

# Implementação de Interação em Sistemas Virtuais para Simulação de Exames de Biópsia

Cléber Gimenez Corrêa, Fátima de L. S. Nunes, Adriano Bezerra

Centro Universitário Eurípides de Marília – UNIVEM  
Laboratório de Aplicações de Informática em Saúde - LApIS  
Av. Hygino Muzzi Filho, 529, Marília, SP, Brasil - CEP: 17.525-901

{correacleber, adrianobezerra}@yahoo.com.br, fatima@univem.edu.br

**Abstract.** *This paper presents the implementation of interaction in Virtual Reality (VR) systems to simulate biopsy exams, aiming at building an interaction module with support to conventional and non-conventional devices. This module will be integrated in a framework, offering different resources for interaction, according to desired degree of realism and financial availability of the user.*

**Resumo.** Este artigo apresenta a implementação de interação em sistemas de Realidade Virtual (RV) para simular exames de biópsia, visando a construção de um módulo de interação com suporte a dispositivos convencionais e não-convencionais. Este módulo será integrado a um *framework*, oferecendo diferentes recursos para interação, de acordo com o grau de realismo desejado e a disponibilidade financeira do usuário.

## 1. Introdução

Ultimamente, vem ocorrendo a expansão dos sistemas de RV em diversas áreas, auxiliando os profissionais em suas atividades. Na área de saúde, aplicações empregando esta tecnologia são utilizadas na educação médica, planejamento e simulação de cirurgias e outros procedimentos médicos, reabilitação e avaliação neuropsicológica [1]. No que diz respeito à simulação de procedimentos, como exames de biópsia, por exemplo, a interação entre o usuário e o sistema computacional deve propiciar um treinamento virtual semelhante à execução do procedimento em questão no mundo real.

Por isso, as características deste exame precisam ser levadas em consideração durante o projeto do Ambiente Virtual (AV), a citar: modelagem adequada de objetos tridimensionais, reprodução de ações executadas (movimentos da seringa ou agulha para coleta de material e movimentos da mão do profissional da área médica na manipulação do órgão), e reações possíveis (deformação do órgão, perfuração de diversos tipos de tecidos).

Um outro ponto do projeto a ser destacado é o uso de *frameworks* na construção de Ambientes Virtuais (AVs), que são definidos como o projeto e implementação abstratos para desenvolvimento de aplicações em um determinado domínio de problemas, permitindo a reutilização de componentes [2], o que pode significar aumento de produtividade quando se trata do desenvolvimento de *software*.

Dentro deste contexto e levando-se em conta a importância de dispositivos que realizem a comunicação entre ser humano e máquina, este trabalho apresenta a implementação de um módulo de interação com suporte a dispositivos convencionais (teclado e *mouse*, por exemplo) e não-convencionais (luva de dados e equipamento háptico, por exemplo), a ser acoplado a um *framework* de RV, denominado ViMeT (*Virtual Medical Training*) [3]. O ViMeT fornece uma estrutura de classes que permite a geração de aplicações semi-acabadas para simulação de exames de biópsia. Na Seção 2 são apresentados conceitos sobre o exame de biópsia e interação em AVs, o ViMeT, bem como os dispositivos adotados no projeto, além de detalhes importantes da implementação, como a integração entre linguagens de programação. Na Seção 3 são disponibilizados os resultados alcançados até o momento e as ferramentas utilizadas. A Seção 4 apresenta uma avaliação do projeto, levando em consideração aspectos computacionais, como desempenho da aplicação gerada, e aspectos humanos, como a opinião do usuário.

## 2. Planejamento da Interação

O exame de biópsia, objeto da simulação a ser implementada, consiste em um procedimento para coletar amostras de células de partes do corpo humano que serão analisadas com o objetivo de verificar a existência de anomalias.

