# HAPTICS AND EXTENSIBLE 3D IN WEB-BASED ENVIRONMENTS FOR E-LEARNING AND SIMULATION

Felix G. Hamza-Lup and Ivan Sopin
*Computer Science, Armstrong Atlantic State University, Savannah, Georgia, U.S.A.*

Keywords: Haptics, H3D, X3D, 3D Graphics.

Abstract: Knowledge creation occurs in the process of social interaction. As our service-based society is evolving into a knowledge-based society, there is an acute need for more effective collaboration and more effective knowledge-sharing systems to be used by geographically scattered people. We present the use of 3D components and standards, such as Web 3D in combination with the haptic paradigm, for e-learning and simulation.

## 1 INTRODUCTION

Web-based knowledge transfer is becoming a field of research which deserves the attention of the research community, regardless of their domain of expertise, especially because of the potential of advanced technologies such as Web 3D and haptics.

In the context of global communication, these technologies are becoming more stimulating through the possibility of creating collaborative spaces for e-learning and simulation.

In this paper we present several advanced features of Web 3D in conjunction with two successful projects employing those features. The paper is structured as follows: In section 2 we provide a brief introduction to the e-learning concept. In section 3 we discuss the details of different modalities to enrich user interaction with web-based 3D and haptics. In section 4 we introduce two case studies demonstrating the potential of X3D in simulation and training: 3DRTT, a radiation therapy medical simulator; and HaptEK16, an e-learning module which provides interaction through haptic feedback for teaching high-school physics concepts. We conclude in section 5 with a set of remarks and conclusions.

## 2 BACKGROUND AND RELATED WORK

Let us take a look at the notion of e-learning. According to Anohina (2005), as illustrated in figure 1, the concept of Internet-based learning is broader than Web-based learning. The Web is only one of the Internet services that uses a unified document language (HTML), unified resource locator (URL), browsers, and is based on the HTTP protocol.

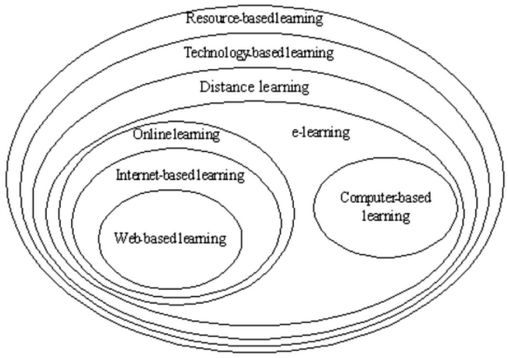


Figure 1: Subsets relationships among the group of terms.

As the largest network in the world, the Internet offers other services besides Web: e-mail, file transfer facilities, etc. Hence, learning could be organized not only on the Web basis, but, for example, as correspondence via e-mail as well. Furthermore, the Internet employs a multitude of proprietary protocols along with HTTP.



G. Hamza-Lup F. and Sopin I. (2008).
HAPTICS AND EXTENSIBLE 3D IN WEB-BASED ENVIRONMENTS FOR E-LEARNING AND SIMULATION.
In *Proceedings of the Fourth International Conference on Web Information Systems and Technologies*, pages 309-315
DOI: 10.5220/0001514303090315

Due to the advances in 3D technology, it is now possible to develop 3D interfaces and environments to enhance the learning process and deploy these interfaces on the Web. An example is Extensible 3D (X3D), an ISO standard for real-time 3D computer graphics and the successor to Virtual Reality Modelling Language (VRML). X3D combines both 3D geometry and runtime behavioral descriptions into a single file, encoded in a particular format such as Extensible Markup Language (XML). It is an initiative to leverage 3D as digital media as easily as text and 2D graphics. It provides a means of associating behaviors and dynamic scripts with 3D objects, so that users can interact with those objects.

On the other hand, advanced interfaces are undergoing a shift towards the incorporation of a new paradigm: haptics. Interfaces combining 3D graphics and haptics have the potential to facilitate our understanding of various concepts and phenomena as well as to promote new methods for teaching and learning.

Haptic technologies offer a new way of creating and manipulating 3D objects. For instance, in Interactive Molecular Dynamics (Stone, Gullingsrud, and Schulten, 2001) the users manipulate molecules with real-time force feedback and a 3D graphical display. Another example, SCIRun (Durbeck et al, 1998), is a problem-solving environment for scientific computation which is used to display flow and vector fields such as fluid flow models for airplane wings.

