

3DIVS: 3-Dimensional Immersive Virtual Sculpting

Falko Kuester

Center for Image Processing and
Integrated Computing
Department of Computer Science
University of California, Davis
+1-530-754-9470

kuester@cs.ucdavis.edu

Mark A. Duchaineau

Lawrence Livermore National
Laboratory
7000 East Ave.
Livermore, CA 94550
+1-510-423-1320

duchaine@llnl.gov

Bernd Hamann

Center for Image Processing and
Integrated Computing
Department of Computer Science
University of California, Davis
+1-530-754-9157

hamann@cs.ucdavis.edu

Kenneth I. Joy

Center for Image Processing and
Integrated Computing
Department of Computer Science
University of California, Davis
+1-530-752-1077

joy@cs.ucdavis.edu

Antonio E. Uva

Dipartimento di Progettazione e
Produzione Industriale
Politecnico di Bari
Viale Japigia, 182, 70126 Bari, Italy
+1-530-754-9470

uva@poliba.it

ABSTRACT

Virtual Environments (VEs) have the potential to revolutionize traditional product design by enabling the transition from conventional CAD to fully digital product development. The presented prototype system targets closing the "digital gap" as introduced by the need for physical models such as clay models or mockups in the traditional product design and evaluation cycle. We describe a design environment that provides an intuitive human-machine interface for the creation and manipulation of three-dimensional (3D) models in a semi-immersive design space, focussing on ease of use and increased productivity for both designer and CAD engineers.

Keywords

Virtual Reality, Immersive Environments, 3D Sculpting, Computer Aided Geometric Design (CAGD).

1. INTRODUCTION

As companies focus on streamlining productivity as a response to global competition, the migration to computer-aided design (CAD), computer-aided manufacturing (CAM), and computer-aided engineering (CAE) systems has established a new backbone of modern industrial product development. While most of these technological advances are of high benefit to the engineering and design community [1], they still lack some of the important visual

and haptic features crucial to product development. A car design, for example, traditionally originates from a clay model, which after digitization, forms the basis for a numerical CAD description in Bezier, B-spline, or NURBS format [9,10]. Furthermore, physical models, so called mock-ups, still play a key role in the areas of design evaluation and verification, introducing a discontinuity in the otherwise CAD-centered development cycle. 3DIVS aims at closing this technology gap experienced by design and CAD engineers by transforming the classical design paradigm into its fully integrated digital and virtual analog (Figure 1). The goal is to facilitate the cooperation between designer and engineer and to aid in streamlining design, analysis and testing phases. This is achieved by providing a real-time 3D semi-immersive design environment that supports modeling metaphors such as intuitive hand gestures and virtual tools for the creation and modification of geometric and CAD primitives. This approach allows the preservation of the hands-on experience from the physical world while overcoming the well known classical 2D constraints introduced through the keyboard. A powerful feature of this implementation is its ability to provide an unprecedented amount of real estate for the user in the form of a 3D desktop. Anyone with experience in working with multiple open and overlapping windows or virtual 2D desktops on a regular display will appreciate that objects, tools, and other components can now be placed and arranged in an almost unlimited 3D domain.

2. BASIC SYSTEM LAYOUT

3DIVS was specifically designed to work with a new generation of stereo projection systems currently marketed under names like *Immersive Workbench*, *Responsive Workbench* and *ImmersaDesk* [2,4]. We used the *Immersive Workbench* from Fakespace which allows stereo projection of 3D computer-generated images onto an approximately 2x1.5m projection area. A 4-processor SGI Onyx2 InfiniteReality system (225MHz, R10000 processor) was used as the rendering engine. The basic hardware setup is illustrated in Figure 2. The user is wearing shutter glasses with integrated head

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

NPIVM 99 Kansas City Mo USA

Copyright ACM 2000 1-58113-254-9/99/11...\$5.00

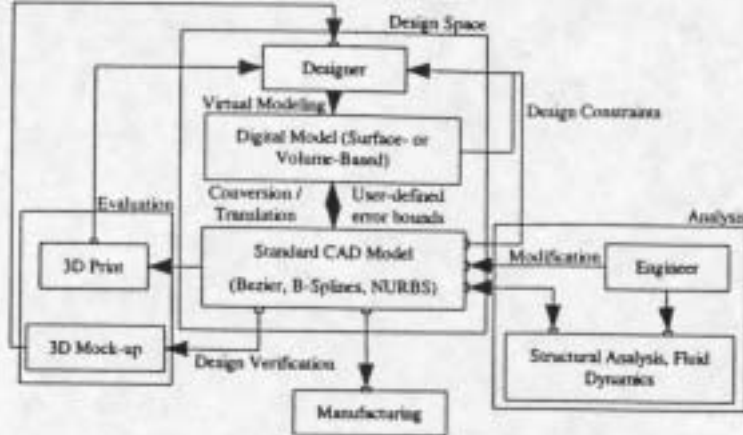


Figure 1: Integrated Modeling Environment

tracking for stereoscopic viewing and uses a set of pinch gloves combined with a stylus device for interaction with the VE. The spatial data describing the user's head position and hand movements is fully incorporated into the VE. We briefly describe the input devices:

