Information-Rich Virtual Environments: Theory, Tools, and Research Agenda

Doug A. Bowman, Chris North, Jian Chen, Nicholas F. Polys, Pardha S. Pyla, and Umur Yilmaz
Department of Computer Science and Center for Human-Computer Interaction
Virginia Tech
Blacksburg, Virginia, USA
1-540-231-2058
{bowman, north, jichen8, npolys, ppyla, uyilmaz}@vt.edu

ABSTRACT

Virtual environments (VEs) allow users to experience and interact with a rich sensory environment, but most virtual worlds contain only sensory information similar to that which we experience in the physical world. *Information-rich* virtual environments (IRVEs) combine the power of VEs and information visualization, augmenting VEs with additional abstract information such as text, numbers, or graphs. IRVEs can be useful in many contexts, such as education, medicine, or construction. In our work, we are developing a theoretical foundation for the study of IRVEs and tools for their development and evaluation. We present a working definition of IRVEs, a discussion of information display and interaction in IRVEs. We also describe a software framework for IRVE development and a testbed enabling evaluation of text display techniques for IRVEs. Finally, we present a research agenda for this area.

Categories and Subject Descriptors

I.3.7 [**Computer Graphics**]: Three-Dimensional Graphics and Realism - *Virtual reality*.

H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems - *Artificial, augmented, and virtual realities*.

General Terms

Human Factors.

Keywords

Information-rich virtual environments, information visualization.

1. INTRODUCTION

Virtual environments (VEs) can be quite effective in immersing the user within a realistic three-dimensional world, and in providing a high level of sensory “richness” to the user. Thus, an art student can find herself transported back to ancient Greece, where she can experience for herself what it would be like to walk through one of the famous temples. In most VEs, however, this sensory experience would be the extent of the information presented to the user – the VE would contain no descriptive text or audio, data about the materials used in the temple’s construction, or historical context for the building.

In our work, we are developing and studying *information-rich* virtual environments (IRVEs). In a nutshell, IRVEs are a combination of traditional VEs and information visualization; that is, they provide a realistic sensory experience that is enhanced with some representation of related abstract information (a more precise definition is given in section 3.1). In this way, they can provide for a better understanding of the relationships between perceptual and abstract information, improved learning of educational material, greater enjoyment and engagement with the VE, and a decreased dependence on other tools for the viewing of abstract data.

The basic concept of IRVEs is not new; many VE and visualization applications have included related abstract information. However, there has been a lack of precise definition, systematic research, and development tools for IRVEs. Our research generalizes from prior work to define the area and produces principles and tools for the development of effective and usable IRVEs.

Some of our prior research has established that IRVEs can be useful in domains such as design education [13]. In this paper, we describe the steps we are taking to establish a more systematic study of IRVEs. After discussing some related research, we present a theoretical framework for IRVE research, two tools we have developed to allow the implementation and evaluation of IRVEs, and a research agenda for this relatively unexplored field.

2. RELATED WORK

Many systems have been developed that use a three–dimensional (3D) space to present some form of information to the user. These include both immersive virtual reality systems and desktop 3D applications. Let us consider two categories of such systems: information and scientific visualizations.

3D information visualizations take a complex and abstract dataset and organize it into an understandable visual representation, which can be navigated and accessed by the user (e.g. [6, 17, 28, 29]). Here abstract properties of the data are mapped into perceptual qualities, such as position, orientation, size, shape, color, or motion, and relationships between pieces of data are represented spatially. The resulting multi-dimensional visualization can reveal patterns in the data that may not be obvious from the original dataset.

Information visualizations present abstract information using a perceptual (usually visual) form. In contrast, IRVEs embed abstract information within a realistic 3D environment. In this way, abstract and perceptual information are integrated in a single environment [8].

Scientific visualizations present abstract visual representations of scientific data within a 3D environment. They may consist of objects that are too small for the naked eye, such as atoms [7], too large to be comprehended, such as the solar system [31], or invisible, such as electromagnetic fields [16] or fluid flow lines [14]. Users can examine these environments from various positions, detect patterns that would not be obvious without the visualization, and make changes to conditions and immediately visualize the results.

As we will discuss in the next section, many scientific visualizations can be considered IRVEs. These visualizations display realistic objects, but modify perceptual information to represent abstract information.

Our previous work in the area of IRVEs involved two proof-of-concept applications: the Virtual Venue [11], and the Virtual Habitat [13]. In the Virtual Venue, users could move about a realistic model of an aquatic center, and obtain various types of information regarding the design and use of the venue and the sports of swimming and diving. A usability study revealed that the most effective types of information were those that were "tightly coupled" to the environment. The Virtual Habitat taught students about environmental design issues by immersing them in an information-rich zoo exhibit model. This work showed the promise of IRVEs for education.

These projects demonstrated that IRVEs could be effective presentations of combined perceptual and abstract information. In the current research, we have developed a theoretical basis for further systematic research in this area, and tools to enable such research to be performed.

Finally, we note that IRVEs share a great deal in common with augmented reality (AR) (e.g. [20]). AR applications enhance the physical world with additional information, much of it abstract, while IRVEs enhance the virtual world with abstract information. Prior work in AR has included research on information display and layout, and user interaction, similar to the research we propose in section 5. The key difference between IRVEs and AR, however, is that IRVEs are purely synthetic, which gives them much more flexibility. Information objects in an IRVE can be perfectly registered with the world, and realistic objects in an IRVE can be animated, moved, scaled, and manipulated by the user at a distance, for example. Thus, while prior work in AR provides a good starting point (and we have used some of this work in the tools described below), the design of IRVEs should also be studied separately.

3. THEORETICAL FOUNDATIONS

While prior work in IRVEs has focused on the development and evaluation of proof-of-concept applications, we need a more precise theoretical framework if we are to approach IRVE research more systematically, and if we desire generalizable results. This section presents four aspects of such a framework.

3.1 Definition of IRVEs

The most crucial step towards a more complete understanding of IRVEs is a precise definition of the term. Previously, we have written that IRVEs "…consist not only of three-dimensional graphics and other spatial data, but also include information of an abstract or symbolic nature that is related to the space," and that IRVEs "embed symbolic information within a realistic 3D environment" [13]. These statements convey the sense of what we mean by IRVE, but they leave significant room for interpretation. What is meant by "spatial," "abstract," and "symbolic" information? What makes a VE "realistic?" The definitions given below serve to disambiguate these terms.

