

Gesture3DFramework: A Generic Gesture-based Interaction Middleware applied to 3D Environments

Diego Passos Costa
Computing and Systems Graduate Program
Salvador University (UNIFACS)
Salvador, Bahia, Brazil
diegopassoscosta@gmail.com

Paulo N. M. Sampaio
Computing and Systems Graduate Program
Salvador University (UNIFACS)
Salvador, Bahia, Brazil
pnms.funchal@gmail.com

Valéria Farinazzo Martins
School of Computing and Informatics
Marckenzie Presbyterian University
São Paulo, São Paulo, Brazil
valfarinazzo@gmail.com

Abstract—The technological advances provide the development and wide adoption of different kinds of human-machine interfaces, which leads to the creation of new applications such as those based on multimedia and virtual reality (3D). In particular, the proposal of interaction metaphors applied to 3D environments which aim at replicating real world concepts into the virtual environment, facilitates user's interaction. The utilization of gestural interaction metaphors within a 3D environment can turn the user experience more familiar and concrete, making the training curb smaller. However, in order to apply interaction metaphors it is necessary their classification and generalization, so that they can be widely deployed in different applications. This paper introduces the development of a generic and customizable solution for the mediation of user gestural interaction (selection, manipulation and navigation) with heterogeneous rendering engines for virtual reality environments. This solution, called *Gesture3DFramework*, allows the users context and gesture-metaphors configuration to be easily customized so that it can be adaptable to multiple 3D virtual environments. With *Gesture3DFramework*, the final user (and developer) will be provided with a higher level of abstraction when it comes to the development of interactive virtual reality applications, since once the configuration directives have been described, the system will adapt itself to the specific interaction routines of the applied rendering engine.

Keywords— *Virtual Reality, Gesture Interaction, Selection, Manipulation, Navigation, Immersion, 3D environment*

I. INTRODUÇÃO

Metáfora é uma figura de linguagem, uma forma de se expressar, onde um termo é substituído por outro [22]. A principal razão para o uso de uma metáfora é a busca por uma melhor comunicação, facilitar o processo de entendimento, ou proporcionar alguma nova experiência para os participantes. No caso da comunicação humano-máquina, a máquina ainda não possui a mesma capacidade cognitiva do ser humano, oferecendo muitas limitações para a comunicação. Portanto, os desafios são maiores quando o processo de interação (seleção, manipulação e navegação) humano-máquina não ocorre através dos dispositivos mais convencionais eg.: Mouse e Teclado [2].

No caso da interação com ambientes computacionais de Realidade Virtual, também conhecidos como ambientes tridimensionais (3D), ela deve promover (em grande parte das aplicações) um maior grau de imersão para o usuário, proporcionando a ele uma experiência mais realista e rica.

A metáfora de interação em um ambiente 3D torna a experiência do usuário mais familiar e concreta, diminuindo o tempo de treinamento do usuário. Da mesma forma, conforme argumentado por diferentes autores

[1][3][4][9][12], a metáforas de interação aplicadas aos ambientes virtuais (3D) também visam replicar os conceitos que já são conhecidos pelo usuário em um contexto real para o contexto do ambiente virtual. A metáfora, se for familiar para o usuário, pode tornar a interação muito mais intuitiva [37].

Diferentes contribuições são apresentadas na literatura relacionadas ao estudo, proposta e implementação de metáforas de interação com ambientes 3D [1][3][4][9][12][30][31][32][34]. No entanto, a maior parte desses trabalhos aplicam as metáforas utilizadas orientadas ao contexto específico da aplicação implementada. Além do mais, nesses casos as *Application Programming Interfaces* (APIs) desenvolvidas para as metáforas não são disponibilizadas ou reutilizadas para a construção de metáforas mais elaboradas ou para a aplicação sistemática em outras aplicações de Realidade Virtual.

A partir do estudo da literatura é observada a necessidade de soluções que permitam a prototipação mais rápida de mecanismos genéricos para a interação gestual. Uma alternativa para esses mecanismos pode ser proposta através da disponibilidade de bibliotecas que permitam a implementação de diferentes metáforas para a interação gestual, e que sejam facilmente adaptáveis a qualquer plataforma de desenvolvimento ou motor de renderização. A partir dessa necessidade, surgiu a proposta do desenvolvimento de uma *middleware* chamada *Gesture3DFramework*.

O *Gesture3DFramework* visa ser um meio facilitador para que os desenvolvedores sejam capazes de abstrair conceitos complexos de interação existentes em diferentes motores de renderização 3D, de forma que tais interações possam ser mais facilmente aplicadas. A aplicação do *Gesture3DFramework* vem impactar na redução da dificuldade de um usuário final em interagir com múltiplos motores de renderização 3D, uma vez que sua configuração contextual poderá ser lida em qualquer ambiente que use o *Gesture3DFramework* como intermediador.

O objetivo deste artigo é, portanto, apresentar as principais etapas do desenvolvimento do *middleware Gesture3DFramework*. Portanto, as principais contribuições apresentadas neste trabalho são:

- I. A partir do levantamento na literatura, a identificação e compreensão das características de uma série de metáforas disponíveis para a interação (seleção, manipulação e navegação);
- II. Apresentação da arquitetura implementada da solução *Gesture3DFramework*;
- III. Ilustração da modelagem de dados e utilização das mensagens declarativas de configuração do

Gesture3DFramework e da mensagem de integração com os motores de renderização, e;

- IV. Desenvolvimento e ilustração de um cenário de estudo de caso da aplicação do *Gesture3DFramework* para viabilizar a interação gestual utilizando uma metáfora de interação.

Este artigo está estruturado da seguinte forma: a seção 2 apresenta o levantamento das metáforas para a interação com ambientes 3D. A seção 3 apresenta os detalhes de modelagem e implementação do *Gesture3DFramework*. A Seção 4 ilustra a utilização da solução implementada através de um estudo de caso. A seção 5 apresenta uma discussão sobre as lições aprendidas após o desenvolvimento e testes sobre o *Gesture3DFramework*. Finalmente, a seção 6 apresenta algumas conclusões e perspectivas futuras.

II. METÁFORAS PARA A INTERAÇÃO COM AMBIENTES 3D

As metáforas de interação gestual permitem que os usuários realizem uma interação mais natural com o ambiente de realidade virtual através de gestos mais próximos daqueles utilizados no dia a dia. O estudo e a identificação das metáforas de interação possíveis associadas à *Seleção*, *Navegação* e *Manipulação* (SNM) em ambientes computacionais 3D são fundamentais para determinar o nível de acessibilidade do usuário. Nos ambientes 3D os usuários precisam estar aptos para indicar com qual objeto desejam interagir e como eles desejam interagir para que a interação possa ocorrer [6].

