

A High-Level Haptic Interface for Enhanced Interaction within Virtools™

Matthieu Poyade¹, Arcadio Reyes-Lecuona¹, Simo-Pekka Leino², Sauli Kiviranta², Raquel Viciania-Abad³, and Salla Lind²

¹ Departamento de Tecnología Electrónica, ETSI de Telecomunicación, Universidad de Málaga, Campus de Teatinos, s/n, 29071 Málaga, Spain
{matthieu.poyade, areyes}@uma.es

² Human-Machine Systems, VTT Technical Research Centre of Finland
Tekniikankatu 1, FIN-33101, Tampere, Finland

{Simo-pekka.Leino, Sauli.Kiviranta, Salla.Lind}@vtt.fi

³ Departamento de Ingeniería de Telecomunicación, Escuela Politécnica Superior de Linares, C/ Alfonso X El Sabio, 28, 23700 Linares, Spain
rviciania@ujaen.es

Abstract. Haptics is the outstanding technology to provide tri-dimensional interaction within Virtual Environments (VE). Nevertheless, many software solutions are not fully prepared to support Haptics. This paper presents a user-friendly implementation of Sensable Phantom haptic interfaces onto the interactive VE authoring platform, Virtools 4.0. Haptics implementation was realized using the Haptic Library (HLAPI) from OpenHaptics toolkit 2.0 which provides highly satisfactory custom forces effects. The integration of Phantom interaction at end-user development fulfils logical VE interactive authoring under Virtools. Haptics implementation was qualitatively assessed in a manual maintenance case, a welding task, as a part of the national Finnish project, VIRVO. Manipulation enhancements provided by the integration of Phantom interaction in Virtools suggest many further improvements for more complicated industrial pilot experiments as a part of the European Commission funded project ManuVAR.

Keywords: Virtual Reality, Haptics, OpenHaptics, Virtools™, Force Feedback.

1 Introduction

Haptic displays have been widely used for interaction technique to support manipulation tasks in Virtual Environments (VE) [1]. The haptic modality is included in a large number of commercial and non-commercial Virtual Reality (VR) applications, implicated in several industrial domains such as mechanical engineering. For instance, haptics was a significant issue in the national Finnish project VIRVO [2] [3], for the development of industrial virtual engineering tools and methods in order to improve maintainability design and maintenance task planning. Usability and benefits of virtual engineering tools were tested in two industrial case-studies related to mining and construction as well as elevator manufacturing and service business. In VIRVO project, haptics was also tested in several generic test cases. Furthermore, VR and

haptics are strongly considered as a potential industrial solution in the European Commission funded project ManuVAR¹, to support manual work through Product Lifecycle Management (PLM).

The implementation of high performance Force Feedback (FF) haptic interfaces onto VE development platforms may strongly improve tri-dimensional interaction within VE. Nevertheless, most of haptic devices are not implemented onto VE development platforms graphical engines. This is the case of the Sensable Technologies OpenHaptics[®] toolkit onto the commercial VE development platform Virtools[™] 4.0. So, it appears necessary to facilitate access to commercial VE development platform to inexpensive haptic interfaces users.

This paper presents a friendly end-user Sensable Technologies Phantom haptic interface implementation for Virtools[™] 4.0 using Haptic Library (HLAPI) from OpenHaptics toolkit 2.0.

2 Technical Background

2.1 Virtual Environment Development Platform

The development of VEs requires powerful programming tools as software interfaces that ensure an optimum use of graphics hardware. The most commonly used is OpenGL [4] [5] which consists in a standard C++ based graphical platform API able to render two and three-dimensional objects geometric primitives through a series of processing stages known as OpenGL Rendering Pipeline. OpenGL programming is enhanced by additional libraries as the OpenGL Utility Library (GLU) and the OpenGL Utility Toolkit (GLUT). Recent developments concerning OpenGL have led to the rise of several toolkits as OpenSceneGraph [6], a high performance tri-dimensional graphics toolkit which uses a scene graph data structure to render objects geometric primitives.

VE development is strongly eased by visual programming software as Virtools[™] [7], a development platform able to support key industry standard OpenGL, that allows intuitive authoring of tri-dimensional interactive graphic applications.