No presente trabalho, foi inicialmente considerado como estudo de caso um procedimento de biópsia mamária para o treinamento médico virtual. Neste caso, a agulha com a seringa constituem o instrumento médico utilizado no exame, manuseada pelo médico ou estudante de Medicina, e é inserida no órgão (mama), até atingir uma determinada região (nódulo), de onde se aspira o líquido de um cisto ou um pequeno número de tecido mamário [4]. No entanto, quando há a presença de nódulos que apresentam conteúdo sólido, algumas células podem ser obtidas por meio de movimentos de vaivém da agulha no interior do mesmo, permitindo desta forma, a aspiração de material para análise em laboratório, que determinará se o nódulo em questão é benigno ou maligno, caracterizando a presença ou ausência do câncer de mama.

Para simular as tarefas descritas, a simulação para o treinamento deve prever todas as ações necessárias desde o início do procedimento até a sua finalização. Em um AV, cada uma dessas fases merece especial atenção em relação à interação do usuário com a aplicação.

As técnicas de interação podem ser classificadas, segundo [5], em três categorias: navegação, que se refere ao movimento do usuário pelo AV; seleção e manipulação, que envolve inicialmente o ato de escolher um objeto dentre diversos e, posteriormente, a alteração de suas características, tais como: translação, rotação e tamanho; e controle do sistema, que representa a alteração do estado do sistema ou o modo de interação.

Nesta aplicação, a interação ocorre quando o usuário realiza determinadas operações no AV por meio das chamadas técnicas de interação, as quais consistem em comandos e dados definidos na etapa de implementação do sistema de RV, enviados a este para execução de certas tarefas.

## 2.1 O ViMeT

O ViMeT [3] é um *framework* de RV orientado a objetos, elaborado para facilitar a geração de aplicações voltadas ao treinamento médico, tendo como domínio os sistemas para simular exames de biópsia. Foi implementado com o auxílio das linguagens de programação Java e C, a API (*Application Programming Interface*) Java3D, e a *interface* de programação nativa JNI (*Java Native Interface*), e fornece características e funcionalidades básicas por meio de classes e métodos, entretanto, de relevância para gerar AVs desta natureza.

Estas funcionalidades são: criação de um AV dinâmico que disponibiliza objetos virtuais que representam um órgão humano, um instrumento médico e a mão do usuário, deformação do objeto tridimensional que representa o órgão humano, detecção de colisão entre os objetos presentes na cena e estereoscopia. A Figura 1 apresenta um AV gerado com os objetos virtuais citados.



Figura 1. AV gerado por meio do ViMeT

A aplicação é responsável pela importação dos objetos virtuais, que são definidos pelo usuário durante a geração da mesma, o qual também define algumas características, como: posição, rotação e tamanho dos objetos, bem como os dispositivos a serem utilizados. O usuário pode manipular a seringa e a mão virtual por meio de tais dispositivos, e quando a aplicação identifica uma colisão, é realizada a deformação do objeto que representa o órgão humano, partindo do ponto onde foi identificado o atrito.

## 2.2 Dispositivos

Os dispositivos adotados no projeto incluem três equipamentos classificados como convencionais: monitor de vídeo, teclado e *mouse* comum, e dois não-convencionais: dispositivo háptico *PHANTOM OMNI* (Figura 2), desenvolvido pela empresa *SensAble Technologies*, e luva de dados *5DT Data Glove 5 Ultra* (Figura 3), fabricada por sua vez, pela empresa *5DT (Fifth Dimension Technologies)*.

**Figura 2. PHANTOM OMNI****Figura 3: Data Glove 5 Ultra**

O *mouse* tem como função capturar as coordenadas  $x$  e  $z$  dos movimentos executados no plano pelo participante da simulação, e enviar tais informações ao computador, onde a aplicação movimenta a seringa virtual nos eixos  $x$  e  $y$  na tela do monitor. Assim, este dispositivo permite que o usuário manipule a seringa.