As interações são subdivididas em três categorias: (i) A **Seleção** consiste no processo que permite o usuário poder escolher qual ou quais objetos serão selecionados e como serão selecionados; (ii) **Navegação** é o processo que permite o usuário construir um caminho, um mapa cognitivo do espaço, de forma a poder se locomover, e; (iii) A **Manipulação**, descreve a mudança de estado do objeto selecionado, tais como através da rotação, translação, redimensionamento, zoom e até a alteração das suas propriedades eg.: texturas [7].

Além destes paradigmas para interação, existem ainda duas formas ortogonais de classificar a interação em ambientes 3D: **Egocêntrica** onde o usuário intervém diretamente dentro do ambiente virtual. Normalmente tal paradigma é usado quando há necessidade de precisão na interação (geralmente associadas à utilização das metáforas **Virtual Hand** e **Virtual Pointer**), e; **Exocêntricas**, usadas para interação a partir de fora do ambiente virtual, onde as proporções entre o usuário e os objetos não são preservadas, assumindo assim que o usuário irá interagir a partir de fora do sistema de referência, denominada *God's Eye* [8].

A seguir são introduzidas as principais metáforas levantadas na literatura de acordo com cada tipo de interação.

A. Seleção

T Através das metáforas de seleção o usuário pode indicar com qual objeto no ambiente 3D ele deseja interagir. Dentre as metáforas de seleção identificadas na literatura, podemos citar *Virtual Hand* [33], *Virtual Pointer* [35], *Ray-Casting* [11], *Go-Go* [9], *Rotate-Pan-zoom* [24], *Point of Interest*, *Depth Ray* [6], *Lock Ray* [6], *Flower Ray* [6], *Smart Ray* [6], *Flash light* [12], *Shadow Cone* [18], *Aperture* [19],

ViewFinder [20], *IntenSelect* [21], *Bubble Cursor* [15], *Silk Cursor* [17], *VirtualPad* [16], *Direct Image Plane* [13].

Em particular, neste trabalho é dada ênfase inicialmente às metáforas *Virtual Hand* e *Ray Casting*. A metáfora *Virtual Hand* é considerada como uma das metáforas mais comuns, dado que é uma forma de interação intrínseca do ser humano. Consiste na utilização de uma mão virtual que poderá ser utilizada para interagir com os objetos [33].

Várias técnicas de seleção foram criadas ao longo dos anos e, uma das formas mais utilizadas para representar determinadas técnicas de seleção é a partir da aplicação da metáfora de raios (*Ray Casting*) [10][11]. Trata-se de uma metáfora utilizada para seleção de objetos à longa distância sem o usuário se deslocar no cenário ou “*estender o seu braço virtual*”. Naturalmente existem outras variações desta metáfora, contudo, para o presente trabalho será usado apenas a *RayCasting*, visto que ela está disponível na maioria das aplicações e API's. No entanto, essa metáfora pode não ser apropriada para a seleção precisa de mais de um objeto simultaneamente, dado que pode haver oclusão entre objetos sobrepostos, por isso, pode ser dificultada a seleção de vários objetos e de objetos distantes. Naturalmente a partir desta metáfora houveram outras soluções tentando superar suas limitações.

Conforme discutido em [6], foram observados alguns aspectos da utilização da seleção por raios. Segundo os autores, este tipo de representação é significativamente mais rápido que a seleção com cursores, por exemplo. Nesse trabalho foram abordadas outras variações desta metáfora, tais como *Depth Ray*, *Lock Ray*, *Flower Ray* e *Smart Ray* as quais foram discutidas em [6].

De uma forma geral, todas as seleções oriundas de seleção por raios parecem ser mais rápidas, contudo, o grande desafio está relacionado às múltiplas seleções, que tendem se tornar comuns à medida que a distância aumenta. Tal dificuldade talvez pudesse ser minimizada se fosse possível acionar um binóculo a fim de que o usuário pudesse ver o objeto à distância e, sob tal perspectiva o raio se comportaria como se ele estivesse na nova distância.

B. Navegação

No que concerne a **Navegação**, ela é definida como a capacidade de conduzir o usuário de um lugar para o outro numa determinada direção num cenário. Pode ser classificada em (i) *Explorativa*, onde não há um propósito ou caminho definido, (ii). *Busca*, quando o usuário já possui um objetivo definido ou o usuário possui o controle dos seus movimentos; (iii). *Passivas*, onde o participante é guiado pelo ambiente, por outro participante ou pelo sistema [1].

Dentre as metáforas de navegação identificadas na literatura, podemos citar *Virtual Hand*, *Ray-Casting*, *Go-Go*, *Point of Interest* [26], *Walking/Driving/Flying* [1], *Navigation by Hyperlink* [1], *World in Miniature* [23], *Navigation By Query* [25] e *Path Drawing* [27].

Em particular, a partir das metáforas *Virtual Hand* e *Ray Casting* é possível implementar outras metáforas, como por exemplo, *Point of Interest (POI)* e *Go-to/Fly-To*. Essas metáforas requerem que o usuário de alguma forma escolha um objeto desejado na cena (seu ponto de interesse) usando um dispositivo de apontamento e, após a seleção, a câmera se movimenta até o objeto escolhido [26].

Outra metáfora que merece destaque neste trabalho é *Walking/Driving/Flying*, resultado de uma composição de metáforas que permitam o usuário poder andar, dirigir ou voar no cenário 3D desejado. Essas metáforas exigem que a câmera acompanhe o usuário de forma que a ação desejada possa simular tal efeito [1].

C. Manipulação

A manipulação ocorre após a tarefa de seleção ter sido efetuada, pois, sem o objeto ser selecionado, não há como o mesmo ser manipulado. Naturalmente, as metáforas de seleção por si só aceitam as tarefas de manipulação.

As metáforas de manipulação devem permitir o usuário a mover um objeto, rotacioná-lo e a redimensioná-lo. Dentre as metáforas de manipulação identificadas na literatura estão *Virtual Hand*, *Ray-Casting*, *Go-Go*, *Rotate-Pan-zoom*, *Flash light* e *Silk Cursor*.

Dessa forma, a tarefa de manipulação tende a estar ligada a tarefa de seleção e, muito naturalmente, a metáfora usada na seleção, por vezes é a mesma a ser usada na manipulação. O mesmo ocorre com a navegação.

Por exemplo, a metáfora *Virtual Hand* pode ser usada para selecionar um objeto, e logo depois o usuário deseja efetuar a manipulação do mesmo através da mesma metáfora. É importante a compreensão que a utilização das metáforas para o usuário deve ser o mais natural possível, deixando-o a vontade para continuar utilizando a mesma metáfora em uma interação diferente ou combiná-la com outra metáfora para produzir outro tipo de interação. Neste caso, o sistema deverá suportar os tipos de metáforas e interações desejadas pelo usuário.

