Z.Y. Alpaslan and A.A. Sawchuk, "Three-Dimensional Interaction with Autostereoscopic Displays," Proc. Stereoscopic Displays and Virtual Reality Systems XI Symposium, Proc. SPIE, Vol. 5291, San Jose, CA, 2004.

Three-Dimensional Interaction with Autostereoscopic Displays

Zahir Y. Alpaslan and Alexander A. Sawchuk[1,2]
Integrated Media Systems Center, University of Southern California, Mail Code 2564
Los Angeles, CA 90089-2564.

ABSTRACT

We describe new techniques for interactive input and manipulation of three-dimensional data using a motion tracking system combined with an autostereoscopic display. Users interact with the system by means of video cameras that track a light source or a user's hand motions in space. We process this 3D tracking data with OpenGL to create or manipulate objects in virtual space. We then synthesize two to nine images as seen by virtual cameras observing the objects and interlace them to drive the autostereoscopic display. The light source is tracked within a separate interaction space, so that users interact with images appearing both inside and outside the display. With some displays that use nine images inside a viewing zone (such as the SG 202 autostereoscopic display from StereoGraphics), user head tracking is not necessary because there is a built-in left right look-around capability. With such multi-view autostereoscopic displays, more than one user can see the interaction at the same time and more than one person can interact with the display.

Keywords: Autostereoscopic displays, interactive displays, video tracking, 3-D display, stereo display.

1. INTRODUCTION

Recently, stereoscopic systems have gained popularity due to increases in the processing power of computers and the reduced price of high quality displays. With the emergence of autostereoscopic displays it has been possible to view images, movies and other kinds of data in 3D space without wearing glasses. Several researchers have investigated stereoscopic viewing technology for better 3D data visualization and manipulation, and work on stereoscopic imaging has been increasing.

Autostereoscopic (AS) displays make the stereo experience more pleasant by removing the necessity of using glasses. Table 1 summarizes the two main types of current commercially available AS displays along with their image quality and applications. Lenticular based AS displays [1-4] have higher brightness and are better for multi-view (more than two displayed images at the same time) and multi-viewer (more than one person can see the effect) applications. However, because the slanted lenticular screen is used to display multiple views the image resolution of each view is reduced horizontally and vertically, making it difficult to read displayed small text. Barrier screen based systems [4,5] have lower brightness and are more appropriate for single viewer and single view situations. However, their only reduction in resolution is in the horizontal dimension.

AS Display Type	Resolution	Text	Image quality	Depth Effect	Application	Cost
Lenticular (StereoGraphics, DDD etc.)	Reduced to 1/3 both in horizontal and vertical	Small text unreadable	Image edges are blurred	Movement parallax and image parallax	Multi view and multi viewer	$4,000 - $18,000
Barrier (Sharp)	Reduced to half only in horizontal	Readable	Sharp clean edges	Only image parallax	Single view, single viewer	$3,300

Table 1. Differences between lenticular and barrier technologies in two commercially available AS displays.

[1] Author emails: {Alpaslan, Sawchuk}@sipi.usc.edu

[2] Phone: (213) 740-4622

The earliest reference to interaction with an autostereoscopic display is by DeWitt [6]. He refers to the possibility of interaction but describes no details to our knowledge. In 2000, the MUltimo3D group at the Heinrich-Hertz-Institute in Germany built a special autostereoscopic display [7,8]. The system consists of a gaze tracker, a head tracker, a hand tracker and an autostereoscopic display for viewing and manipulating objects in 3D. The head tracker makes the images appear in a fixed position so the user has a look-around capability (being able to see around the sides of an image). Gaze tracking activates different applications in the desktop, and the hand tracker navigates and manipulates objects in space rather than a mouse. The Multimo3D hand tracking system is accurate to approximately 2-3 cm. More recently, Berkel at the Philips Labs [9] has shown that it is possible to track a user's hand and fingers with magnetic fields using sensing electrodes around the edges of a display. They use this information to interact with an autostereoscopic display.

There are also commercial companies that combine haptics and stereoscopic viewing technology [10]. These highly accurate systems supply 3D navigation and manipulation data using haptic devices instead of a mouse. For haptic interaction the reader can refer to [11] and references contained therein.