Virtools[™] authoring development, also known as end-user development, is graphically-based on linking logical components, the Behavior Building Blocks (BBs), described into software libraries. High-end development consists in advanced programming using the Virtools[™] development tools: Virtools[™] Software Development Kit (SDK) and Virtools[™] Script Language (VSL). Virtools[™] SDK deals with standard C++ programming that provides access to rendering and behavioral engines allowing BBs programming in order to improve end-user development by implementing custom functions and VR device interfaces.

Virtools[™] context is a real-time engine supported by a standalone threaded process loop, which includes behavioral and hierarchical graphical rendering sub-processes. So, primitive shapes are first vertex-based rendered and then hierarchically processed.

¹ ManuVAR is a European Union funded project, envisaged to start in spring 2009.

2.2 Haptics and Implementation on Virtual Environment Development Platforms

Haptics. Haptics refers to manual interaction with environments [1]. More specifically, it is associated to the ability of sensing and manipulating environment components through the sense of touch characterized by tactile and proprioceptive information [8].

Haptics in VEs is characterized by a multimodal real-time sensory-motor interaction between the human and the machine motor channels [1] [8], inducing reciprocal Force Feedback (FF). It is proved that haptic FF strongly improves task performance due to a higher maneuverability and enhances interaction making it more pleasurable [9] and environmentally more immersive [1].

Even though many works have successfully implemented the haptic modality in tri-dimensional interaction within VEs [9] [10] [11], managing with haptics interfaces keeps being a delicate labor that requires advanced programming knowledge.

Implementation of Haptics. The OpenHaptics® toolkit 2.0 [12] is an OpenGL-based haptic library for Phantom devices by SensAble Technologies Inc® providing a Phantom-based interaction [13].

OpenHaptics® toolkit has a two layered architecture: the Haptic Device API (HDAPI) and the Haptic Library API (HLAPI). HDAPI provides a low-level access to haptic device; it requires advanced programmer skills. HLAPI is a friendly high level rendering engine that uses a description of OpenGL's graphics stores graphic card buffers to display haptics. It offers several extension modules such as custom force effects (stiffness, damping, friction, dynamic friction, viscosity...) and threads management to support haptic rendering. The client thread (~30 Hz) supports graphical rendering. The collision thread (~100 Hz) supports collision detection. The servo thread (~1000 Hz) handles the position and orientation of the haptic device and updates forces.

Recent improvements in the brand new version OpenHaptics® toolkit 3.0 proposed an implementation of HLAPI pre-written functions and a top third layer API, Quick Haptics which shortens and simplifies OpenHaptics® programming in OpenGL.

There are plenty of other haptic APIs for non-commercial VR platforms. The most prominent are the SenseGraphics² H3DAPI and VHTK for OpenGL and osgHaptics for OpenSceneGraph.

The Haption Company [14] provides integration modules using haptic libraries corresponding to the Virtuouse™ haptic devices onto various Dassault Systèmes industrial VR solutions like Virtools™ and Catia.

Concerning Phantom interaction onto Virtools™ platform, few works can be found. The Phantom interface was successfully implemented using the obsolete Sensable Technologies Ghost® SDK³. A much recent approach [15] used OpenHaptics® toolkit 2.0 HDAPI allowing a low level control of forces.

Our approach uses HLAPI which simplifies Virtools™ end-user authoring development and provides a wide set of customizable haptic force effects.

² SenseGraphics AB - Home, <http://www.sensegraphics.com>

³ <http://www.theswapmeet-forum.com/forum/download/file.php?id=1854>

3 Description of the Developed Interface

3.1 Presentation of the Developed Interface

The proposed interface is a set of extension modules (BBs) for Virtools™ 4.0 platform that facilitate the integration of Sensable Technologies Phantom Omni and Desktop⁴ haptic interfaces at end-user development stage. The integration of the extension modules at the end-user development stage can be easily realized regarding the Virtools™ logical components interlinking policy. Furthermore, the proposed extension allows adding and removing objects from the VE at end-user development stage.