Initial pilot demonstrations with biology students using augmented graphical models and haptic feedback support the hypothesis that this method provides an intuitive and natural way of understanding difficult concepts and phenomena (Sankaranarayanan et al, 2003). Another research group, at the University of Patras, Greece, is involved in designing simulations to aid children in comprehending ideas concerning several subject areas of science such as Newtonian Laws, Space Phenomena, and Mechanics Assembly (Pentelios, Christodoulou, and Papatheodorou, 2004). Tests show that haptics technology improves the level of human perception due to the deeper immersion provided.

Other fields, such as mathematics and especially geometry, also benefit from haptic interaction. Recently, a system was proposed to allow the haptic 3D representation of a geometric problem's construction and solution (Kaufmann, Schmalstieg, and Wagner, 2000). Initial performance evaluation indicates the system's amplified user-friendliness and higher efficiency compared with the traditional learning approach.

National Aeronautics Space Administration (NASA) has shown interest in the potential use of haptics in educational technology. The Learning Technologies Project at the Langley Research Center is concerned with innovative approaches for supporting K-16 education. Pilot study results from the use of simple haptics-augmented machines have yielded positive feedback with 83% of the elementary school and 97% of the college students, rating the software from "Somewhat Effective" to "Very effective" (Williams, M.-Y.C., and Seaton, 2000, 2003).

# 3 USER INTERACTION

User interaction can be enriched through 3D content and haptics. In what follows we explore in more detail the potential of combining X3D with additional web-enabled instruments, such as HTML and JavaScript, to provide control over the 3D world. We also explain and explore the haptic paradigm and its potential applications in the web-based environment.

## 3.1 X3D Graphics Visualization

On their own, X3D files are simply formatted lines of code. To visualize the graphical content of X3D online, a web browser needs a special plug-in. Most of the X3D plug-in vendors release their software at no cost for public use and a small license fee for commercial use. One example is the BitManagement Contact X3D player.

Usually, X3D plug-ins are equipped with a set of basic controls for customizing the user interface and specifying the properties of user interaction: navigational tools, graphics modes, and rendering settings, just to name a few. While useful, these features only facilitate the user in exploring the visual content, but do not provide any means of altering it. It is the X3D standard itself that allows users to dynamically modify and interact with the 3D graphical scene. There are several alternatives to implement such systems. In the following two subsections we discuss the advantages and drawbacks of a stand-alone X3D based simulation environment in comparison with an environment where the functionality is enriched with JavaScript functions and HTML.

### 3.2 X3D-based GUI

An X3D-based graphical user interface (GUI) implies that the entire functionality is embedded in the X3D and no control is possible outside the X3D content.

If radical changes are made to the application, the code of the file with graphical content has to be altered. This entails the necessity to provide an updated version of the file and also to manually refresh the X3D scene. Such organization largely corresponds to the common client/server interaction on the web. When an HTML form is populated with data, the user has to press the "Submit" button to request the server response.

The three-dimensional scope introduced by X3D brings into play new aspects of GUI/user interaction. For instance, volumetric controls, easily implemented in X3D, can better mimic the behavior of the objects. A multitude of components can also be controlled by simply clicking, dragging, rotating, or actuating them through a system of specifically designated sensors. The scripting capabilities of X3D enrich the GUI interactivity, enabling developers to create efficient control panels meeting the project-specific tasks. The potential to create 3D GUIs and organize information in the third dimension is considerable.

### 3.3 HTML/JavaScript-based GUI

A different approach to improving the GUI interactivity is involving external tools that could effectively communicate with the 3D graphical scene. A good example of such tools is HTML and JavaScript, most commonly used to build web pages.

In HTML/JavaScript-based GUI, JavaScript is the driving force of most of the features while HTML only serves as its operating environment. However, JavaScript makes it difficult to encode unconventional GUI components needed to closer represent the dynamics of the virtual objects in the 3D scene. These difficulties arise because browsers usually do not support such tools that go beyond the facilities of regular HTML widgets. Therefore, creation of powerful and flexible task-oriented GUI components requires the usage of the traditional HTML objects (layers, inputs, etc.) combined with extensive JavaScript code.

