Isto seria uma obrigação de projeto, uma vez que não há espaço físico para inserção de sensores sobre seu botão ^[14]. Há que se avaliar, por outro lado, que a falta de sensores fixos ao SpaceNavigator (i.e. que tenham movimentos acompanhando o SpaceNavigator) pode ser desvantajosa devido ao fato que gestos para movimentar a mão virtual e gestos para movimentar os dedos virtuais podem acabar por não poder ser executados com boa sincronia. A solução básica com a utilização da forma do SpaceNavigator seria utilizar sensores fixos mas sem vínculo de movimentação com o SpaceNavigator. Denominar-se-á esta solução como “Solução inicial” para a “Forma do SpaceNavigator”.

Outra solução poderia ser proposta, porém, para solucionar o problema de falta de sincronia dos movimentos. Como tal falta de sincronia seria efetivamente causada devido ao fato de não se pode controlar os sensores dos dedos ao mesmo tempo em que se segura o SpaceNavigator, a idéia seria uma fixação entre a mão real e o botão do SpaceNavigator, de tal maneira que o acionamento do sensor não afetasse o controle da mão. Seu esquema segue abaixo, no Diagrama 8:

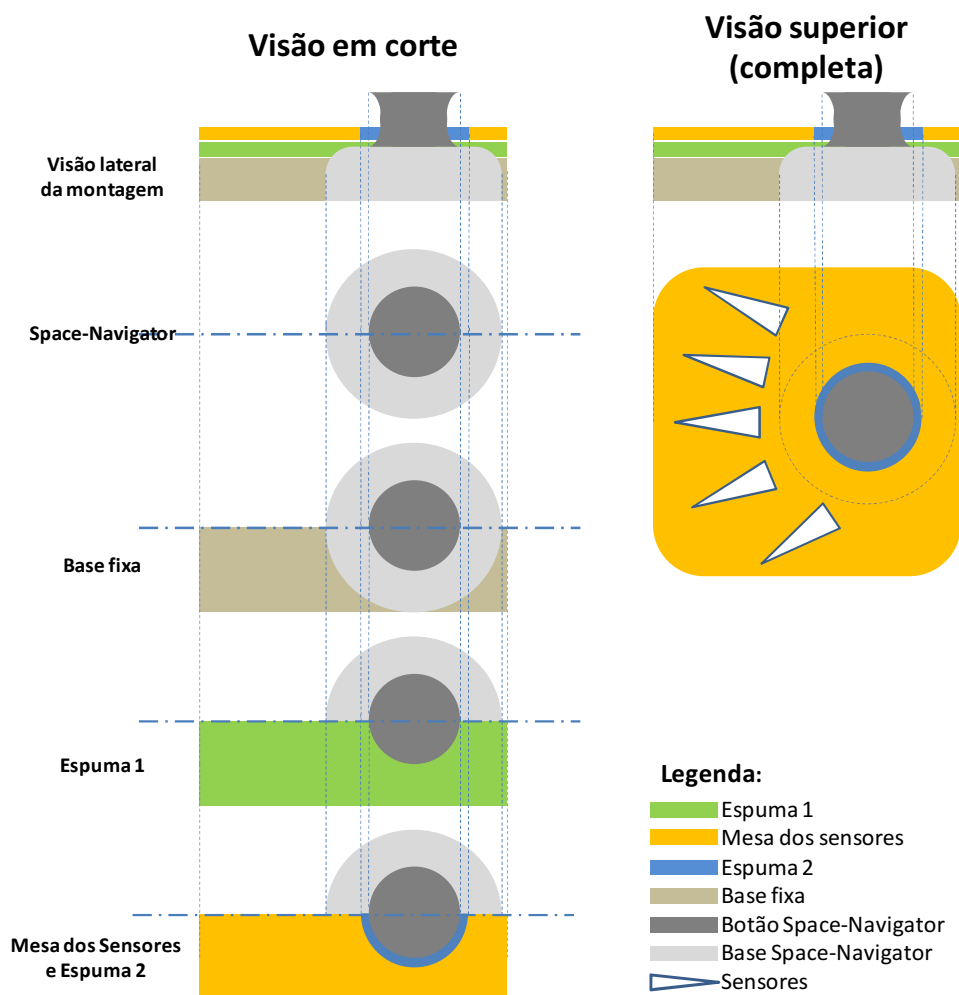


Diagrama 8 - Esquema de montagem de uma possível solução do HandNavigator

A espuma 2 que faz a ligação entre a mesa dos sensores e o botão do SpaceNavigator, serve para que movimentos de um não interfiram nos movimentos do outro, exceto durante a rotação do botão do SpaceNavigator, que acarretará na rotação da mesa e, conseqüentemente, se poderá coordenar movimentos dos dedos virtuais com movimentos das mãos virtuais. A Espuma 1 serve para que a mesa de sensores tenha um apoio, mas que também se mova um pouco para permitir que o SpaceNavigator possa se inclinar para frente e para os lados. Denominar-se-á esta solução como “Solução proposta” para a “Forma do SpaceNavigator”.

Poderia ser considerada nesta solução também a presença de um prendedor entre a mão real e o botão do SpaceNavigator, que serviria para possibilitar o movimento dos dedos mesmo quando se soltar o SpaceNavigator. Por outro lado, há

que se avaliar que isso não seria extremamente necessário uma vez que pelo menos um dedo poderá ser solto do HandNavigator para controlar seu equivalente virtual, além do fato que, em geral, a manipulação dos dedos virtuais com a mão virtual parada é mais eficiente. O prendedor, porém, poderia acarretar um pequeno tempo para início de utilização, além de poder causar a movimentação o periférico em um momento onde possivelmente deseja-se soltá-lo. O problema de maior tempo para início de utilização seria algo a se considerar como ponto negativo deste prendedor.

4.1.2. Decisões

As três sugestões de solução descritas acima serão avaliadas segundo os seguintes critérios estabelecidos (pesos designados de 0 a 5):

- **Critério 1: Cancelamento de forças:** o maior problema de todos os protótipos do HandNavigator até hoje fabricados foi o cancelamento de forças (vide Tabela 1). Este critério influi altamente na usabilidade do dispositivo final, e portanto terá peso 5, pois afeta muito a usabilidade do periférico final.
- **Critério 2: Tempo para início de utilização:** entre os objetivos descritos, comentou-se que o tempo para início de utilização sempre foi principalmente, mas não só, um problema dos dispositivos com captura óptica dos movimentos, ou com retorno háptico, pois estas características geralmente necessitam de calibração e instalação. O tempo para início de utilização é portanto uma avaliação a se considerar, e terá peso 3.
- **Critério 3: Tamanho:** o tamanho do dispositivo influi em sua portabilidade, um dos objetivos gerais do projeto. Da mesma forma que o critério anterior, portanto, o peso será 3.
- **Critério 4: Fácil compreensão do dispositivo:** é importante que o usuário possa olhar o dispositivo e facilmente insinuar sobre como utilizá-lo. Isso foi um dos pontos analisados no relatório de ergonomia dos protótipos V1 e V2. Como, porém, não foi definido como objetivo principal, será considerado o peso 2.

- **Critério 5: Posição fixa dos sensores em relação ao botão do SpaceNavigator:** este fator é altamente afetado pelo Critério 1 (notas inversamente proporcionais). Mesmo assim, sua análise é importante, até mesmo por que é o que difere essencialmente as duas soluções que possuem forma do SpaceNavigator utilizada. Considerar-se-á peso 2, por não ser um dos objetivos principais do projeto.

Chegamos a seguinte tabela de decisão:

Tabela 2 - Matriz de decisão para forma do periférico

	Peso	Forma Mouse	Forma Space-Navigator	
			Solução inicial	Solução proposta
Cancelamento de forças	5	1	5	5
Tempo para início de utilização	3	5	5	4
Tamanho	3	3	2	2
Fácil compreensão do dispositivo	2	4	3	3
Posição fixa dos sensores em relação ao botão do Space-Navigator	2	5	2	5
Nota final		47	56	59

O cancelamento de forças é muito fraco para a solução com forma de mouse, conforme pôde ser visto nos protótipos V3 analisados anteriormente, devido aos sensores acoplados a forma de mouse. O tempo para início de utilização só foi afetado na solução proposta com forma de Space-Navigator, uma vez que nela poderia ser incluída a utilização de um fixador entre a mão real e o botão do Space-Navigator (e.g. ímã, fita etc), o que atrasaria o início de uma operação para fixação da mão real, apesar disto não ser uma obrigatoriedade. O tamanho dos dispositivos com forma de SpaceNavigator são planos, mais baixos, porém mais largos, pois a mão fica mais aberta neles, e por isto as notas tendiam a ser próximas. A fácil compreensão do dispositivo é lógica para um Mouse, porém não tão difícil para os outros, pois a mesa dos sensores pode ter forma de mão, por exemplo, facilmente identificada por um possível usuário. No critério “posição fixa dos sensores em relação ao SpaceNavigator”, como era de se esperar, foram consideradas notas altas para a solução com forma de mouse e para a solução proposta, pois movimento dos sensores de acordo com o movimento do SpaceNavigator é interessante no controle simultâneo da mão e dos dedos virtuais.

O periférico que será desenvolvido será o que foi explicado em “solução proposta”, com forma de SpaceNavigator e sensores não fixos a ele, porém afixados em uma mesa de sensores. Tal mesa de sensores será externa, e se liga com o botão do SpaceNavigator através de uma espuma, para não afetar seu movimento porém poder rotacionar junto com o botão, para que os dedos possam sempre estar sobre os sensores. Um fixador entre a mão-real e o botão do SpaceNavigator está previsto, para auxiliar na manipulação, porém seu uso não será de caráter obrigatório, mas apenas para teste. Ainda faltariam analisar melhoras para a forma do SpaceNavigator.