Embora a variedade das metáforas existentes possa ser expressiva para o desenvolvedor, muitas dessas aparentemente estão relacionadas entre si e, compartilham muitas propriedades. Com o objetivo de definir um conjunto de metáforas básicas para futuramente poderem ser disponibilizadas em uma biblioteca de APIs genéricas implementadas através de técnicas gestuais, este estudo buscou as diferentes metáforas de SNM apresentadas na literatura, e foi realizada a análise de suas características e similaridades [29]. Com essa proposta, é possível utilizar a biblioteca de metáforas implementada e combiná-las de forma a se obter metáforas de SNM mais elaboradas.

As metáforas estudadas neste trabalho foram classificadas como básicas, intermediárias e avançadas de acordo com o nível de complexidade de implementação e disponibilidade de APIs nativas nos motores de renderização. Dentre as metáforas básicas adotadas neste trabalho estão *Ray Casting* e *Virtual Hand*. Como marcadas na tabela, elas são aplicadas para a seleção e manipulação de objetos virtuais, já com suporte nativo dos motores de renderização. A navegação utilizando essas metáforas pode ser ainda enriquecida com alguma implementação adicional.

Tabela 1: Metáforas básicas para a SNM

Nome	Interação			Complexidade
	Seleção	Manipulação	Navegação	
Virtual Hand	•	•	○	básica
Ray-Casting	•	•	○	básica
Go-Go	•	•	○	básica
Rotate-Pan-zoom	•	•	○	básica
Point of Interest (Go-To/Fly-To)	•		•	básica
Walking/Driving/Flying			•	básica

Na próxima seção são apresentados os detalhes da implementação do arcabouço *Gesture3DFramework*, como solução genérica para intermediar a interação gestual do usuário

III. GESTURE3DFRAMEWORK: IMPLEMENTAÇÃO DO PROTÓTIPO

Esta seção aborda as principais etapas do desenvolvimento do arcabouço *Gesture3DFramework*, proposto como uma solução capaz de permitir a associação entre gestos/metáforas, de forma que tais associações possam ser configuradas pelo desenvolvedor através de um arquivo baseado em XML [28]. Após o arquivo de configuração ser lido e interpretado pelo arcabouço proposto, a respectiva interação deverá ser convertida em um comando para um determinado motor de renderização 3D, e este, por sua vez, executará a mensagem renderizando visualmente a interação indicada assim como o seu resultado.

Para que o arcabouço proposto possa responder às expectativas do usuário, sendo uma ferramenta genérica para a otimização da prototipação de ambientes virtuais interativos através de gestos, é importante compreender alguns de seus principais requisitos para implementação:

- O sistema deverá dar suporte às interações gestuais permitindo a seleção, manipulação e navegação com o ambiente 3D;
- O sistema inicialmente deverá dar suporte às metáforas *RayCasting* e *Virtual Hand*;
- O sistema deverá dar suporte à interação gestual customizável a qualquer tipo de aplicação de Realidade Virtual, permitindo a Seleção, Navegação e Manipulação;
- O sistema deverá oferecer suporte à depuração visual das interações gestuais realizadas;
- O sistema deverá permitir a fácil configuração dos gestos e metáforas suportadas;
- O sistema deverá fornecer *Application Programming Interfaces (API's)* a serem integradas com o código-fonte da aplicação de ambiente virtual para que motores de renderização 3D possam interpretar as diretivas de interação recebidas do *Gesture3dInterface*, e;
- O sistema deve possuir rápido tempo de resposta para a atualização do ambiente virtual com o resultado da interação do usuário.

Nas próximas seções são apresentadas a arquitetura da solução proposta, assim como a descrição da estrutura de dados manipulada pelo sistema,

A. Arquitetura do Projeto

A arquitetura de um sistema ilustra os seus principais módulos e componentes implementados, assim como as relações entre eles. A Figura 1 ilustra as camadas que compõem a arquitetura proposta:

- **Camada de apresentação.** É representada pelos motores de renderização 3D através dos quais são apresentados os ambientes virtuais. Possui uma interface com o *Gesture3DFramework* através de API's disponibilizadas;

- **Camada lógica.** Contém as funcionalidades do arcabouço propriamente dito, esta camada é composta pelos seguintes sub-módulos:

- **Middleware Gestual (Engine Middleware),** realiza a conexão entre o sub-módulo de *Contextualizador Gestual (Gesture-Oriented Context Processing)* e o motor de renderização 3D. Este sub-módulo recebe as diretivas do sub-módulo *Contextualizador Gestual* para a execução da interação de acordo com a metáfora utilizada, e gera uma mensagem de comunicação passando as diretivas à API de interação do motor de renderização;
- **Contextualizador Gestual (Gesture-Oriented Context Processing).** Associa o gesto reconhecido pela API de captura de gestos da câmera (*API Gestos*), com as metáforas existentes no arquivo de configuração do *Gesture3DFramework*, reconhecendo a metáfora de interação desejada pelo usuário;
- **API Gestos (Gesture Capture API),** realiza a captura do gesto através do dispositivo, no caso em particular, câmeras, mapeia o gesto realizado e o encaminha para o módulo *Gesture-Oriented Context Processing*.

- **Camada de Dados.** Esta camada descreve a persistência dos dados mantidos pelo *Gesture3DFramework*, através da descrição declarativa da informação de configuração (*Gesture Configuration*) para o mapeamento gesto/metáfora, e para a geração da mensagem para indicação da metáfora de interação (*Gesture Interaction*) desejada pelo usuário, a ser lida pela a API do motor de renderização.

Na Figura 1 é ilustrada a arquitetura do *middleware Gesture3DFramework*.

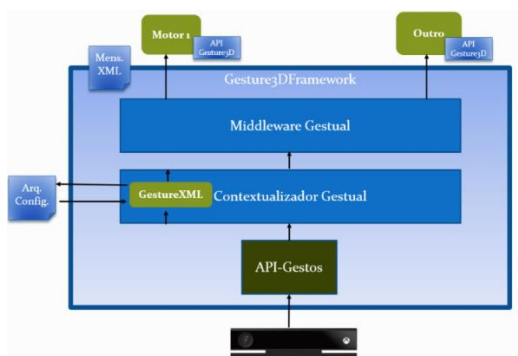


Figura 1. Arquitetura do *Gesture3DFramework*

Para que o processo de reconhecimento gesto-metáfora seja eficaz e a interação do usuário possa ser mapeada corretamente, é importante a descrição dos possíveis gestos suportados pelo arcabouço e do tipo de interação desejado através das mudanças de gestos (ou estados do usuário). Para essa finalidade, para cada nova metáfora suportada pelo *Gesture3DFramework* é necessário descrever as mudanças de estados entre os gestos e as respectivas interações associadas no arquivo de configuração *GestureConfiguration*.