We begin with a set of definitions of terms that will then be used to define an IRVE:

1. A *virtual environment* (VE) is a synthetic, spatial (usually 3-dimensional) world seen from a first-person point of view. The view in a VE is under the real-time control of the user.
2. *Abstract information* is information that is not normally directly perceptible in the physical world. For example, information about the visual appearance or surface texture of a table is directly perceptible, while information about its date and place of manufacture is not (this information is thus abstract). Taken together, the abstract information can form abstract structures distinct from the sensory or spatial structure of the VE. Shneiderman [30] defines a taxonomy of such abstract structures including temporal, 1D, 2D, 3D, multi-dimensional, tree, and network. Information visualization techniques [15] provide methods for the display of such structures.
3. A VE is said to be *realistic* if its perceptible components represent components that would normally be perceptible in the physical world. If a VE's components represent abstract information (see #2) then the VE is not realistic, but abstract. For example, a virtual Greek temple (existed in the past), Statue of Liberty (exists in the present), DNA molecule (exists at an unfamiliar scale), or city of Atlantis (exists in fantasy) could all be considered realistic. On the other hand, a VE displaying spheres at various points in 3D space to represent three parameters of the items in a library's collection would be abstract.

These three terms allow us to define IRVEs:

4. An *information-rich virtual environment* (IRVE) is a realistic VE that is enhanced with the addition of <u>related</u> abstract information.

We also further define the space of IRVEs:

5. IRVEs exist along a continuum that measures the *fidelity of the perceptual information mapping*. In other words, how faithfully does the IRVE represent the perceptual information from the physical world in the virtual world? In some cases, perceptual information will be <u>changed</u> to show some abstract information about a location or object. In other cases, new information/objects will be <u>added</u> to the environment without changing the perceptual information of the original environment. The two extremes of this continuum are "Pure scientific visualization," which changes perceptual information (e.g. the color of a wall) to represent some abstract information (e.g. the air pressure at each point on the wall), and "Information-enhanced VEs," which represent the physical environment with as much perceptual

fidelity as possible, and add additional abstract information in the form of text, audio, video, graphs, etc.

6. Other dimensions in the space of IRVEs include the *variety of abstract information types* present in the environment, and the *density of abstract information* in the environment. Density will be very hard to define quantitatively, but it could still be useful as a qualitative measure.
7. “Pure information visualization” (e.g. a 3D scatterplot of census data) is not an IRVE because the VE is abstract, not realistic. All of the information in the environment is abstract information that has been mapped to a perceptual form. IRVEs, on the other hand, add information visualization to realistic VEs to provide richness.

To make this definition more concrete, consider the example of VEs for design and review in building construction. Architects need to design and review complex plans for the construction of a home. Good plans must take into account the spatial layout of the home as well as other information such as cost, materials, and schedule. The spatial and abstract information are tightly interrelated, and must be considered and understood together by the architect or contractor. Users of such an application need a) immersion in the space for perceptual fidelity, b) access to related abstract information, and c) an understanding of the relationships between the perceptual and abstract information. Typically, users want to examine abstract information within context in the space. In some cases, they may also use abstract information as an index into space. For example, from a display of the production schedule, elements to be completed by a certain date are selected by the architect. The VE responds by highlighting those elements in the 3D architectural plan, or temporarily filtering other elements from the plan.

An IRVE addressing home design review would allow users to perform tasks such as “navigate to the northwest corner of the master bedroom,” or “attach a note for the client saying that this wall could be thinned to reduce cost.” It would also provide answers to questions such as: “How can we reduce the cost of this room? Which items are most costly? Are these columns aesthetic or essential for load bearing support?”

3.2 Design Space for Information Display in IRVEs

Research in VEs and information visualization provides methods for the display of perceptual and abstract information respectively. However, further new methods are needed for the display of the combination of these information types in IRVEs. We have identified three major decisions designers must make when embedding abstract information in a perceptual VE. Figures 1 and 2 illustrate some of the possible options.

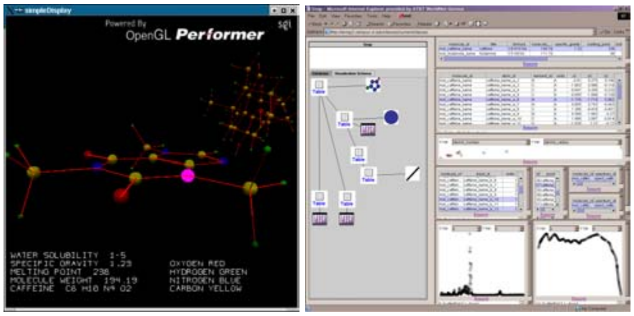
3.2.1 Display Location

Where should abstract information be displayed? Following from the AR work of Feiner and colleagues [18], we divide display locations into *world-fixed*, *display-fixed*, *object-fixed*, and *user-fixed* categories. World-fixed information is attached to a specific 3D (world-coordinate) location in the VE. Display-fixed information remains at the same location on the display (screen) surface. Information that stays attached to an object in the environment (even if that object is moved) is termed object-fixed. Finally, information may be fixed to the user’s view, so that it’s always available as the user navigates the environment. In some display systems, such as head-mounted displays, display-fixed and user-fixed may be equivalent.

When abstract information is world- or object-fixed, we can choose to represent it in at least two ways. We can embed the information by adding a sensory representation (usually a visible object) of it to the VE. The new objects can represent the abstract information as text, glyphs, graphs, and other visualizations, or through audio clips or haptic feedback. For example, each wall in an architectural VE might have a label or small glyph hovering in front of it representing the wall’s cost. Alternatively, we can embed the abstract information by changing the appearance of the associated existing objects in the VE. For example, the color of the walls in the architectural VE could represent their cost.

Figure 1 illustrates world-fixed (the audio cube on the left) and object-fixed (the sign on the tree) information. The abstract information in figure 2 is display-fixed (fixed to the bottom of the viewing window, or located in a separate window).