- **Stylus:** Using a fixed transmitter as reference, this pencil-like system accurately computes position (x, y and z coordinates) and orientation (yaw, pitch and roll) of a tiny receiver contained in the stylus. In addition, it provides an integrated button that can be used for picking actions.
- **Gloves:** The Pinch System uses cloth gloves with electrical sensors in each fingertip. Contact between any two or more digits completes a conductive path, providing a variety of possible "pinch" gestures that can be associated with distinct actions. Additionally, an attached electromagnetic tracker captures the position of each glove.

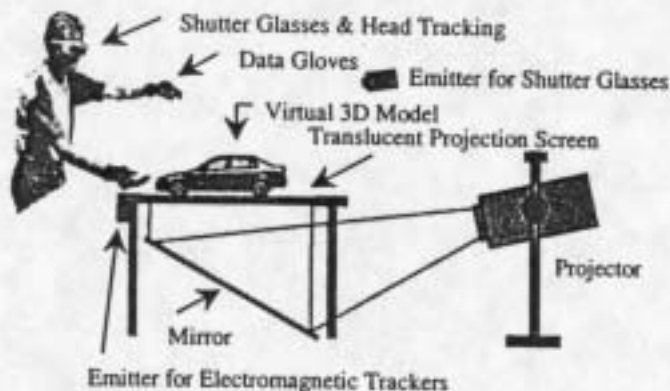


Figure 2: Hardware setup

3. MODELING METAPHORS

The creation, visualization and modification of CAD data within a VE requires a new type of human-computer interface. The observation that humans develop certain patterns on how to distribute tasks between their hands [6] has triggered the development of a natural two-handed interface using spatially tracked input devices such as data gloves [3,7] and pointers. In our implementation, the user is always wearing two gloves and the stylus can be used as an additional tool with situation-specific meaning.

Devices used	Interaction performed
--------------	-----------------------

Head tracker + stereo rendering	Navigation
Stylus	Selection, manipulation
One glove	Object navigation mode
Two gloves	Scene navigation mode
One glove + stylus	Natural object design

Table 1: Interactions performed using different devices

The basic set of controls that an immersive modeling environment has to provide include:

- Virtual menus
- Object creation
- Object selection
- Object manipulation
- Scene/Object navigation

3.1 Virtual Menus

Menus are a vital component of all modeling systems. Given the new set of VE input devices, new concepts have to be implemented. Different solutions to this problem were proposed during recent years opting for either a direct port from the classical 2D menu to its 3D counterpart or new implementations designed specifically for 3D space [5]. Commonly observed problems are interference between the 3D menus and the scene and that of reaching options in highly cascading menus. We distinguish between gesture-based trigger and invocation events that allow the user to activate and select from various menus. For the novice user, the "watch-menu" option is supported, which displays the base menu as soon as the user pretends to check the time on his/her wristwatch. The menu is composed of 3D buttons assembled on a rectangular palette and is attached to the "watch" glove. Besides its clarity and accessibility, we choose this 2½D "palette" menu since its structure is well accepted in the drafting community. All the sub-menus pop up from the original palette in a different plane and can be closed by

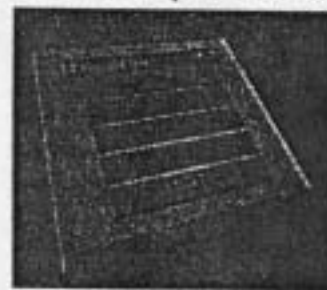


Figure 3: Virtual file menu

squeezing them with the other hand. Each plane defines a sub-menu level while the higher level is slightly faded to help the user focus on the selection. Since the user can actually see his/her physical hand, the menu has to be translated slightly to be visible. Furthermore, since the menu and all its sub-menus are treated as regular objects, the user can translate, rotate and scale them as desired. The menu items can be text, a graphical presentation of the associated function, or a combination thereof.

