# Bringing Haptics to Second Life for Visually Impaired People

Maurizio de Pascale, Sara Mulatto, and Domenico Prattichizzo

Siena Robotics and Systems Lab,
University of Siena, Via Roma 56, Siena, Italy
{mdepascale,mulatto,prattichizzo}@dii.unisi.it

**Abstract.** Potential applications of online virtual worlds are attracting the interest of many researchers around the world. One and perhaps the most famous example of such systems is Linden Lab's *Second Life*. Last year, sources for its client application have been released under the GPL license, allowing anyone to extend it building a modified client. This work presents an effort to explore the possibilities that haptic technologies can offer to multiuser online virtual worlds, to provide users with an easier, more interactive and immersive experience. A haptic-enabled version of the Second Life Client, supporting major haptic devices, is proposed. Two haptic-based input modes have been added which help visually impaired people to navigate and explore the simulated 3D environment by exploiting force feedback capabilities of these devices.

**Key words:** Second Life, Haptics, Blind Walk, Blind Vision.

## 1 Introduction

So-called *Virtual Worlds*, computer-simulated realistic 3D worlds where people can experience fully immersive alternative lives, have long been a central theme in science fiction literature. Nowadays, the increasing network bandwidth and computational power, cheaply available in home systems, are making possible and enjoyable such a multi-player online experience, allowing remote users to connect and interact in complex 3D environments. Currently, most of these systems employ very simple graphics and approximated physics simulation, but their technological development is only a matter of time. As far as the number of users is concerned, despite the recent birth of most of these virtual worlds (the oldest one dating back no more than a decade), they already count millions of players worldwide, and this number is rapidly increasing, growing so fast that it is even getting under the lens of mainstream media. A further evidence of the importance of online multiuser experience can be seen in the ongoing battle between last generation gaming consoles, where the challenge is focusing on providing users with community for online multi-player, and developers with software infrastructures for supporting these services.

Among existing virtual worlds, one of the most famous is Second Life [14] from Linden Lab. Second Life (SL) users connect to a network of servers using a viewer which displays their *avatars* and the surrounding simulated world using 3D graphics. Servers maintain the virtual world representation (e.g. objects, buildings, landscapes) and simulate the physical evolution of all entities. Once connected, users can move their avatars around, play animations, move objects or create new ones, and mainly can communicate with other avatars by typing text as in common instant messaging applications. Chat messages are received depending on distance, i.e. like in real life, users can hear chats only if they are close enough. Moreover, the underlying infrastructure has also been recently enhanced to include voice capabilities, allowing people to talk and listen each other voices.

Second Life is certainly the most known virtual world, although not the most populated. Despite claiming around 9 millions of registered users, which has allowed Linden Lab to attract a great attention from many societies among which Microsoft, Nissan, Sony BMG Music and Coca Cola, just to name a few, a recent article appeared on Wired [8] highlighted users online at the same time are usually no more than 50.000. This is far below the hundreds of thousands of users concurrently using massive multi-player online systems like, for instance, Blizzard's World of Warcraft [3]. However, it's worth noting that these more crowded worlds mainly attract users for their gaming nature, while Second Life is basically focused on social communication only. The social aspect of virtual worlds is attracting and will attract a large interest in the near future, thus its potential impact should not be ignored.

Social interaction in online virtual worlds occurs through text messaging, audio and visual modalities, as for almost any other current digital entertainment system. Tactile-based interaction has not yet been explored. The purpose of this paper is to investigate potential applications of haptic feedback in virtual worlds and the benefits their users could achieve. Indeed, this appears to be the right time to evaluate such potentials, given the recent release of low cost haptic devices [21] on the consumer market.

The main contribution of this paper is to propose a modified Second Life Client supporting haptic interfaces and some new input modes exploiting these devices. Unfortunately, the current version of SL has not been designed to support tactile feedback thus limiting the first use of haptics in this virtual world to few possible applications. Among these, we focus on navigation of visually-impaired people and propose haptic-based input modalities to observe and walk around in the virtual environment.