B. Descrição do arquivo de configuração

O *Gesture3DFramework* visa facilitar o processo de desenvolvimento e integração de recursos de interação gestual com motores de renderização 3D. Portanto, é importante a utilização de uma representação de dados que permitam que o usuário (desenvolvedor) possa facilmente descrever os possíveis gestos suportados pelo arcabouço e a correspondência para as suas respectivas metáforas. Para isso, o arquivo de configuração de gestos (*GestureConfiguration*) é utilizado para a configuração inicial do *Gesture3DFramework*.

Na Figura 2 é ilustrado um trecho do arquivo de configuração de gestos (*GestureConfiguration*) para descrever os parâmetros de referência para os mecanismos de interação por seleção, manipulação (redimensionar, rotacionar e mover um objeto) e navegação.

```
<Gestures>
<!-- Seleção, Manipulação (Rotação e Deslocamento) - -->
<MainHand>
<hand>Left</hand>
<Speed_Factor_X> 0.1 </Speed_Factor_X >
<Speed_Factor_Y> 0.1 </Speed_Factor_Y >
<Speed_Factor_Z> 0.1 </Speed_Factor_Z >
<Moving_Factor> 0.3 </Moving_Factor>

<CurrentState>Closed</CurrentState>
<Metaphor>Virtual Hand</Metaphor>
<Interaction>Selection</Interaction>
<Action>Select</Action>

<SecondaryHand>
<hand>Right</hand>
<State>Open</State>
<Metaphor>Virtual Hand</Metaphor>
<Interaction>Manipulation</Interaction>
<Action>Rotate</Action>
<Speed_Factor_X> 0.1 </Speed_Factor_X>
<Speed_Factor_Y> 0.1 </Speed_Factor_Y>
<Rotation_Factor> 0.3 </Rotation_Factor>
</SecondaryHand>

<SecondaryHand>
<hand>Right</hand>
<State>Closed</State>
<Metaphor>Virtual Hand</Metaphor>
<Interaction>Manipulation</Interaction>
<Action>Resize</Action>
<Speed_Factor_X> 0.1 </Speed_Factor_X >
<Speed_Factor_Y> 0.1 </Speed_Factor_Y >
<Speed_Factor_Z> 0.1 </Speed_Factor_Z >
<Scaling_Factor> 0.5 </Scaling_Factor>
</SecondaryHand>

<SecondaryHand>
<hand>Right</hand>
<State>Lasso</State>
<Metaphor>Ray Casting</Metaphor>
<Interaction>Manipulation</Interaction>
<Action>Moving</Action>
<Moving_Factor> 0.3 </Moving_Factor>
</SecondaryHand>
```

Figura 2. Configuração de gestos - *GestureConfiguration*

Na interação gestual, o usuário poderá utilizar ambas as mãos para descrever o tipo de interação desejada. Por essa razão, é importante que usuário possa definir qual a mão principal para sua interação, caso ele seja destro ou canhoto. Assim, é possível ilustrar na Figura 2 que o elemento *<MainHand>* determina todas as características de interação para a mão principal sendo utilizada pelo usuário, e este elemento também engloba todos os possíveis estados da mão secundária *<SecondaryHand>*. Mais precisamente, o elemento *<hand>* possibilita definir qual mão está sendo utilizada, como principal ou secundária, seja ela direita ou esquerda.

De forma a caracterizar o movimento que será renderizado pelo motor de renderização, o fator de velocidade de movimento *<Speed_Factor_>* entre os eixos

X, Y e Z, e o fator de movimento *<Move_Factor>* também podem ser configurados através desse arquivo. Permitindo assim, um processo de refinamento maior, caso o usuário deseje.

As demais características da interação são definidas através dos próximos elementos: estado atual *<CurrentState>*, metáfora *<Metaphor>*, interação *<Interaction>* e ação *<Action>*. É neste ponto a partir do processo de identificação, reconhecimento e combinação entre gestos e metáforas que verdadeiramente começa o seu processo de tratamento da interação.

O elemento *<CurrentState>* determina o último gesto capturado para a mão, compatível com um estado existente e fornecido pelo dispositivo de captura de movimento. Os estados capturados podem ser *open* (aberta), *closed* (Fechada) e *lasso* (indicador).

Este estado estará diretamente relacionado com a metáfora a ser determinada e indicada através do elemento *<Metaphor>*. Os possíveis valores para o elemento *<Metaphor>* são *Virtual Hand* e *Ray Casting*, indicando respectivamente, a utilização da metáfora de uma mão virtual ou de um apontador virtual. Até o momento, neste trabalho propomos a utilização de apenas essas duas metáforas, podendo ser estendidas posteriormente para o suporte a outras metáforas de interação.

Dependendo da metáfora escolhida pelo usuário, os seguintes valores para o elemento Interação *<Interaction>* podem ser definidos, seleção (*Selection*), manipulação (*Manipulation*) e navegação (*Navigation*).

Para toda interação escolhida, deverá ser indicada uma ação *<Action>*, que poderá ter os seguintes valores selecionar (*Select*), rotacionar (*Rotate*), redimensionar (*Resize*), move (*Moving*) e navegar (*Navigate*).

Por exemplo, no caso deste trecho do arquivo ilustrado na Figura 2, o desenvolvedor do ambiente 3D configurou este arquivo descrevendo qual o tipo de interação poderá ocorrer se a mão principal estiver fechada, relacionada à metáfora *Virtual Hand*, e a interação *<Interaction>* for do tipo de seleção (*selection*), associada a uma ação *<Action>*, com o valor de *select*. Observe que após a definição do comando de seleção para a mão principal, existem 3 elementos *<SecondaryHand>* indicando 3 (três) formas de interação para a mão secundária, explicadas a seguir:

- 1ª possibilidade: O elemento *<SecondaryHand>* define a mão secundária como a mão direita e, como o seu estado definido no elemento *<state>* é o de mão aberta (*open*), a metáfora *Virtual Hand* é associada ao elemento *<Metaphor>*. Neste caso, a interação *<Interaction>* do tipo manipulação (*Manipulation*) do objeto selecionado no ambiente 3D foi estabelecida, e o usuário poderá realizar a rotação do objeto, já que a ação *<Action>* possui o valor *Rotate*. Neste caso, também é definido o elemento *<Rotate_Factor>*, indicado por uma valor percentual em pixels, para caracterizar o fator de rotação durante a renderização do objeto 3D;
- 2ª possibilidade: O elemento *<SecondaryHand>* define a mão secundária como a mão direita e, como o seu estado definido no elemento *<state>* é o de mão fechada (*closed*), a metáfora *Virtual Hand* é associada ao elemento *<Metaphor>*. Neste caso, a

interação *<Interaction>* do tipo manipulação (*Manipulation*) do objeto selecionado no ambiente 3D também foi estabelecida, e o usuário poderá realizar o redimensionamento do objeto, já que a ação *<Action>* possui o valor *Resize*. Da mesma forma, neste caso, é definido o elemento *<Scaling_Factor>* para caracterizar o fator de redimensionamento durante a renderização do objeto 3D, e;

3ª possibilidade: O elemento *<SecondaryHand>* define a mão secundária como a mão direita e, como o seu estado definido no elemento *<state>* é o de apontar (*lasso*), a metáfora *Ray Casting* é associada ao elemento *<Metaphor>*. Neste caso, a interação *<Interaction>* do tipo manipulação (*Manipulation*) do objeto selecionado no ambiente 3D também foi estabelecida, e o usuário poderá realizar a movimentação do objeto, já que a ação *<Action>* possui o valor *Moving*. Para esta interação também é definido o elemento *<Moving_Factor>* para caracterizar o fator de deslocamento durante a renderização do objeto 3D.

