

Autostereoscopic and Haptic Visualization for Space Exploration and Mission Design

Cagatay Basdogan, Mitchell Lum, Jose Salcedo, Edward Chow
*JPL-Virtual Environments Laboratory
California Institute of Technology
Pasadena, CA 91109*

Stephen A. Kupiec, Andrew Kostrewski
*Information Systems Division
Physical Optics Corporation
Torrance, CA 90501*

Abstract

We have developed a multi-modal virtual environment set-up by fusing visual and haptic images through the use of a new autostereoscopic display and a force-feedback haptic device. Most of the earlier visualization systems that integrate stereo vision and touch have utilized polarized or shutter glasses for stereovision. In this paper, we discuss the development stages and components of our set-up that allows a user to touch, feel, and manipulate virtual objects through a haptic device while seeing them in stereo without using any special eyewear. We also discuss the transformations involved in mapping the absolute coordinates of virtual objects into visual and haptic workspaces and the synchronization of cursor movements in these workspaces. Future applications of this work will include a) multi-modal visualization of planetary data and b) planning of space mission operations in virtual environments.

Set-Up

Our set-up is designed to create a multi-modal virtual environment that integrates vision and touch with minimum obstruction to the user. We believe that the next generation of user interfaces for virtual environments will be non-intrusive and more natural. With this aim, we have developed a set-up (see Figure 1) that includes a new projection table for autostereoscopic visualization and a PHANToM haptic device (available from Sensable Technologies Inc) for simulating touch interactions.



Figure 1. Our set-up includes a projection table for stereo visualization of 3D objects without using any special eyewear and a haptic device for force feedback.

Our autostereoscopic display system consists of two LCD projectors (one for each eye) and mirrors housed in a rectangular enclosure, topped with a holographic plate (see Figure 2). Each LCD projector reflects its image off a pair of mirrors, casting the image onto the hologram at the tabletop. Our holographic plate is a typical parallax display and allows each eye to see its corresponding image, but not the image meant for the other eye. In order to render two separate images, a graphics card with a dual ported output was used (one output for each projector). The left and right images projected from the LCD projectors on the holographic display were swapped to create a negative horizontal parallax such that a displayed 3D image appears as if it is floating slightly above the holographic plate (see Figure 2).

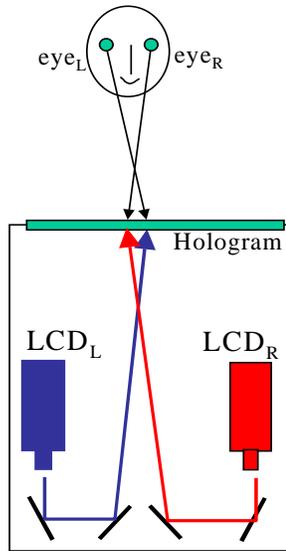


Figure 2. Cross section of the projection table: The images projected by the LCD projectors were swapped at the holographic plate to create a negative horizontal parallax.

Autostereoscopic Visualization

Stereo visualization is not a new problem. Many of us are familiar with the view-master, which displays two images, one to each eye, to create a sense of a three-dimensional scene. Since the apparent depth of objects is a function of the difference in their positions from the left and right eye views, stereo visualization can be achieved by enabling each eye to see only the view created for it. The most well known stereo displays include shuttered or passively polarized eyeglasses which block one of two displayed images from each eye. Shutter-glasses close the left and right eyes of the user in turn 60 times per second or faster. They are synchronized with a computer via an infrared transmitter. When the brain receives the images in rapid enough succession, it fuses them into a single scene and perceives depth. In passive polarized glasses, two slightly offset images are projected through differently polarized filters and visualized through polarized glasses so that correct image is visible to each eye. In general, these devices are light-weighted, cheap, and easy to use. However, they are obtrusive, and viewers usually dislike wearing any invasive equipment. These reasons have recently led to the development of non-invasive approaches such as autostereoscopic visualization systems (Perlin et al., 2000). Autostereoscopic visualization is a relatively new technology that allows the viewer to

see 3D images without the aid of special glasses. Various types of autostereoscopic display systems have been developed during the last decade (see the review in Hale, 1997). Among these, parallax displays are the most common autostereoscopic displays. Parallax displays consist of a surface covered with display elements that can emit light of varying intensity in different directions. Our autostereoscopic display table uses a holographic glass plate to create a parallax effect.