## 2 Related Work

The very first networked virtual world in which there were people represented as avatars and able to communicate and form a virtual community was Habitat [19]. From then, many virtual words have been implemented here listed: the *ActiveWorlds* [1] platform, which claims to host the oldest collaborative virtual world on the Internet; Sony's announced *Home* service, reserved for playstation3 users; The scifi-themed *Entropia Universe* [18]; the Sims Online [7], spinoff of the well known life-simulator game; *There* [17]; *Moove* [15]; the cinese Second Life clone *HiPiHi* [10].

Many more virtual worlds are expected to become operative in the next years, given the increasing availability of ad-hoc tools for development of Massive Multiuser Online (MMO) simulated worlds as for instance the *Croquet system* [5], a set of SmallTalk based technologies for developing collaborative multi-user virtual environments, promoted by the Croquet consortium, an international alliance of industry and academic institutions. Multiverse [20] is a commercial company providing client and server software, as well as tools and assets for building and using virtual worlds, with the peculiarity that the multiverse client can connect to any multiverse based system. Even SUN Microsystem is showing its interest on the subject, funding the open source *DarkStar Project* [24], a java based runtime for building the server side of virtual worlds. *Interreality* [27] is another similar open source project, driven by a small community of volunteers, built around a C++ library providing an infrastructure for sharing a dynamically changing state over a network of distributed hosts, with support for both peer-to-peer and client/server architectures.

The potentials of using haptic devices for blind people have been studied more in-depth since the end of the nineties. Visually disabled persons have now access to text through computer controlled Braille-displays or speech synthesis. However, the introduction of Graphical User Interfaces (GUI), which has greatly simplified the use of computers, achieved the paradoxical effect of making its use more difficult for blind people. Being unable to deal with graphics-only metaphors like desktop and icons, visual impaired people find hard or even impossible to control the computer.

From these considerations, the idea of using haptic devices as a computer interface for blind people was born. At the beginning these devices were used for applications like digital painting with a finger, feel mathematical curves and surfaces, or haptic variants of well-know games [23]. Later on, several studies have been conducted with the aim of improving computer accessibility for visually disabled users. Lahav and Mioduser in [13] created a multi-sensory virtual environment enabling blind people to learn how to explore real life spaces (e.g. public buildings, school or work place). The user interface of their proposed virtual environment consists of a real rooms and objects simulation where users can navigate using a force feedback joystick. Afterwards, further studies about the interaction with virtual objects and navigation in virtual environments have been presented. In [16] test recognition of geometrical and VRML objects, mathematical surfaces and traffic environment exploration are reported. In [25] the authors presented a more interactive and extensible haptic virtual reality system. The proposed method offers several advantages to blind users as compared with previous techniques: providing the ability to use virtual training environments with large workspace, supporting a rather natural users interaction thanks to the choice of using the CyberGrasp haptic device.

As far as texture perception is concerned, in [4, 12] authors investigates the usefulness of haptic devices for blind-folded sight subject to judge the roughness of real and virtual sandpapers.

Similar issues naturally arise about the use of the internet and access to information through the web. Making easier access to this information channel for blind subjects has been the subject for several studies. Hardwick et al. [9] proposed to use haptic devices to perceive the 3D images of internet pages, represented in the VRML way. Ad-hoc haptic interfaces have also been created to improve access for visually impaired people to 3D computer graphic, exploiting the sense of touch. In [2, 11] authors presented the GRAB system, a new haptic device provided with a set of utilities and applications that allow blind persons to explore a 3D world through touch and audio help.

Very few projects have been focused on using haptic devices to navigate in virtual worlds. Studies about walking in digital environments are available, as in [22], where the virtual scene interactively rotates about the user, such that the user is made to continuously walk towards the farthest "wall" of the tracker.

In [26], the authors studied the sense of presence of subjects immersed in a virtual environment during a real walking, a virtual walking (walking-in-place), and a virtual flying too.