C. Descrição dos dados para a Integração com o Motor de Renderização 3D

A Integração do *Gesture3DFramework* com a motor de renderização 3D é viabilizada através do envio de mensagens entre esses módulos. Uma vez que todo o processo de configuração e associação gesto/metáfora já é realizado pelo arcabouço, a mensagem pode ser simplificada, otimizando assim o tempo de processamento do motor de renderização. Na Figura 3 é descrito um trecho da mensagem trocada com o motor de renderização, chamada de *<Gesture3DInteraction>*. Essa mensagem indicará ao motor de renderização, de forma sintetizada, qual a metáfora e a interação que o usuário deseja realizar dentro do ambiente 3D.

```
<Gesture3DInteraction>
<Message>
  <Metaphor> RayCasting </Metaphor>
  <Interaction> Selection </Interaction>
  <x value="30.0" speed_factor="0.1"/>
  <y value="40.0" speed_factor="0.1"/>
  <z value="70.0" speed_factor="0.1"/>
</Message>
</Gesture3DInteraction >
```

Figura 3. Trecho da mensagem *Gesture3DInteraction*

A mensagem trocada é declarativa (baseada em XML) e é composta por um conjunto de elementos: (i) A metáfora definida é descrita no elemento *<Metaphor>*; (ii) O elemento *<Interaction>* indica o tipo de interação a ser realizada; (iii) Os valores referentes aos fatores da interação estão descritos nos elementos *<X>*, *<Y>* e *<Z>*. Note que os valores referentes aos fatores das interações são oriundos do processo de leitura do arquivo de configuração *GestureConfiguration*, realizado previamente pelo *Gesture3DFramework*.

A mensagem é em seguida processada e enviada para o motor de renderização pelo *Gesture3DFramework*. O processo de envio da mensagem é relativamente simples. Após o processo de montagem das mensagens, elas são encaminhadas para o motor de renderização através de uma fila utilizando um canal de comunicação de rede. Nesta fase,

este processo teve como objetivo garantir a existência do canal de comunicação e envio de mensagens, local e remoto.

O processo de leitura é efetuado através das API no motor de renderização, fornecidas pelo *Gesture3DFramework*. Estas API são desenvolvidas para poderem efetuar a leitura da mensagem, permitindo que o motor de renderização 3D trate a mensagem simplificada, para que então o desenvolvedor possa associar a mensagem aos comandos a serem realizados no motor de renderização 3D.”,

IV. ESTUDOS DE CASO - GESTURE3DFRAMEWORK

Após a concepção e desenvolvimento do arcabouço *Gesture3DFramework* é importante demonstrar a proposta e o funcionamento da interação gestual associada aos motores de renderização 3D, em particular, com o motor *Irrlicht*. Para isso, foram propostos cenários de estudo de caso durante o levantamento das metáforas de interação mais utilizadas nos ambientes virtuais 3D.

Na Figura 4 é ilustrado o ambiente utilizado para testes. Trata-se de um ambiente 3D simples ilustrando um aposento de um castelo medieval, modelado e disponível em [39]. Este cenário é adequado para o teste uma vez que possui proporções suficientes para a adição e manipulação de um personagem e objetos 3D, seleção e manipulação desses objetos, como também, a navegação do usuário.



Figura 4. Ambiente virtual de teste renderizado no Irrlicht

As metáforas implementadas inicialmente para a validação do arcabouço *Gesture3DFramework* são *Raycasting* e *VirtualHand*. Essas metáforas foram selecionadas uma vez elas são suportadas de forma nativa pelo motor de renderização gráfico *IrrLicht*.

Por motivo de espaço, neste artigo ilustramos o *Gesture3DFramework* para a mediação de diversas técnicas de interação (seleção, redimensionamento, deslocamento, rotação e navegação) utilizando a metáfora *VirtualHand*.

A. Interação utilizando a metáfora *VirtualHand*

Neste estudo de caso o arcabouço *Gesture3DFramework* é utilizado para ilustrar como um objeto é selecionado através da metáfora *VirtualHand*. Para isso, é necessário primeiro indicar o objeto no ambiente virtual e, posteriormente efetuar a sua seleção propriamente dita.

Na Figura 5 as imagens numeradas através de (1) e (2) também ilustram um personagem localizado no ambiente 3D, que será utilizado para os testes. Neste ambiente 3D também é possível observar a existência de uma mão que indica a metáfora *VirtualHand* em execução, ilustrado em (3). Quando esta mão toca um personagem, o personagem fica iluminado conforme ilustrado em (4).

Da mesma forma, esta simples interação ocorre mediante o tratamento do gesto realizado pelo usuário, feito em várias etapas pelo *Gesture3DFramework*. Estas etapas estão numeradas a seguir, ilustradas na Figura 5:

- Ao ser inicializado, o *Gesture3DFramework* lê o arquivo de configuração (9) para carregar todas as variáveis e mapas de gestos de metáforas suportadas pelo sistema (10);

- Quando o usuário faz o gesto de mão aberta com a mão principal (13), em (12) o dispositivo efetua a leitura do gesto fornecido pelo usuário. Em (14) é possível observar depuração em tempo real do funcionamento do *Gesture3DFramework*, através do aplicativo de depuração desenvolvido, onde a ação de tocar objetos é indicada em (15);

- Em seguida, a API de gestos envia a diretiva básica gestual (11) para o módulo de *Contextualizador Gestual* (8), que é responsável por realizar o mapeamento do gesto do usuário para a correta metáfora e ação de interação a serem utilizadas;

- A partir das diretivas da metáfora e ação selecionadas, o módulo de *Middleware Gestual* (7) irá montar a mensagem de interação (6) simplificada (*Gesture3DInteraction*) a ser enviada para o a API (5) do motor de renderização, e;

- Por fim, o motor de renderização apresenta a atuação da interação realizada (4), no caso, a indicação do objeto.