Haptic Displays and Rendering

Haptic rendering enables the user to touch, feel, and manipulate objects in virtual environments through a haptic device. Various haptic rendering techniques have been developed during the last few years (see the review in Basdogan and Srinivasan, 2001). In our simulations, we used the point-based rendering technique developed in our earlier studies (Ho et al., 1999).

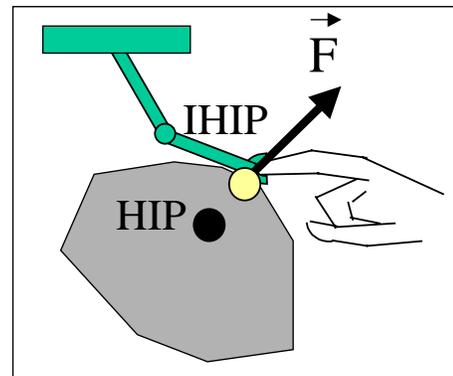


Figure 3. Point-based haptic rendering: the Haptic Interface point (HIP) coordinates are digitized through the position encoders. HIP can penetrate into a virtual object while the Ideal Haptic Interface Point (IHIP) is constrained to stay on the surface.

In point-based haptic rendering, the tip point of the end-effector (Haptic Interface Point - HIP) is digitized via encoders and used to detect collisions with virtual objects as the user manipulates the haptic device. Once a collision is detected, the contact point (Ideal Haptic Interface Point - IHIP) is determined and the penetration vector is calculated as the difference between the IHIP and HIP coordinates. Then, reaction force is calculated based on the depth of penetration reflected to a user through a haptic device (see Figure 3).

Autostereoscopic Rendering

Creating stereo images can be achieved in two parts: first, we create views for the left and right eye, which leads to the rendering of each eye. Then, we deal with how each rendering is displayed to a user in order to create the desired stereoscopic effect (see TGS Open Inventor notes on stereo displays as well as Grinberg and Siegel, 1994). In order to create two slightly different views of the same image, we utilize two perspective cameras, one for each eye. If the view volume of a perspective camera is represented by a pyramid (also known as the frustum) whose apex is the camera lens, then the further from the camera the more points will be projected, creating the perspective effect (see Figure 4a). To create a stereo effect, we take advantage of the asymmetric frustum concept (see Figure 4b).

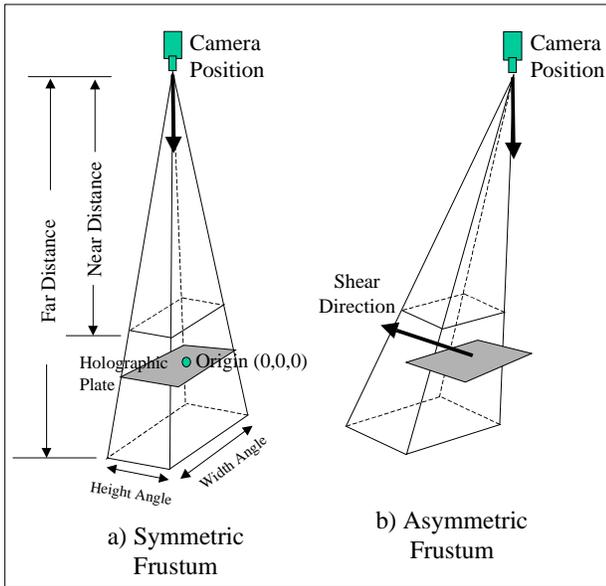


Figure 4. a) Symmetric and b) asymmetric (skewed) frustums of a perspective camera.

An asymmetric frustum is obtained by “skewing” the view frustum of a camera such that the view volume still contains the object and the viewing direction is kept unchanged. To achieve this effect, we first create two perspective cameras positioned above the holographic glass plate (see Figure 5: the cameras are placed at the eye level, separated by the distance of left and right eye, and point towards the plate) and then, apply a shear transform to the objects displayed through these cameras. One can think that placing

perspective cameras at the eye location and orienting them such that they point to the center of the object could also create a stereo effect. However, this approach has a couple of problems: a) the point of focus has to be updated as the objects move in the scene and b) the orientation of one view will be different than the orientation of the other view.