## 3 Bringing Haptics to Second Life

Using a new device, with an application not designed for it, is never an easy task. Yet, when the new device is functionally equivalent with a previously supported one, this can be often achieved without modifying the original executable, for example providing emulation drivers. However, in the case of a haptic device there is the need to compute forces, that strongly depends on the current internal state of the application. Therefore, modifying the program itself becomes mandatory. At the beginning of 2007, Linden Lab decided to publicly release the C++ sources for its Second Life Client under the GPL license, making it into an open source project. Source availability has been one of the main factors driving our choice of Second Life as the platform to test applications of haptics within virtual worlds. This effort started by adding support for haptic devices to the SL client. However, to fully exploit capabilities of haptic devices new interfacing paradigms were needed, since simply mapping keyboard, mouse, or joystick input on a force capable device would have been mostly useless.

Therefore, we started by sketching and designing several possible interfacing modalities to use haptic devices in the Second Life world context, the most fascinating one certainly was the idea of controlling the avatar hands using two haptic devices, with arms and body following accordingly using inverse kinematics. Unfortunately, most of planned ideas were too demanding to be implemented given the current status of the Second Life Viewer: despite its original commercial nature, and similarly to most open-source projects, the Second Life Client is practically undocumented. The online wiki contains very scarce information dealing with minor aspects or common patterns for 3D applications, while leaving


**Fig. 1.** A screenshot from our modified Second Life [H] client. The user is currently perceiving through Blind Vision the avatar at the end of the environment. The newly added *Haptics* floater provides quick access to all configurable options and settings as well as visual feedback and debug output.

the most complex parts, such as rendering and objects management, completely uncovered. Additionally, the small community grown around the mailing list, on which also several Linden Lab developers post messages, didn't proved supportive, and all our tech-related questions, from the simplest to non trivial ones, have simply gone unanswered. Finally, source code are almost completely uncommented, exception made for few notes in places where bugs were corrected. Several parts of the code look extremely messy and poorly designed, sometimes with logic spread all along a call path instead of being encapsulated in a single, well defined, place.

As a consequence we started implementing a small subset of these input modes, selecting the ones that could prove potentially useful for visually impaired people. Allowing blind people to use Second Life, in fact, has been for long a known request to Linden Lab thought, so far, related to the development of a voice based user interface.

### 3.1 Second Life Architecture

The Second Life infrastructure follows a standard client-server architecture. A cluster of servers called *sims*, and collectively referred as the *main grid*, is hosted by Linden Lab and runs a closed proprietary software. Each sim runs one or more *simulators*, server processes responsible to simulate a fixed-size piece of the virtual world, independently from the number of avatars within. The grid stores the full state (in particular position, velocity and shape) of all entities in the world since all the simulation is performed on the server side: in particular collision detection and physics update needed to approximate a realistic behavior for objects. Additionally, the grid is in charge of storing in a database all digital content representing the world (i.e. 3D meshes for objects and avatars, textures, animations, audio samples) that will be streamed on-demand to clients.

Each user downloads and runs on its platform (currently Windows, Linux and MacOS X are supported with varying degrees) the Second Life Client application, called simply the *Viewer*, which communicates with the grid using a known protocol. The viewer can be seen as a streaming visualizer of 3D objects and environments, and is responsible for collecting input from the user and displaying data received from the server. It sends camera orientation and controlled avatar movements and actions to the server, which in turn computes collisions and dynamics update, returning new positions and velocities of objects back to the client. In order to minimize the data bandwidth, the client usually interpolates objects' state and, in the absence of server communication, guess positions accordingly, assuming that no collision occurred. For the same reason the server sends to the viewer only data and digital content for objects that are in the view/hear range of the client avatar.

### 3.2 Haptic Based Input Modes

With the objective of helping blind people to be able to interact with the Second Life world we have implemented two new input modes that exploit the force feedback capabilities of haptic devices. We call them *Blind Walk* and *Blind Vision*, and they allow respectively to navigate and explore the virtual environment.

