

Ferramentas de Realidade Virtual para os Profissionais da Área Médica

Aline M. Moraes¹, Liliane S. Machado², Ronei M. Moraes³

¹Instituto Federal de Educação, Ciência e Tecnologia da Paraíba (IFPB)

^{1,2,3}Laboratório de Tecnologia para o Ensino Virtual e Estatística (LabTEVE)

Universidade Federal da Paraíba (UFPB)

João Pessoa, Paraíba, Brasil

alinemmcefet@gmail.com

Resumo — O uso de ferramentas computacionais pela comunidade médica pode ser um importante aliado em processos de aprendizado e treinamento. Sistemas baseados em Realidade Virtual (RV) permitem recriar situações que facilitam o processo de entendimento de estruturas anatômicas e sua fisiologia. Porém, os *frameworks* que geram essas aplicações podem apresentar uma série de fatores que distanciam o usuário deste tipo de ferramenta. Neste trabalho serão analisadas algumas necessidades quanto ao uso de ferramentas baseadas em RV pela comunidade médica bem como ferramentas desenvolvidas para o desenvolvimento de aplicações com RV. A partir de uma proposta de Sistema de Programação Visual (SPV) para *frameworks* de RV é apresentado o processo de migração de um *framework* para posterior uso em testes de um SPV.

Abstract — *The use of computational tools in the medical community can be important to processes of learning and training. Systems based on VR allow to design scenarios that can facilitate the comprehension of anatomical structures and human physiology. However, frameworks that generate these applications can present a number of factors that separate the user of these kind of tool. This paper will present an analysis of computer tools needs from medical professionals as well as tools developed to the development of VR systems. Thus, is presented the migration of a framework to support a propose of a visual programming system (VPS).*

Palavras-chave: Realidade Virtual, Programação Visual, *framework*.

I. INTRODUÇÃO

O processo de formação acadêmica do profissional da área médica em instituições de ensino público e privado é composto por diversas etapas. Durante esse período é indispensável o acesso do profissional da área médica a recursos materiais, como peças anatômicas e cirúrgicas, e recursos físicos, como salas para estudo e treinamento práticos [1]. Porém, a realidade mostra que o acesso a esses recursos tem se tornado cada vez mais difícil por motivos que variam desde fatores financeiros até a dificuldade jurídica em obter materiais como cadáveres [2], por exemplo. Uma pesquisa realizada por Fiorotti *et al.* [3] mostra que, de acordo com os alunos da área médica, grande parte das disciplinas é teórica e centrada apenas no professor.

Com base na problemática apresentada, vale destacar que a utilização de ferramentas computacionais tem sido aplicada como uma alternativa para auxiliar estudos e treinamentos médicos. Essas ferramentas podem ser, por exemplo,

componentes que utilizam técnicas de Realidade Virtual (RV) para simular procedimentos [4] e técnicas médicas ou até mesmo ferramentas de visualização. Um levantamento recente [3] mostra que o perfil majoritário dos profissionais e estudantes da área médica é caracterizado por um poder sócio-econômico e cultural elevado em relação aos outros estudantes e profissionais. Em decorrência desse dado, é possível sugerir que o acesso à tecnologia e à informática está próximo desse grupo de usuários.

II. REALIDADE VIRTUAL

A RV é definida por Brooks [5] como uma experiência no qual o usuário sente-se imerso no Ambiente Virtual (AV). Isso acontece como resposta à estrutura apresentada no AV que explora os diversos sentidos humanos. Além disso, a RV é utilizada para representar cenários reais que são, muitas vezes, difíceis de ser experimentados [6]. Na área médica, a visualização em aplicações em RV possibilita, para as habilidades do usuário, mensurar e explorar áreas que são classificadas como difíceis ou impossíveis de serem percebidas com exatidão [7].

Além disso, a RV tem como objetivo oferecer sistemas que integrem aspectos de imersão e interatividade para simular ambientes realistas que permitam ao usuário imergir, navegar e interagir com objetos e outros elementos do ambiente virtual [8]. As aplicações que possuem recursos em RV podem ser geradas por pacotes de *software* chamados de *frameworks*. Tais pacotes são estruturas de suporte no quais outros *softwares* podem ser organizados e desenvolvidos [9].

A geração de aplicações médicas por meio dos *frameworks* depende, na sua maioria, de comandos em linguagens específicas. Porém, a maioria dos profissionais médicos é leiga em linguagens computacionais específicas. Para tornar o processo de manipulação dos *frameworks* uma tarefa mais simples para os usuários da área de saúde seria importante adotar técnicas que tornassem a interação gráfica e simplificada. Uma das técnicas de visualização mais conhecidas por sua eficiência é a PV.