2. APPROACH

2.1. System Overview

The basic block diagram of our system is pictured in Fig. 1. Our current focus is on the unshaded blocks of Fig. 1. Our system consists of acquisition, display, interaction and tracking parts. Starting from the interaction volume on the lower right side, two cameras image a light source cursor held by the user. A tracking algorithm analyzes the images and extracts the position information. This information is combined with graphical models to create a scene with virtual objects. Virtual cameras produce synthesized views of the virtual scene while the graphics card interlaces up to nine views that are sent to the display. When we put these parts together they form a closed loop as seen in Fig. 1. Figure 2 shows details of the two cameras tracking a light source (small flashlight) used as a cursor in the interaction volume.

The acquisition and display parts together are a 3D drawing and image manipulation program based on OpenGL and Visual C++. The acquisition part accepts drawing commands in the form of 3D (x, y, z) coordinates and it draws 3D virtual objects in virtual space using these coordinates. The display part produces nine views of the virtual object scene created in the drawing part by shifting a virtual camera horizontally, and interlaces these nine views at the sub-pixel level to create the final interlaced image to be displayed on the autostereoscopic display.

The interaction and tracking parts work like a 3D mouse. To interact with the system a user moves a small hand-held light source in the interaction volume defined by the field of view of two FireWire cameras. A tracking algorithm analyzes this live video frame-by-frame and finds the coordinate of the light source's center in the frames. When the cameras are perpendicular, simple combination of two axes from one camera and the third axis from the other camera gives the 3D coordinates of the light source in camera pixel coordinates. In general, the cameras do not have to be perpendicular.

In our current system we focus on implementing the interaction and display parts in separate physical volumes, although they could be overlapped and placed in register. Thus we do not have to track the user hand movements right in front of the display where the images appear; instead we are free to place the cameras anywhere we want. Another advantage of this implementation is the increased interaction volume. Removing the restriction of interaction with the objects appearing in front of the display plane allows us to interact with objects appearing behind the display plane. In this case interaction volume at least doubles in size. Implementation of not-in-register interaction requires the user to interact with the virtual objects with the help of a cursor that represents the light source in virtual space.

A cursor also avoids the accommodation vergence conflict of the human observer. To perceive a 3D sensation, a user's eyes have to focus on the display plane. However, if the user wants to do in-register interaction and interact without the help of a cursor such as [7, 8], he needs to focus on his hand and the display plane at the same time. Since the hand cannot touch the display plane, this forces the user to focus on two different planes, causing eyestrain. One solution to this problem uses active display systems with moving lenses in front of the display plane. While we do not have the ability to move the lenticular sheet in our display, if we decide to add in-register interaction we may modify the display