O teclado está associado à mão virtual, no qual teclas específicas podem ser escolhidas para representar cada componente (dedo virtual) deste objeto. Desta forma, o pressionamento de tais teclas indica que a mão está sendo fechada para segurar o órgão humano virtual, objeto do exame de punção; e a liberação das teclas indica que a mão está sendo aberta, soltando o órgão virtual. O teclado também deve permitir a navegação pelo AV por meio das setas, fazendo com que o usuário movimente seu ponto de vista pelo ambiente.

A luva de dados por sua vez, é um equipamento constituído de cinco sensores, os quais permitem captar informações sobre a flexão dos dedos do usuário [6]. Tais informações são enviadas ao computador, e indicam o ato de segurar e soltar o órgão humano, como o teclado, isto é, o ato de abrir e fechar a mão virtual. Os dispositivos não-convencionais podem propiciar um maior grau de realismo, pois no caso da luva de dados, por exemplo, segundo [7], o reconhecimento de gesto significa uma forma de interação altamente intuitiva e eficiente quando se trata de comunicação efetuada entre usuários e AVs.

Por último, o dispositivo háptico, que pode imitar um instrumento médico, como a seringa necessária no exame de biópsia, capta informações, como movimentos de translação e rotação nos eixos  $x$ ,  $y$  e  $z$ ; por isso, ele é considerado um equipamento com seis graus de liberdade. Além disso, este tipo de dispositivo oferece um mecanismo de *feedback* de força nos eixos  $x$ ,  $y$  e  $z$ , permitindo desta forma, uma interação eletromecânica entre o participante da simulação e a aplicação [8]. O termo háptico se refere à ciência do toque, envolvendo força e sensações ocasionadas pelo tato [9].

Neste caso, a aplicação recebe as informações do dispositivo háptico, e realiza o mapeamento destas para as coordenadas no monitor do computador, alterando a translação e a rotação do objeto que representa o instrumento médico. A aplicação também deve calcular o retorno de força por meio de fórmulas matemáticas, e enviá-las ao equipamento, para que este forneça sensações táteis ao participante do treinamento. Tais fórmulas precisam contemplar os tipos de tecido, as camadas de tecidos até o nódulo, a existência de anomalias, além do ângulo em que a agulha perfura o órgão.

Um obstáculo a ser encontrado nesta etapa do projeto é a diversidade de linguagens de programação, uma vez que o ViMeT está sendo implementado em Java, e

os dispositivos não-convencionais (luva de dados e dispositivo háptico), possuem *drivers* e bibliotecas em linguagem C, disponibilizados pelos fabricantes. Estas bibliotecas têm funções pré-definidas, que facilitam o trabalho com tais equipamentos.

Desta forma, duas abordagens podem ser adotadas: a construção de *drivers* próprios, no caso, em linguagem Java, ou a integração de diferentes linguagens de programação, possibilitando uma *interface* entre a aplicação em Java e as bibliotecas e *drivers* em C. A primeira engloba alguns problemas, tais como: programação em baixo nível, abrangência de diversos tipos de portas de entrada para conexão dos equipamentos, tempo de resposta e tempo dedicado à implementação e testes, que poderia ser utilizado no aperfeiçoamento da aplicação. A segunda abordagem elimina alguns destes problemas, pois os *drivers* e funções já estão testados e aprovados, entretanto, exige conhecimento sobre as linguagens adotadas e bibliotecas disponíveis e pesquisa intensa sobre as funcionalidades de ambas.

Outro aspecto importante relacionado à luva de dados é a modelagem da mão virtual, cuja estrutura pode ser entendida por meio da Figura 4, que mostra a maneira como os membros da mão foram divididos, visando à obtenção de uma maior sensação de realismo, diferentemente da mama e da seringa, objetos que não apresentam divisão nas suas estruturas.