Uma melhora que foi constantemente apontada como fator problemático em protótipos da Tabela 1 fora a falta de conforto devido a falta de **apoio de braço**. Essa solução é citada nesta parte do texto, pois poderia ser aplicada a qualquer tipo de solução que fosse escolhida. Mesmo que este apoio possa atrapalhar a portabilidade, ele será considerado para maior conforto do utilizador, até mesmo pelo fato de que a mesa de sensores precisa estar apoiada em algum objeto, que poderia ser estendido lateralmente para funcionar como um apoio de braço, como mostra o seguinte esquema final da solução:

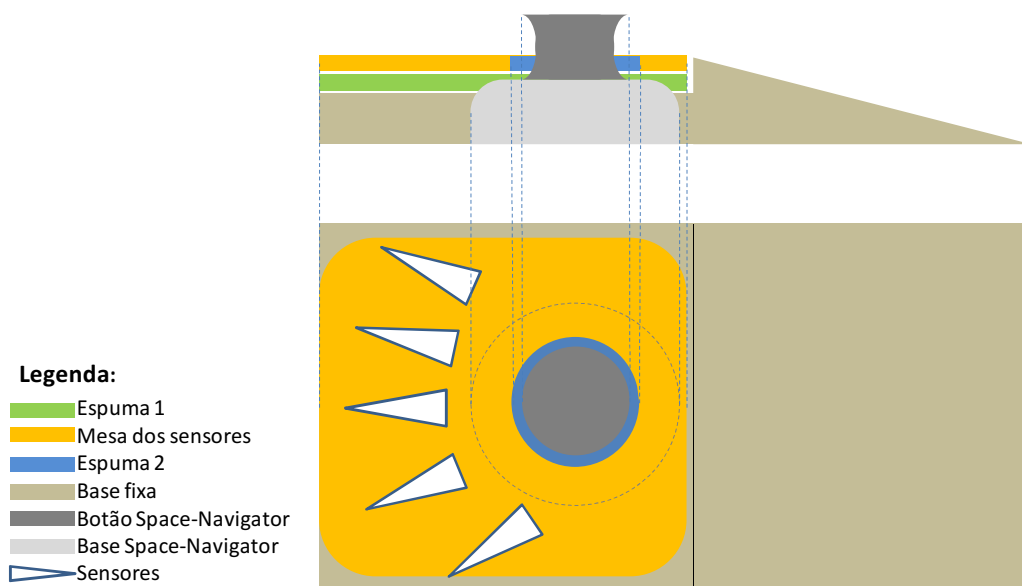


Diagrama 9 - Desenho ilustrativo da idéia escolhida, com apoio de braço incluso.

Material para construção

Alguns pontos importantes na análise para utilização de certo material são:

- Leveza
- Facilidade de montagem para protótipo
- Preço

O primeiro peso foi considerado 3, devido ao fato de que leveza pode ajudar na portabilidade, conforme especificado em objetivos. Os outros pesos foram considerados como 2, por serem de menor importância, porém ainda passíveis de análise.

Três materiais comumente utilizados para fazer protótipos são:

- Madeira
- Chapas de metal
- Plástico

Tabela 3 - Matriz de decisão para o material de construção do protótipo

	Peso	Madeira	Chapas de metal	Plástico
Leveza (para mesma resistência)	3	2	4	2
Facilidade de montagem para protótipo	2	5	3	1
Preço	2	5	1	3
Nota final		26	20	14

A madeira será a escolhida, de acordo com as notas e pesos atribuídos na Tabela 3. É importante salientar que a madeira ganhou muitos pontos devido à facilidade em manuseio e seu preço.

4.2. Sensor e tipo de feedback para os dedos

4.2.1. Sensores disponíveis para utilização na solução

Os sensores disponíveis para possível utilização na solução a ser desenvolvida são mostrados no Diagrama 10:



Diagrama 10 - Os sensores disponibilizados pelo INRIA

Da esquerda para a direita, têm-se:

- TrackBall
- ScrollPad
- Alavanca Mono-Eixo
- Sensor de Pressão

4.2.2. Análise dos sensores disponíveis segundo dados da Tabela 1.

A análise da Tabela 1 mostra muitos comentários à respeito ao sensor de pressão. Em todas as utilizações o sensor de pressão mostrou responder somente a forças de alta intensidade, sempre piorando a independência entre o movimento dos dedos e da mão. No caso deste trabalho, porém, já está decidido que o sensor será colocado fora do contato com o botão do SpaceNavigator. É ainda sim importante avaliar a interação deste sensor com os dedos do utilizador uma vez que a suavidade de movimentos é um dos objetivos do projeto, conforme especificado em 2 - *Objetivos*.

Uma boa vantagem do sensor de pressão foi de poder ajudar na implementação do retorno semi-háptico a partir de resistências mecânicas. Como a questão da usabilidade vem antes da questão do feedback, segundo os objetivos descritos, esta boa característica do sensor de pressão não será analisado nesta seção.

O TrackBall foi utilizado em algumas soluções da tabela 1. Em particular, na solução V3a do HandNavigator, ele foi extensivamente utilizado. Pode-se dizer que seu ponto forte é que não necessita de alta força para ser acionado, e seu maior déficit é sua pequena discretização com grandes “gaps”.

O ScrollPad não foi ainda testado. Seus pontos fortes são: baixa pressão para funcionamento e possibilidade de integração com retorno por vibração. Antigamente esta opção de sensor não existia, pois o ScrollPad disponível tinha mais de 10 cm, porém agora a nova opção encontrada possui apenas 5 cm. Assim como o sensor de pressão, outro de seus pontos fortes é a simplicidade de seu circuito, pois o ScrollPad, assim como o sensor de pressão, é apenas uma resistência. Isto é muito bom devido a sua característica analógica. Outra boa característica é que ele pode ser utilizado no controle tanto por posição quanto por velocidade (e outras maiores derivadas da posição).

Finalmente, tem-se a alavanca mono eixo, também não testada. Sua característica marcante seria a possibilidade de um controle o movimento dos dedos virtuais em velocidade através de uma posição como grandeza imposta (diferentemente do sensor de pressão, que pode fazer o mesmo porém através de uma força). Não é possível controle em posição, mas poderia ser em aceleração ou força. Há que se avaliar que sua implementação é mais complicada que as outras devido ao circuito necessário para tratar as informações de seus cinco pinos de ativação. Seu ponto negativo é a pequena discretização. Controle seria por duas velocidades (ou acelerações) em cada direção.

4.2.3. Decisão do sensor a ser utilizado

Os critérios de análise são poucos, portanto:

- **Suavidade de movimentos na utilização do sensor:** conforme descrito, é um dos objetivos, portanto terá peso 3. A força para acionamento será contabilizada neste item.
- **Discretização do sensor:** aos sensores cuja discretização é baixa ou que tenham grandes “gaps” em sua aquisição serão atribuídas menores notas. Será atribuído peso 2, pois esta característica pode afetar consideravelmente o controle da manipulação virtual.
- **Facilidade de construção no protótipo:** Este item deve ter peso baixo, menor que o dos objetivos, portanto peso 1.
- **Facilidade de integração no protótipo:** Este item deve ter peso baixo, menor que o dos objetivos, portanto peso 1.

E a matriz de decisão fica:

Tabela 4 - Matriz de decisões para escolha do sensor

	Peso	Sensor Pressão	TrackBall	ScrollPad	Alavanca
Suavidade dos movimentos	3	1	3	5	3
Discretização do sensor	2	5	3	5	2
Facilidade na construção do protótipo	1	5	4	5	4
Facilidade na construção do circuito	1	5	1	5	2
Nota final		23	20	35	19

Em suavidade de movimentos, o ScrollPad lidera, pois necessita de força quase nula para ser pressionado. O TrackBall e a Alavanca Mono-Eixo necessitam de força intermediária, enquanto o sensor de pressão necessita de alta força. Com relação a discretização do sensor, o TrackBall perde pontos por ter possibilidade de implementação com discretização de muitos níveis porém com altos “gaps” para se transitar entre eles. A alavanca, por outro lado possui apenas 4 estados em discretização e um estado estacionário. Sensor de pressão e ScrollPad, por serem resistências, portanto analógicos, possuem as maiores notas neste quesito.

A facilidade de construção do protótipo e do circuito acabaram somente por deixar mais claro a escolha pelo ScrollPad. Ele pode ser instalado no protótipo apenas pela colagem de uma fita dupla face, e em seu circuito ele tem o papel de resistência variável, mostrando sua simplicidade. A alavanca e o TrackBall tem necessidade de circuitos complexos de controle, em particular este segundo. Além disso, a instalação destes envolve a inclusão de seu formato no design 3D, enquanto o ScrollPad pode ser instalado simplesmente a partir de sua colagem com fita dupla face. O sensor de pressão, assim como o Scrollad, é fácil de se instalar e possui circuito simples, porém perde em outros quesitos, conforme apresentado.

Optou-se pela utilização do ScrollPad, devido a sua elevada pontuação na matriz de decisão da Tabela 4.

Após a escolha do sensor principal, poderia ser considerada a hipótese da inclusão de um feedback duplo com utilização de dois tipos de sensores (e.g. semi-háptico de vibração no toque, onde o controle estaria sendo feito via ScrollPad, e háptico ou semi-háptico para atuação de força sobre objetos, via sensores de pressão por exemplo). Porém, a maior complexidade para a implementação e o curto prazo para desenvolvimento deste projeto impedem esta realização, além do fato dos grandes problemas que o sensor de pressão possui, conforme foi explicado na seção anterior.

4.2.4. Definição do tipo de controle

O ScrollPad capta uma posição ϵ . Controle em posição, velocidade ou aceleração são possíveis.

Ao analisar os prós e contras da Tabela 1, pode-se inferir que um grande problema do TrackBall foi o controle em posição. Isso se deve ao fato de que em grandes movimentações são necessárias várias voltas eram necessárias naquele sensor devido aos seus altos “gaps” para geração sinal digital. Aqui, porém, tem-se um sensor analógico, onde isso não seria um problema.