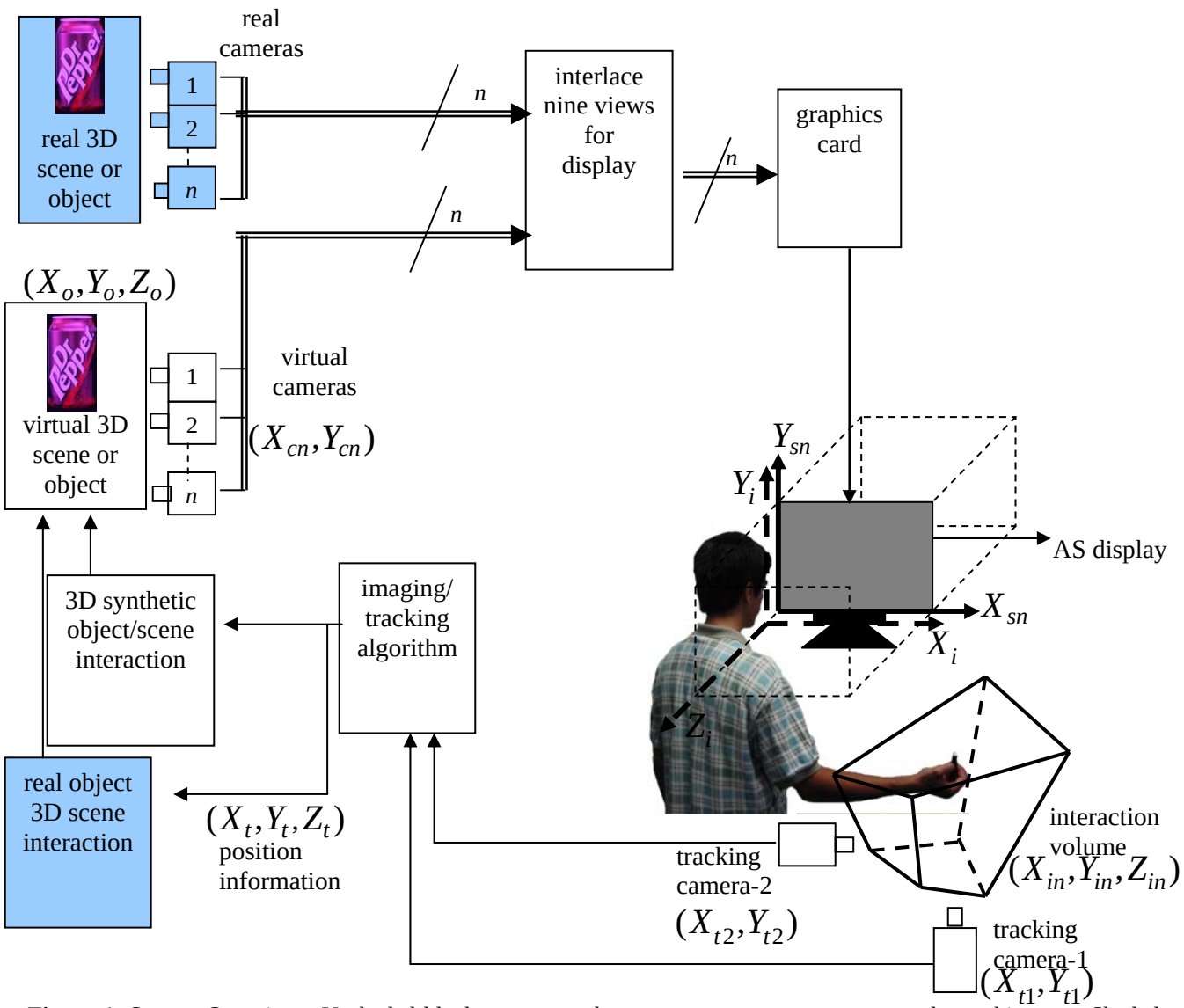

Figure 1. System Overview. Unshaded blocks represent the two components we are currently working on. Shaded blocks are the future system components. (X_o, Y_o, Z_o) are the coordinates of the virtual 3D scene. (X_{cn}, Y_{cn}) are the coordinates of the virtual 3D scene as measured by the detector array of camera *n*. Our system uses from two to nine virtual cameras in order to create the sensation of a 3D display. (X_{sn}, Y_{sn}) are the coordinates of view *n* at the AS display surface. (X_i, Y_i, Z_i) are the coordinates of the displayed points as perceived by a human observer. These coordinates depend on viewing distance, eye separation and many other parameters. (X_{in}, Y_{in}, Z_{in}) are the coordinates within the effective interaction volume. A cursor or user's hand must be in this volume for both cameras to track its position. (X_{t_1}, Y_{t_1}) and (X_{t_2}, Y_{t_2}) are the coordinates of the objects in the interaction volume as measured by the detector array of cameras 1 and 2 respectively. We process these coordinates to create (X_t, Y_t, Z_t), the 3D location of an object in the interaction volume as seen by both cameras.

for a moving lenticular sheet following a similar method to [7, 8, 12-14]. Another possibility is to add a virtual lens in software similar to [15].

Another future application of our system involves the shaded blocks in Fig. 1. By adding multiple camera real-time video capture, it is possible to interact with 3D video of live scenes.