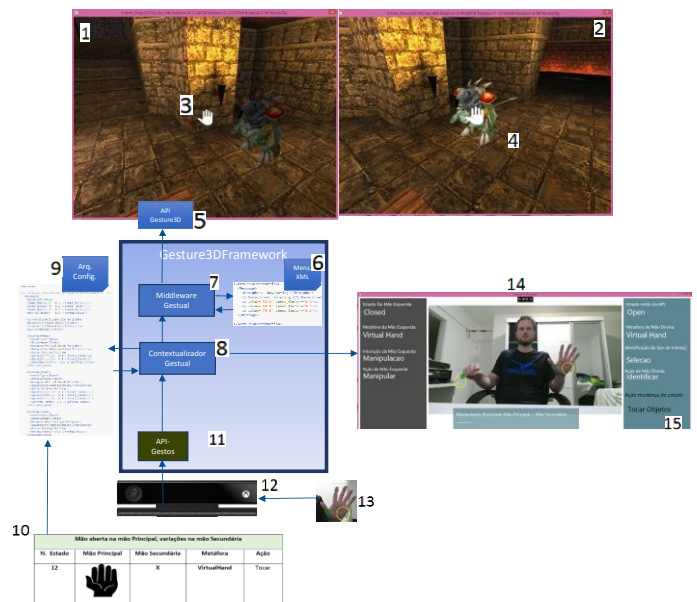


Figura 5. Interação de indicação utilizando *VirtualHand*

Uma vez a indicação ter sido realizada, torna-se possível efetuar a seleção propriamente dita. A associação entre gesto/metáfora foi definida após a indicação do objeto pela mão principal ser fechada, conforme ilustrado na Figura 6

(10). Após a realização desta ação o objeto pode ser manipulado.

A seguir é ilustrado o processo de seleção por *VirtualHand*, onde um objeto estará pronto para ser manipulado. As etapas do processo também estão numeradas na Figura 6, como segue:

- Como ilustrado anteriormente, também nas imagens numeradas em (1) e (2) é possível observar novamente o personagem localizado no ambiente 3D. Também é possível notar o ícone da mão indicando a metáfora *VirtualHand* em execução, ilustrado em (3). No momento em que o usuário fechou a mão principal, o ícone da mão também passou a ser representado como mão fechada, utilizando a metáfora *VirtualHand* ilustrado em (3), quando este ícone toca um personagem, o personagem, fica iluminado conforme ilustrado em (4), indicando sua seleção;

- Ao ser inicializado, o *Gesture3DFramework* lê o arquivo de configuração (9) para carregar todas as variáveis e mapas de gestos de metáforas suportadas pelo sistema (10);

- Quando o usuário faz o gesto de fechar a mão com a mão principal (13), em (12) o dispositivo efetua a leitura do gesto fornecido pelo usuário. Em (14) é possível observar depuração em tempo real do funcionamento do *Gesture3DFramework*, através do aplicativo de depuração desenvolvido, onde a ação de seleção por *VirtualHand* é indicada em (15);

- Em seguida, a API de gestos envia a diretiva básica gestual (11) para o módulo de *Contextualizador Gestual* (8), que é responsável por realizar o mapeamento do gesto do usuário para a correta metáfora e ação de interação a serem utilizadas;

- A partir das diretivas da metáfora e ação selecionadas, o módulo de *Middleware Gestual* (7) irá montar a mensagem de interação (6) simplificada (*Gesture3DInteraction*) a ser enviada para a API (5) do motor de renderização, e;

- Por fim, o motor de renderização apresenta a atuação da interação realizada (4), no caso, a seleção do objeto.

ilustrado anteriormente para a indicação e seleção do objeto virtual. A combinação de gestos da mão principal e secundária que irá determinar a técnica de interação desejada pelo usuário. De forma a não tornar a apresentação desse estudo de caso para as técnicas (SNM) baseadas em *VirtualHand* repetitiva, a seguir discutimos a combinação de gestos para cada tipo de interação:

- Após o usuário ter determinado a metáfora de seleção por *VirtualHand* (**mão fechada**) com a mão principal, o dispositivo deverá efetuar a leitura do gesto fornecido pela mão secundária do usuário. Se o usuário fizer o gesto de **mão fechada** com a mão secundária, ele irá determinar a ação de manipulação com redimensionamento. Uma vez que o usuário move a mão secundária para a esquerda ou direita, o objeto sofrerá o processo de **redimensionamento**;
- Após o usuário ter determinado a metáfora de seleção por *VirtualHand* (**mão fechada**) com a mão principal, o dispositivo deverá efetuar a leitura do gesto fornecido pela mão secundária do usuário. Se o usuário fizer o gesto de **mão aberta** com a mão secundária, ele irá determinar a ação de manipulação por rotação. Uma vez que o usuário move a mão secundária para a esquerda ou direita, o objeto sofrerá o processo de **rotação**;
- Após o usuário ter determinado a metáfora de seleção por *VirtualHand* (**mão fechada**) com a mão principal, o dispositivo deverá efetuar a leitura do gesto fornecido pela mão secundária do usuário. Se o usuário fizer o gesto de **indicar** com a mão secundária, ele irá determinar a ação de manipulação por **deslocamento**. Uma vez que o usuário move a mão secundária para a esquerda ou direita, o objeto sofrerá o processo de deslocamento;
- Após o usuário ter determinado a metáfora de seleção por *VirtualHand* (**mão fechada**) com a mão principal, o dispositivo deverá efetuar a leitura dos gesto fornecidos pelas mãos primária e secundária do usuário. Se o usuário fizer o gesto de **indicar com ambas as mãos**, ele irá determinar a ação de **navegação**. Neste momento o usuário irá navegar no ambiente 3D.

V. DISCUSSÕES E LIÇÕES APRENDIDAS

Os cenários elaborados como estudo de caso visaram ilustrar as funcionalidades do *Gesture3DFramework* no suporte à metáfora *VirtualHand*, e validar os requisitos propostos para o seu desenvolvimento, assim como demonstrar sua capacidade de extensão para demais metáforas. Os exemplos demonstraram que o sistema permite a execução e suporte a interações básicas como seleção, navegação e manipulação.

Como o objetivo principal do trabalho foi a aplicação do *Middleware Gesture3DFramework* para a mediação da interação gestual em cenários e motores de renderização 3D, não foi dada muita ênfase neste trabalho para os aspectos de modelagem do cenário virtual. Portanto, para finalidade de teste e de validação, foi utilizado um cenário virtual já existente em [38].