Desta forma, é necessária uma divisão nos membros da mão, pois há uma hierarquia no movimento das falanges dos dedos. A Figura 4 mostra a divisão exata da mão, cujos dedos são ligados à palma da mão da seguinte forma: as pontas dos dedos são ligadas ao corpo dos respectivos dedos, os corpos dos dedos são ligados às esferas de rotação, e as esferas são ligadas à palma da mão. Em testes realizados em outras aplicações, verificou-se que os dedos não executavam a rotação corretamente, por isso a necessidade da inserção de esferas, permitindo que a rotação se limite ao centro de cada esfera, para propiciar um maior realismo.

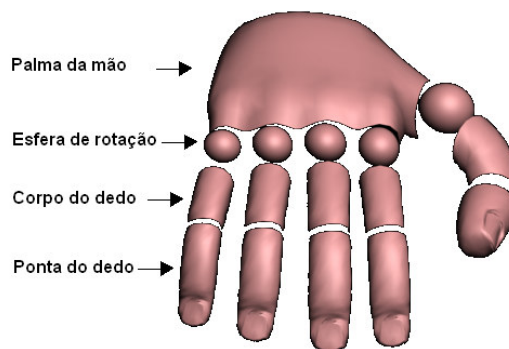


Figura 4: Estrutura da mão modelada

### 2.3 Integração entre linguagens de programação

Para a realização da integração entre as linguagens de programação C e Java, pode ser adotada uma interface de programação nativa, denominada de JNI, que permite a interoperação entre a aplicação escrita em Java e bibliotecas e programas desenvolvidos em outras linguagens, como C, C++, *Assembly* [10].

Para tornar a integração possível, o JDK (*Java Development Kit*) oferece arquivos e ferramentas que permitem a construção de métodos nativos, a compilação dos programas em Java e a geração de arquivos de cabeçalho para a criação de bibliotecas de ligação, o carregamento de tais bibliotecas, a especificação de chamadas e ligações entre métodos JNI (Java) e as funções nativas (Linguagem C) [11].

As bibliotecas de ligação, responsáveis pela *interface* entre a aplicação em Java e a aplicação em C, são geradas por meio de compiladores. No presente trabalho, a plataforma utilizada foi *Windows XP*, e o compilador escolhido para gerar a biblioteca de ligação foi o aplicativo *Microsoft Visual C++ 6.0*, devido à familiaridade com estas tecnologias, sendo que neste tipo de plataforma as bibliotecas de ligação são denominadas DLLs (*Dynamic-Link Libraries*). No entanto, outros Sistemas Operacionais e compiladores podem ser utilizados. Neste contexto, a linguagem Java apresenta-se como uma importante ferramenta, visto que é disponibilizada gratuitamente e oferece recursos interessantes, como portabilidade, classes e métodos prontos [12].

As Figuras 5 e 6 apresentam os diagramas de integração e interação para a luva de dados e o dispositivo háptico. Ambos representam os fluxos de informações entre as partes da aplicação no computador, o usuário e os dispositivos, representados pelas setas. No primeiro diagrama pode-se observar, as informações da luva de dados sendo captadas por funções escritas em linguagem C, provenientes de bibliotecas oferecidas pelo fabricante e transferidas à aplicação por meio de uma biblioteca de ligação, no caso, *5DTGlove.dll*. Tais informações são tratadas pela aplicação desenvolvida em Java e Java3D, que mostra ao usuário por meio do monitor de vídeo, os movimentos do objeto virtual associado ao equipamento.