With the interface functionality programmed as JavaScript functions, the 3D scene still derives its maneuverability from the methods implemented in the X3D scripting nodes. The browser and X3D environments communicate through mutual function calls. The browser refers to the virtual scene as to an HTML document's object with a number of public functions. Different X3D plug-in manufacturers provide their own sets of such functions. X3D feedback is composed of dynamic injections of JavaScript code. Because both, visual content and HTML/JavaScript GUI, are synchronized automatically, no manual page updates are necessary. However, synchronization between HTML and X3D can be an issue in such implementation. The synchronization involves continuous calls between the two media and may consume considerable CPU power.

New client/server communication techniques are also feasible for the development of various dynamic environments. For instance, asynchronous JavaScript (AJAX) is used in our HTML/JavaScript-based GUI (details in section 4.1) to obtain the listing of external X3D components loaded in the scene. Therefore, without "page refresh", the user is able to visualize external 3D objects and manipulate them as if they were initially in the 3D scene.

### 3.4 Multimodal, Haptic-based GUI

Besides traditional GUIs, a novel paradigm, *haptic* feedback (from the Greek haptesthai, meaning "contact" or "touch"), may improve the interface usability and interactivity. The tactile sense is the most sophisticated of all our senses as it incorporates pressure, heat, texture, hardness, weight, and the form of objects.

Lederman, Klatzky, and Metzger (1985) summarized four basic procedures for haptic exploration, each one eliciting a different set of object characteristics:

- *Lateral motion* (stroking) provides information about the surface texture of the object.
- *Pressure* gives information about how firm the material is.
- *Contour following* elicits information on the form of the object.
- *Enclosure* reflects the volume of the object.

Recent haptic technologies are capable of delivering realistic sensory stimuli at a reasonable cost, opening new opportunities for academic research and commercial developments (Stone, 2000). Such devices have a distinct set of performance measures (Wall, 2004):

- *Degrees of freedom* (DOF) – the set of independent displacements that specify completely the position of a body or a system.

- *Workspace* – the volume within which the joints of the device will permit operation.
- *Position resolution* – the minimum detectable change in position possible within the workspace.
- *Maximum force/torque* – the maximum possible output of the device, determined by such factors as the power of the actuators and efficiency of any gearing systems.
- *Maximum stiffness* – the maximum stiffness of virtual surfaces presented on the device. It depends on the maximum force/torque, but is also related to the dynamic behaviour of the device, sensor resolution, and the sampling period of the controlling processor.

The addition of haptic feedback enables the users to feel the virtual objects they manipulate. We have experimented with the PHANTOM® Omni™ device, developed by Sensable Technologies (illustrated in figure 2). The Omni™ became one of our choices due to its low cost and force-feedback qualities. It is also backed by an open source Application Programming Interface (API).

Figure 2: Phantom® Omni™ Device.

## 4 CASE STUDIES

To illustrate the concepts discussed so far, we will describe two successful projects developed at our laboratory (www.cs.armstrong.edu/felix/news) using X3D and haptic technologies: a web-based medical simulator (3DRTT) and a haptic-based module for teaching physics (HapteK16).

### 4.1 Medical Simulators – 3DRTT

Visual simulation in medicine plays a very important role as the success of an operation relies upon practical procedures and the physician's (or surgeon's) experience. Many complex treatment processes are preplanned well in advance of the operation. This is especially the case with radiation therapy. Medical personnel concerned with the planning part (e.g. correct radiation dosages, appropriate patient setup) are sometimes frustrated by the fact that a theoretically sound plan proves inconsistent with the current hardware and patient constraints (e.g. collisions with the patient and treatment hardware may occur).

3D Radiation Therapy Training (3DRTT) is a web-based 3D graphical simulator for radiation therapy. It simulates linear accelerators (linacs) used to deliver radiation doses to an internal tumor, therefore destroying cancerous cells. The project focuses on improving the efficiency and reliability of the radiation treatment planning and delivery process by providing accurate visualization of the linacs hardware components as well as careful imaging of their interactive motion.

The virtual representation of the treatment settings (figure 3) provides patients and therapists with a clear understanding of the procedure. Equipped with patient CT data, the treatment planner can simulate a series of patient-specific setups and, also detect unforeseen collision scenarios for complex beam arrangements. Hence the necessary adjustments in the treatment plan can be made beforehand and validated.