III. PROGRAMAÇÃO VISUAL

A PV é comumente definida como o uso de expressões visuais tais como gráficos, desenhos, animação ou ícones na programação [10]. Algumas características da PV [11] são:

a) as imagens representam um maior nível de abstração, por isso a sintaxe não está envolvida; b) quando projetadas adequadamente, as imagens podem ser compreendidas pelas

peças, independentemente da língua que falam e c) as imagens podem transmitir um significado em uma unidade mais concisa do que uma expressão em texto.

De acordo com Green [12], existem três motivos pelos quais a comunidade computacional estuda PV até hoje, independentemente de sua técnica. São elas:

- A PV é tecnicamente desafiadora, pois existem muitos pontos que precisam ser estudados em pesquisa;
- A PV suporta o modelo de fluxos de dados da computação muito melhor do que programação textual convencional;
- Embora os programadores sejam usuários, a Interação Humano Computador (IHC) convencional tem agido pouco em projetos de ambientes de programação.

IV. FRAMEWORKS DA ÁREA MÉDICA

Os *frameworks* em RV se caracterizam por ser uma implementação abstrata para o desenvolvimento de aplicações em um problema de domínio específico e com a vantagem de reutilizar componentes [13]. Isso propicia que um usuário consiga executar de maneira simplificada uma série de passos comuns.

Uma parcela desses *frameworks* não é projetada para área médica exclusivamente como é o caso do SCIRun [14] e o LabView [15], por exemplo. Essa característica pode gerar uma restrição em funcionalidades para a área médica. Outro fato que merece ressaltar é que muitas dessas ferramentas computacionais são desenvolvidas por profissionais da computação que implementam e testam os recursos de interação. Provavelmente os recursos gráficos e de interação serão insuficientes para usuários leigos, incluindo os profissionais médicos.

Realizou-se um levantamento das características gerais dessas ferramentas computacionais voltadas especificamente para a área médica no qual foram investigadas quatro características principais, como pode ser visto na Tabela I: a) a existência de recursos em RV, b) a existência de PV, c) os pontos negativos de tais *frameworks* e d) os Sistemas Operacionais (SO) compatíveis. A presença de RV é um elemento importante para garantir que essas ferramentas computacionais possuam características imersivas e interativas em tempo-real para o usuário da área médica. Outro aspecto diz respeito à presença de PV porque essa técnica facilita o processo de interação em *frameworks* para os usuários leigos em computação quando inserida em uma ferramenta computacional. Outra característica importante observada foi se o *framework* é multiplataforma, porque quanto maior for o número de SOs compatíveis, maior será a quantidade de usuários com diferentes níveis de conhecimento que utilizarão o *framework*. Entretanto, o tipo de plataforma não deve ser um agravante adicional na interação para os profissionais da área médica.

Na pesquisa realizada, sete *frameworks* foram encontrados que são utilizados ou possuem potencialidades para serem usados pelos usuários da área médica, como pode ser visto na Tabela I. Com relação às características

apresentadas, é possível observar que não se encontrou uma ferramenta computacional completa para os usuários da área médica, ou seja, que fosse de fácil manipulação para os usuários leigos em computação, que ofereça recursos de visualização e treinamento e que tenha uma licença de uso *open-source*. Dentre os *frameworks* baseados em RV da área médica é possível destacar um que atende a maioria dos requisitos solicitados por um profissional da saúde abordados anteriormente e ele é chamado de CyberMed¹.

TABELA I. COMPARATIVO ENTRE OS FRAMEWORKS NA ÁREA MÉDICA

Framework	Possui RV?	Possui PV?	Desvantagens	SO compatível
MeVisLab	Sim	Sim	Licença fechada	Windows Linux Mac OS X
ViMeT Wizard	Sim	Não	Recursos limitados do ViMeT	Windows
SCIRun	Sim	Sim	Recursos limitados para a medicina	Windows Linux Mac OS X
NeatVision	Não	Sim	A forma de criar novos componentes não é intuitiva [17]	Windows Linux Solaris
VisProg	Não	Sim	Licença fechada	Windows Linux Mac OS X
LabView	Não	Sim	Licença fechada	Windows Mac OS X
CyberMed	Sim	Não	Não tem uma interface visual	Windows

De acordo com Machado *et al.* [18], o CyberMed é um *framework* que possui como principais recursos em RV a visualização estéreo, a interação através de dispositivos convencionais e hápticos, o uso de modelos deformáveis e o suporte à colaboração. Além disso, tal *framework* apresenta características que o destaca em relação aos demais *frameworks* em RV [19]. São as seguintes:

- É um *framework open-source* projetado para ser executado na plataforma PC;
- Possui integração de métodos de avaliação *online* do usuário.