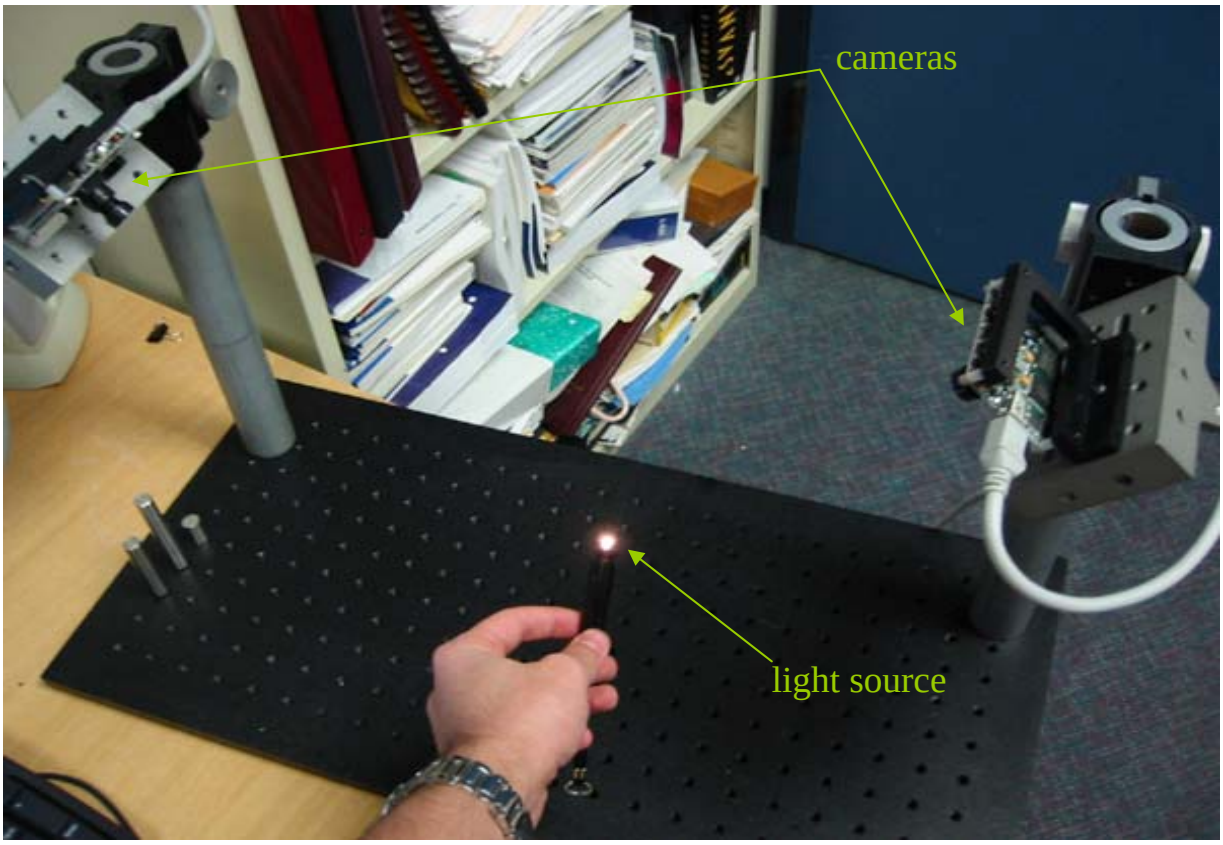

Figure 2. Two cameras tracking a light source in the interaction volume.

In order to find the location of the light source in a frame we used brightness thresholding. We adjusted the threshold so that the light source appears as a small point. Since the light source is very bright we can reduce the camera gain and still get a very good image of the light source. The software finds the location of the brightest pixels in the image for every frame. Because of the thresholding nearly all pixels are black except at the light source image, therefore finding the white pixels is very accurate. Next, the software does simple averaging to find the center of mass of the whitest area and this center is the location of the light source in one camera. The software repeats the same operation for the frame acquired by the other camera. Finding the 3D location of the light source in pixel coordinates requires taking two (such as z and y) coordinates from one camera and the remaining (x) coordinate from the other camera after applying rotation to the camera axes in software. If cameras are perfectly perpendicular, one coordinate (such as z) acquired from both cameras are the same.

2.2. Interaction Volume

Figure 3 shows details of the interaction volume and cameras. Currently, our cameras can record video at a 640x480 resolution with 30 fps frame rate. Thus they can track the light source with an accuracy of 640 pixels in two dimensions and an accuracy of 480 pixels in the third dimension. Since the light source cannot go inside the monitor, the user interaction volume is the region of space where the FOVs (field of view) of both cameras intersect.