When the Blind Walk mode is active, the haptic device is used as a standard joystick to control walking and flying. However, appropriate force feedback is rendered when collisions with obstacles occur, such as bumping into a wall, another avatar or objects. In this way a visually impaired user can, for example, follow a wall to find a doorway, instead of getting stuck in a corner walking, with no clue of what is happening. For a more comfortable user experience, a small elastic force is used to keep the device tool-tip back in the origin and a dead zone (i.e. in which device position is ignored) allows to avoid unwanted movements.

While the Blind Walk mode simply enhances a standard input interaction with force feedback, the Blind Vision mode provides something completely new. In this modality, the haptic device is used to provide "vision" for exploring the environment: the device is used to control a virtual sonar which feels objects as vibrations. An outgoing ray from the device stylus, applied in the reference frame of the user avatar, is casted to probe the environment. The first colliding entity is perceived as a sinusoidal force feedback signal, applied along the pointing direction, the amplitude of which depends on the actual distance from the user avatar: distant entities will be perceived as weak vibrations, while close objects as strong ones. Different types of entities are identified by different vibration frequencies. In this way other players' avatars can be easily distinguished from unanimated objects. The sky, and especially, the ground are not considered for collision, while other types of entities can be selectively ignored. Both the range of perception and the max amplitude of the feedback can be configured. In this way even a visually impaired user can look around her/his avatar, to find out items or persons with which she/he can interact.

The Blind Vision mode is controlled in two different ways depending on the underlying hardware. Using a haptic device with a stylus, like the PHANToM, the stylus direction controls the sonar. On devices without the stylus, the vector pointing at the tool-tip position is used to determine the ray casting direction.

While the two input modes have been designed separately and can be selectively activated, they can also be used at the same time. On devices with stylus the input modes are trivially combined: the stylus orientation and the tool-tip position vector provide inputs to the Blind Vision and Walk, respectively. For devices without stylus, the tool-tip controls both modes. In this case a larger dead zone is used, to allow looking around without actually moving the avatar. Additionally, for fully actuated 6DoFs, the Blind Vision modality also provides lock on target, using torque feedback on the device stylus to keep it pointing at a moving target.

### 3.3 Hacking the Second Life Client

Following the release of sources for the Second Life Client under the GPL license, some unofficial alternative versions of the Viewer have been produced by enthusiast users. However, all of these are limited to very small changes, usually providing some extensions to the graphical user interface, additional options, performance improvements or bug-fixes. We started from version 1.18.3.5 of the original viewer to build our modified client, called *"Second Life [H]"*. Access to haptic devices has been achieved using the Haptik Library [6], thus ensuring support for a large range of devices with a single code base.

Given the lack of documentation, the knowledge required to "plug in" new code in the existing sources has been gathered by tracing the execution of the client in a debugger, in order to get at least some guesses on the inner structure of the viewer. For example, founding out how to move the client avatar has been discovered by following the call path responding to an arrow key being pressed. Our approach to code modification has been aimed at being the less invasive possible. All the new code has been confined to a single newly added C++ source file (`haptics.cpp`). Just few function calls to this code have been inserted in places within the original sources, using a small header file (`haptics.h`) to provide needed prototypes. This allowed both for keeping things clearly separated as well as for quicker relinking of the executable.

Three functions have been hooked in the viewer main source file (`viewer.cpp`). A pair of calls to initialization and cleanup routines have been put around the application main loop. The initialization routine looks for an available haptic device and establish a connection to it. In case of failure the function returns an error code that causes the application to log the error and gently terminate. Symmetrically, the cleanup routine simply frees all used resources and releases the haptic device. The bulk of the work, is implemented by a third procedure placed in the main loop between objects update and rendering. As a consequence, this function runs at the same rate of graphic rendering. First, this procedure gathers state from device, then it loops over all items in the global object list (the `gObjectList` variable). Depending on their types and their distance from the avatar, some objects are discarded from further processing, for example terrain patches or entities out of vision range. Objects surviving this culling phase get tested for collision detection with both the user avatar itself as well as the ray casted for vision, of course depending on the actual active modes.