Figure 1. Abstract information in the Virtual Habitat



```
WATER SOLUBILITY  1-5
SPECIFIC GRAVITY  1.23        OXYGEN RED
MELTING POINT  238            HYDROGEN GREEN
MOLECULE WEIGHT  194.19       NITROGEN BLUE
CAFFEINE  C8 H10 N4 O2        CARBON YELLOW
```

Figure 2. Coordinated VE and information visualization

3.2.2 Association

A second information display distinction for IRVEs relates to the display of interrelationships between abstract and perceptual information. The display of these “links” may be explicit or implicit. The audio cube and the sign in figure 1 demonstrate *spatially explicit* association – the abstract information is located at or near the relevant part of the environment. In figure 2, the associations are *visually implicit*. There is no apparent link between the two windows, but when a selection is made in either window, the associated abstract or perceptual information becomes highlighted in the other window. Brushing-and-linking strategies [1] fall into this category. Finally, associations may be *visually explicit*. For example, we could display a graph and draw a line or arrow between the graph and the associated perceptual information.

3.2.3 Level of Aggregation

Third, designers must choose how much abstract information to combine in a single visualization. Individual pieces of abstract information may each be displayed separately (as in figure 1), or abstract information may be aggregated and displayed using more complex visualizations (as in the right side of figure 2). A separated approach allows users immediate access to details, and can easily be combined with world- or object-fixed display location so that each piece of information can be associated with a place in the environment. An aggregated approach gives designers flexibility in choosing any type of information visualization, and is more likely to use a display- or user-fixed location. For example, the production schedule information of an architectural plan might be displayed in a Gantt chart format.

3.3 Design Space for Interaction in IRVEs

Again, task and interaction strategies in VEs and information visualizations provide a starting point [9, 23], but further insight is gained from considering their combination. In general, two high-level goals emerge:

- VE to Abstract Information: In this case, the user wishes to use the VE as an index into the abstract information. An example task is details-on-demand, in which the user desires to retrieve abstract information related to a given element in the VE space.
- Abstract Information to VE: In this case, the user wishes to proceed from abstract to VE. This can enable the user to control the VE through abstract information. For example, by selecting data in a separate abstract display, users could highlight desired objects in the VE, filter uninteresting objects, or automatically travel to a related location.

These high-level goals translate into interaction tasks. For example, in a details-on-demand task, a user must first recognize that an object in the VE has more associated information (perhaps via a glyph). They then select the object (e.g. using ray-casting) and indicate an action to display additional information (e.g. using a menu). Finally they view the additional abstract information (e.g. on a pop-up HUD), and perhaps act to dismiss the additional information display.

We can classify these interaction tasks into multiple levels. At the lowest level are the standard VE interaction tasks of navigation, object selection and manipulation, system control, and symbolic input. Techniques for these basic tasks enable IRVE-specific tasks. For example, users must be able to access abstract information and choose the way it is displayed, either through embedded objects or separate displays. Finally, at the highest level are tasks allowing users to interact with displayed abstract information, including searching embedded information and linking with separate displays.

4. TOOLS FOR IRVE RESEARCH

Although IRVEs can embed any type of abstract information, perhaps the most basic function of IRVEs is the inclusion of related text (or numeric) information into a VE. We have focused much of our initial work on this problem.

We wish to perform research to answer questions about how to design effective and usable IRVEs. Besides the theoretical framework, we also need practical tools to carry out this research. This section describes a software development framework allowing the rapid development of IRVEs and a testbed that can be used to evaluate various aspects of text display and travel in IRVEs.

4.1 Software Development Framework

There are a number of challenges when considering the enhancement of VEs with additional information types. Consider for example the display of the abstract properties of an object. These attributes may be rendered as textual descriptions or textures, which, in turn, may be photos, graphs, videos or other related information. In either case, the display must ensure the perception and association of those properties to the referent, the legibility of those properties, and the minimal occlusion of other objects in the scene. Our goal is to provide a portable system enabling the development of IRVE applications with high levels of usability and task performance for exploration or search tasks in IRVEs.

Since an object's attributes may themselves have a range of overviews and details, the notion of levels-of-detail is also essential in presenting attribute information effectively. In order to provide an efficient means to explore the design space of IRVEs and tackle these issues of information display, we developed a set of programmable objects that encapsulate flexible behaviors as reusable scenegraph nodes. The system is portable in that it is standards-based using Virtual Reality Modeling Language (VRML) and Extensible 3D (X3D) [35], and applicable to the "camera" projection paradigm as employed in both desktop and head-mounted displays (HMDs).

While a number of researchers have investigated the use of image plane representations for the management and manipulation of 3D objects [4, 5, 25], we focus on the behaviors of objects within the world space. This has the advantage that behaviors can be independent of screen resolution and also impose less computational overhead to track and process an object's projection. Within the world space, we support the display of information both in proximity to the referent object, and on a HUD that travels with the user's viewpoint and remains a constant size.

4.1.1 Semantic Objects

Bederson, et al [2] have proposed that interface designers appeal to user's knowledge about the real world, i.e. that objects appear and behave differently depending on the scale of the view and the context. They propose a new "interface physics" called "semantic zooming" where both the content of the representation and the manipulation affordances it provides are directly and naturally available to the user. More recently, Bederson et al [3] used a scenegraph to represent a 2D visualization space as a "zoomable" surface. In 3D, however, there are usually more degrees of freedom and fewer constraints on user motion relative to the space and objects of interest. In many cases, detail attributes may not be legible either because of size, color, or occlusion by the referent or neighbors.

We drew inspiration from this prior work by focusing on the definition of *semantic objects*, which are displayed differently depending on the distance from which they are perceived. Consider the case where some heterogeneous, abstract information is embedded and associated with some objects or an area in the virtual space. From a certain distance, only the title or name of an object may be visible. As a user navigates closer or zooms into smaller scales, he or she may receive more detail about that object such as a full text or numerical description. In order to account for differences in the number of geometrical and abstract attribute definitions for a semantic object, we implemented a Level-of-Detail (LOD) technique that separates the functional descriptions of the 3D object and its abstract attributes.

Semantic objects may also carry affordances similar to Bederson et al.'s notion of an information lens and Shneiderman's mantra of details-on-demand [30]. As a user explores the IRVE, she may point to an object and receive an overview description of the object in her HUD. When the user selects an object with a button press, detail information is displayed in the HUD (figure 3).