Há que se avaliar, porém, que um controle em velocidade poderia permitir pequenos ou grandes deslocamentos virtuais a partir de apenas um pequeno deslocamento do dedo real. Devido a este motivo, irá se utilizar o controle em velocidade. O controle em aceleração não será escolhido devido ao fato do controle em velocidade ser mais simples para um utilizador iniciante que quer compreender o funcionamento e facilmente utilizar a solução.

É o programa do computador que define como será feita o controle, e esta alteração é simples e poderia ser feita em qualquer momento do projeto. Optou-se por um controle relativo, conforme explicitado na seção 1.4.3 (ii). Este tipo de controle se justifica por ser o mais simples para o usuário que esta acostumado com MousePads, onde o controle é idêntico, e pelo fato de que o controle absoluto não é compatível com usuários que venham a retirar seus dedos dos sensores diversas vezes devido aos problemas apontados na seção 1.4.3 (ii).

4.2.5. Definição do tipo de retorno a ser utilizado

Um tipo de sensor pode ser mais adaptável a um tipo específico de retorno táctil ^[14]. O ScrollPad é um sensor que capta posicionamento, e não forças, portanto, o retorno através de vibrações seria funcional. O retorno já imposto pelo design sugerido, porém, sugere o retorno semi-háptico proporcionado pelo material deformável (espuma) abaixo da mesa para os dedos (Espuma 1 do Diagrama 9), que também seria funcional.

É importante considerar que os sensores estarão sobre uma mesma mesa. A qualidade do retorno semi háptico através de material deformável vem do fato de que a mesa inclina para o local onde um dedo a pressiona. Além disso, os outros dedos, por não exercerem força, não estarão sentindo o retorno semi-háptico. No caso de retorno através de vibração, esta poderia se propagar entre os dedos e para a mão. Dessa maneira, seria difícil a implementação de feedbacks independentes para todos os dedos com este tipo de design. A vibração poderia ser implementada, portanto, através de um único vibrador para o protótipo como um todo.

Ambos os retornos são viáveis e seria possível a implementação de ambos. Por questões de simplificação, optou-se portanto pelo sistema de retorno através de material deformável, pois este não necessita de feedback do computador (simplicidade) e não se propaga entre os dedos nem para a mão (individual), além de que por haver um retorno semi-háptico, obtêm-se uma melhora em relação dispositivo sem retorno ^[2], porém sem o alto custo e complexidade do retorno háptico.

4.2.6. Definição do circuito para um sensor

A instalação dos sensores tipo ScrollPad é muito simples, bastante conectar um fio em +V e outro no terra e medir-se a corrente circulante. O circuito pode ser visto no Diagrama 11:

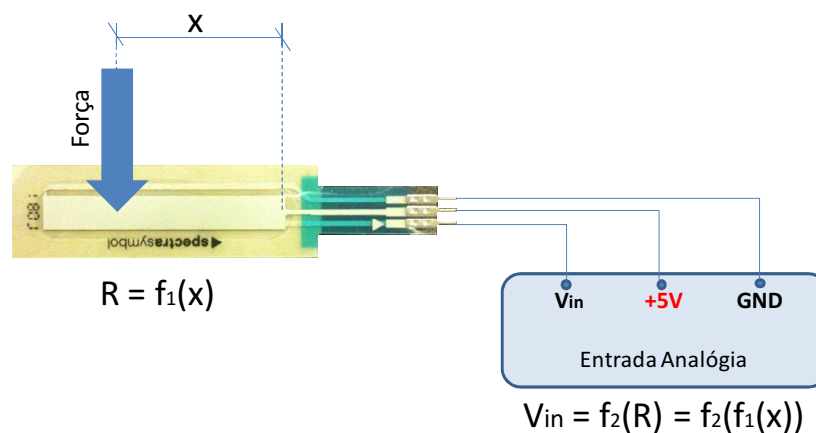


Diagrama 11 - Funcionamento de circuito do ScrollPad

4.3. Programa a ser desenvolvido

4.3.1. Descrição do programa

Pretende-se utilizar o Microsoft Visual Studio 2008 (VS08) para desenvolvimento do programa simplificado. Optou-se pela linguagem de programação com qual o autor tem familiaridade e que comporte uma interface

gráfica a partir de uma fácil implementação. O VS08 está de acordo com ambas as premissas.

Abaixo se explicará como será feita a programação. Todas as decisões selecionadas na descrição abaixo foram feitas baseadas na simplicidade de implementação desejada. Um modelo plano será utilizado para representar a mão virtual na tela do computador, conforme mostram o Diagrama 12:

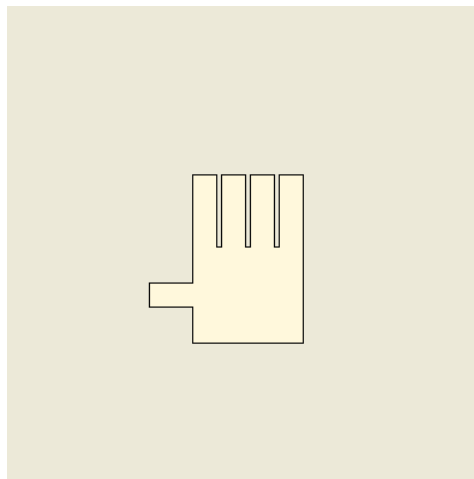


Diagrama 12 – Modelo plano da mão virtual na tela do computador

Por simplicidade, optou-se por fazer o representar o fechamento dos dedos através da diminuição dos mesmos, conforme mostra o Diagrama :

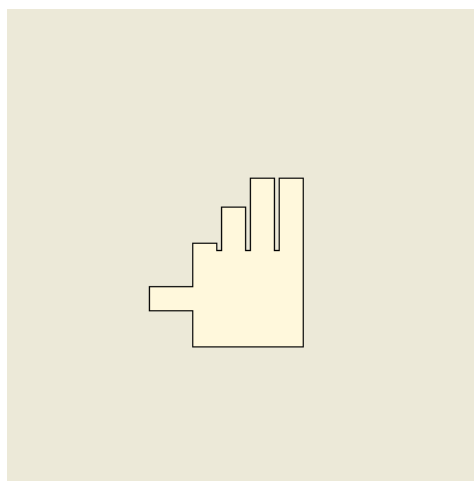


Diagrama 13 - Representação do dobramento parcial ou total de alguns dedos no modelo da mão virtual

A translação da mão nos eixos x, y e z pode ser visualizada no Diagrama 14:

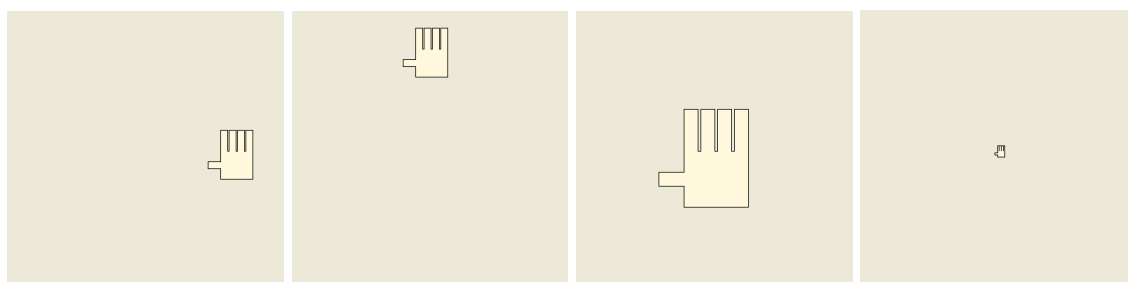


Diagrama 14 – Representação das translações nos eixos x (positivo), y (positivo) e z (positivo e negativo)

Analogamente, a rotação da mão nos eixos x, y e z pode ser visualizada no Diagrama 15:

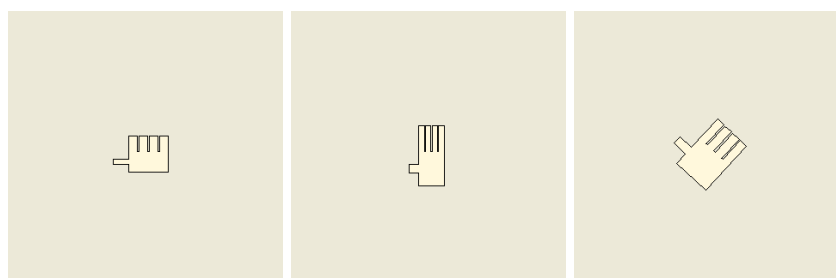


Diagrama 15 - Representação das rotações nos eixos x (positivo), y (positivo) e z (positivo)

É importante observar que, por simplificação e maior facilidade de uso no programa simplificado, optou-se pelo sistema de coordenadas fixas. Isto significa que os eixos x, y e z serão sempre fixos com respeito a seu sentido, independente de eventuais rotações ao redor deles mesmos. Desta forma, a translação no eixo y

será sempre no sentido vertical, a translação no eixo x será sempre no sentido horizontal, e a translação no eixo z será sempre evidenciada como um zoom. Analogamente, o mesmo ocorre com respeito ao sentido e orientação das rotações, porém é preciso citar que o centro de rotação deverá ser um ponto definido no centro da palma da mão virtual.