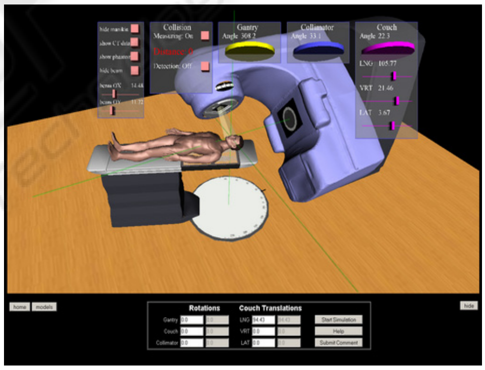

Figure 3: 3DRTT Simulator with X3D GUI.

Another important application of 3DRTT is improving the current level of radiation therapy education and training. With such web-based 3D simulation tools at the disposal of radiation therapy staff, there is plenty of room for exploring various treatment procedures (linac components, motion limitations, associated accessories, etc.) and gaining experience for future operations.

Currently, two versions of the simulator, with X3D and HTML/JavaScript-based GUIs (refer to sections 3.2 and 3.3), are available on the project's website (http://www.3drtt.org). The X3D-based version provides tools for controlling the angles and locations of the machine's parts (figure 3). The GUI is composed of several semitransparent pannels containing various volumetric controls. The controls are designed to logically correspond to the assigned operations (specifically, scrolls for rotations, slides for translations, and buttons for switching between different simulation modes) and therefore improve the overall intuitiveness (Hamza-Lup et al, 2007). The user can easily rearrange the GUI components to avoid occlusions of important parts of the scene.

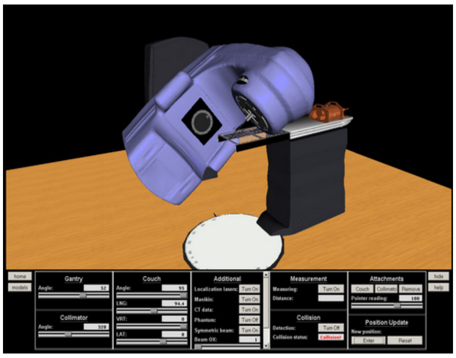

Figure 4: 3DRTT with HTML/JavaScript-based GUI.

The HTML/JavaScript-based GUI alternative supports the same functionality and also provides new and highly useful features (figure 4) as follows. Instead of floating X3D menus, the simulator controls are shifted to the HTML document scope. The set of GUI elements includes sliders, buttons, and displays that support the learnability of the interface. For the convenience of navigation in the virtual space, the control panel can be hidden and brought back per user request, improving in this way the navigation throughout the simulator. HTML introduces new methods of accessing and dynamically processing external modules. For instance, the user may load various hardware attachments for the linacs directly in the virtual world. The source X3D file for the simulator does not "know" how many attachments are available at the moment, what their names are, etc. However, at the user's request, an AJAX function makes a call to a Java Servlet Pages code stored on the server and receives the listing of available files as the response. This listing is transmitted to an X3D script that handles the loading and embedding of specified files into the virtual environment. Therefore, no alterations of the X3D source code are necessary when new attachments are uploaded to the server because they become immediately accessible.

## 4.2 E-Learning Module – HaptEK16

HaptEK16 is designed to assist students in understanding Pascal's principle and other difficult concepts of hydraulics. The simulator includes three simulation modules: pressure measurement, hydraulic machine, and hydraulic lifting simulation. Students can interact with the 3D scene using a haptic device, as illustrated in figure 5. The functionality of the simulator is implemented through Python, X3D, and the Sense Graphics' H3D API (http://sensegraphics.com), discussed further.

Figure 5: Students using the HaptEK16 hydraulics module.

Python is an object-oriented programming language that offers strong support for integration with other toolkits and APIs. It is a rapidly growing open source programming language. According to InfoWorld (McAllister, 2004), Python's user base nearly doubled in 2004 and currently includes about 14% of all programmers. Python is available for most operating systems, including Windows, UNIX, Linux, and Mac. Some of the Python's strengths which were considered when selecting the language to implement the system functionality include

- *Low complexity*: wxPython (an auxiliary library for GUI) was selected because of its ease of use and reduced complexity compared with Java/Swing;
- *Prototyping*: Prototyping in Python is quick and simple and often leads to a quick prototype that can be adapted for the development of the final system;
- *Maintainability*: The code in Python is easy to modify and/or redesign. Less time is spent understanding and rewriting code which leads to an efficient integration of new features.