4.1.2 Level of Detail Techniques

Our software design attempts to improve techniques and technologies for managing the rendered form and context of information objects over wide ranges of scale. There are existing specifications and implementations for level-of-detail (LOD) scenegraph nodes, but they have typically only been applied to an object's geometric resolution – not to multiple information types such as annotations or numbers associated with that object.

The VRML97 LOD was designed to address network, memory, and performance tradeoffs in managing a VE's complexity. Related work on managing LOD in wide-scale geographic models has been done [19, 26], but we want to improve these approaches in the context of IRVEs. In addition, few of the current approaches balance the detail and scale with respect to the user's scale and viewing frustum, especially as concerns the layout and legibility of text or textures. The semantic object approach integrates and manages geometry details and attribute details simultaneously.

4.1.3 Implementation

Our implementation uses VRML and X3D PROTOTYPE nodes to define labels and semantic objects that can be easily populated from a database or XML (e.g. using XSL Transformations (XSLT)). Authors simply declare the EXTERNPROTO's file location and interface, and semantic objects may be instantiated in any scene. Semantic objects can be animated with set methods for position and orientation.

The label object, or Text Panel, is a basic unit in our system that can support arbitrary lines of text aligned left, center, or right, normal, bolded, or italic. The text can be further customized by color, transparency, or serif or sans serif font. Text Panels may or may not have a panel background. If they do, this background is automatically sized to cover the span of the text; the panel background can also be customized for color and transparency, aiding legibility in a variety of environments. In addition, we defined a simple Image Panel object to support textual annotations.

World authors have a great degree of flexibility in defining the behavior of semantic objects with our framework. There are separate distance ranges and description levels for information and geometry LODs. The label is associated to its referent by an optional connector line and by its proximity to the referent in 3D space. The connector line may be colored and its origin on the referent's geometry customized.

In order to support legibility over a variety of distances and angles, information labels are defined as true 3D billboards which always appear 'right side up' and orthogonal to the viewers' orientation. In addition, we implemented three scaling modes for information panels: fixed size, constant size, and periodic size. Fixed size means the information panel retains its defined size regardless of the user's distance. Constant size scales the annotation continuously by a factor of the user's distance. This helps guarantee legibility, but also can give confusing depth cues. A middle ground is provided by the periodic size option that proportionally scales the panel at intervals of user distance. Figure 3 shows a test environment in a VRML browser [24] instantiating semantic object behaviors.

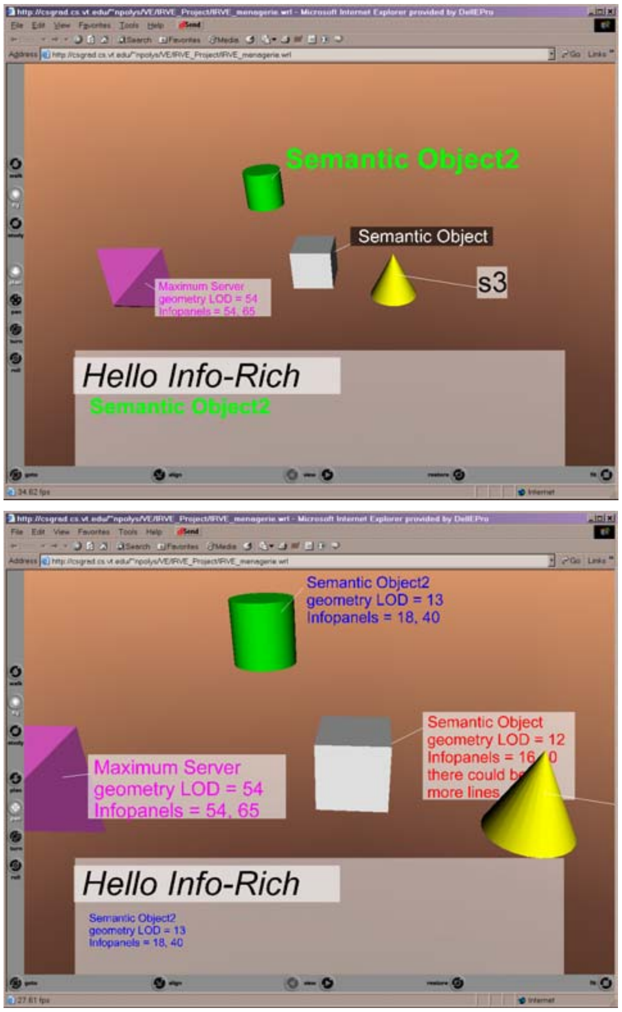


Figure 3. Semantic objects in a test environment. The top image shows the result of pointing to an object in a zoomed-out view; the bottom image shows the result of clicking on an object in a zoomed-in view.

In grappling with the requirement for attribute panels to maintain proximity to their referent, we implemented a variety of layout schemes. The first approach locates panels by a specified 3D offset relative to the referent geometry's origin regardless of the user's location. The second approach takes the user's position into account by placing the panel on the nearest corner of an author-defined rectangular prism that encloses the geometry's bounds. Both of these approaches are useful in different types of environments that may have oddly-shaped objects or objects that are close by.

Occlusion is a major problem in IRVE implementations. Since we did not implement an image plane representation of objects' projection, there was no notion of a view 'manager' to layout the information panel locations. In order to mitigate this situation, we decided to implement an emergent behavior algorithm similar to flocking [27] where panels' goal location is the nearest prism corner and obstacles are other semantic objects. In this scheme there is no centralized control and "intelligent" behavior is the result of simple attraction and avoidance rules. Figure 4 shows a comparison of a corner layout and a flocking layout.