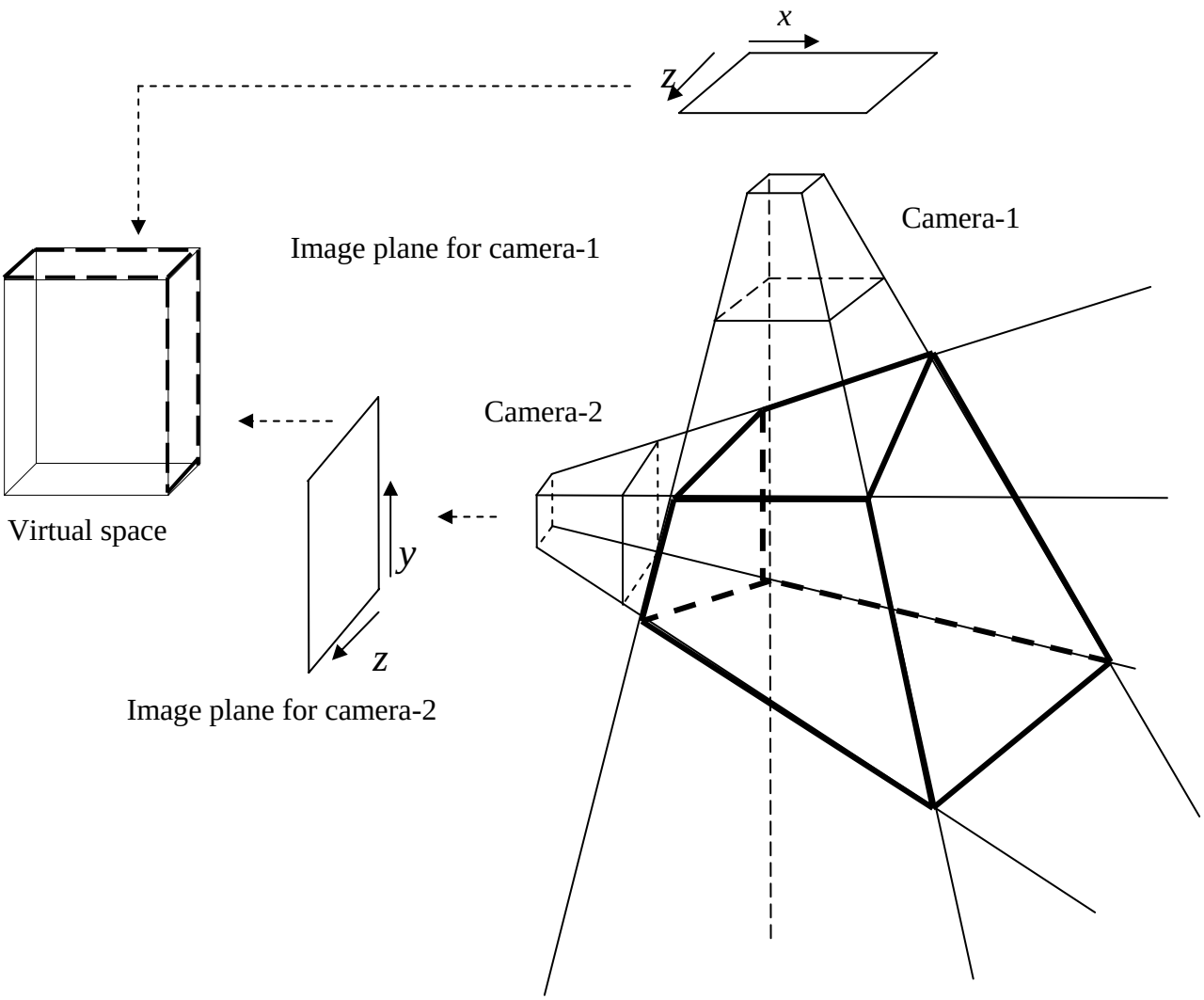

Figure 3. The relationship of the interaction volume to the AS display space.

2.3. Drawing and Navigation

Cameras track the light source movements in the interaction volume. A pointer in the virtual space mimics the motion of the light source to produce a 3D shape consisting of points or lines. The program converts the tracking information coming from the cameras to horizontal, vertical and depth (parallax) coordinates for the display. This information is processed and supplied as the necessary nine images for the AS display. When the software operates in moving mode, it uses the 3D coordinate information coming from the tracking mode to translate the position of the virtual objects.

2.4. Interlacing

Interlacing is the process of multiplexing multiple images onto a display covered by a lenticular sheet. When the lenticular sheet is vertical, interlacing is done by cutting each input image into thin vertical slices and placing them side by side horizontally so that exactly one slice from each input image is under a lenticula. Figure 4 shows the interlacing for a vertical lenticular sheet. Interlacing for vertical lenticular sheets creates an image called parallax panoramagram. The width S_{pp} of a parallax panoramagram is the sum of the width S_{image} of all the images that contribute to it as given by

$$S_{pp} = \sum S_{image} \cdot \tag{2.3}$$

Assuming all the input images have the same size, then

$$S_{pp} = S_{image} N \tag{2.4}$$

where N is total number of images interlaced.