O CyberMed é capaz de gerar aplicações em RV que contem imagens anatômicas com recursos hápticos e estereoscópicos e trabalha com a biblioteca gráfica OpenGL. Além disso, esse *framework* é compatível com diversos dispositivos não-convencionais de interação, como o *phantom* e o *shutter*, que aumentam o grau de imersão do usuário na visualização e interação das aplicações em RV. Contudo, esse *framework* não possui um módulo gráfico que propicie facilidade na programação de aplicações médicas, principalmente por parte dos usuários da saúde. Baseando-se no empecilho de interação do CyberMed agregado a escassez

¹ <http://cybermed.sourceforge.net/>

de ferramentas computacionais que auxiliem o estudo de profissionais médicos, mostrado na sessão anterior, será proposta a utilização de um Sistema de Programação Visual (SPV) no CyberMed chamado de CyberMedVPS.

V. CYBERMEDVPS

O sistema proposto por Moraes, 2011 [20] é um SPV que se define como um módulo visual de *frameworks* em RV e permitirá que usuários leigos em programação consigam executar e criar suas próprias aplicações em RV. Além disso, ele será construído de tal maneira que outros *frameworks* em RV possam ser integrados a esse SPV.

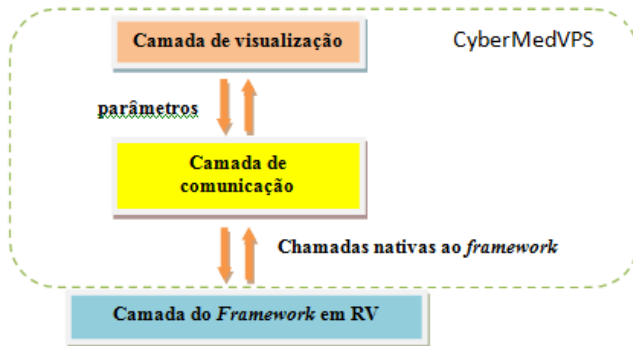


Figura 1. Arquitetura do SPV proposto [20]

Como pode ser visto na Figura 1, a arquitetura do sistema é composta por três componentes principais: a) a camada de visualização que corresponde ao componente que contém as técnicas de PV, b) a camada de comunicação que é responsável pela tradução entre a camada de visualização e a camada referente ao *framework* e c) a camada do *framework* que corresponde à ferramenta computacional para geração de aplicações médicas com RV. Em um primeiro momento o SPV trabalhará apenas com o CyberMed.

A criação do SPV possivelmente irá facilitar o uso de ferramentas computacionais baseados em RV, pois deverá permitir que o usuário médico consiga programar aplicações com recursos em RV pelo modo gráfico. Embora a PV torne a interação do CyberMed mais intuitiva para usuários leigos, ainda existe uma característica que merece ressalva: o *framework* que será utilizado para testes não é multiplataforma. Tal fato ocorre porque o desenvolvimento do CyberMed voltou-se para o SO Linux. A limitação de plataforma pode ser um fator excludente para usuários que desejam interagir com o CyberMed, pois entre usuários comuns o uso do Windows ainda é predominante [21]. Com base nesse aspecto é proposta neste trabalho a migração do CyberMed para a plataforma Windows.

VI. PLANEJAMENTO E MIGRAÇÃO DO CYBERMED

Para definir as etapas da migração do CyberMed será necessário:

- fazer um levantamento das dependências para a compilação do CyberMed no Linux;
- verificar se existe alguma alternativa análoga para as dependências serem compiladas no Windows;
- aplicar as mudanças das dependências no Linux;

- gerar uma aplicação de exemplo;
- executar a aplicação do CyberMed no Windows.

O CyberMed para ser executado no Linux precisa das seguintes dependências para compilar: a) do compilador *g++*, já que o *framework* é todo feito na linguagem C++; b) da biblioteca *glut-devel*, para trabalhar com OpenGL; c) da biblioteca de *threads*, nativa do SO Linux. d) da biblioteca *OpenHaptics* para compilar o módulo do CyberMed referente ao suporte háptico.