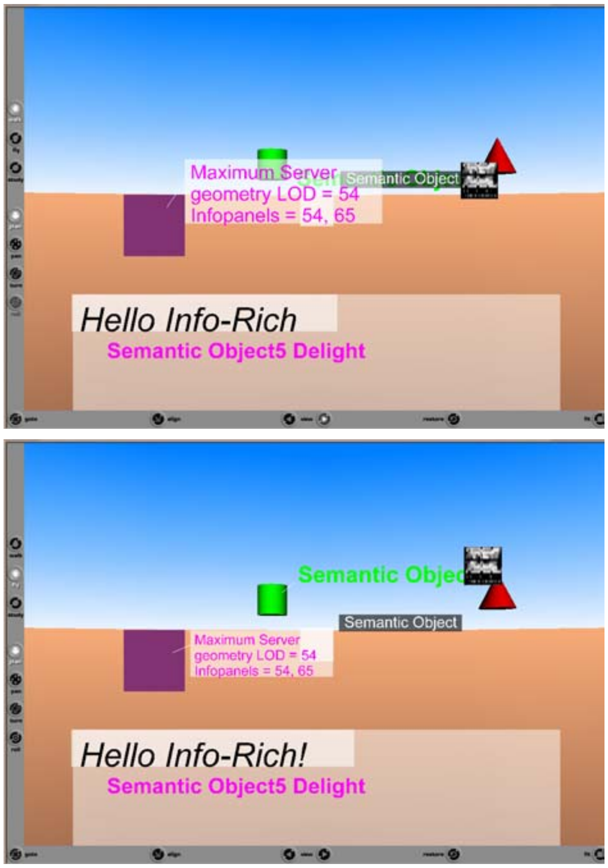


Figure 4. Comparison of deterministic (top) and emergent (bottom) layout techniques

Emergent behavior as we implemented it has some disadvantages, however. Most importantly, it requires tuning per environment – parameter sets do not generalize. For example, attraction and repulsion forces need to be balanced by the number and density of semantic objects in the scene (i.e. the degree of crowding). In addition, parameters are interdependent and finding the proper values for thresholds and movement steps can be difficult.

Finally, to test the portability of our framework to other immersive displays such as an HMD, we implemented an interface for head tracker data to be fed into the VRML scene via the External Authoring Interface (EAI). We chose the ParallelGraphics Cortona engine to implement this functionality since it has good performance and standards compliance. While the combination of technologies required (dlls, JNI, and JavaApplets) leads to some added latency, non-stereo display of semantic objects appears to be successfully transferable from desktop to HMDs. Future work will involve testing other engines that support VRML or X3D and native tracker input functionality.

4.1.4 Example Applications

We believe that researchers, developers, and users can benefit from structured, encapsulated behaviors for semantic objects. The flexibility and portability of this framework leads to possible applications in a number of domains such as HCI research testbeds, CAD and architecture applications, and medical applications such as anatomy visualization and biological simulation.

By encapsulating information-rich display behaviors in reusable scenegraph nodes, we have been able to apply this work to another research project in the domain of anatomical visualization with little extra cost. In this application, abstract information such as surface area, blood count, etc. are all associated with various organs and tissue sections with users being able to navigate down to the cellular level. Figure 5 shows our system integrated with a human anatomical model.

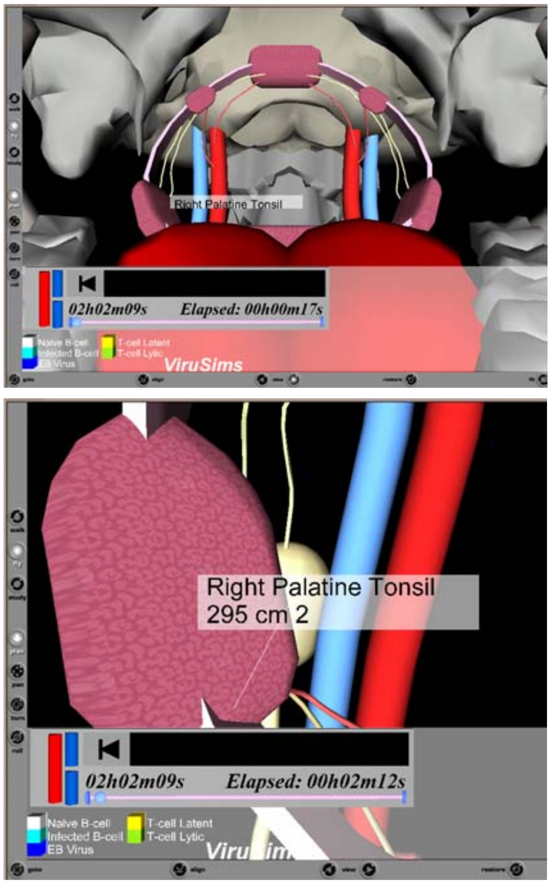


Figure 5. Zooming in on an anatomical model defined with Semantic objects

Our implementation has led us to recognize a deficiency in the current standards specifications – in the current standards, HUDs can only be defined in world space, but authors need scenegraph control of last-rendered objects or objects at the near clipping plane. We hope this work will lead advocacy for an X3D specification component that addresses this type of compositing functionality.

While our system has shown some useful applications and promising approaches, more work is needed to identify optimal display and interaction conditions for successful IRVEs.

4.2 IRVE Travel Testbed

Another useful tool for IRVE research is an environment for the evaluation of various display or interaction techniques. We chose to implement a testbed environment to serve this purpose. The advantages of testbed evaluations are that they combine multiple tasks, multiple independent variables, and multiple response measures to obtain a more complete picture of interaction technique performance and usability.

Specifically, our testbed allows the evaluation of travel techniques for IRVEs with embedded text information. There is at least one travel testbed existing for traditional VE evaluations [12]. Using this testbed, seven different travel techniques (pointing, torso-directed steering, HOMER travel, 2D map dragging, ray-casting travel, and Go-Go travel) were studied for speed, accuracy, spatial awareness, ease of learning, ease of use, information gathering ability, and presence.

4.2.1 Classification of Text Layout Techniques

Text layout and display can be classified quantitatively (amount of text embedded), qualitatively (by varying the visual attributes), spatially (based on its location in the space) and temporally (varying the display on a time scale). We created a taxonomy reflecting these categories as a theoretical basis for our testbed implementation. The high-level entries of our taxonomy are shown in Figure 6.