Quick collision checks are first performed using sphere–sphere and segment–sphere tests to provide fast discard. Then, collision detection between the avatar and in-world entities is performed using oriented bounding boxes obb–obb tests, while the ray cast collision detection is performed with full segment–triangle tests on all faces of the entity's mesh. Detected interpenetrations with in–contact objects become reaction forces being accumulated. Similarly distance and type of the closest object intersected by the sonar ray are used to drive the vibration component of the force feedback. This is passed back to the high-rate haptic thread in charge of rendering forces at 1000Hz, using a synchronized data structure. Given this multi-rate approach, forces are filtered, in order to mitigate transitions due to the lower-rate at which collision detection is performed.

To provide the user with an easy way of enabling modes as well as configuring parameters, we added to the viewer an in-game dialog box, a so-called *floater* in SL terminology. This has been achieved implementing code with logic for the controls on the floater and providing/modifing resource files which describe the look and placement of GUI elements. The main viewer menu `menu_viewer.xml` has been slightly modified to add the menu item for displaying our added haptics-related floater (see Fig 1). The layout of this floater is described in the `floater_haptics.xml` file.

## 4 Preliminary Results

In order to validate the new haptic based input modes, we have performed some preliminary tests using a small set of 8 temporary blind-folded users, with varying degrees of prior haptic experience. All users were asked to perform some very simple tasks, such as telling whether there is at least another avatar around, looking for and reaching a group of avatars, and counting people in these groups. A PHANToM premium was used as the haptic device for all the experiments. All tests took place within the Second Life world in the main square of the well known Parioli quarter, which is large, flat and always quite crowded by groups of avatars. Tests requiring to find a single avatar were conducted in a remote uncrowded place. Perception of objects was disabled, thus having feedback signals generated only from avatars' presence.

All users found it very easy to detect and reach a group of avatars. After a couple of minutes of training, all subjects spent less than a minute to perform this task, given any starting position and orientation of their avatar. Similar experiences were reported by all the subjects when they were asked to look for a single avatar instead of a group of avatars. This, of course, took more time. While users were able to correctly detect the direction in which avatars where present, estimating the size of these groups proved to be hard task. The users were only able to state that a large group was present in a given direction but cannot find out the actual number of elements. This can be easily explained by the fact that a group of avatars close to each other, are more or less at the same distance from the user's avatar, thus resulting in an almost identical feedback signal, making it impossible to recognize how many actually distinct characters are being perceived. As a final comment, it is remarkable that all users were satisfied with their experience and reported as easy the task of finding and reaching avatars using only the tactile feedback provided by the haptic device.

## 5 Conclusions and Future Work

Second Life and similar virtual worlds are rapidly growing, both in number of users and in graphical and physical realism. It's easy to foresee that they will soon be an important part of our connected society, maybe even competing with current social network portals, and instant messaging communities. In this work we have started exploring the potentials of introducing haptics in virtual worlds. In particular we have enhanced the Second Life client with two input modes explicitly designed for exploiting the force feedback capabilities of haptic devices to allow blind people to start experiencing the Second Life world.

Results of this work are preliminary, since so far we have tested the modified client only with temporary blind-folded users. Notwithstanding that, results are encouraging and show that the haptic-based input modalities allow to easily find and reach avatars in the surrounding environment. We plan to enhance the client and have it tested by permanently blinded people to further validate the effectiveness of this interface to allow visually impaired persons to experience virtual worlds. Future work will also be aimed at implementing enhanced paradigms of haptic interaction, not only restricted to a particular class of users, such as controlling avatar hands using two haptic devices.

## References

1. Activeworlds Inc. Active Worlds. *http://www.activeworlds.com.*
2. C. Avizzano, S. Marcheschi, M. Angerilli, M. Fontana, M. Bergamasco, T. Gutierrez, and M. Mannegeis. A multi-finger haptic interface for visually impaired people. *Robot and Human Interactive Communication, 2003. Proceedings. ROMAN 2003. The 12th IEEE International Workshop on*, pages 165–170.