Para garantir a boa compreensão da posição da mão por parte do utilizador, foi utilizada uma cor diferente para a palma da mão, conforme mostra a seqüência abaixo, em um exemplo de rotação ao redor do eixo x (horizontal):

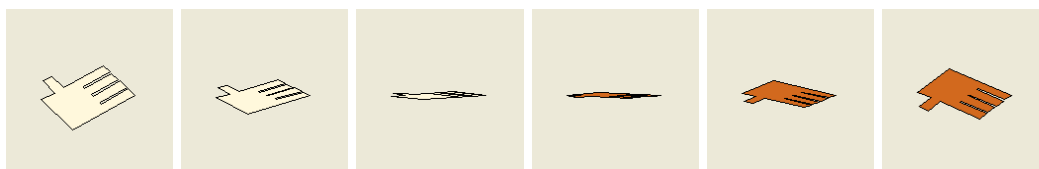


Diagrama 16 - Representação da rotação no eixo x, exibindo a diferença entre as cores de cima da mão (à esquerda, mais clara) e da palma da mão (à direita, mais escura).

Para indicar a posição 3D, três quadros serão apresentados no programa, cada um com um ângulo de visão (x, y e z), conforme mostra o Diagrama 17 abaixo:

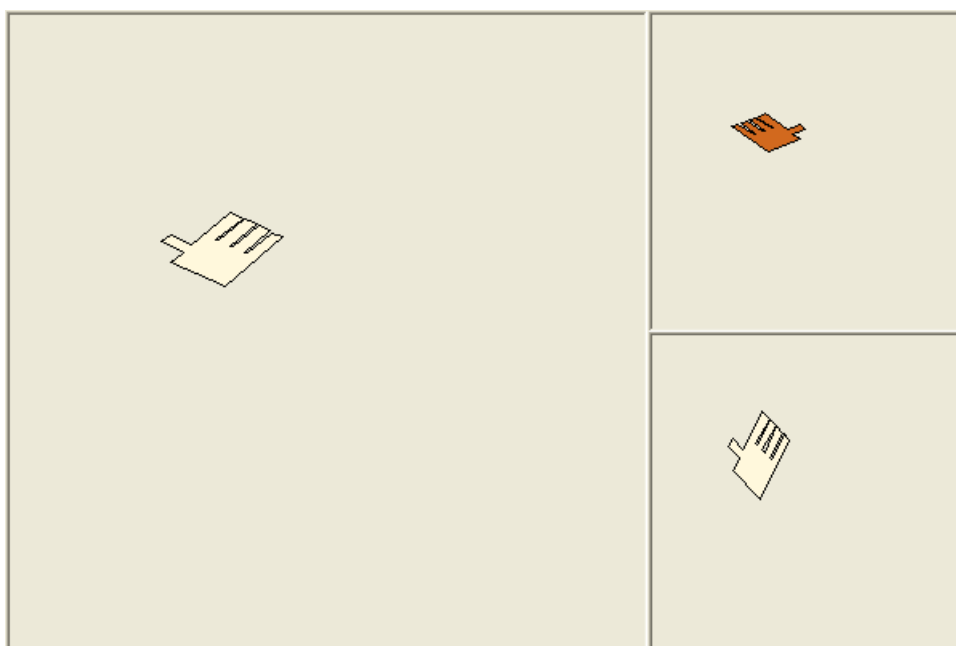


Diagrama 17 - Diagramas que serão apresentados na tela do programa

Finalmente, é importante observar que opção pelo design de mão virtual plana possui pontos fracos. O principal deles é a falta de noção de 3D devido a sua planicidade. Isso, porém, foi considerado como secundário no programa (conforme descrito em 2 - Objetivos), uma vez que se procurava uma solução simples e propriamente desenvolvida, que pudesse comprovar minimamente todas as funcionalidades do protótipo mecânico. A utilização do protótipo para simulação de uma mão virtual 3D em um ambiente virtual 3D pode ser alcançado com a utilização do programa original.

4.3.2. Definição variáveis do programa

As variáveis utilizadas podem ser de três tipos: de entrada, de desenho e auxiliares. Abaixo se descreve o processo simplificado para apresentação das imagens no frame que representa o plano projetado XY (definido como visão principal).

As variáveis de entrada definem um estado da mão virtual (coordenadas da mão virtual como um todo). Elas são calculadas após aquisição de sinais do protótipo do HandNavigator via interface USB (SpaceNavigator) ou via Placa de Aquisição (sensores). Abaixo segue uma descrição delas:

- **Variáveis que representam posição da mão virtual no frame da tela:** serão **iX e iY**. Variam em um intervalo [-100;100]. Inicialmente, todas são definidas como nulas, e devido a isso a mão se posiciona ao centro do frame que ilustra o mundo virtual na tela do programa. Essas variáveis controlam a translação da mão no plano XY.
- **Variável que representam profundidade da mão virtual no frame da tela:** será **zoom_Z1**. Varia em um intervalo [-100;100]. Inicialmente, é definida como nula, o que deixa a mão com tamanho médio na tela. Se essa variável tiver seu valor aumentado, a mão aumentará na tela, como se posicionasse mais próxima; se tiver seu valor diminuído, a mão diminuirá na tela, como se ficasse mais longínqua. Esta variável controla, portanto, a translação da mão no sentido do eixo Z.

- **Variáveis que representam a rotação da mão:** serão **ang_X**, **ang_Y** e **ang_Z**, que variam no intervalo [-180;180] no espaço virtual. Os valores representam a rotação em graus nos eixos x, y e z virtuais. É importante observar que o centro de rotação na mão é constante e varia junto com as variáveis de translação acima mencionadas.
- **Variáveis que representam a abertura ou fechamento dos dedos:** serão variáveis com nomes na forma **trkDedo1**, **trkDedo2**, **trkDedo3** e **trkDedo4**, representando respectivamente a abertura do Dedão, Indicador, Médio e Anelar / Mínimo (mais uma vez lembrando que estes dois dedos tem movimentos acoplados). Seus valores podem indicar que o dedo está completamente aberto (valor igual a 0) ou completamente fechado (valor igual a -100), ou ainda valores intermediários que representam posições intermediárias.

O outro tipo de variável do programa foi chamado de variável de desenho. As variáveis de desenho são responsáveis por interpretar os dados das variáveis de entrada e transformá-las em coordenadas dos vértices da figura geométrica que aparece na tela representando a mão virtual. Para simplificar, ao invés de uma variável por vértice, foi utilizado um vetor de pontos, conforme descrito abaixo:

- **Vetor de pontos que representa os vértices da mão:** A mão será representada como uma figura geométrica plana na tela. Os vértices desta figura plana são representados por pontos com coordenadas X e Y. Esses vértices são todos armazenados no **vetor p de pontos**, no qual cada ponto pode armazenar uma coordenada X e uma coordenada Y. Os vértices da mão virtual que aparece na tela são apresentados na Figura 38 abaixo.

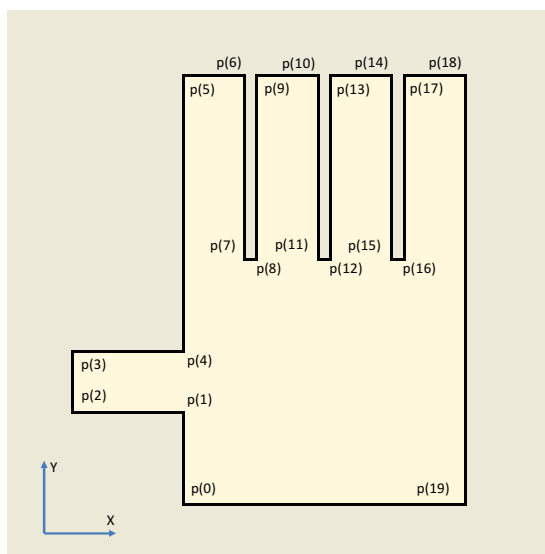


Figura 38 – Vetor $p(19)$ representa os vértices da mão

O vetor p de pontos é calculado a partir de uma base de pontos que representam a mão em sua planicidade. Esta base chamada **molde** é um vetor de pontos com coordenadas X e Y que possui os vértices calculados em função das variáveis auxiliares. Esta base molde é calculada no início do programa e não poderá ser alterada durante sua execução. As variáveis auxiliares servem para o dimensionamento da mão virtual que aparece na tela. Elas aparecem na Figura 39 abaixo, que também representa entre os dedos maior e anular o centro de rotação definido que será definido para a mão. A função responsável pela representação da mão virtual na tela utiliza os valores de cada ponto de **molde(19)** e calcula **$p(19)$** em função das variáveis de entrada e da dimensão do painel de representação.

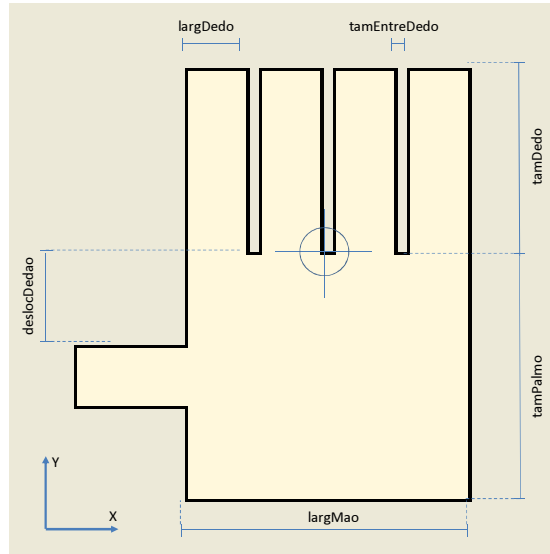


Figura 39 – Variáveis auxiliares para cálculo de molde(19)

Finalmente, ainda foi gerado código que permitisse saber quando a mão estaria mostrando sua palma (e não sua parte superior) a partir da análise da análise do resultado, no caso do plano XY, de $\cos(\text{ang_x}) * \cos(\text{ang_y})$. A cor do polígono representando a mão virtual se altera em função deste resultado, e assim consegue-se a projeção de uma mão virtual com diferentes cores em cada um de seus lados.