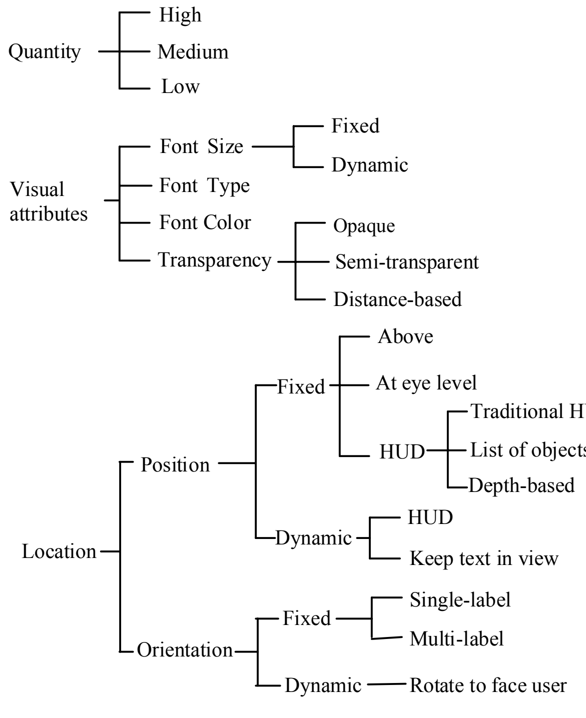


Figure 6. Taxonomy of Text Display Techniques for IRVEs

In this classification, labels with few words are considered low in text content. Similarly, a brief description of the objects was considered to be medium quantity and a more detailed description as a high quantity of text content.

The visual attributes of the text are those aspects that directly affect the legibility of the text and the overall perception of the VE itself. These include size of the text, the type of font, the transparency of the text frame, and color. Some of these factors were further classified into lower-level attributes. The size of the text may be fixed or it may change dynamically. In the dynamic condition, text is scaled based on the user's distance from it, allowing text to be read even when it is attached to a distant object. The transparency attribute refers to the panel on which the text is drawn (not to the text itself). This box may be opaque or semi-transparent, or the transparency may be varied according to distance from the user.

The position of the text may be fixed in space or dynamic, based on the user's view. In the fixed case, we consider text fixed above the object, fixed at the eye level of the user, and fixed on a heads-up display (HUD). The fixed HUD condition was further classified into variable-size text displayed at the closest point to the object (to produce the illusion of depth), text displayed as a list of all objects in view, and finally fixed-size text displayed on the image plane. The dynamic position category includes a HUD and a "keep text in view" technique. In the dynamic HUD, the text display changes based on the user's view of the environment, i.e. the content on the HUD is dynamically updated based on the user's view. In the "keep text in view" case, the position of the text display is dynamically updated to always be in the user's view if the corresponding object is within the view volume.

The second spatial attribute (orientation) was decomposed into fixed and dynamic categories. In the fixed case, we include the use of a single label (like a billboard) and the use of multiple labels (e.g. a cube with all faces having the same text content). In the dynamic orientation condition, the text is rotated to always face the user.

We have not yet pursued the temporal classification, but possibilities would include effects such as animation, marquee, and blinking.

4.2.2 Testbed Implementation

Our testbed uses the same travel techniques implemented in the previous testbed but added text layout techniques from the taxonomy above. This testbed was developed to support any combination of layout parameters (except color and font type) that can be created from the taxonomy. The rationale for not providing control to vary the text color is that this characteristic, unlike the other properties described in the taxonomy, is almost entirely application-dependent.

The system was implemented using SVE [21], OpenGL and GLF. The GLF font-rendering library [32] supports numerous text rendering techniques including bitmapped, outlined, and texture mapped. The font can be displayed as wire-frame, solid, etc. It is easy to use and is cross-platform compatible. The system is flexible to allow the experimenter to vary parameters at run-time.

A *view management* component similar to some previous AR work [5] was implemented to decide where to put the information in a dynamic scene. In our system, the inputs to the view management component are the various display constraints from the taxonomy. The output, based on the current viewpoint of the user, is the location where information can be displayed.

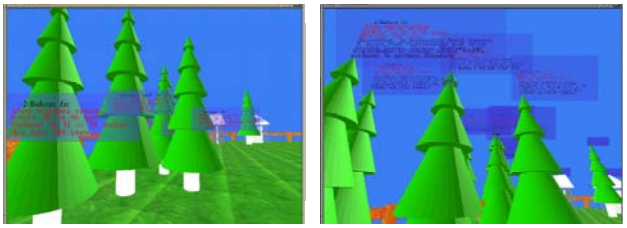


Figure 7. Text Layout from the View Component in the Testbed

Figure 7 shows examples of the testbed display. In the left part of the picture, the label is semi-transparent, placed at the center of the objects, and fixed in orientation. A medium amount of information (tree species name and characteristics) is displayed. In the right part of the picture, the transparency is varied with distance. In addition, the information is placed on the top of the tree and always faces the user.

Figure 8 shows an example of a HUD. All objects that are wholly or partially visible are displayed on the viewplane. The font is displayed at the same size regardless of the distance of the object to the user (dynamic display). The text labels follow the user's viewpoint and disappear when that object is completely outside the user's view. Another type of HUD implemented has the names of the objects displayed on the top part of the screen in a transparent view plane and has lines joining the labels and their referents.

Using this testbed, an experimenter can study any combination of text display techniques from our taxonomy and travel techniques from the list above. Of particular interest are the interactions between IRVE text layout and travel techniques. No single layout technique is likely to produce good user task performance and usability for all travel techniques, and vice-versa.

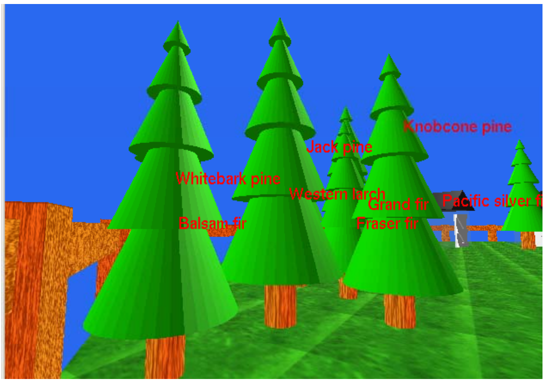


Figure 8. A HUD Display from the View Component in the Testbed

4.2.3 Proof-of-Concept Experiment

To demonstrate the use of the testbed, we conducted an HMD-based experiment to investigate the effects of varying amounts of text (medium and large) and two different travel techniques (HOMER and gaze-directed steering) on usability and task performance. In the HOMER technique [10], the user selects a remote object by shooting a ray from the stylus, and the selected object becomes the center of the world around which the user can navigate by moving his arms. In gaze-directed steering [22], the user looks in the direction he wants to travel and presses a button. We set the text layout to fixed orientation, half of object height position, semi-transparent display, and fixed size font. For this experiment, font type was fixed to Arial. The color of the font was varied according to the information levels. The title of the object (low level of text) was provided in black, the short description (medium) in red, and the detailed (high) description in brown.