3. Blizzard. World of Warcraft. *http://www.worldofwarcraft.com.*
4. C. Colwell, H. Petrie, D. Kornbrot, A. Hardwick, and S. Furner. Haptic virtual reality for blind computer users. *Proceedings of the third international ACM conference on Assistive technologies*, pages 92–99, 1998.
5. Croquet Consortium. Croquet. *http://www.opencroquet.org.*
6. M. De Pascale and D. Prattichizzo. The Haptik Library, a Component Architecture for Uniform Access to Haptic Devices. *Robotics and Automation Magazine.*, 14(4):64–75, December 2007.
7. Electronic Arts Inc. Sims Online. *http://www.ea.com/official/thesims/thesimsonline.*
8. Frank Rose. How madison avenue is wasting millions on a deserted second life. *Wired Magazine*, 2007.
9. A. Hardwick, S. Furner, and J. Rush. Tactile display of virtual reality from the World Wide Web–a potential access method for blind people. *Displays*, 18(3):153–161, 1998.
10. HiPiHi. Hipihi World. *http://www.hipihi.com.*
11. R. Iglesias, S. Casado, T. Gutierrez, J. Barbero, C. Avizzano, S. Marcheschi, M. Bergamasco, F. Labein, and S. Derio. Computer graphics access for blind people through a haptic and audio virtual environment. *Haptic, Audio and Visual Environments and Their Applications, 2004. HAVE 2004. Proceedings. The 3rd IEEE International Workshop on*, pages 13–18, 2004.
12. G. Jansson. Can a Haptic Force Feedback Display Provide Visually Impaired people with Useful Information about Texture Roughness and 3D Form of Virtual Objects. *Euro. Conf. Disability, Sweden*, pages 105–111, 1998.
13. O. Lahav and D. Mioduser. Multisensory virtual environment for supporting blind persons acquisition of spatial cognitive mapping, orientation, and mobility skills. *Proceedings of the Third International Conference on Disability, Virtual Reality and Associated Technologies, ICDVRAT 2000*, pages 23–25, 2000.
14. Linden Lab. Second Life. *http://www.secondlife.com.*
15. Lothar Bongartz and Dr. A. Burak Kozan. Moove. *http://www.moove.com.*
16. C. Magnusson, K. Rassmus-Gröhn, C. Sjöström, and H. Danielsson. Navigation and recognition in complex haptic virtual environments–reports from an extensive study with blind users. *Proceedings of Eurohaptics*, 2002, 2002.
17. Makena Technologies, Inc. There. *http://www.there.com.*
18. MindArk. Entropia Universe. *http://www.entropiauniverse.com.*
19. C. Morningstar and F. Farmer. The lessons of Lucasfilm's habitat. *Cyberspace: first steps table of contents*, pages 273–302, 1991.
20. Multiverse Network, Inc. . Multiverse. *http://www.multiverse.net.*
21. Novint Technologies. Falcon Controller. *http://home.novint.com.*
22. S. Razzaque, Z. Kohn, and M. Whitton. Redirected Walking, Eurographics 2001 Proc., 2001. *On-line available: http://www.cs.unc.edu/eve/pubs.html*, 2001.
23. C. Sjöström. The Phantasticon-Haptic Interfaces Give New Possibilities for Blind People. *Master's Thesis, Certec, Lund University, Sweden*, 1997.
24. Sun Microsystems, Inc. Project Darkstar. *http://www.projectdarkstar.com.*
25. D. Tzovaras, G. Nikolakis, G. Fergadis, S. Malasiotis, and M. Stavrakis. Design and implementation of haptic virtual environments for the training of the visually impaired. *IEEE Trans. on Neural Systems and Rehabilitation Engineering*, 12(2):266–278, 2004.
26. M. Usoh, K. Arthur, M. Whitton, R. Bastos, A. Steed, M. Slater, and F. Brooks Jr. Walking, walking-in-place, flying, in virtual environments. *Proc. of the 26th annual conference on Computer graphics and interactive techniques*, pages 359–364, 1999.
27. Virtual Object System. Interreality. *http://interreality.org.*