4.3.3. Tratamento do tipo de controle do SpaceNavigator

Para o controle da mão virtual, o controle será em velocidade. Quando se detecta um input de sinal do SpaceNavigator, o programa é responsável por incrementar no tempo a posição / rotação atual da mão virtual. Ao detectar um input do SpaceNavigator em uma direção e um sentido o programa se encarrega de incrementar posições na representação de maneira constante. Dessa forma, o controle em velocidade é implementado.

4.3.4. Tratamento do tipo de controle dos sensores dos dedos

No caso dos sensores, o controle será em velocidade relativa, conforme já discutido na seção 4.2.4. Para se implementar um controle relativo no sensor, utilizou-se o esquema abaixo (as variáveis utilizam nomes fictícios representativos):

- Seja `posicao_sensor` a posição captada pelo sensor;
- Seja `posicao_dedo` a posição relativa calculada do dedo virtual.

O calculo da `posicao_dedo` em função da `posicao_sensor` em um controle por velocidade relativa pode ser implementado em um Timer com o seguinte código:

```

SE (aquisição_sensor = 0) ENTÃO (definir_base = 1)
(se o sensor captar 0, o programa está pronto para definir uma nova base)

SENÃO
(só se entra nesse caso se definir_base = 1, ou seja, se a posição era igual a 0 e mudou)
  SE (definir_base = 1) ENTÃO (base = aquisição_sensor) E (definir_base = 0)
  (se define a base)
  SENÃO (posição_dedo = posição_dedo(anterior) + posicao_sensor - base)
  (se define a variação relativa em função da base definida)

```

Será ainda necessário impor condições de contorno para evitar exceções no programa que aceita valores de posição de dedos entre 0 (aberto) e -100 (fechado).

4.3.5. Estrutura Geral do Programa

As atualizações de cada uma das variáveis de estado em função das variáveis de entrada se darão a cada tempo t_0 definido por um Timer. O tempo t_0 de aquisição é definido como uma propriedade deste componente. As funções de aquisição atualizam as variáveis de estado de acordo com os últimos inputs recebidos.

4.3.6. Diagrama das funções que serão utilizadas

Para o programa simplificado, pode-se traçar esquema de funções e sub-rotinas segundo o Diagrama 18 a seguir:

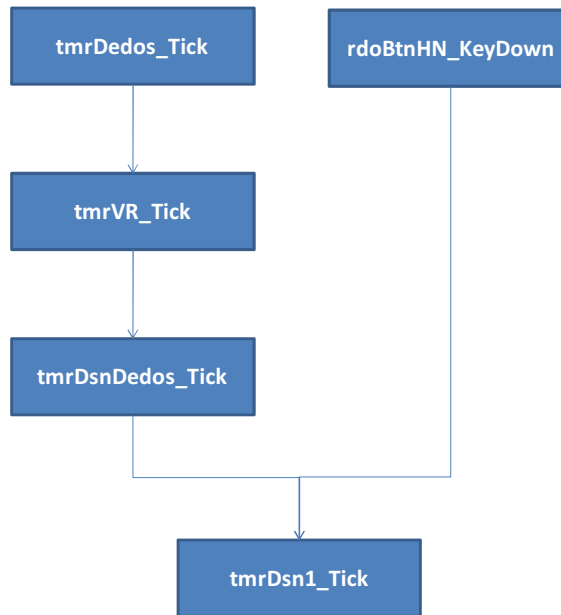


Diagrama 18 – Diagrama de funções que serão utilizadas no programa (simplificado)

As descrições das sub-rotinas acima seguem abaixo:

- **Sub tmrDedos_tick:** esta função faz a aquisição do sinal que o sensor faz aquisição naquele momento. É ativada por um timer, que a executa em tempos freqüentes para determinar o input do usuário;
- **Sub tmrVR_tick:** esta função é responsável pela implementação do controle em velocidade relativa dos dedos virtuais. Esta sub-rotina é ativada por um timer que a executa em tempos freqüentes. A função calcula a posição dos dedos virtuais em função do input do sensor (que a sub-rotina tmrDedos_tick fez a aquisição e gravou os dados na memória) utilizando o conceito de velocidade relativa;
- **Sub tmrDsnDedos_tick:** esta função está presente para que eventuais outros controles (que não sejam em velocidade relativa) possam ser implementados. Esta sub-rotina tem um input relativo ao controle que deve ser selecionado. Ela faz com que os dados que serão utilizados para desenhar a mão virtual na tela sejam alimentados por dados relativos ao controle escolhido;
- **rdoBtnHN_KeyDown:** analogamente a aquisição dos dedos, esta sub-rotina faz a aquisição do sinal do SpaceNavigator. O sinal do HandNavigator é processado pelo driver RBC9-SpaceNav. Este driver é configurável e tem a

capacidade de fazer com que o computador enxergue o SpaceNavigator como um periférico comum (no caso deste trabalho, como um teclado). Desta forma, esta sub-rotina é ativada pelo evento KeyDown do VS08.

- **Sub tnrDsn1_tick:** após as informações de aquisição e tipo de controle terem sido tratadas nas funções anteriores, esta sub-rotina avalia as variáveis de entrada previamente calculadas (que definem o estado da mão) e calcula as variáveis de desenho. Finalmente, o programa insere a imagem calculada na tela;

5. Desenvolvimento

5.1. Desenvolvimento da solução

5.1.1. Desenvolvimento modelo 3D (CATIA)

As imagens do Diagrama 19 abaixo mostram as peças desenvolvidas com programa de desenho 3D (CATIA V5 release 19) e sua ordem de montagem.

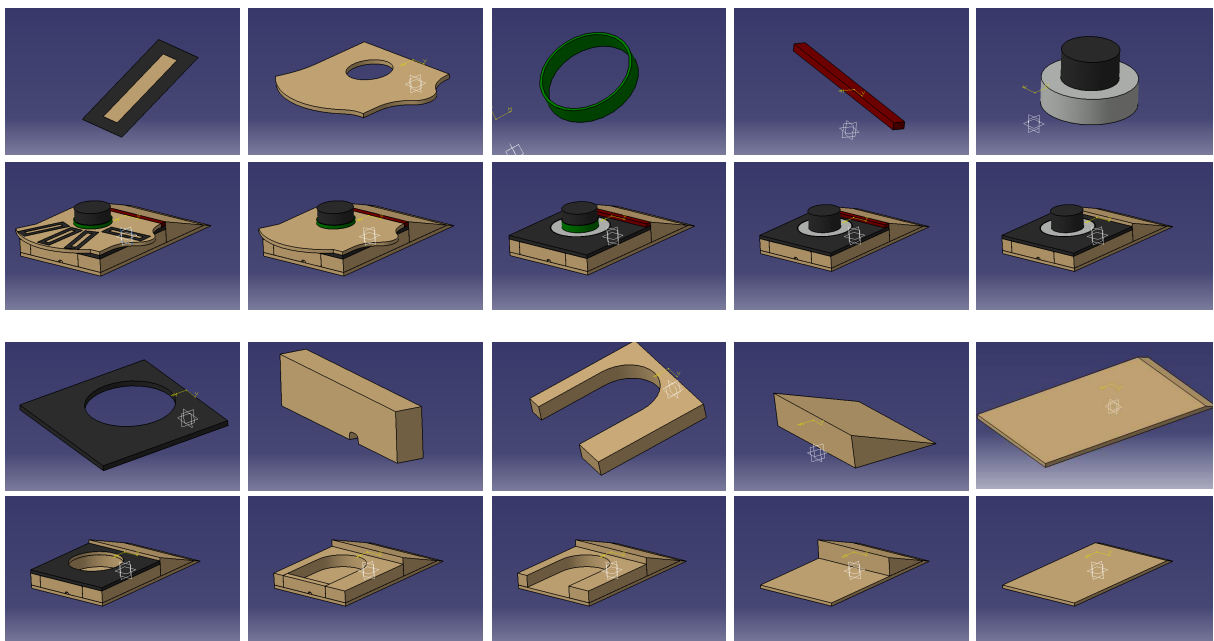


Diagrama 19 - Peças do protótipo feitas em CATIA v5r19 e a ordem de montagem do protótipo final

5.1.2. Projeto de circuito eletrônico

Conforme previamente citado, o sensor ScrollPad a ser utilizado não necessita de circuito complexo. Tal circuito já fora apresentado no Diagrama 11. A manufatura de uma placa, neste caso, torna-se desnecessária. Os fios provenientes do sensor são ligados diretamente na placa eletrônica, conforme o esquema do Diagrama 20 abaixo:

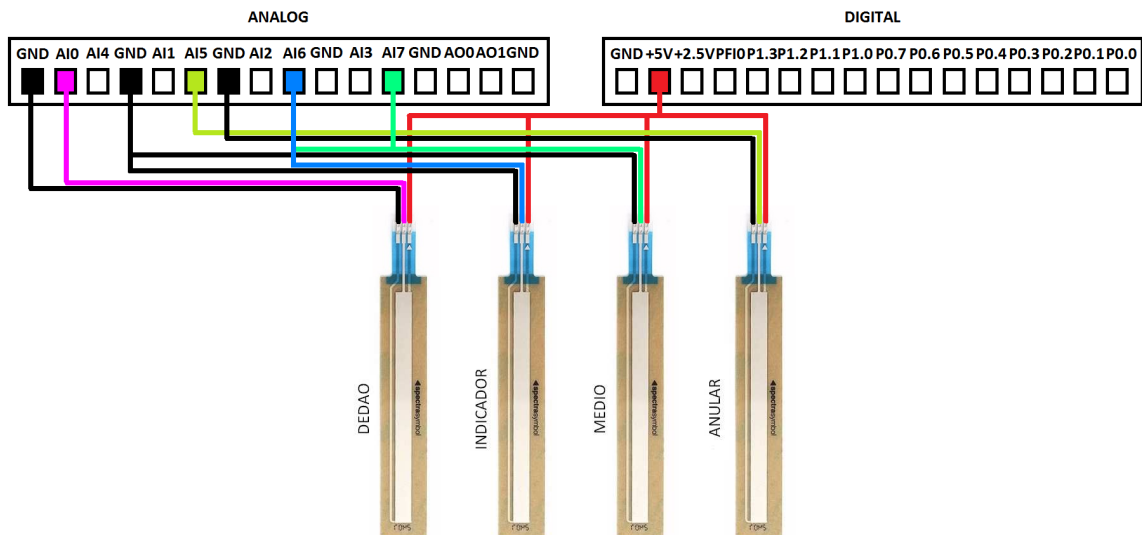


Diagrama 20 – Circuito final indicando portas da placa de aquisição NI-USB6009 a serem utilizadas

O circuito representado no diagrama acima serve apenas para o programa simplificado, uma vez que o programa original desenvolvido na França possui outra configuração de entrada.