3.2 Object Creation and Manipulation

3DVIS provides a variety of mechanisms for the creation and manipulation of objects. Objects can be either imported from a file, interactively created by using a set of CAD primitives, or modeled using drawing primitives and freeform shapes. The CAD primitives are represented in Bezier, B-Spline or NURBS form, and for smaller models tessellated on the fly. Supported CAD primitives and operations are: point, line, curve, circular arc, elliptic arc; surface, ruled surface, surface of revolution (cylinder, sphere, etc.); symmetric geometry creation by using reflection, curve fitting, surface fitting and Coons patches. Drawing primitives and freeform shapes support the conventional line- and polygon-oriented sketching concept. Most CAD primitives have a predefined set of default properties assigned to them. In the case of the NURBS surface primitive, the system creates a predefined set of control points, knot vectors, weights, orders, etc. Once an object is created, its visual representation is added to a hierarchical scene graph. All graphics components within the VE are treated as objects, which can be freely positioned, manipulated or grouped. Special behavioral actions can be attached to any object and turned into a tool for the manipulation of other objects. Any regular non-static object within the scene graph can be directly accessed and manipulated. Non-static objects come without any attached constraints, whereas static objects come with a set of distinct attributes defining which manipulations and interactions are allowed within certain boundaries. These objects are of particular interest for industrial design since they preserve predefined design constraints. All visible objects contained in the scene graph can be selected and their properties visualized using a simple hand gesture. Selected non-static objects can be directly accessed, manipulated and grouped, whereas static objects come with a set of distinct attributes defining which interactions and manipulations are allowed within the specified boundary conditions.

Furthermore, we distinguish between engineering and design metaphors in the context of object creation and manipulation. For example, the engineering metaphor for a NURBS surface provides the user with direct access to control points, weights, degrees and knot vectors, whereas the design metaphor mask this information using hand-surface interpolation with local feature control. In other words, the designer uses high-level tools such as magnets and stamps to sculpt the surface without actually having to deal with its mathematical description.

Another important prerequisite was that an object design history must be accessible. This approach allows us to enable important features including undo and redo operations but also supports tracing of efficient artistic or engineering design patterns.

3.3 Virtual Toolbox

The virtual toolbox is merging the advantages of conventional physical tools with the high-precision components of today's CAD systems. Instead of actually defining tools, we define actions and functionality, which can be applied to arbitrary objects in the VE. By doing this, we provide the designer with unlimited space for creativity and the means for the creation of new tools and design concepts. In our object oriented framework tools can be used to shape models which then can be turned into tools on their own.

Manipulators	Translation, rotation, scale, merge, etc.
--------------	---

Masks	Modeling constraints applied to objects
Editors	Color, material and certain properties
Smoothers	Surface smoothing operations
Hotwires	Cutting tool operations
Stamps	3D printing stock
Magnets	Attractive or repulsive forces
Brushes	Apply properties to touched objects
Jigs	Movement constraints

Table 2: Components of the virtual toolbox

3.4 Object Selection

This operation is the starting point for any object-specific interaction task. The basic idea is to use a 3D input device and to select the closest object to the spatial position of the device when device-specific action is invoked (button press or pinch). While this operation can be easily performed in 2D by using the intersection of a ray with the screen this approach is not satisfying for 3D environments. Specifying a target based on the absolute (x, y, z) position of the tracker can be a tiresome task and better solutions using *ray casting* and *cone casting* metaphors exist [6]. Instead of directly specifying the 3D point, the spatial input device is used to shoot a ray or a spotlight into the scene, allowing the user to hold the input device in a comfortable position and rotate it to change the ray direction. Nevertheless, since accuracy and ease-of-use are key design factors for our application, we decided to use the absolute position of the tracker mapped to world coordinates. In the case of the stylus, a virtual proxy at the end of its tip visualizes the target pointer. When the proxy intersects the bounding box of a selected object, the object is highlighted and ready for selection. Improved interactivity is achieved by the possibility to move or scale the entire scene or particular selected objects. Once an object is selected it can be rotated, translated, scaled or re-shaped, provided the object is declared as non-static.