We used a between-subjects design with 16 subjects (four per condition – see Table 1), 11 male and 5 female. All subjects were computer science students. Each user performed two tasks, with the order of the tasks counterbalanced. We measured the time per task and interviewed the subjects to obtain subjective usability and comfort data. The tasks required the users to read the text on the display and to navigate around the environment to do so. We designed one task that required the user to navigate through our "forest" environment and browse through all the trees. The second task required the user to travel long distances in the space and then read the information displayed on the objects. The rationale for selecting such tasks was to investigate the usability of the travel and text layout techniques for each of these extreme conditions.

Table 1: Experimental Design

	HOMER	Gaze-directed
Medium amount of text	4	4
Large amount of text	4	4

We used a two-factor ANOVA with replication for analysis of the time metric. The analysis showed us that HOMER performed better than gaze-directed travel in both text conditions. However, there was no statistical significance in this finding ($p > 0.05$). This may have been caused by the low sample size in our experiment. Also, there was no statistically significant difference between the medium and large text quantity conditions.

Feedback from user interviews revealed the following usabillity issues for HMD-based text IRVEs:

1. For text objects that are densely packed, transparency in the text display hinders readability. Conversely, for environments having a lower density of text objects, transparency can enhance the spatial orientation of the user, since she can see the objects behind the text frame.
2. For environments that constrain the user to travel at a fixed height only, displaying the text at higher than eye level can be fatiguing for the user. If, for a particular application, it is necessary to display text at a higher level, providing the user the ability to fly would be necessary.
3. For environments in which the user is permitted or required to come very close to text objects with high information content, scaling down the text would make it easier to read. However, the change in scale might disorient the user.

This experiment is moderately complex (four conditions, multiple tasks per condition, complex travel and text layout techniques). Using our testbed, however, implementing and running an experiment of this type is almost trivial. The four conditions can be described with four simple initialization files, the number of trials can be selected at runtime by the experimenter, and the techniques are already implemented. Changing the techniques to be tested or modifying the choices for the fixed attributes of the environment simply requires changing the initialization file. This testbed will allow us to quickly and easily perform a large number of experiments as we address the issues related to text layout and travel in IRVEs.

5. RESEARCH AGENDA

With a theoretical framework (section 3) and a set of tools (section 4) in place, we can begin to address general research questions related to IRVEs. The ultimate goals of the research are a complete understanding of the nature of IRVEs – what makes them effective, what makes them usable, and how users think and act when using them – and a set of principles or guidelines for designers and developers based on this knowledge.

There is a huge amount of work to be done, but we can at least begin to map out a strategy for research in this area. We have identified three key areas for future research.

5.1 Implementation Issues

The first category of research questions deals with the underlying implementation of IRVEs. One key consideration is the database that is used to store the abstract and/or spatial information. Our research will address questions such as:

- How do we manage the content creation issues for IRVEs? Can we provide usable and efficient tools for building the VE, building the database, and linking the two?
- How can we populate a 3D scene with embedded information from a live spatiotemporal database?
- How can we manage huge amounts of spatiotemporal information at multiple scales?
- How do we embed various types of abstract information (such as text, numbers, images, audio, and graphs) in the same 3D environment?

5.2 Information Display Issues

A second category of research questions addresses the display of the abstract information in the IRVE. This display does not have to be visual only – it may include audio, haptics, or other sensory information. We will explore questions such as:

- How do we maintain the legibility of embedded information of different types?
- What should be the mapping between the database representation and the sensory representation of the abstract information?
- How can we encourage a consistent and correct mental model of the data in the user?
- How do we decide where information should be displayed (e.g. on a heads-up display, on a hand-held tablet, in the scene itself)?
- What is the appropriate level of detail for embedded information? How should level of detail be managed? Should the user have control over these parameters?
- Should information display be different in immersive and non-immersive VE systems? If so, how?
- How should we embed information only indirectly related to an object or location in the environment?
- How should we embed "ambient" information that exists at all points in an environment (e.g. temperature)?

5.3 Interaction Issues

At a minimum, IRVEs allow the user to navigate freely through the 3D environment, gathering information about the world. But navigation in IRVEs may have different characteristics than navigation in traditional VEs. In addition, the user may want to interact with the IRVE in other ways. We should perform research on questions such as:

- How does the travel technique affect the user's ability to browse or search the abstract information?
- What are the best techniques for 3D navigation through environments with multiple scales?
- How should users access information that is not immediately visible in the environment (i.e. details-on-demand)? Should the access techniques change based on how the information is displayed?
- What techniques can we use to interact with the information (e.g. annotation, query, search, filtering, modifying viewing parameters, changing the initial conditions of a simulation, etc.)?

6. CONCLUSIONS

IRVEs provide exciting opportunities for extending the use of VEs for more complex, information-demanding tasks in many domains. We have generalized prior research and provided a theoretical framework for systematic research in IRVEs. We have also presented a set of tools for the development and evaluation of IRVEs, and a research agenda for this burgeoning area.

Our work so far has demonstrated the potential of IRVEs, and the work presented here should enable a deeper understanding of IRVEs from the point of view of effectiveness and usability. Ultimately, we hope that this research will open the door for an entirely new set of VE applications.

7. ACKNOWLEDGEMENTS

The authors would like to thank the people involved with the ViruSims project: Dr. Karen Duca, Dr. Reinhard Laubenbacher and their research team at The Virginia Bioinformatics Institute: Jignesh Shah, Rohan Luktuke, and Purvi Saraiya. We also acknowledge the help of the subjects in our experiment. Finally, we thank Glenn Fink for his contribution to our discussions of IRVEs.

8. REFERENCES

[1] Becker, R. & Cleveland, W., "Brushing Scatterplots", Technometrics, 29(2), 127-142, 1987.

[2] Bederson, B.B., Hollan, J.D., Perlin, K., Meyer, J., David, B., & Furnas, G., "Pad++: A Zoomable Graphical Sketchpad for Exploring Alternate Interface Physics", Journal of Visual Languages and Computing, 7(1), 3-32, 1996.