5.1.3. Projeto do programa

A organização estrutural do programa e suas funções já foram discutidas na seção 4.3 - *Programa a ser desenvolvido*. As decisões tomadas naquele momento foram feitas para se obter um programa simples.

5.2. Desenvolvimento do protótipo final

5.2.1. Construção periférico

As figuras abaixo mostram o protótipo final (Figura 40) e seu projeto (Figura 41). Pode-se notar a fidelidade de construção das peças e do protótipo.



Figura 40 - Protótipo construído

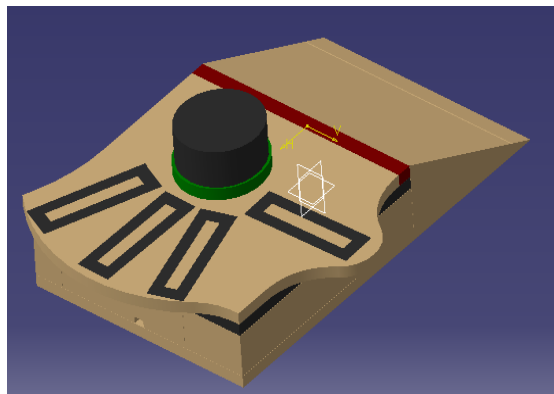


Figura 41 - Projeto 3D em CATIA do protótipo

Pequenas mudanças que foram impostas durante ou após a construção:

- Modificação da base de apoio para o pulso para ser feita uma extrusão triangular em apenas uma peça. A base de apoio diminuiu e sua parte correspondente que estava abaixo da antiga base de apoio para o pulso foi anexado a forma final para da nova base de pulso para construção;
- Devido à alteração acima, foi feita interligação lateral entre a base de apoio para o pulso e o resto do dispositivo. Antigamente, a ligação era feita inferiormente a base de apoio, pois lá havia uma madeira interligada ao resto do dispositivo;
- Aumento da grossura de Espuma 1 devido ao fato que a diminuição feita na mesa dos sensores não foi pequena o suficiente para que a rotação do

SpaceNavigator não tocasse superiormente a base de apoio para o pulso. Devido a este fato, o HandNavigator se tornou mais alto;

- Por se tornar mais alto, uma espuma foi colocada estrategicamente sobre a base de apoio dos pulsos para que estes pudessem efetivamente ficar apoiados;
- A direção dos sensores foi selecionada de acordo com a análise instantânea do posicionamento dos dedos em fechamento sobre a mesa (houve modificação do sentido do sensor do dedão).

5.2.2. Construção circuito eletrônico

Conforme já especificado, não será necessária construção de placa eletrônica. A inclusão da fiação dos sensores no projeto foi feita exteriormente. A conexão com os sensores tipo ScrollPad foi feita através de conectores para evitar danificação dos mesmos. A Figura 42 abaixo mostra o protótipo completo montado e pronto para funcionamento:

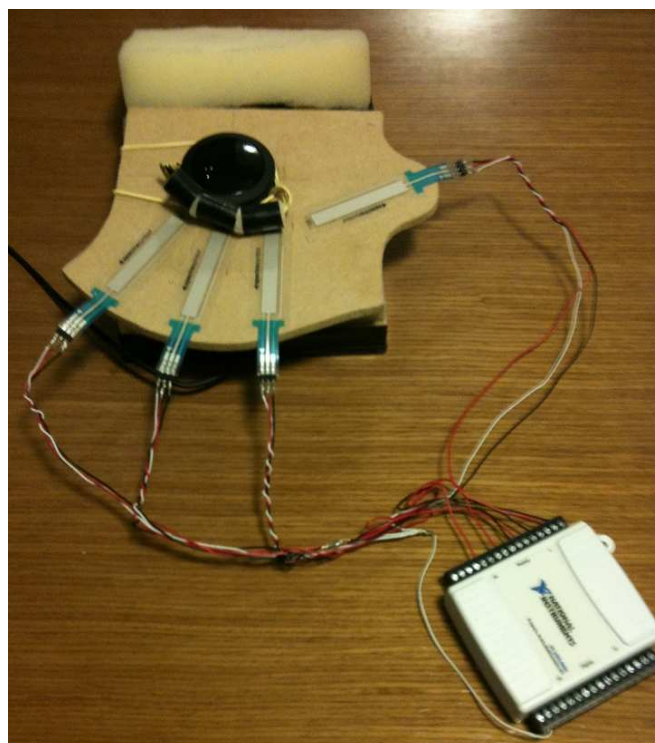


Figura 42 – Protótipo final conectado ao circuito

5.2.3. Programação interface computacional

A Figura 43 abaixo mostra o programa final implementado:

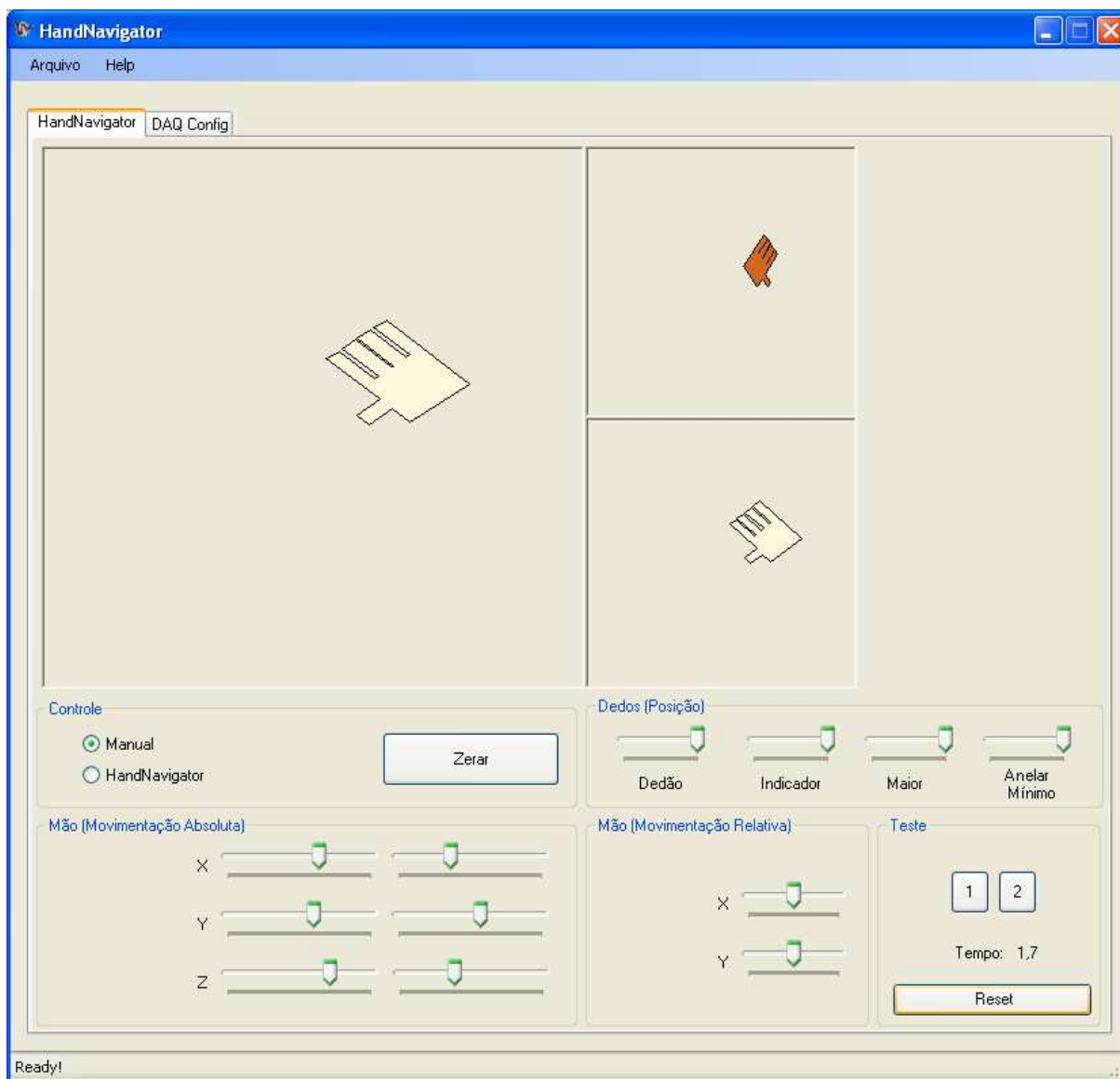


Figura 43 – Tela final do programa simplificado

O Anexo I mostra o código implementado em VS08 para o programa simplificado representado na Figura 43 acima. A Figura 44 abaixo mostra o programa simplificado em utilização



Figura 44 – Programa simplificado em utilização

5.2.4. *Integração com interface antiga*

A integração com o programa antigo foi realizada. A única alteração relevante feita ao programa original foi para que suportasse o sensor tipo ScrollPad e que o controle fosse feito por velocidade. Além disso, as duas das conexões da placa devem ser alteradas. A Figura 45 abaixo mostra o programa antigo sendo utilizado com o novo protótipo:

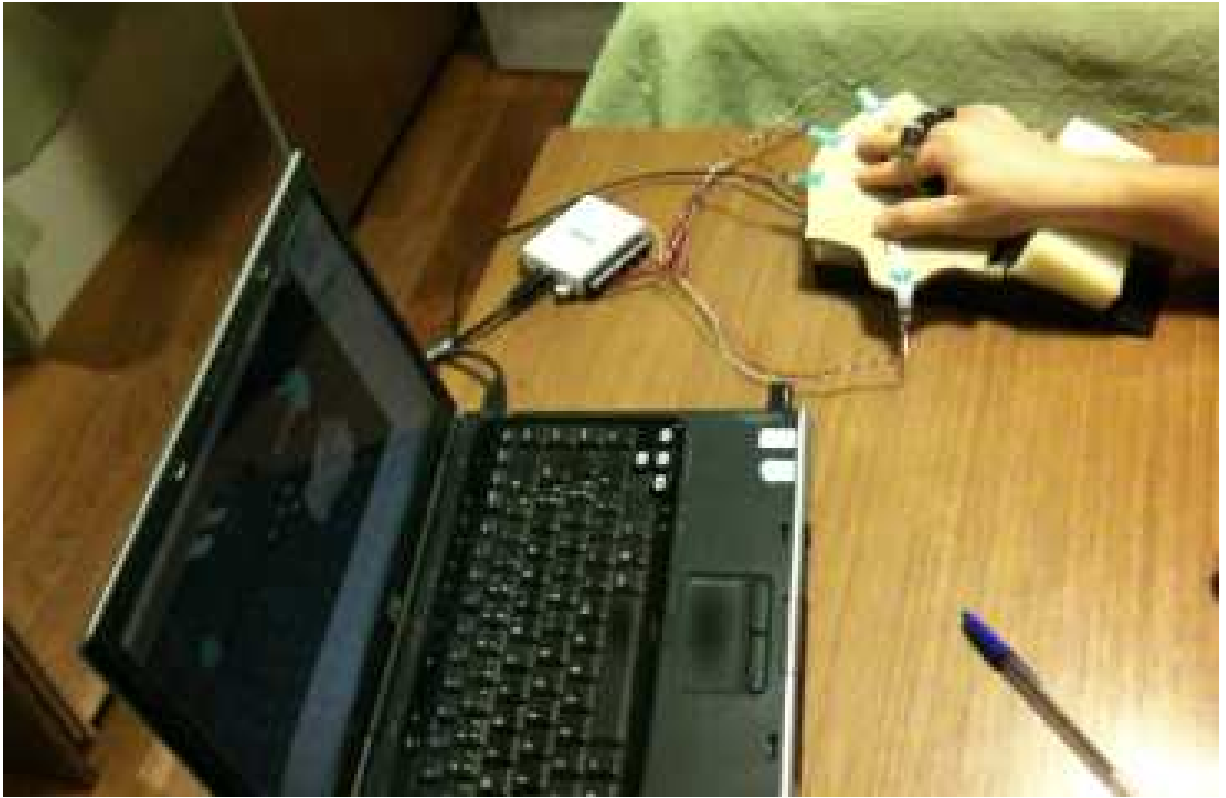


Figura 45 – Protótipo sendo utilizado com programa antigo

6. Validação

Os testes de validação vêm descritos abaixo. Eles serão executados com o programa simplificado atual:

- Teste 1: Tempo para abrir e fechar completamente todos os dedos utilizando controle com velocidade relativa;
- Teste 2: Tempo para fazer circuito de um S deitado na tela sem tocar nos quadrados desenhados na tela. A mão será mantida fixa durante o teste.

Finalmente, também foi elaborada uma pesquisa para averiguar os principais problemas identificados pelos usuários. As opções incluíam: dificuldade de visualização 3D, sensibilidade do controle da mão, sensibilidade do controle dos dedos, interferência entre controle da mão e dos dedos.

Foram avaliadas 17 pessoas e os resultados dos testes seguem no Gráfico 1 e no Gráfico 2 abaixo:

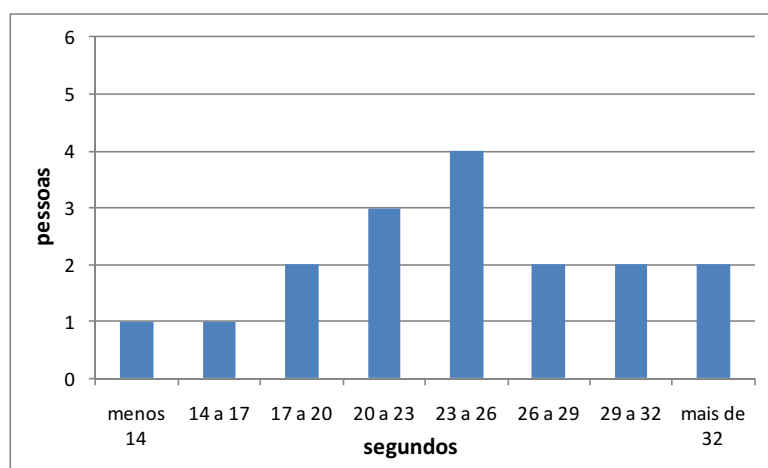


Gráfico 1 - Resultados Teste 1

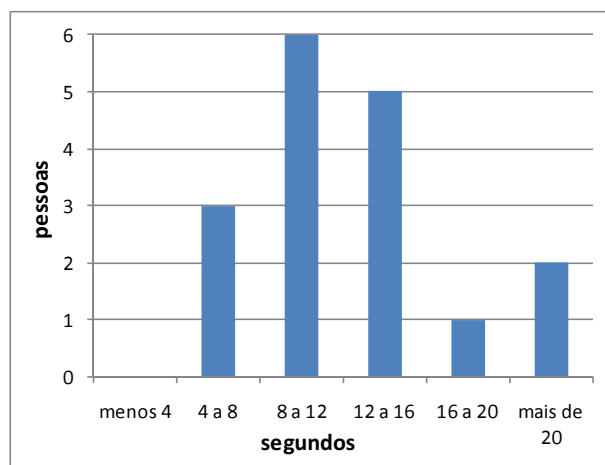


Gráfico 2 – Resultados Teste 2

O teste 1 mostra uma média de 23 a 26 segundos para a execução da tarefa previamente definida. Duas pessoas precisaram de mais de 32 segundos para executar tal tarefa (os resultados oficiais foram 60,7s e 101,3s). A maior idade destas pessoas talvez justifique pelo fato de estarem menos acostumadas com computadores ou ambientes 3D.

No teste 2 a maioria dos avaliados conseguiu executar a tarefa entre 8 e 16 segundos. Novamente, pode-se notar que há resultados acima de 20 segundos. Esses resultados (24,3 e 28,3 segundos) não foram tão extremos quanto o anterior, porém foram exatamente das mesmas pessoas que precisaram de mais de 32 segundos para executar o teste 1. Dessa forma, o fato de serem pessoas que desconheciam a simulação de ambientes virtuais pesou novamente.

Um dado que chama atenção é a grande disparidade entre a média do teste 1 e seu valor extremo quando comparado ao teste 2. Esperava-se o contrário devido ao fato de que os sensores que devem ser utilizados no teste 2 são unidirecionais, enquanto o SpaceNavigator pode fazer a aquisição de diversos graus de liberdade. O que provavelmente pesou foi o fato de o driver do sensores terem sido configurados com uma constante de multiplicação de variação de velocidade possivelmente baixa.

Finalmente, a pesquisa mostrou os resultados representados na Gráfico 3 abaixo. As pessoas eram permitidas de utilizar o protótipo durante três minutos

antes de respondê-la. As pessoas que respondiam que o maior problema era a dificuldade de visualização 3D, foi perguntado o segundo maior problema. Fez-se isto pois o programa foi desde o começo deste trabalho descrito como simplificado, e portanto a boa imagem 3D não era um objetivo. Dessa forma temos dois resultados: em azul, incluindo a opção de visualização 3D como principal problema, e em vermelho sem sua inclusão.

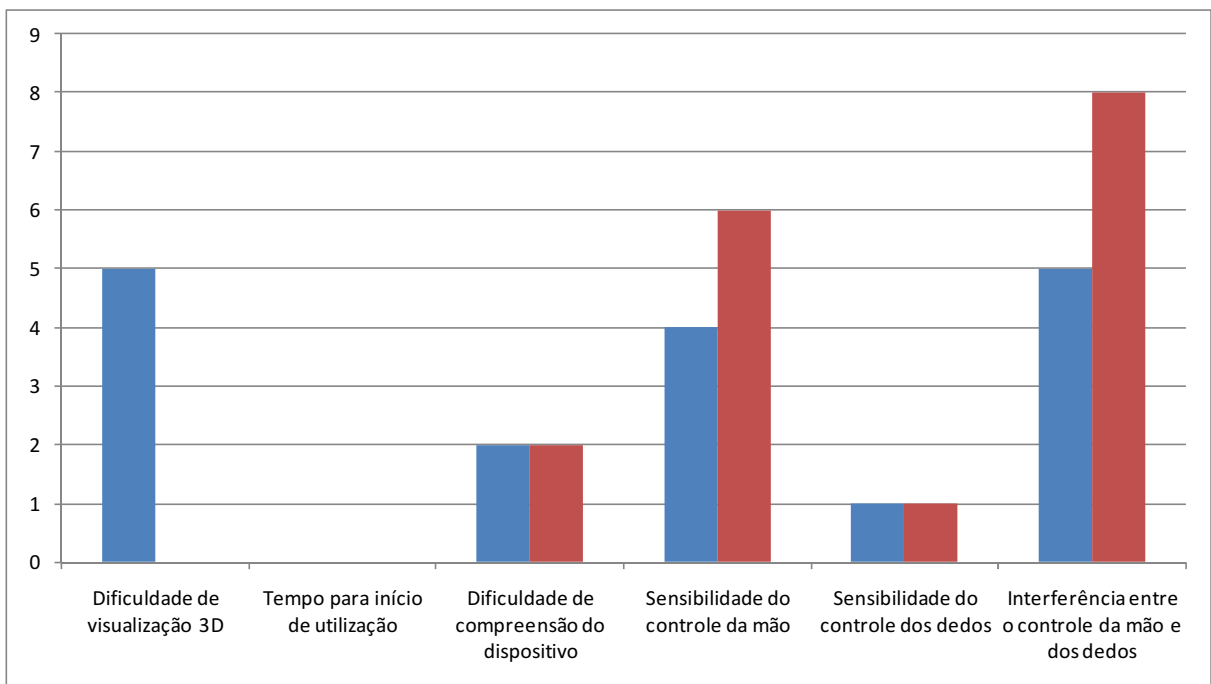


Gráfico 3 – Pesquisa opinião sobre o HandNavigator

A partir da análise dos gráficos acima, pode-se evidenciar alguns pontos:

- O controle dos dedos, mesmo que mais demorado, é fácil de se utilizar e pouco sensível. A sensibilidade é considerada um aspecto negativo uma vez que o principal objetivo é obter controlabilidade do protótipo e não a agilidade de utilização, conforme descrito nos objetivos. Desta forma, o sensor pode ser avaliado como um ponto forte do projeto.
- O tempo para início de utilização não foi citado como um problema por nenhum usuário. Um medo comum antigo deste projeto, muitas vezes citado neste trabalho, foi um maior tempo para início da utilização. Porém, mesmo com a utilização do prendedor, isto não foi um problema e o prendedor pode ser considerado um ponto forte do

projeto que auxilia a utilização sem causar grandes problemas ao projeto.

- A interferência entre o controle da mão e dos dedos foi algo crítico, sendo o ponto mais citado. É facilmente perceptível no projeto que muitas pessoas quer-se movimentar os dedos virtuais e acaba-se movimentando a mão. O contrário, isto é, tentar movimentar as mão e acabar movimentando os dedos, não foi muito comentado, porém. A interferência entre os graus de liberdade já era um problema em diversas versões anteriores, conforme evidenciado na seção 1.6 - Problemas das soluções existentes para captura do movimento das mãos.
- A sensibilidade do controle da mão foi um ponto bastante citado também. Além da questão óbvia de que o SpaceNavigator é extremamente sensível e tentou-se criar DeadZones que coibissem tal sensibilidade extrema, deve-se observar que a dificuldade de visualização 3D pode ter eventualmente afetado a manipulação da mão no espaço virtual, levando muitas pessoas a assinalar a opção de dificuldade do controle da mão como principal problema. Uma eventual aprimoração para controle por velocidade relativa variável, por parte do programa, também poderia ajudar, uma vez que, de acordo com a implementação atual, uma vez adquirido o sinal do SpaceNavigator, sua velocidade era constante e talvez alta demais para o início de um movimento, que acabava por se tornar muito brusco.
- A visualização 3D evidentemente poderia ser melhor, porém, novamente, não era um objetivo deste projeto. Foi devido a este fato que optou-se por perguntar uma segunda opção as pessoas que escolheram tal item como principal problema do projeto. A idéia do programa é que se pudesse evidenciar as funções do protótipo junto ao programa, além de que ele permitisse a realização de testes, o que foi obtido.

7. Conclusão

Algumas considerações importantes a se fazer sobre a solução final são:

Sobre o periférico

O projeto inovou ao propor um design diferente dos que foram tradicionalmente desenvolvidos, o que foi resultado de uma longa análise não evolutiva porém qualitativa ao analisar as qualidades de uma boa quantidade de soluções atualmente disponíveis.

Um ponto importante que foi visado neste projeto foi a questão da possibilidade de acionamento do movimento da mão e dos dedos de forma simultânea. Todos os designs propostos neste trabalho propunham isto, assim como as versões antigas do HandNavigator já possuíam tal característica, e ela foi mantida.

A construção do dispositivo foi bem executada. O SpaceNavigator é porém um dispositivo altamente sensível, e as grandes DeadZones impostas durante a configuração de seu driver (software) também foram conseqüência de características menos precisas do periférico. A idéia da espuma abaixo da mesa, idealmente, possa vir a ser aproveitada uma vez que ela colaborava para evitar que nem o acionamento dos sensor e nem o peso da mesa afetassem no controle da mão virtual.

Uma coisa que sempre foi evitada foi a utilização de um prendedor, pois este poderia acarretar tempo extra para início de utilização. Constatou-se que o pouco tempo perdido era insignificante para os utilizadores, e este tempo talvez fosse significativo para casos onde haja necessidade de colocar uma luva ou de grande e demorada preparação para utilização do protótipo.

O apoio de pulso que foi inserido ao fim colaborou altamente para a confortabilidade do dispositivo. O dispositivo, mesmo que já possuísse uma base para apoiar o pulso, esta ficava abaixo da altura ideal.

Entre os pontos que ficaram a desejar, analisando os objetivos propostos, pode-se citar a portabilidade. O dispositivo é pequeno, porém é relativamente grande para se carregar ou se comparado ao tamanho de um laptop, além de ser pesado também. Ele não é como um Mouse, por exemplo, que se pode levar para qualquer local sem maiores problemas. Sua instalação, devido a placa eletrônica, também contribui para sua característica não-portátil.

Um eventual evolução deste projeto poderia incluir um feedback semi-háptico a partir de vibradores, mesmo que apenas um vibrador pra mão e todos os dedos.

Sobre o programa e o circuito

A possibilidade de não haver circuito complexo devido a escolha do sensor do tipo ScrollPad facilitou o projeto e sua construção, sem afetar a usabilidade do projeto pois ele é um sensor de alta qualidade. Além disso, tal qualidade do sensor pode ser verificada devido ao pequeno esforço necessário para ativá-lo.

A escolha do controle em velocidade por parte dos sensores foi de bom proveito para o projeto, uma vez que os dedos podem controlar os dedos virtuais a partir de pequenos movimentos procuram afetar o controle da mão virtual o mínimo possível.

Voltando a falar sobre portabilidade, a opção por circuito externo foi um ponto negativo neste aspecto. Finalmente, ficou claro pela pesquisa de validação que a visualização 3D ficou a desejar.

Sobre os objetivos em geral

Neste momento não se irá avaliar os pontos positivos/negativos dos projeto, conforme feito acima, mas apenas comparar se os objetivos foram cumpridos. Desta forma:

- Conseguiu-se fazer uma análise mais qualitativa e menos evolutiva;
- Tem-se o SpaceNavigator para o controle semi-háptico das mãos;
- Controlabilidade da mão virtual deve ainda melhorar;
- Portabilidade foi afetada em relação aos protótipos anteriores;
- Dispositivo, circuito e programas simples, mas levam tempo para instalação;
- Programa simplificado para validação em 2D foi executado apesar de que não teve boa aceitação durante a validação.

Finalmente, pode-se dizer que os objetivos do projeto pautaram todo o desenvolvimento e construção do protótipo e do programa.

8. Referências Bibliográficas

- [1] EVASION. Evasion - Laboratoire Jean Kuntzmann, 2010. Disponível em: <<http://www-evasion.imag.fr/>>. Acesso em Abr / 2010.
- [2] KRY, P. G.; PIHUIT, A.; BERNHARDT, A. et al. *HandNavigator: Hands-on interaction for desktop virtual reality*. ACM symposium on Virtual Reality Software and Technology, 2008.
- [3] PMR, POLI USP. Departamento de Engenharia Mecatrônica. <<http://www.pmr.poli.usp.br>>. Abr / 2010.
- [5] INSKO, B. E. *Passive haptics significantly enhances virtual environments*. Technical Report, University of North Carolina, 2001.
- [6] LECUYER, A.; COQUILLART S.; KHEDDAR A. et al. *Pseudo-haptic feedback: can isometric input devices simulate force feedback?* IEEE Virtual Reality, pp. 83–90, New Brunswick, NJ, 18–22 Mar / 2000.
- [7] ZHAI S.; MILGRAM P.; BUXTON W. *The influence of muscle groups on performance of multiple degree-of-freedom input*. CHI, 1996. pp. 308–315.
- [8] PAI, D. K.; VANDERLOO, E. W., SADHUKHAN, S.; KRY, P. G. *The tango: A tangible tangoreceptive whole-hand human interface*, in *Joint Eurohaptics .Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pp. 141–147, 2005.
- [9] IBM. User System Ergonomics Research. Disponível em: <http://www.almaden.ibm.com/cs/user/tp/tp.html>. Acesso em Nov / 2010.

- [10] STURMAN D. J.; ZEITZER D.; PIEPER S. *Hands-on interaction with virtual environments*. ACM Siggraph symposium on User interface software and technology, pp. 19–24, Nova-York, NY, 1989.
- [11] KIM S.; ISHII M.; KOIKE Y. et al. *Development of tension based haptic interface and possibility of its application to virtual reality*. ACM symposium on Virtual Reality Software and Technology, pp. 199–205, 2000.
- [12] BOUZIT, M.; BURDEA G.; POPESCU, G.. et al. *The rutgers master II-new design force-feedback glove*. IEEE/ASME Transactions on Mechatronics, vol. 7, Jun / 2002.
- [13] CHARDONNET, Jean Rémy. *Designing and evolving hands-on interaction prototypes for virtual reality*, 2010.
- [14] AMARO A. C.; CHARDONNET J. R.; LEON J. C.. et AL. *HandNavigator: HandNavigator: Hands on interaction for desktop virtual reality*. Defesa de Trabalho de Conclusão de Curso Francês no Institute Polytechnique de Grenoble, INRIA, EVASION, Sep / 2009.