Em relação ao cumprimento dos requisitos propostos, podemos dar ênfase: ao suporte das técnicas de interação gestual orientadas à seleção, manipulação (deslocamento, redimensionamento e rotação) e navegação; implementação

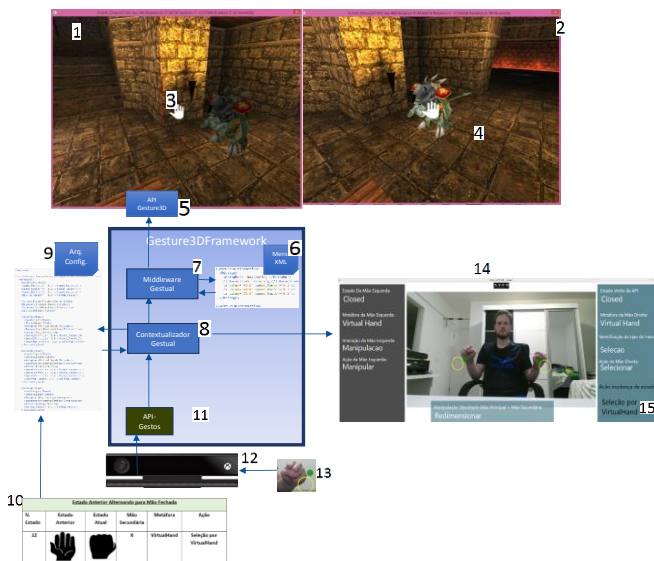


Figura 6 Interação de seleção utilizando VirtualHand.

As demais técnicas de interação baseadas na metáfora *VirtualHand* (redimensionamento, deslocamento, rotação e navegação) seguem o mesmo processo de tratamento que foi

das metáforas *RayCasting* e *Virtual Hand*; a utilização genérica do middleware desenvolvido com qualquer motor de renderização 3D; rápido tempo de resposta entre a interação gestual e a modificação do ambiente 3D, entre outras.

As extensões também podem ocorrer com as interações de seleção e manipulação através de outras metáforas, como por exemplo, a *Go-Go*, que é uma extensão da metáfora *VirtualHand*, onde a mão é substituída por um braço virtual, indicando a capacidade de alcançar objetos à uma determinada distância. As funcionalidades e requisitos que foram identificadas e não puderam ser implementados, são considerados como perspectivas de continuação do trabalho;

IV CONCLUSÕES

O *Gesture3DFramework* conseguiu ser concretizado através do mapeamento prévio dos gestos e associações com as metáforas definidas pelo usuário. Esse arcabouço foi concebido como um *middleware* entre o dispositivo de captura que informa os valores básicos referentes aos gestos, e o motor de renderização 3D, que executará a metáfora propriamente dita, abstraindo assim, todo o processamento no momento em que houver a combinação de um determinado gesto com uma determinada metáfora.

Naturalmente, os estudos de casos estão voltados para os gestos e metáforas mais básicos encontrados nos motores de renderização 3D. Mas a característica de extensibilidade do *Gesture3DFramework*, está fortemente ligada ao arquivo de configuração, tornando possível expansão a combinação entre gestos e metáforas dando uma maior flexibilidade ao desenvolvedor e implementação de novas metáforas.

Dessa forma, o *Gesture3DFramework* pode ser considerado como uma solução para mediar a interação gestual do usuário com motores de renderização 3D. Voltados para Interações gestuais customizável para qualquer aplicação, suporte nativo as metáforas mais básicas, como possibilidade de extensão para as suas variações, como a adaptação para novas metáforas. Como perspectivas futuras, pretende-se estender a sua utilização para o suporte de outras metáforas de interação e aplicar esse arcabouço a outros motores de renderização.

REFERENCIAS

[1] J. Jankowski, M. Hachet, *A Survey of Interaction Techniques for Interactive 3D Environments*, in *Eurographics 2013 - STAR* (Girona, Spain, 2013), <https://hal.inria.fr/hal-00789413>. Clerk Maxwell, *A Treatise on Electricity and Magnetism*, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] I IDGNews, *Microsoft usa Kinect para interpretar a linguagem de sinais de surdos* (2008), <https://tinyurl.com/lawst5z>

[3] T.M. Takala, *RUIS: A Toolkit for Developing Virtual Reality Applications with Spatial Interaction*, in *Proceedings of the 2Nd ACM Symposium on Spatial User Interaction* (ACM, New York, NY, USA, 2014), SUI '14, pp. 94–103, ISBN 978-1-4503-2820-3, <http://doi.acm.org/10.1145/2659766.2659774>

[4] R.J. Jacob, A. Girouard, L.M. Hirshfield, M.S. Horn, O. Shaer, E.T. Solovey, J. Zigelbaum, *Reality-based Interaction: A Framework for post-WIMP Interfaces*, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, NY, USA, 2008), CHI '08, pp. 201–210, ISBN 978-1-60558-011-1, <http://doi.acm.org/10.1145/1357054.1357089>

[5] Poupyrev, Ivan, et al. "Egocentric object manipulation in virtual environments: empirical evaluation of interaction

techniques". *Computer graphics forum*. Vol. 17, No. 3. Oxford, UK and Boston, USA: Blackwell Publishers Ltd, 1998.

[6] T. Grossman, R. Balakrishnan, *The Design and Evaluation of Selection Techniques for 3D Volumetric Displays*, in *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology* (ACM, New York, NY, USA, 2006), UIST '06, pp. 3–12, ISBN 1-59593-313-1, <http://doi.acm.org/10.1145/1166253.1166257>

[7] Hrimech, Hamid, Leila Alem, and Frederic Merienne. "How 3D interaction metaphors affect user experience in collaborative virtual environment". *Advances in Human-Computer Interaction* 2011 (2011): 8.

[8] C.S. U ghini, F.R. Blanco, F.M. Pinto, C. Freitas, L.P. Nedel, *EyeScope: a 3D interaction technique for accurate object selection in immersive environments*, in *Proceedings of the SBC Symposium on Virtual Reality* 2006 (2006), pp. 77–88.

[9] I. Poupyrev, M. Billingham, S. Weghorst, T. Ichikawa, *The Go-go Interaction Technique: Non-linear Mapping for Direct Manipulation in VR*, in *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology* (ACM, New York, NY, USA, 1996), UIST '96, pp. 79–80, ISBN 0-89791-798-7, <http://doi.acm.org/10.1145/237091.237102>

[10] Lee, Sangyoon, et al. "Evaluation of pointing techniques for ray casting selection in virtual environments." *Third international conference on virtual reality and its application in industry*. Vol. 4756. International Society for Optics and Photonics, 2003.

[11] Richard A. Bolt. 1980. "Put-that-there": Voice and gesture at the graphics interface. In *Proceedings of the 7th annual conference on Computer graphics and interactive techniques (SIGGRAPH '80)*. ACM, New York, NY, USA, 262-270. DOI=<http://dx.doi.org/10.1145/800250.807503>.

[12] P. Fuchs, G. Moreau, P. Guitton, *Virtual Reality: Concepts and Technologies*, A Balkema book (Taylor & Francis, 2011), ISBN 9780415684712.

A.J.S. Pierce, A.S. Forsberg, M.J. Conway, S. Hong, R.C. Zeleznik, M.R. Mine, *Image Plane Interaction Techniques in 3D Immersive Environments*, in *Proceedings of the 1997 Symposium on Interactive 3D Graphics* (ACM, New York, NY, USA, 1997), I3D '97, pp. 39–ff., ISBN 0-89791-884-3, <http://doi.acm.org/10.1145/253284.253303>

[13] Wozniowski, Mike. - The SPIN Framework, Networked Framework for Distributed Virtual Environments & 3D Visualization.- Disponível em: <http://www.spinframework.org/> > (2012).