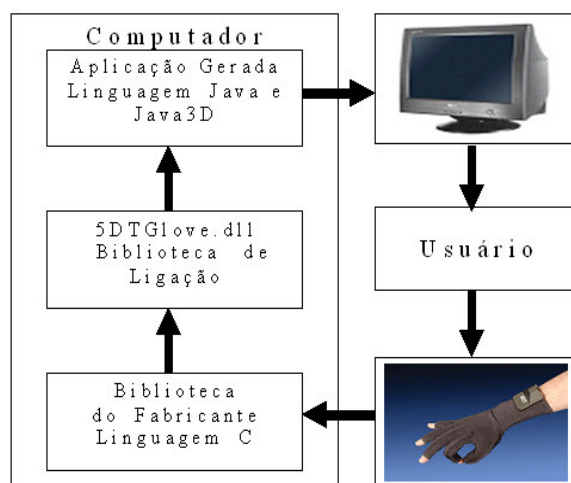


Figura 5. Diagrama de integração e interação para a luva de dados

No diagrama da Figura 6 ocorre uma situação semelhante, entretanto a biblioteca de ligação é denominada *Haptic.dll*, e há a existência de setas em ambas as direções, indicando que a aplicação, além de receber informações do dispositivo háptico, captadas pelas funções do *software OpenHaptics Toolkit*, fornecido pela *SensAble Technologies*, também processa e envia informações ao dispositivo para promover o *feedback* tátil, utilizando a biblioteca de ligação. Desta forma, o equipamento háptico se comporta como um dispositivo de entrada e saída ao mesmo tempo.



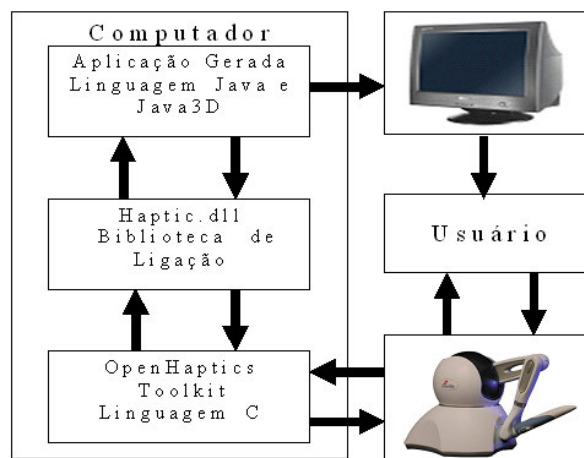


Figura 6. Diagrama de integração e interação para o dispositivo háptico

O monitor de vídeo, presente em ambos diagramas, apresenta os objetos virtuais, suas características, ações e reações dentro do AV. Desta forma, pode-se dizer que há a ocorrência dos retornos visual e tátil durante a interação.

### 3. Resultados e Discussões

Para a inserção do teclado e do *mouse* no *framework*, foram usados *interfaces* e métodos fornecidos pela linguagem de programação Java e a API Java3D. Para a inserção dos dispositivos não-convencionais foi adotada a integração entre as linguagens Java e C.

O módulo de interação com suporte a dispositivos convencionais e não-convencionais foi implementado e incorporado ao ViMeT. A Figura 7 mostra a dinâmica do módulo de interação, com os dispositivos associados aos seus respectivos objetos virtuais, sendo que as setas vermelhas indicam o fluxo de informações, observando-se também que entre o equipamento háptico e o instrumento médico virtual há uma seta bidirecional, que contempla, além do recebimento de informações, o retorno de força. Pode-se notar ainda as combinações de dispositivos disponíveis ao desenvolvedor: *mouse* e teclado, *mouse* e luva de dados, equipamento háptico e teclado, equipamento háptico e luva de dados, mesclando desta maneira, a utilização de equipamentos convencionais e não-convencionais.