3.2 Development of the Extension Modules

Basics of High-end Development. The Development of extension Modules for Virtools™ 4.0 platform results in a standard C++ programming using the Virtools™4.0 SDK pre-written functions. The whole project was compiled with Microsoft Visual Studio.NET 2003.

OpenHaptics programming using HLAPI requires setting up an alternative OpenGL context to store a copy of graphics vertices into specific buffers. This provides support to the average 1 KHz haptic rendering, which would be hardly achievable in a Virtools™ based context. The solution consists in supporting haptics within a hidden OpenGL context located on the top of the Virtools™ 3D Layout and then, proceeding with haptic coding using OpenHaptics® toolkit HLAPI programming.

Virtual objects geometries from Virtools™ are selectively processed in the alternative OpenGL context to render a vertex-based mapped haptic model which is a unique haptic shape that gathers all the Virtools™ objects to be hapticallized. The developed interface deals then with a real-time recognition of haptic geometries and their respective haptic characteristics while touching objects in the VE.

Basics of End-user Development. The interface extension results in a pair of BBs which facilitate end-user programming. These BBs were named “Phantom Sensable Waiter” and “Hapticallize Object”, respectively shown in Fig 1.a and Fig 1.b.

“Phantom Sensable Waiter” BB launches the OpenGL context, initializes the Phantom haptic device, manages and outputs Phantom device 6 DOF sensing, handles object position and orientation, deals with collision detection and force feedback effects, scales manipulation area and sets up VE’s viscosity. Fig 1.a details the input and output parameters of the “Phantom Sensable Waiter” BB that allow a natural integration into Virtools™ scenarios with interconnected BBs. The “Phantom Sensable Waiter” BB input parameters set up the haptic bounding box and its location into a VE in order to scale and position the manipulation area, to specify the group of touchable objects to which all hapticallized objects belong, and the viscosity parameter.

⁴ The Phantom Omni and Desktop haptic interfaces are non-expensive 6 Degree-of-Freedom (DOF) position sensing electromechanical desktop devices with high fidelity Force Feedback output onto 3 DOF, with a maximum respectively set to 3.3N and 7.9N. Typical Phantom device configuration includes a stylus grip so that users can handle it as a pen.

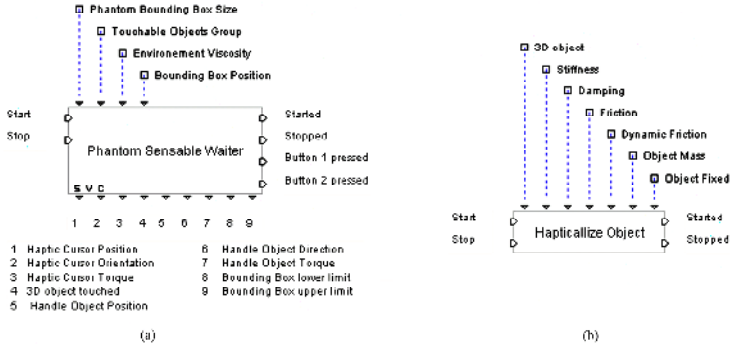


Fig. 1. (a) The Phantom Sensable Waiter BB and (b) the Hapticallize Object BB

“Hapticallize Object” BB supports as input parameters, the specification of the object to hapticallize, its mobility state and haptic characteristics as Stiffness, Damping, Friction, Dynamic Friction and Object Mass used to define object’s weight and inertia. Haptic characteristics in the “Hapticallize Object” BB are quantified from 0 to 1, respectively representative of the null force level and the maximum force level applied by the Phantom haptic device². Fig 1.b details the input parameters of the “Hapticallize Object” BB.

The implementation of the developed interface was tested in a simple Virtools™ generated VE consisting of basic geometric shapes as shown in Fig 2. This first trial allows examining the functionality of the implemented interface, so that objects contact and handling can be ensure as suitable.

The interconnectivity with other BBs is a key issue of the integration of the developed modules in Virtools™. “Phantom Sensable Waiter” and “Hapticallize Object” BBs must be connected into Virtools™ scenarios as shown in Fig 3.

“Hapticallize Object” BB specifies haptic characteristics of a unique object, so it is necessary to declare one BB per object.