3.5 Scene Navigation

Using head tracking, the designer can study an entire model in a VE by simply moving his/her head or physically walking around the model. The navigation mode is enhanced in two different ways. By performing a pinch action the user can select and re-position an object by using either one of the data gloves. This mode, called *object-navigation mode*, allows the user to freely re-position and analyze the active object. If both gloves are pinching at the same time, 3DVIS switches into *scene-navigation mode*. The imaginary segment between the pinching points is used as a five-degrees-of-freedom manipulator. The length of this segment, i.e., the distance of the two picking points, controls the scale of the model. In the case that no object has previously been selected, the navigation action is applied to the entire scene, which, by definition, is just another object composed of a group of objects. This scheme also supports a user-defined level of accuracy in which finer or coarser levels of precision can be defined by scaling the workspace. This navigation scheme is intuitive and versatile and new users are able to easily examine even complex scenes after only a few minutes of practice.

4. APPLICATION FRAMEWORK

The application framework of 3DVIS is depicted in Figure 4. The conceptual idea was to develop a fully object-oriented framework that can handle a variety of different CAD objects and drawing primitives on a plug-in basis. Each object provides the system with information about its type, properties and visual representation. CAD objects additionally furnish the description of their control grid or control mesh, which is used to compute the visual representation for the particular object. The visual representation of any object is stored in a rooted hierarchical scene graph that maintains distinct callbacks to the creating objects. The object type is used to associate the object with a particular virtual menu providing access to a set of common options, and its property field provides the means for adding object-specific controls to the VE.

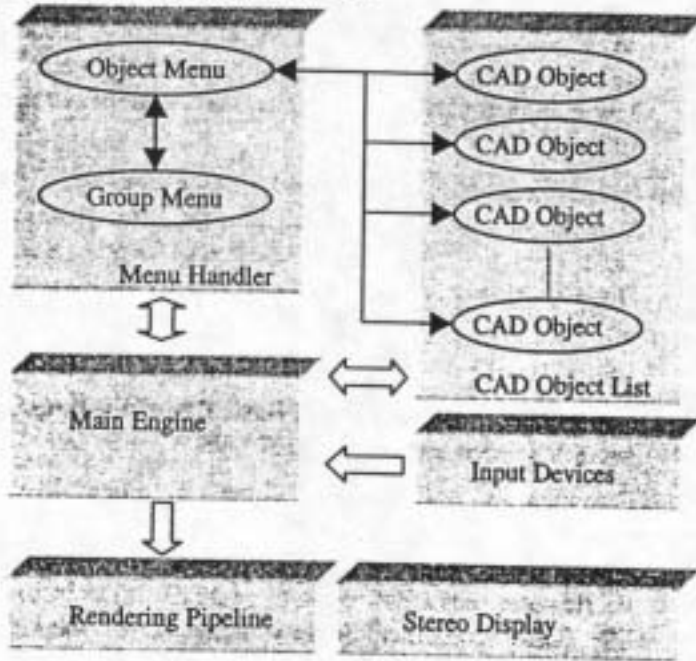


Figure 4: Object oriented application framework

5. CONCLUSIONS

We have described some of the components required for a fully integrated semi-immersive VE. Although only at a preliminary stage, 3DVIS has shown to be very efficient, both for quick sketching and more complicated tasks like NURBS modeling. The current challenge lies in the development of more complex interaction and modeling schemes in the form of new virtual tools and input devices, using voice and gesture (pattern) recognition. One of the core tasks for the near future is to provide the necessary modeling precision required for standard engineering design tasks. The object-oriented framework is easily extendable and provides a user-friendly prototyping environment. An enlarged CAD feature set is currently under development. Eventually, this will be accompanied by a collection of CAD file converters.



Figure 5: Example of navigation (a) and editing (b)

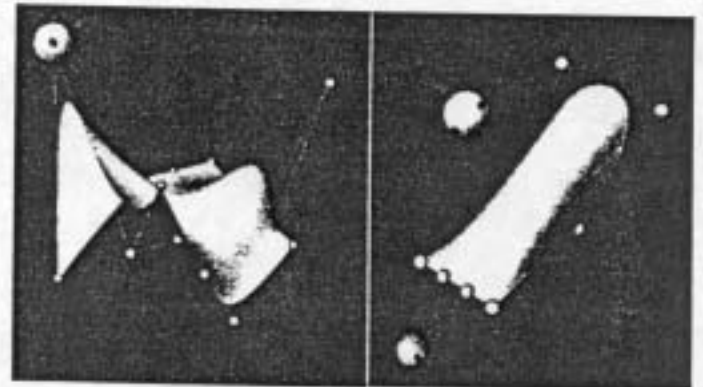


Figure 6: Nurbs surface and data glove proxies

of objects. This scheme also supports a user-defined level of accuracy in which finer or coarser levels of precision can be defined by scaling the workspace. This navigation scheme is intuitive and versatile and new users are able to easily examine even complex scenes after only a few minutes of practice.