H3D is an opened X3D-based haptic API. It is written entirely in C++ and uses OpenGL for graphics rendering and OpenHaptics (de-facto industry standard haptic library) for haptic rendering. With its haptic extensions to X3D, H3D API is an excellent tool for writing hapto-visual applications that combine the sense of touch and 3D graphics. The main advantage of H3D to

OpenHaptics users is that, being a unified scene graph API, it makes management of both graphics and haptics rendering easy.

The scene graph concept facilitates application development, but it can still be time consuming. For this reason, SenseGraphics extended their API with scripting capabilities in order to empower the user with the ability of rapid prototyping. The design approach used in HaptEK16 was the one recommended by SenseGraphics: i.e. geometry and scene-graph structure for a particular application were defined using X3D, and application and user interface behaviors were described using Python and wxPython.

Programming the sense of touch for a virtual object involves two steps. First, the programmer must specify the haptic device to use; second, a set of haptic properties must be defined for each "touchable" object. To specify the haptic device, an instance of the *DeviceInfo* node is created, and the haptic device is added to it. *HLHapticsDevice* is the node used to manipulate a Phantom device. The graphical representation of the device (in case of HaptEK16 a sphere) is also specified in the *containerField* group, as illustrated in figure 6.

```
<DeviceInfo>
  <HLHapticsDevice
    positionCalibration="1e-3 0 0 -.15
    0  2e-3 0 .05 0 0 1e-3 0 0 0 0 1">
    <Group containerField="stylus">
      <Shape>
        <Appearance>
          <Material />
        </Appearance>
        <Sphere radius="0.0025" />
      </Shape>
      <Transform translation="0 0 0.08"
        rotation="1 0 0 1.570796">
        <Shape>
          <Appearance>
            <Material />
          </Appearance>
          <Cylinder radius="0.005"
            height="0.1" />
        </Shape>
      </Transform>
    </Group>
  </HLHapticsDevice>
</DeviceInfo>
```

Figure 6: Specifying the haptic device.

To implement the tactile sensation for a generic shape, one must add a surface node with haptic properties to the shape's *Appearance* node. In HaptEK16 this is accomplished with a fictional surface node added to the cylinder's *Appearance* node. The *DynamicTransform* node is added to define properties for rigid body motion, as illustrated in figure 7.

```
<DynamicTransform DEF="DYN1"
   mass=".05" inertiaTensor=".1 0 0 .1 0 0 0 .1">
   <Shape>
     <Appearance>
       <Material
         diffuseColor="0 .8 .8" />
       <FrictionalSurface
         dynamicFriction=".6"
         staticFriction=".2" />
     </Appearance>
     <Cylinder DEF="LEFTCYL"
       height=".085" radius=".045" />
   </Shape>
 </DynamicTransform>
```

Figure 7: Implementing haptic properties.

The X3D file format is used by H3D as an easy way to define geometry and arrange scene-graph elements such as user interfaces. A screenshot of the HaptEK16 e-learning module is illustrated in figure8.

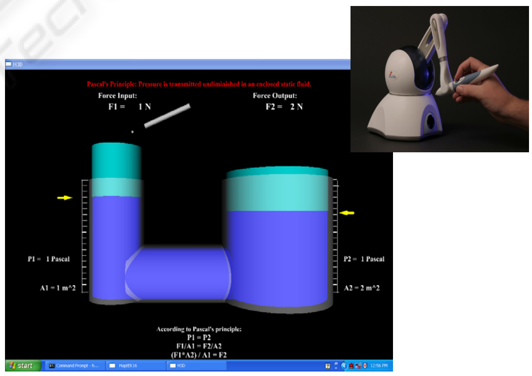

Figure 8: HaptEK16 screenshot and corresponding Phantom® Omni™ device from Sensable Technologies.