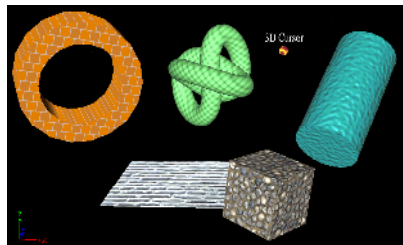


Fig. 2. The trial scene for rapid examination of the developed interface functionality

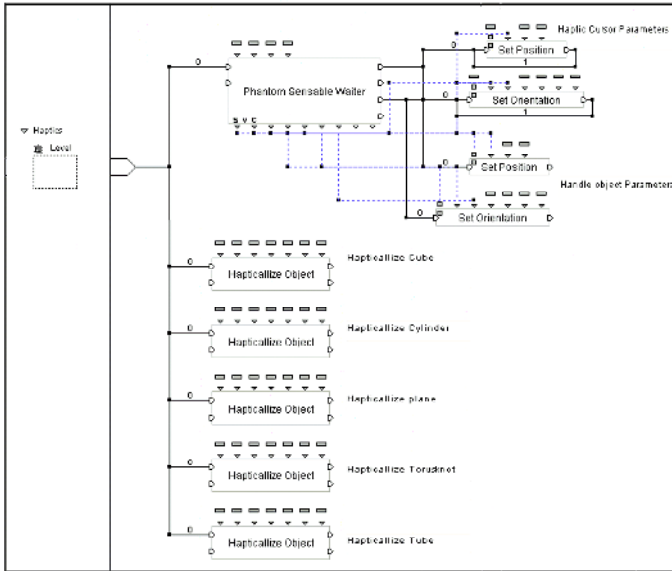


Fig. 3. An example Virotools™ scenario with interconnection on the developed interface

“Phantom Sensable Waiter” returns the haptic cursor and touched object position, direction and torque values. In order to see graphical shapes moving following haptics, it is necessary to assign position and orientation values to graphical shapes that represent the haptic cursor and the touched objects, by linking “Phantom Sensable Waiter” position and orientation output parameters to appropriate BBs input parameters. “Set Position” and “Set Orientation” BBs are required and connected to “Phantom Sensable Waiter” outputs to receive position and orientation data in order to graphically set up position and orientation of the haptic cursor and touched objects.

4 Assessment of the Developed Interface

4.1 Experimental Methods

In VIRVO project [3], a usability industrial test of new haptic user interface in VE was carried out. The test consisted in a virtual welding task using haptic device (SensAble Phantom) as a welding torch. In the welding task, testing subjects connected two parts of a rock crusher during reassembly task of crusher maintenance. The virtual model was built from CAD models imported into Virotools simulation software. Haptic interface was built using the developed haptics interface between Virotools and SensAble Phantom device. Fig. 4 shows a virtual welding task model dedicated to public audience.

There were 11 participants involved in usability test, three Development engineers, two project managers, two machine engineers, one mechanics engineer, one mechanic, one documentation engineer and one design engineer. All the testing subjects were employees of Metso Minerals Company.



Fig. 4. A virtual welding task for public audience

All participants had at least little experience of tri-dimensional models and CAD. Development engineers, mechanics engineer and design engineer got most experience. Most of the participants did not have experience of VR or simulation at all, while two development engineers and the mechanics engineer got little experience. Two development engineers got some experience of haptics, while the rest of tests persons had no experience at all.

Before the actual usability test, participants had possibility to practice onto a general assembly demo using haptics and leading with replacing a car pneumatic. Every test person did the welding task, and after the task, filled in a questionnaire form to qualitatively assess the haptic implementation. In the questionnaire, participants' experience level concerning tri-dimensional models, CAD, Virtual Environments, simulation, haptics was self-evaluated. Possible answers were no experience, little experience or professional.