Shear Transformation

It is necessary to describe necessary perspectives for stereo imaging in terms of the combination of the monoscopic perspective and a shear transformation that skews the camera’s view frustum. The general form of the shear matrix is:

$$S = \begin{bmatrix} I + \tan(\varphi)(v^T w) & 0 \\ -(Q \cdot v)w & 1 \end{bmatrix} \quad (1)$$

where, Q is a point on the shear plane, v is a unit vector perpendicular to the shear plane, w is a unit vector in the direction of the shear and perpendicular to v (i.e. $w \cdot v = 0$) and φ is the angle of the shear (see Figure 5). We wish to obtain a shear transformation which shifts the point of perspective from $[0 \ 0 \ r_z]$ to $[r_x \ r_y \ r_z]$ while not shifting the x-y plane at $z = 0$. This is obtained by using the following values (see Graphics Gems II):

$$Q = [0 \ 0 \ 0] \quad (2)$$

$$v = [0 \ 0 \ 1] \quad (3)$$

$$w = \begin{bmatrix} \frac{r_x}{\sqrt{r_x^2 + r_y^2}} & -\frac{r_y}{\sqrt{r_x^2 + r_y^2}} & 0 \end{bmatrix} \quad (4)$$

$$\tan(\varphi) = \frac{\sqrt{r_x^2 + r_y^2}}{r_z} \quad (5)$$

where, $[r_x \ r_y \ r_z]$ is the distance of eyes from the table (see Figure 5; note that r_y has a negative value if the coordinate frame is defined as in the figure and the r_x value is different for each eye which results in

a different shear vector and transform for each camera)

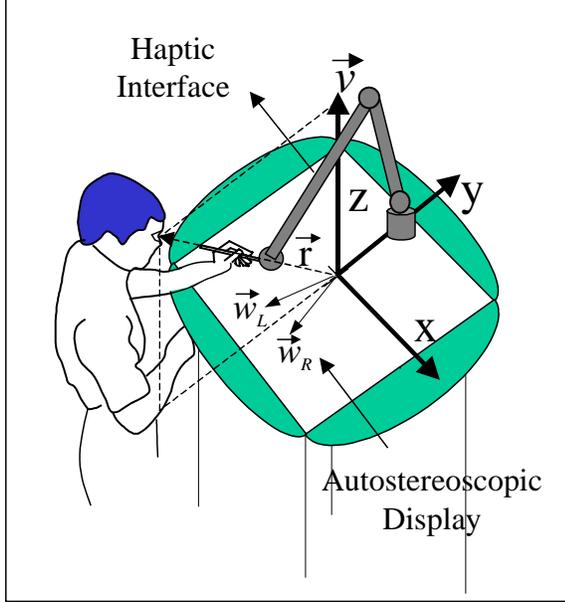


Figure 5. The coordinate frame and the relative distances used in the computation of shear matrix.

This results in the following matrix:

$$\mathbf{S}_{camera} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \frac{r_x}{r_z} & -\frac{r_y}{r_z} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6)$$

If we begin with a standard perspective camera at $l = r_z$:

$$\mathbf{P}_{monoscopic}(l = r_z) = \begin{bmatrix} r_z & 0 & 0 & 0 \\ 0 & r_z & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & r_z \end{bmatrix} \quad (7)$$

If we first apply the shear transformation and then the projection matrix, we obtain the sheared projection matrix as

$$\mathbf{P}_{shear} = \begin{bmatrix} r_z & 0 & 0 & 0 \\ 0 & r_z & 0 & 0 \\ r_x & -r_y & 0 & -1 \\ 0 & 0 & 0 & r_z \end{bmatrix} \quad (8)$$

In order to implement this concept using Open Inventor Graphics Toolkit, we first construct two perspective cameras located at $[0 \ 0 \ r_z]$ and pointing to the origin. Then, we apply the shear transform associated with the left and right eye locations to the identical copies of the object. This process generates two slightly different and skewed views of the same object. Finally, we display the sheared copies of the same object in two separate viewports. The viewports were constructed by dividing the display window into two equal halves. A different view of the same object was displayed in each viewport using a separate perspective camera.