A set of test questionnaires were designed and implemented for the assessment of the e-learning module. The results from the assessment tests proved that the students who had the opportunity to use HaptEK16 scored better (13% higher total scores) than the group that did not. Such results prove the potential of using X3D and haptics to develop novel simulation and training environments.

# 5 CONCLUSIONS

3DRTT serves as an example of a web-based system extensively taking advantage of X3D to improve the efficiency of the user-interface interaction as well as to provide powerful means of professional education and training. Naturally, complex concepts and settings are better understood when provided with visual support, especially in complex scenarios. Easy access, simple control, and advanced capabilities of visualizing in 3D and online radiation therapy treatment scenarios proved to be of great value to the radiation therapists using our system. Currently, 3DRTT has over sixty registered users and keeps attracting the attention of other professionals working in the radiation therapy field.

Another development, the haptic e-learning module (HapteK16) facilitates student understanding of difficult concepts (e.g. in physics) and has the potential to augment or replace traditional laboratory instruction with an interactive interface offering enhanced motivation, retention, and intellectual stimulation. HaptEK16's haptics-augmented activities allow students to interact and feel the effects of their choices. We believe the force feedback will lead to more effective learning and that the HaptEK16 project has significant educational potential.

Considering the advances in software and hardware technology, we will see many applications of haptics and 3D graphics in near future web-based information systems and applications.

# REFERENCES


Anohina, A. (2005) Analysis of the terminology used in the field of virtual learning. Journal of Educational Technology & Society, 8(3) July, pp.91-102.

Durbeck, L.; Macias, N.J.; Weinstein, D.M.; Johnson, C.R.; Hollerbach, J.M. (1998) SCIRun haptic display for scientific visualization. In: Salisbury, K.J. and Srinivasan, M.A. ed. Proceedings of the Third Phantom User's Group Workshop, October 3-6, Cambridge USA. Massachusetts Institute of Technology.

Hamza-Lup, F.; Sopin, I.; Lipsa, D.; Zeidan, O. (2007) X3D in radiation therapy procedure planning. In: Proceedings of the International Conference on Web Information Systems and Technologies, March 3-6, 2007, Barcelona, Spain. pp.359-64.

Kaufmann, H.; Schmalstieg, D.; Wagner, M. (2000) Construct3D: a virtual reality application for mathematics and geometry education. Education and Information Technologies, 5(4) December, pp.263-76.

Klatzky, R. L.; Lederman, S.J.; Metzger, V. A. (1985) Identifying objects by touch: An "expert system." Perception and Psychophysics, 37(4), pp.299-302.

McAllister, N. (2004) What do developers want? [Internet] Available from: <http://www.infoworld.com/article/04/09/24/39FErrdev_1.html> [Accessed 24 September 2007].

Pantelios, M.; Tsiknas, L.; Christodoulou, S.; Papatheodorou, T. (2004) Haptics technology in Educational Applications, a Case Study. Journal of Digital Information Management, 2(4), pp.171-179.

Sankaranarayanan, G.; Weghorst, S.; Sanner, M.; Gillet, A.; Olson, A. (2003) Role of haptics in teaching structural molecular biology. In: Proceedings of the 11th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS '03), March 22-23, Los Angeles, CA, pp.363-366.

Stone, J.E.; Gullingsrud, J.; Schulten, K. (2001) A system for interactive molecular dynamics. In: Hughes, J.F., and Sequin, C.H. ed. Proceedings of Symposium on Interactive 3D Graphics, March 19-21, Research Triangle Park USA. ACM SIGGRAPH, pp.191-194.

Stone, R. J. (2000) Haptic feedback: a brief history from telepresence to virtual reality. In: Proceedings of the First International Workshop on Haptic Human-Computer Interaction, August 31 – September 1, Glasgow Scotland. Springer-Verlag, pp.1-16.

Williams II, R.L.; Chen, M.-Y.; Seaton, J.M. (2001) Haptics-augmented high school physics tutorials. International Journal of Virtual Reality, 5(1).

Williams II, R.L.; Chen, M.-Y.; Seaton, J.M. (2003) Haptics-augmented simple machines educational tools. Journal of Science Education and Technology, 12(1), pp.16-27.

Wall, S. (2004) An investigation of temporal and spatial limitations of haptic interfaces. Ph.D. Thesis, Department of Cybernetics, University of Reading.