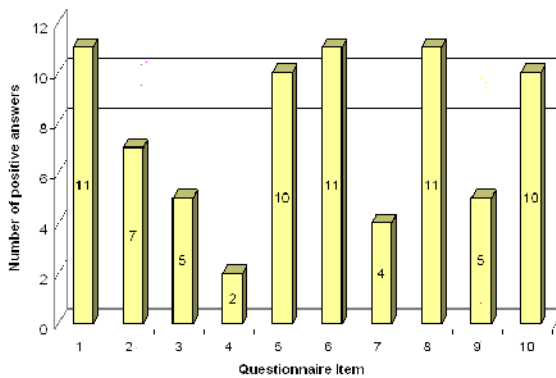


Fig. 5. Number of positive answers for each questionnaire item

The participants answered in questionnaire of 10 dichotomic items in which they have to report whether they agree or not to following statements: 1) Haptic interface is easy to use, 2) user interface is natural, 3) improved 6DOF navigation, 4) easier to

use than tri-dimensional-mouse, 5) sense feedback improves interface, 6) force feedback improves interface, 7) haptics improves design work, 8) haptics improves training of work tasks, 9) I could adopt haptics as working tool and 10) positive attitude. The participants had also the possibility to give their free comments about haptics and ideas about haptics applications in industry.

4.2 Experimental Results

Users Experimentation Results. Fig. 5 summarizes all the experimental results. All participants assessed haptic user interface to be easy to use. Project managers and design engineers did not find haptics as natural user interface. Machine engineers and design engineers thought that haptic interface improves 6 DOF navigation in VE. Two development engineers found haptic interface easier to use than tri-dimensional mouse. All, but one project manager, assessed sense feedback of VE to be good and all participants appreciated force feedback. One development engineer, documentation engineer, design engineer and machine engineer assessed haptics to improve their design work. All persons thought that haptics improves work task training in industrial environments. Half of persons thought that they could adopt haptic interface into their work tools. Only one project manager did not have positive attitude against haptic interface.

The participants reported that haptic manipulation was surprisingly easy to learn but required more training in order to be considered its potential as design or training tool. As well, participants underlined that haptics is far away from realism but provide hints about mass of big machine parts, enhancing the objects manipulation and perception within VE.

End-User Development Comments. Performance of the haptics driver was well sufficient for end-user development of training applications. Performance of the driver was suitable for training task and overall stability was very satisfactory.

Haptic engine presented OpenHaptics limitations on capability to process high accuracy models. Direct CAD models from industrial partners were too complicated for the haptic engine, so reducing models' details was compulsory to use such device.

One drawback of the current implementation was the lack of capability to supply feedback between manipulated object and environment, needed in assembly tasks. On the contrary, welding task simulator using objects contact feedback received exceptionally good results from users.

5 Conclusion and Future Works

This paper has presented a friendly implementation of Haptic Library (HLAPI) from OpenHaptics[®] toolkit 2.0 onto Virtools[™] 4.0, resulting in a set of Virtools BBs allowing an easy end-user development.

The developed interface was qualitatively assessed by VTT and industrial partners in a usability test case dealing with a virtual welding task onto a rock crusher during a reassembly task of crusher maintenance.

Experimental results showed usability of the developed haptic interface at user level. Those results supported haptics as a powerful interaction tool in virtual

maintenance tasks even if participants felt that haptic force feedback was far away from realistic force models.

Comments by users and end-user developers from VTT suggested further improvements in order to improve end-user development and tri-dimensional haptic interaction in VEs generated with Virtools™ 4.0 and newer versions.

A firstly suggested improvement would consist in enhancing BBs with a scalable haptic bounding box so that haptic workspace would automatically fit to visual workspace. A secondly suggested improvement would provide haptic force feedback between manipulated objects and environment. This improvement would require solving the problem of complex collision detection at a high rate and the use of torque forces, but would result in strongly enhancing manipulation realism. A thirdly suggested improvement would remain in the implementation of a both hands interaction within VE through a set of two Phantom devices. Effectively, objects manipulation using both hands might appear more natural and immersive.

Furthermore end-user development would be strongly improved by implementing a database of most common haptic materials so that setting up materials of Virtools generated Objects would results as easy as selecting them from a collection of already defined OpenHaptics material.

Acknowledgements

This works was partially supported by the Spanish Ministry of Education and Sciences (Project TIN2006-15202-C03-02). Authors would like to thank Prof. Dr. Christian Geiger from the Düsseldorf University of Applied Sciences for sharing his knowledge about the integration of OpenHaptics on Virtools.