[3] Bederson, B. B., Meyer, J., & Good, L., "Jazz: An Extensible Zoomable User Interface Graphics Toolkit in Java", Proceedings of the ACM Symposium on User Interface Software and Technology, 171-180, 2000.

[4] Bell, B.A. & Feiner, S.K., "Dynamic Space Management for User Interfaces", Proceedings of the ACM Symposium on User Interface Software and Technology, 238-248, 2000.

[5] Bell, B.A., Feiner, S.K., & Hollerer, T., "View Management for Virtual and Augmented Reality", Proceedings of the ACM Symposium on User Interface Software and Technology, 101-110, 2001.

[6] Benford, S., Snowdon, D., & Mariani, J., "Populated Information Terrains: First Steps", In R. Earnshaw, J. Vince, & H. Jones (Eds.), Virtual Reality Applications, 27-39, Academic Press, 1995.

[7] Bergman, L., Richardson, J., Richardson, D., & Brooks, F., "VIEW - An Exploratory Molecular Visualization System with User-Definable Interaction Sequences", Computer Graphics (Proceedings of SIGGRAPH 1993), 27, 117-126, 1993.

[8] Bolter, J., Hodges, L.F., Meyer, T., & Nichols, A., "Integrating Perceptual and Symbolic Information in VR", IEEE Computer Graphics and Applications, 15(4), 8-11, 1995.

[9] Bowman, D., "Principles for the Design of Performance-Oriented Interaction Techniques", In Handbook of Virtual Environment, K.M. Stanney (Ed), Lawrence Erlbaum Associates, 2002.

[10] Bowman, D., & Hodges, L. "An Evaluation of Techniques for Grabbing and Manipulating Remote Objects in Immersive Virtual Environments." Proceedings of the ACM Symposium on Interactive 3D Graphics, 35-38, 1997.

[11] Bowman, D., Hodges, L.F., & Bolter, J., "The Virtual Venue: User-Computer Interaction in an Information-Rich Virtual Environment", Presence: Teleoperators and Virtual Environments, 7(5), 478-493, 1998.

[12] Bowman, D., Johnson, D., & Hodges, L.F., "Testbed Evaluation of Virtual Environment Interaction Techniques", Presence: Teleoperators and Virtual Environments, 10(1), 75-95, 2001.

[13] Bowman, D., Wineman, J., Hodges, L., & Allison, D. "The Educational Value of an Information-Rich Virtual Environment." Presence: Teleoperators and Virtual Environments, 8(3), 317-331, 1999.

[14] Bryson, S. & Levit, C., "The Virtual Wind Tunnel", IEEE Computer Graphics and Applications, 12(4), 25-34, 1992.

[15] Card, S., Mackinlay, J., & Shneiderman, B. (Eds), "Information Visualization: Using Vision to Think", Morgan Kaufmann, 1999.

[16] Dede, C., Salzman, M., & Loftin, R., "ScienceSpace: Virtual Realities for Learning Complex and Abstract Scientific Concept", Proceedings of the Virtual Reality Annual International Symposium, 246-252, 1996.

[17] Fairchild, K.M., Poltrock, S.E., & Furnas, G.W., "Semnet: Three-Dimensional Graphic Representations of Large Knowledge Bases", Cognitive Science and its Applications for Human-Computer Interaction, Lawrence Erlbaum Associates, 201-233, 1988.

[18] Feiner, S., Macintyre, B., Haupt, M., & Solomon, E. "Windows on the World: 2D Windows for 3D Augmented Reality", Proceedings of the ACM Symposium on User Interface Software and Technology, 145-155, 1993.

[19] GeoVRML, http://www.geovrml.org (Available May 2003).

[20] Hoellerer, T., Feiner, S., Terauchi, T., Rashid, G., & Hallaway, D., "Exploring MARS: Developing Indoor and Outdoor User Interfaces to a Mobile Augmented Reality System." Computers and Graphics, 23(6), 779-785, 1999.

[21] Kessler, G.D., Bowman, D., & Hodges, L.F., "The Simple Virtual Environment Library: An Extensible Framework for Building VE Applications". Presence: Teleoperators and Virtual Environments, 9(2), 187-208, 2000.

[22] Mine, M., "Virtual Environment Interaction Techniques." Technical Report TR95-018, UNC Chapel Hill CS Dept, 1995.

[23] North, C., "Multiple Views and Tight Coupling in Visualization: A Language, Taxonomy, and System", Proc. CSREA CISST Workshop of Fundamental Issues in Visualization, 626-632, 2001.

[24] Parallel Graphics Cortona VRML browser http://www.parallelgraphics.com. (Available May 2003).

[25] Pierce, J.S., Forsberg, A., Conway, M.G., Hong, S., Zeleznik, R., & Mine, M.R., "Image-Plane Interaction Techniques in 3D Immersive Environments", Proceedings of the ACM Symposium on Interactive 3D Graphics, 39-43, 1997.

[26] Reddy, M., Iverson, L., & Leclerc, Y.G., "Under the Hood of GeoVRML 1.0", Proceedings of VRML0 Symposium, 2000.

[27] Reynolds, C.W., "Flocks, Herds, and Schools: A Distributed Behavioral Model", Computer Graphics (Proceedings of SIGGRAPH 1987), 21(4), 25-34, 1987.

[28] Risch, J., May, R., Thomas, J., & Dowson, S., "Interactive Information Visualization for Exploratory Intelligence Data Analysis", Proceedings of the Virtual Reality Annual International Symposium, 230-238, 1996.

[29] Robertson, G., Card, S., & Mackinlay, J., "Information Visualization Using 3D Interactive Animation", Communications of the ACM, 36(4), 57-71, 1993.

[30] Shneiderman, B., "The eyes have it: A task by data type taxonomy for information visualizations", Proceedings of IEEE Visual Languages, 336-343, 1996.

[31] Song, D. & Norman, M., "Cosmic Explorer: A Virtual Reality Environment for Exploring Cosmic Data", Proceedings of the IEEE Symposium on Research Frontiers in Virtual Reality, 75-79, 1993.

[32] Survey of OpenGL Font Technology, http://www.opengl.org/developers/code/features/fontsurvey/ (Available May 2003).

[33] Web3D Consortium, http://www.web3d.org; Specifications: http://www.web3d.org/fs_specifications.htm (Available May 2003).