[14] T. Grossman, R. Balakrishnan, *The Bubble Cursor: Enhancing Target Acquisition by Dynamic Resizing of the Cursor's Activation Area*, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, NY, USA, 2005), CHI '05, pp. 281–290, ISBN 1-58113-998-5, <http://doi.acm.org/10.1145/1054972.1055012>

[15] C. Andujar, F. Argelaguet, *Virtual pads: Decoupling motor space and visual space for flexible manipulation of 2d windows within ves*, in *3D User Interfaces, 2007. 3DUI'07. IEEE Symposium on* (IEEE, 2007)

[16] S. Zhai, W. Buxton, P. Milgram, *The Bubble Cursor: Investigating Transparency for 3D Target Acquisition*, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM, New York, NY, USA, 1994), CHI '94, pp. 459–464, ISBN 0-89791-650-6, <http://doi.acm.org/10.1145/191666.191822>

[17] A. Steed, C. Parker, *3D selection strategies for head tracked and non-head tracked operation of spatially immersive displays*, in *8th International Immersive Projection Technology Workshop* (2004), pp. 13–14

A. Forsberg, K. Herndon, R. Zeleznik, *Aperture Based Selection for Immersive Virtual Environments*, in *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology* (ACM, New York, NY, USA, 1996), UIST '96, pp. 95–96, ISBN 0-89791-798-7, <http://doi.acm.org/10.1145/237091.237105>

[18] F. Argelaguet, C. Andujar, *Visual Feedback Techniques for Virtual Pointing on Stereoscopic Displays*, in *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology* (ACM, New York, NY, USA, 2009), VRST '09, pp. 163–170, ISBN 978-1-60558-869-8, <http://doi.acm.org/10.1145/1643928.1643966>

- G.d. Haan, M. Koutek, F.H. Post, *IntenSelect: Using Dynamic Object Rating for Assisting 3D Object Selection*, in *Eurographics Symposium on Virtual Environments*, edited by E. Kjems, R. Blach (The Eurographics Association, 2005), ISBN 978-3-905674-06-4, ISSN 1727-530X
- [19] Dicionário Aurélio – Metáfora - Disponível em: <<https://dicionarioaurelio.com/metafora>>. Acesso em 28/04/2018
- [20] Poupyrev, Ivan. *3D Manipulation Techniques Why Manipulation?* 2008, Disponível em: <https://pdfs.semanticscholar.org/1e3f/e0c3f2640c9862f26503bc74d7d3de840336.pdf>.
- [21] R. Stoakley, M.J. Conway, R. Pausch, *Virtual Reality on a WIM: Interactive Worlds in Miniature*, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 1995), CHI '95, pp. 265–272, ISBN 0-201-84705-1, <http://dx.doi.org/10.1145/223904.223938>
- [22] Chen, Michael, S. Joy Mountford, and Abigail Sellen. "A study in interactive 3-D rotation using 2-D control devices." *ACM SIGGRAPH Computer Graphics* 22.4 (1988): 121-129.
- [23] A. van Ballegooij, A. Eliéns, *Navigation by Query in Virtual Worlds*, in *Proceedings of the Sixth International Conference on 3D Web Technology* (ACM, New York, NY, USA, 2001), Web3D '01, pp. 77–83, ISBN 1-58113-339-1, <http://doi.acm.org/10.1145/363361.363380>
- [24] Mackinlay, Jock D., Stuart K. Card, and George G. Robertson. "Rapid controlled movement through a virtual 3D workspace." *ACM SIGGRAPH computer graphics*. Vol. 24. No. 4. ACM, 1990.
- [25] W3.ORG. Extensible Markup Language XML. 2018. Disponível em: <<https://www.w3.org/XML/>>.
- [26] COSTA, D. P.; SAMPAIO, P. N.; MARTINS, V. F. Gesture interaction metaphors within 3d environments: Revisiting the literature. In: IEEE. *Computer Conference (CLEI), 2017 XLIII Latin American*. [S.l.], 2017. p. 1–10.
- [27] Daniel Kharlamov, Krzysztof Pietroszek, and Liudmila Tahai. 2016. TickTockRay Demo: Smartwatch Raycasting for Mobile HMDs. In *Proceedings of the 2016 Symposium on Spatial User Interaction (SUI '16)*. ACM, New York, NY, USA, 169-169. DOI: <https://doi.org/10.1145/2983310.2989206>
- [28] Daniel Kharlamov, Krzysztof Pietroszek, and Liudmila Tahai. 2016. TickTockRay Demo: Smartwatch Raycasting for Mobile HMDs. In *Proceedings of the 2016 Symposium on Spatial User Interaction (SUI '16)*. ACM, New York, NY, USA, 169-169. DOI: <https://doi.org/10.1145/2983310.2989206>
- [29] Imad Afyouni, Ahmad Muaz Qamar, Syed Osama Hussain, Faizan Ur Rehman, Bilal Sadiq, and Abdullah Murad. 2017. Motion-Based Serious Games for Hand Assistive Rehabilitation. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces Companion (IUI '17 Companion)*. ACM, New York, NY, USA, 133-136. DOI:<https://doi.org/10.1145/3030024.3040977>
- [30] Hand, Chris. "A survey of 3D interaction techniques." *Computer graphics forum*. Vol. 16. No. 5. Oxford, UK and Boston, USA: Blackwell Publishers, 1997.
- [31] F. Dermody, A. Sutherland, *Multimodal System for Public Speaking with Real Time Feedback: A Positive Computing Perspective*, in *Proceedings of the 18th ACM International Conference on Multimodal Interaction* (ACM, New York, NY, USA, 2016), ICM 2016, pp. 408–409, ISBN 978-1-4503-4556-9,
- [32] D.A. Bowman, D.B. Johnson, L.F. Hodges, *Testbed Evaluation of Virtual Environment Interaction Techniques*, in *Proceedings of the ACM Symposium on Virtual Reality Software and Technology* (ACM, New York, NY, USA, 1999), VRST '99, pp. 26–33, ISBN 1-58113-141-0, <http://doi.acm.org/10.1145/323663.323667>
- [33] I. Poupyrev, 3D User Interface Design, Lecture Slides. SIGGRAPH 2000 (2000)
- [34] Boeck, J. De; **Raymaekers**, C.; Coninx, K. Are Existing Metaphors in Virtual Environments Suitable for Haptic Interaction. In *Proceedings of the 7th International Conference on Virtual Reality (VRIC2005)*,. p. 261-268. 2005
- [35] IRRLICHTLIME. irrlichtlime Engine. 2018. Disponível em: <<http://irrlichtlime.sourceforge.net/>>.