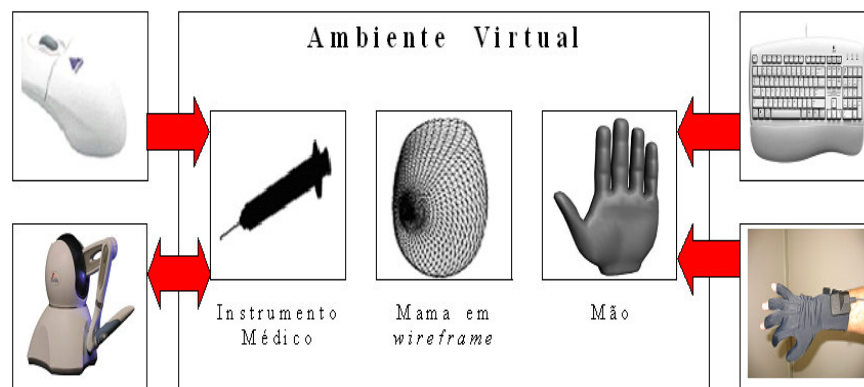
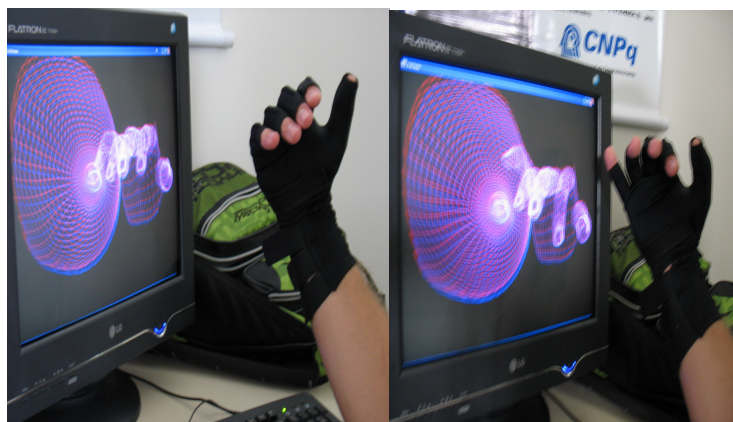


Figura 7. Funcionamento do módulo de interação

Com o uso do teclado, é possível navegar pelo mundo virtual, bastando recorrer às teclas que representam as setas, podendo desta forma, verificar a composição do AV. Além disso, este dispositivo é usado para alterar a rotação dos dedos da mão virtual, representados por teclas específicas, que pressionadas ou liberadas determinam a movimentação dos mesmos, e representam o ato de segurar e soltar o órgão humano durante o procedimento médico. O controle do sistema é realizado automaticamente, visto que pode-se alternar entre navegação e manipulação durante a sessão de treinamento sem interrupção para modificar o modo de interação. Observa-se, no entanto, que não há a ocorrência da operação de seleção, devido à associação existente entre os objetos virtuais e os dispositivos.

O *mouse* e o dispositivo háptico movimentam o instrumento médico virtual de acordo com os movimentos do usuário, entretanto, o dispositivo háptico trabalha com um número maior de informações (translação e rotação nos três eixos), do que o *mouse* (movimentos de translação em dois eixos). Além disso, o dispositivo háptico retorna força ao usuário, causando sensações táteis a este quando a aplicação detecta uma colisão, indicando que o instrumento usado no procedimento tocou o órgão humano virtual. A luva de dados, por sua vez, representa o ato de segurar e soltar o órgão por meio da mão virtual. Os dedos desta última são rotacionados de acordo com a flexão captada pelos sensores da luva, conforme a Figura 8, que apresenta as alterações causadas pela aplicação no objeto virtual (que representa a mão) quando os dedos do usuário são flexionados.



**Figura 8. Demonstração de movimentos captados pela luva dados**

Nos sistemas de RV, principalmente naqueles relacionados ao treinamento médico, é desejável, entre diversas características, precisão e respostas em tempo real, para que a interação durante o treinamento seja mais próxima da realidade [13], oferecendo experiências positivas a médicos e estudantes de Medicina. Um AV para simular procedimentos médicos, de acordo com [14], apresenta vantagens, como: disponibilidade do AV, minimização da necessidade de obter animais ou pacientes reais toda vez que o treinamento precisar ser executado; inexistência de riscos a pacientes, médicos e estudantes da área, pois pessoas não dotadas da habilidade ou conhecimento necessário podem cometer erros, causando danos a determinados tecidos e ferimentos, por exemplo; e diversidade de casos clínicos, apresentando casos normais ou com anomalias e situações encontradas em exames de biópsia no cotidiano de profissionais que executam tais procedimentos.