References

1. Srinivasan, M.A., Basdogan, C.: Haptics in Virtual Environments: Taxonomy, Research Status, and Challenges. *Computers & Graphics* 21(4), 393–404 (1997)
2. Leino, S.-P., Helin, K., Lind, S., Viitaniemi, J., Multanen, P., Mäkiranta, A., Lahtinen, J., Nuutinen, P., Heikkilä, J., Martikainen, T.: Virtual engineering and remote operation in design, training and completion of demanding maintenance work tasks in challenging industrial plants (VIRVO) - MASIT 15. In: *MASI Programme 2005–2009, Yearbook 2008*, Tekes, pp. 111–120 (2008)
3. Leino, S.-P., Lind, S., Poyade, M., Kiviranta, S., Multanen, P., Reyes-Lecuona, A., Mäkiranta, A., Muhammad, A.: Enhanced industrial maintenance work task planning by using virtual engineering and haptic user interfaces. In: *HCI International 2009, 13th International Conference on Human-Computer Interaction*, Springer, San Diego (2009)
4. Shreiner, D., Woo, M., Neider, J., Davis, T.: *The OpenGL® Programming Guide: The Official Guide to Learning OpenGL®, Version 2, 5th edn.* Addison-Wesley Professional, Reading (2005)
5. Shreiner, D.: *OpenGL® Reference Manual: The Official Reference Document to OpenGL®, Version 1.4, 4th edn.* Addison-Wesley Professional, Reading (2004)
6. Martz, P.: *OpenSceneGraph Quick Start Guide: A Quick Introduction to the Cross-Platform Open Source Scene Graph API.* Skew Matrix software LLC, Louisville (2007)

7. Agil, M., Balbed, M., Ibrahim, N., Yusof, A.M.: Implementation of Virtual Environment using VIRTOOLS. In: 5th Int. Conference on Computer Graphics, Imaging and Visualization, pp. 101–106. IEEE CS Press, Penang (2008)
8. MacLean, K.E., Hayward, V.: Do It yourself Haptics: Part II. IEEE Robotics & Automation Magazine, 104–119 (2008)
9. Dominjon, L., Lécuyer, A., Burkhardt, J.-M., Andrade-Barroso, G., Richir, S.: The Bubble technique: Interacting with Large Virtual Environment Using Haptic Devices with Limited Workspace. In: Proceedings of World Haptics Conference (joint Eurohaptics Conference and Haptics Symposium), pp. 639–640. IEEE CS Press, Pisa (2005)
10. Seth, A., Su, H.-J., Vance, J.M.: SHARP: A system for Haptic Assembly & Realistic Prototyping. In: Proceedings of the DETC 2006, ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Philadelphia (2006)
11. Fischer, A., Vance, J.M.: Phantom Haptic Device Implemented in a Projection Screen Virtual Environment. In: Proceedings of Workshop on Virtual Environments 2003, ACM International Conferences Proceeding Series, Zurich, vol. 39, pp. 225–229 (2003)
12. Itkowitz, B., Handley, J., Zhu, W.: The OpenHapticsTM Toolkit: A Library for Adding 3D TouchTM Navigation and Haptics to Graphics Applications. In: Proceedings of World Haptics Conference (joint Eurohaptics Conference and Haptics Symposium), pp. 590–591. IEEE CS Press, Pisa (2005)
13. Potts, A.: Phantom-based haptic interaction. In: Proceedings of the Computer Sciences discipline Seminary Conference. CSCI 3901, <http://mrs.umn.edu/~lopezdr/seminar/spring2000/potts.pdf>
14. Garrec, P., Friconneau, J.-P., Louvreau, F.: Virtuose 6D: A new Force-Control Master Arm Using Innovative Ball-Screw Actuators. In: 35th International Symposium of Robotics, Paris-Nord-Villepinte (2004)
15. Geiger, C., Klompfner, F., Stoecklein, J., Fritze, R.: Development of an Augmented Reality Game by Extending a 3D Authoring System. In: Proceedings of the International Conference on Advances in Computer entertainment technology, ACM International Conference Proceeding Series, Salzburg, vol. 203, pp. 230–231 (2007)