Mapping Between Visual and Haptic Workspaces

In our set-up, the visual and haptic workspaces defined by the table and the haptic device are unequal and limited by size. Hence, one has to map the absolute coordinates of a 3D object to the visual and haptic workspaces such that the displayed object fits into the view volume and is large enough to be explored with the haptic device.

The visual transformation of an object is trivial with a standard rendering package. For example, in order to make a virtual object smaller in the scene, one can either multiplying its visual coordinates by a scale factor or move the perspective camera away from the scene along the view direction to create the scale effect. However, the positions of perspective cameras used in our autostereoscopic rendering are fixed and cannot be moved (i.e. recall that each camera is located at the eye level and looking down). Moving the camera up or down along the viewing direction would change the shear transform matrix (inspect Eq. 6) and ruin the stereo effect. Moreover, the aspect ratios of our cameras (the ratio of height angle to width angle of the viewing frustum shown in Figure 4) have to match with the aspect ratio (the ratio of length to width) of the holographic plate. Hence, the camera properties cannot be altered much to fit our virtual objects into the view frustum. In addition to these visualization constraints, the haptic device used in our simulations has a limited

workspace. We cannot *directly* feel the objects that are out of our reach or objects that are too small or too big. Hence, the problem is to map the absolute coordinates of a 3D virtual object into haptic and visual workspaces such that it fits into the view frustum and we can fully explore its shape through a haptic device.

For the sake of discussion, let's assume that we have a large 3D model of a space shuttle (see Figure 6). Since our visual and haptic coordinates are stored separately in the database, one approach is to scale the visual size of the shuttle (by multiplying its visual coordinates with a transformation matrix) such that it fits into the view volume of the stereoscopic display. However, the same transformation matrix may not be used to scale the haptic coordinates of the shuttle if it shrinks to a size that is too small to feel through a haptic device (see Figure 6; if T_1 and T_3 are equal, the object may be too small for haptic exploration). Since the size of the haptic workspace is available from the manufacturer of the device, we can estimate the transformation matrix T_1 (for example, the scale factor can be estimated based on the ratio of the haptic workspace to the bounding box of the object). Let's assume that we have transformed our object to the haptic workspace such that we can feel it through a haptic device. Now, we would like to transform our shuttle to visual workspace such that the skewed frustum contains the visual model of the shuttle and it is fully visible to the user standing in front of the table. Here the transformation (T_2) must be done carefully for two reasons; First, we would like to maintain the haptic coordinates of the object while we change its visual coordinates. If the movements of the cursor are not synchronized, this creates a visual-haptic mismatch since the haptic and visual coordinates of the shuttle will be different after the transformation. Second, the transformation matrix (T_2) cannot include any translation along the viewing axis of the perspective cameras (Z-axis in Figure 5). The camera position for each eye is fixed and moving cameras or objects in the scene will damage the shear transform and the stereo effect. A solution to these problems can be obtained if cursor movements of the haptic device are synchronized in visual and haptic workspaces and the proper transformations are applied. This solution that

maintains the haptic size of the object while changing its visual size and can be implemented in three consecutive steps:

1. Multiply the visual coordinates of the object with T_2 such that it fits into the skewed frustum of the cameras (keep the haptic coordinates unchanged).
2. Multiply the HIP coordinates of the haptic device with T_2 at each cycle of the haptic loop to synchronize the visual cursor movements with the haptic one. Also multiply the HIP coordinates separately with the inverse of the transformation matrix ($HIP_r = HIP \cdot T_2^{-1}$) for the purposes of collision detection.
3. Detect collisions between the HIP_r and the 3D object and calculate the reaction force at each cycle of the haptic loop. Multiply the force vector with T_2 and reflect it to the user through the device.