No SO Windows verificou-se que as seguintes modificações serão necessárias para compilar o CyberMed: a) instalação do ambiente MinGW que embarca o compilador *g++* para o Windows; b) instalação da biblioteca OpenGL que possui uma versão voltada para Windows; c) instalação da biblioteca *Pthreads* que compatibiliza as chamadas nativas da biblioteca *Thread* do Linux para o Windows e d) instalação da biblioteca *OpenHaptics* para Windows.

A implementação dessas dependências no Windows começou pela instalação do MinGW, seguida da OpenGL, depois a *Pthreads* e por fim a instalação do *OpenHaptics*. Após instalar as dependências, foram realizadas alterações nos arquivos *Makefile* dos módulos do CyberMed para adaptar as novas versões das dependências para que os arquivos de configuração compilassem adequadamente os projetos desenvolvidos com o *framework*.

VII. RESULTADOS PARCIAIS

Atualmente os módulos *Utils*, *Core*, *View*, *Assess*, *Deformation*, *Collision*, *Network* e *Collaboration* (Figura 2) encontram-se compilados no SO Windows. Com esses módulos disponíveis já foi possível desenvolver uma aplicação de visualização simples utilizando o CyberMed, como pode ser visto na Figura 3.

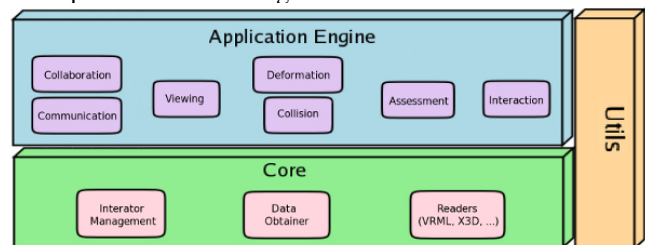


Figura 2. Módulos do CyberMed

Há duas formas de compilar e executar a aplicação de visualização do CyberMed. A primeira é criando um *Makefile* que compile a aplicação com as dependências do Windows. A outra forma é executar a compilação pelo *g++* do MinGW, ou seja, diretamente no *prompt* de comando do SO. No final das duas compilações é gerado um arquivo executável da aplicação cujo resultado pode ser visto na Figura 3.

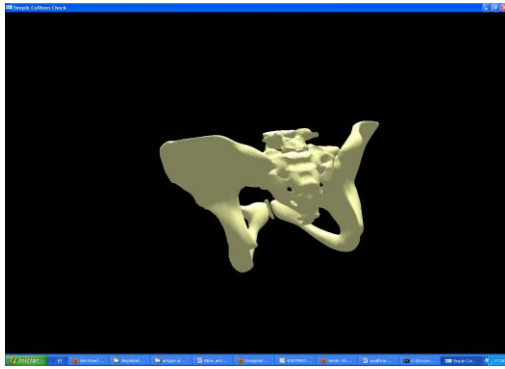


Figura 3. Aplicação de visualização em 3D do CyberMed executada no SO Windows

VIII. CONSIDERAÇÃO FINAL

Atualmente, a migração se encontra em processo de compilação do módulo Haptics referente ao suporte háptico do CyberMed. Após a compilação desse módulo será desenvolvida uma aplicação de teste com suporte háptico e a compilação do módulo Tracker. Com toda a migração concluída será possível disponibilizar o CyberMedVPS também em diversas plataformas.

O desenvolvimento de um SPV para a comunidade médica visa suportar não apenas o CyberMed, mas oferecer uma interface gráfica que possa ser comum a outros frameworks de forma unificada, ou seja, utilizando as mesmas representações gráficas independentemente do framework utilizado de modo que o usuário possa construir suas próprias aplicações com recursos de RV. Neste caso, o SPV deverá prever uma execução em diferentes plataformas e permitir o acoplamento de outros frameworks à sua estrutura, como ilustrado na cor azul na Figura 1.

AGRADECIMENTOS

Este projeto está vinculado ao Instituto Nacional de Ciência e Tecnologia (INCT) – MACC e apoiado pelo Conselho Nacional Brasileiro de Desenvolvimento Científico e Tecnológico, cujo número do processo CNPq é 312375/2009-3.