4. APPLICATION FRAMEWORK

The application framework of 3DVIS is depicted in Figure 4. The conceptual idea was to develop a fully object-oriented framework that can handle a variety of different CAD objects and drawing primitives on a plug-in basis. Each object provides the system with information about its type, properties and visual representation. CAD objects additionally furnish the description of their control grid or control mesh, which is used to compute the visual representation for the particular object. The visual representation of any object is stored in a rooted hierarchical scene graph that maintains distinct callbacks to the creating objects. The object type is used to associate the object with a particular virtual menu providing access to a set of common options, and its property field provides the means for adding object-specific controls to the VE.

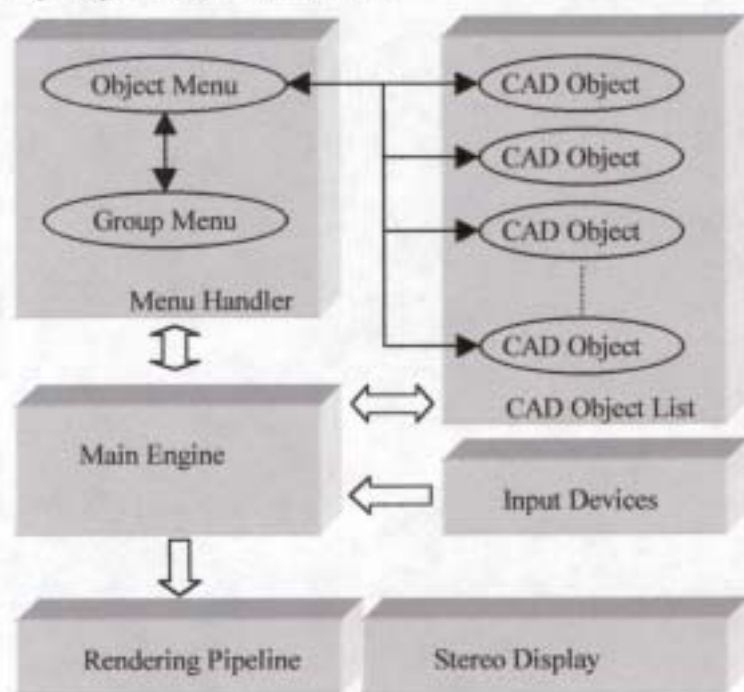


Figure 4: Object oriented application framework

5. CONCLUSIONS

We have described some of the components required for a fully integrated semi-immersive VE. Although only at a preliminary stage, 3DVIS has shown to be very efficient, both for quick sketching and more complicated tasks like NURBS modeling. The current challenge lies in the development of more complex interaction and modeling schemes in the form of new virtual tools and input devices, using voice and gesture (pattern) recognition. One of the core tasks for the near future is to provide the necessary modeling precision required for standard engineering design tasks. The object-oriented framework is easily extendable and provides a user-friendly prototyping environment. An enlarged CAD feature set is currently under

development. Eventually, this will be accompanied by a collection of CAD file converters.



Figure 5: Example of navigation (a) and editing (b)

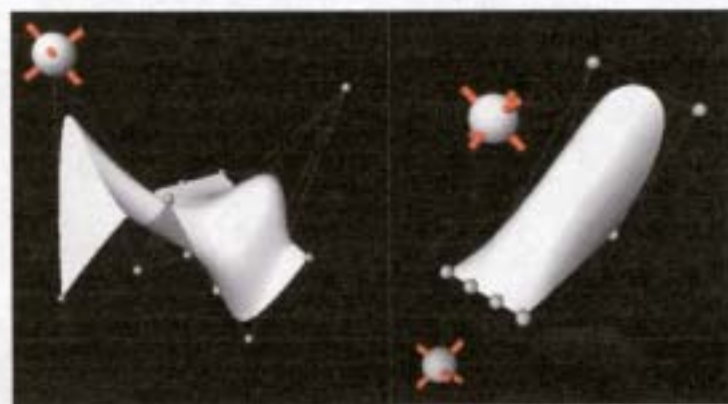


Figure 6: Nurbs surface and data glove proxies

ACKNOWLEDGMENTS

This work was supported by the National Science Foundation under contract ACI 9624034 (CAREER Award) and through the National Partnership for Advanced Computational Infrastructure (NPAC); the Office of Naval Research under contract N00014-97-1-0222; the Army Research Office under contract ARO 36598-MA-RIP; the NASA Ames Research Center through an NRA award under contract NAG2-1216; the Lawrence Livermore