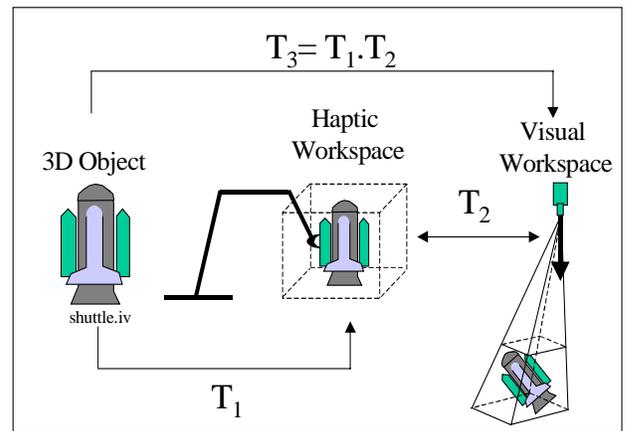


Figure 6. Transformations that take place from object space to visual workspace: first, the 3D object is transformed to haptic workspace such that it can be explored via a haptic device. Then, it is transformed to the visual workspace such that the asymmetric viewing frustum contains the object.

Discussion and Future Development

We have developed a multi-modal visualization system that enables the user to see a virtual object in stereo without using any special eyewear while touching and feeling it through a haptic device. Although the current version of our visualization setup can display 3D models in stereo to a single person, Physical Optics Corporation are currently working on a new display system that will allow the

participation of multiple people also known as auto-multiscopic 3D display. The system will generate four viewing zones, creating a 60 degrees stereo view for each zone. Each viewing zone will be created by one hologram and one set of projectors (see Figure 7).

We opted to use a Phantom (Sensable Tech. Inc.) haptic display to simulate touch interactions in virtual environments. Among the available devices in the market today, Phantom is relatively less intrusive than other devices. However, it provides a limited haptic workspace and the touch interactions with virtual objects are through an end-effector of a robotic arm. In our set-up, the user inserts his/her finger into the thimble-gimbal attachment to interact with virtual objects. We believe that there is a need for less intrusive haptic displays that will make the interface more transparent to the user.

We also proposed a simple but an efficient technique for mapping the absolute coordinate of a virtual object into visual and haptic workspaces that are unequal and have limited size. If the mapping is not one to one, a visual-haptic mismatch occurs. Our mapping technique maintains the haptic coordinates of the object while the visual orientation and size of the object displayed to the user can be changed. In fact, we observed that alternative implementations of this mapping could alter our perception of object shapes in certain conditions, which will be explored in the future.

We plan to explore the applications of the developed set-up in several areas including space science, computer-aided mechanical design, and medical visualization. In planetary exploration for example, we anticipate that our set-up can be useful in two ways: a) multi-modal display of planetary data to scientists such as displaying Mars rocks and soil; b) planning of space mission operations which may involve landing site selection and generating navigation plans for a planetary rover. We envision that mission operators and scientists can stand around the autostereoscopic table and interact with virtual objects and collaborate with each other through haptic displays (see Figure 7) to develop more robust mission plans.

References

1. TGS Open Inventor Documentation, Chapter 7: Stereo Viewing.
2. Perlin, K., Paxia, S., Kollin, J.S., 2000, "An Autostereoscopic Display", SIGGRAPH 2000 Conference Proceedings, July-23-28, (<http://mrl.nyu.edu/projects/autostereo/>).
3. Halle, M., 1997, "Autostereoscopic displays and computer graphics", SIGGRAPH 1997, Vol. 31, No. 2, pp. 58-62.
4. Grinberg, V.S., Siegel, M., 1994, "Geometry of Binocular Imaging", Proceedings of the SPIE Conference on Electronic Imaging, Stereoscopic Displays and Virtual Reality Systems, Vol. 2177, pp. 56 - 65. (http://www.ri.cmu.edu/pubs/pub_1908.html#ext_ref)
5. *Graphics Gems II*, 1991, pp. 338-341, Academic Press, Boston, Ed: J. Arvo.
6. Basdogan, C., Srinivasan, M.A., 2001, "Haptic Rendering In Virtual Environments" to appear in "Virtual Environments Handbook" (<http://eis.jpl.nasa.gov/~basdogan>)
7. Ho, C., Basdogan, C., Srinivasan, M.A., 1999, "An Efficient Haptic Rendering Technique for Displaying 3D Polyhedral Objects and Their Surface Details in Virtual Environments", October'99 Vol. 8, No. 5, pp. 477-491, *Presence*.



Figure 7. The future versions of our system will enable the participation of multiple users.