Nos testes empíricos iniciais, a aplicação apresentou um tempo de resposta satisfatório, ou seja, o usuário recebe as respostas em tempo real. Entretanto, uma fase de avaliação com protocolo mais adequado está sendo elaborada com o intuito de coletar dados importantes no aperfeiçoamento do *framework* e conseqüentemente, da aplicação gerada.

Uma avaliação inicial mostrou que o AV obtém em torno de 30 quadros por segundo durante a interação com o usuário, mesmo após a inclusão de todos os dispositivos citados. O tempo de resposta e a quantidade de quadros gerados por segundo serão mensurados com métodos mais precisos, com o propósito de verificar o desempenho da aplicação, visto que dispositivos não-convencionais são utilizados, e estes exigem um processamento de informações mais elevado se comparados com os convencionais. Além disso, existe a integração entre as linguagens C e Java, que também demanda processamento computacional adicional. O objetivo é que o tempo de resposta seja satisfatório, para não prejudicar a interação com atrasos entre os movimentos do usuário e as reações causadas pela aplicação, não comprometendo conseqüentemente, o treinamento virtual.

Uma característica que é importante destacar é o uso do dispositivo háptico para simular o equipamento utilizado na coleta de material durante o exame. Este dispositivo permite que o usuário tenha sensação tátil e, assim, possibilita o aumento de realismo na simulação. Apesar do seu formato não ser próximo ao formato de alguns instrumentos médicos usados nos exames de biópsia, ainda é o mais utilizado para este tipo de aplicação, visto que não há equipamentos mais específicos disponíveis.

#### 4. Conclusão

Este artigo apresentou a implementação de um módulo de interação com suporte a dispositivos convencionais (teclado e *mouse*), e não-convencionais (luva de dados e equipamento háptico), que foi incorporado ao ViMeT, *framework* de RV, cujo objetivo é disponibilizar uma estrutura de classes para facilitar a geração de aplicações relacionadas ao treinamento médico, inicialmente para simulação de exames de biópsia. Para trabalhos futuros, a detecção de colisão e a deformação para cada dedo virtual deverão ser implementadas, bem como a construção de algoritmos mais precisos de colisão e deformação.

Como a aplicação será avaliada sob aspectos computacionais e humanos, também está prevista uma avaliação por profissionais da área médica. Informações serão coletadas por meio de questionários, como: nível de experiência com dispositivos de RV, nível de experiência na realização de exames de biópsia, facilidade de uso da aplicação com cada dispositivo e cada combinação de dispositivos, uso da intuição durante a interação, sugestões e idéias para aperfeiçoar a simulação, tornando-a mais próxima do real.

Com o módulo de interação implementado, é possível construir aplicações de acordo com a disponibilidade financeira e o grau de realismo desejado, uma vez que dispositivos não-convencionais possuem custo mais elevado quando comparados com os convencionais, entretanto, são projetados para simular ações do usuário no mundo sintético de forma natural, como no mundo real, podendo o desenvolvedor optar por diversos tipos de combinações e realizar um treinamento satisfatório.