REFERÊNCIAS

- [1] G. Merrill, R. Raj e J. Merrill “*Changing the Focus of Surgical Training*”. VR World, March/April 1995, pp. 56-61.
- [2] E. N. Melo e J. T. Pinheiro, “Procedimentos Legais e Protocolos para Utilização de Cadáveres no Ensino de Anatomia em Pernambuco”. Revista Brasileira de Educação Médica (RBEM 2010). Número 34 (2). Pags 315 -323, 2010.
- [3] K. P. Fiorotti, R. R. Rossoni e A. E. Miranda, “Perfil do Estudante de Medicina da Universidade Federal do Espírito Santo, 2007”. Revista Brasileira de Educação Médica (RBEM 2010). Número 34 (3). Pags 355-362, 2010.
- [4] K. E. Augustine, D. R. Holmes III e R. A. ROBB “*ITK And ANALYZE: A Synergistic Integration*”. Proceedings of the SPIE, vol. 5367, Medical Image. pags 6-15, DOI:10.1117/12.533484, 2004.

- [5] F. P. Brooks, “What’s real about virtual reality,” *IEEE Comput. Graph.Appl.*, vol. 19, no. 6, pp. 16–27, Nov./Dec. 1999.
- [6] M. Massiah “*The Use of Virtual Reality in Industry*”. 4th Annual Multimedia Systems, Electronics and Computer Science, University of Southampton, 2003.
- [7] G. D. Lecakes e J. L. Schmalzel, “*Virtual Reality Environments for Integrated Systems Health Management of Rocket Engine Tests*”. IEEE Transactions On Instrumentation And Measurement, vol. 58, Número 9, September 2009.
- [8] W. Greenleaf, “Medical Applications of Virtual Reality”. Disponível em: <http://www.greenleafmed.com/publications/VR%20Med%20overvie w.pdf>. Acessado em Maio de 2011.
- [9] M. M. Rahman, “*An Application Of Prototyping Models For Web Application Development Using Php Framework Cakephp*”, Independent University, Bangladesh, 2007.
- [10] J. Paquet e J. Plaice, “Dimensions And Functions As Values”. Proceedings of the Eleventh International Symposium on Lucid and Intensional Programming, Sun Microsystems, Palo Alto, California, EUA, 1998.
- [11] Dong, L. “*Transforming Visual Programs into Java and Vice Versa*”, Master’s Thesis, Dalhousie University, Halifax, Nova Scotia, Canadá, 2002.
- [12] T. R. G. GREEN, “*Noddy's Guide To ... Visual Programming*”. Interfaces, The British Computer Society, Human-Computer Interaction Group. 1995.
- [13] M. Mattsson, J. Bosch, “*Framework Composition: Problems, Causes and Solutions*”. Course Documentation. Technology of Object-Oriented Languages and Systems, 1997.
- [14] C.R. Johnson e S.G. Parker. “Applications In Computational Medicine Using SCIRun: A Computational Steering Programming Environment”. H.W. Meuer, editor, Supercomputer ‘95, pags 2–19. Saur-Verlag, 1995.
- [15] L. Regan e M. Gregory. “Flux Analysis Of Microbial Metabolic Pathways Using A Visual Programming Environment”. Journal of Biotechnology 42, pags 151-161, 1995.
- [16] L. S. Machado, D. F. L. Souza, L. C. Souza e R. M. MORAES, “Desenvolvimento Rápido de Aplicações de Realidade Virtual Utilizando Software Livre”. X Simpósio em Realidade Virtual e Aumentada, pags 5-33, João Pessoa, 2008.
- [17] J. A. Chávez-Aragón et al., “*JADE: A Graphical Tool for Fast of Imaging Applications*”. Advanced Techniques in Computer Sciences and Software Engineerings, Ed Springer, pg 401, doi: 10.1007/978-90-481-3660-5. 2009.
- [18] R. Moraes e L. Machado, “Development of a Medical Training System with Integration of Users' Assessment”. Book Chapter. In: Jae-Jin Kim (Ed.), Virtual Reality, Ch. 15. Croatia: Intech. 2011.
- [19] A. M. Morais e L. S. Machado, “Programação Visual para um Framework de Desenvolvimento de Aplicações Médicas Baseadas em Realidade Virtual”. Congresso Brasileiro de Informática em Saúde (CBIS 2010). Porto de Galinhas, Brazil, 2010.
- [20] A. M. Morais e L. S. Machado, “*CyberMedVPS: Visual Programming forDevelopment of Simulators*”. Medicine Meets Virtual Reality 18, J.D. Westwood et al. (Eds.), IOS Press, 2011. doi:10.3233/978-1-60750-706-2-386.
- [21] N. Economides e E. Katsamakakos, “Linux vs.Windows: A Comparison of Application and Platform Innovation Incentive for Open Source Proprietary Software Platforms” . The Economics of Open Source Software Developmente. Elsevier. Cap 10. 2006.