## 5. Referências

- [1] G. Riva, “Applications of Virtual Environments in Medicine”, In: MIM – Methods of Information in Medicine, Vol. 42, Número 5, Schattauer GmbH, Alemanha, 2003, pp. 524-534.
- [2] J. Bosh et al., “Framework Problems and Experiences”, In: M. Fayad, R. Johnson, D. Schmidt, “*Building Application Frameworks: Object-Oriented Foundations of Framework Design*”, Nova Iorque, John Wiley and Sons, 1999, pp. 55-82.
- [3] A. C. M. T. G. Oliveira, S. R. Delfino, F. L. S. Nunes. “Virtual Reality Framework for Medical Training: Implementation of a deformation class using Java”. In: *Proceedings of ACM SIGGRAPH International Conference on Virtual-Reality Continuum and its Applications in Industry*, Hong Kong, Nova Iorque: ACM Press, 2006, pp. 347-351.
- [4] R. F. Junior. “Punção Aspirativa por Agulha Fina: Estudo Comparativo entre dois Dispositivos para a Obtenção da Amostra Citológica”, 2001, 136 f., Grau: Tese de Doutorado em Tocoginecologia, Faculdade de Ciências Médicas da Universidade Estadual de Campinas, Campinas, Brasil, Disponível em: <<http://www.rau-tu.unicamp.br/nou-rau/demo-pt/document/?view=57>>, Acesso em: 5 fev. 2008.
- [5] D. Bowman, J. J. L. Jr, E. Kruijff, I. Poupyrev, “An Introduction to 3-D User Interface Design”, In: *Presence: Teleoperators and Virtual Environments*, Volume 10, Número 1, 2001, pp- 96-108.
- [6] 5DT – Fifth Dimension Technologies, Disponível em: <http://www.5dt.com/hardware.html>, Acesso em: Março 2007.
- [7] J. Eisenstein, S. Ghandeharizadeh, L. Golubchil, C. Sahabi, D. Yan, R. Zimmermann, “Device Independence and Extensibility in Gesture Recognition”, In: *Proceedings of the IEEE Virtual Reality*, 2003, pp. 207-214.
- [8] SensAble Technologies, Disponível em: <http://www.sensable.com>, Acesso em: Março 2007.
- [9] J. M. Burns, A. G. Chila, D. C. Eland, J. N. Howell, R. R. C. M. Jr., M. Srivastava, R. I. Willinas, “The Virtual Haptic Back for Palpatory”, In: *Proceedings of the 6<sup>th</sup> International Conference on Multimodal Interaction*, 2004, pp. 191-197.
- [10] G. Cornell, C. S. Horstmann, “Métodos Nativos”, Core Java 2, Volume 2, Recursos Avançados, Pearson Education do Brasil, São Paulo, SP, Brasil, 2003, pp. 755-785.
- [11] JNI, “Java Native Interface Specification”, Disponível em: <http://java.sun.com/j2se/1.4.2/docs/guide/jni/spec/jniTOC.html>, Acesso em: Março 2007.
- [12] C. G. Corrêa, F. L. S. Nunes, A. Bezerra, “Integração de Linguagens de Programação para Uso de Dispositivos Não-Convencionais: Possível Solução para Construir Aplicações com Baixo Custo”, In: *Proceedings of 10<sup>th</sup> SVR Symposium on Virtual and Augmented Reality*, João Pessoa, PB, Brasil, 2008, pp 266-269.
- [13] F. L. S. Nunes, R. M. E. M. Costa, A. C. M. T. G. Oliveira, S. R. Delfino, L. Pavarini, I. A. Rodello, J. R. F. Brega, A. C. Sementille, “Aplicações Médicas Usando Realidade Virtual e Realidade Aumentada”, In: C. Kirner, R. Siscoutto, *Realidade Virtual e Aumentada – Conceitos, Projeto e Aplicações*, Livro do 9<sup>o</sup> SVR Symposium on Virtual and Augmented Reality, Petrópolis, RJ, Brasil, 2007, pp.234-235.
- [14] L. S. Machado, R. M. Moraes, D. F. L. Souza, M. C. O. Valdeck, “SITEG – Sistema Interativo de Treinamento em Exame Ginecológico”, In: *Proceedings of 8<sup>th</sup> SVR Symposium on Virtual and Augmented Reality*, Belém, PA, Brasil, CD-ROM, 2006.