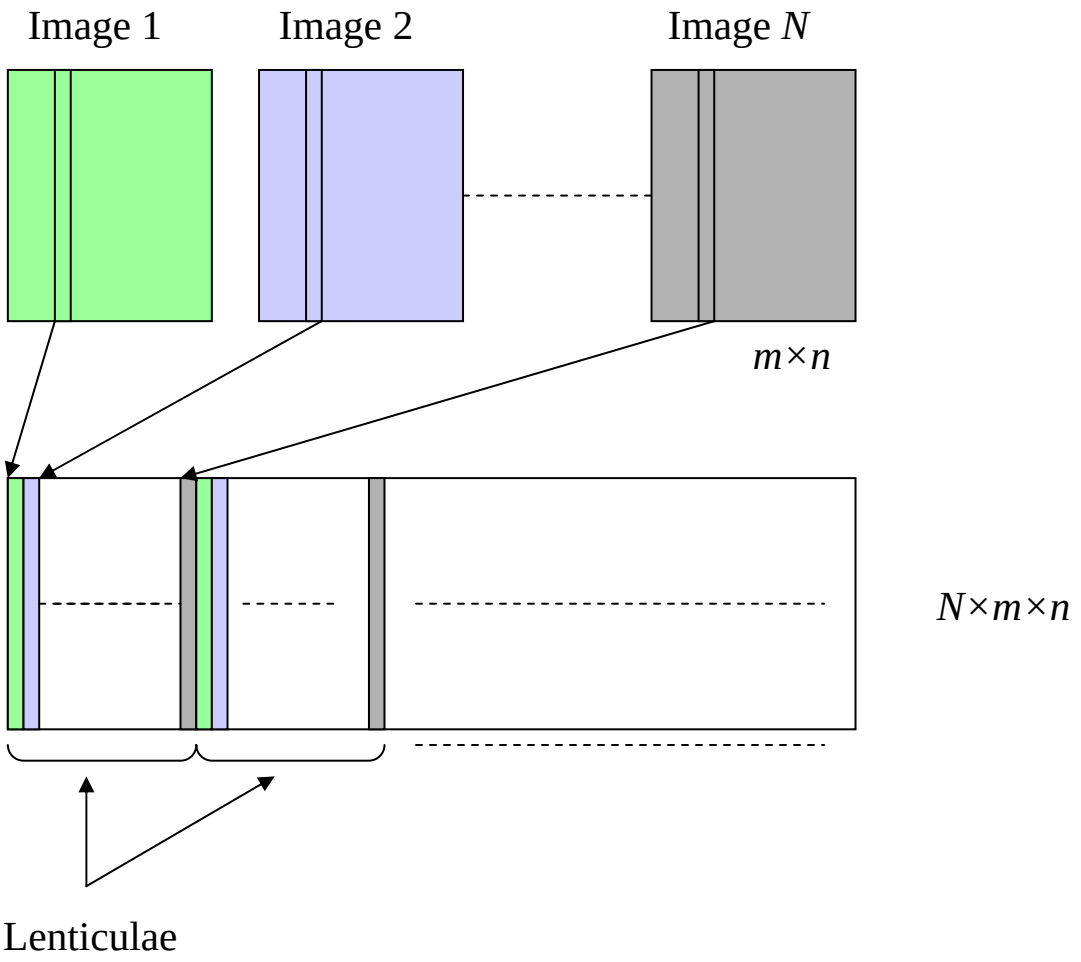

Figure 4. This image shows the interlacing of N images to be displayed under a vertically oriented lenticular sheet. Each group of N slices fits exactly under a lenticula. The total number of pixels is N times the number of pixels of one image. The final image is called a parallax panoramagram.

Lenticular sheet interlacing is done in a different way for LCD panels. It is a well-known fact that vertical lenticular LCDs suffer from a Moiré-like vertical dark band effect [16]. Slanting the lenticular sheet with respect to the vertical at a small angle removes the dark bands in the displayed image and makes the transition between views appear more continuous. However, both the horizontal and vertical effective resolution of each view is reduced because nine images are multiplexed onto the fixed resolution of the LCD display.

Our display has a basic resolution of 1600x1200, with each pixel composed of red, green and blue sub-pixels. The pitch of the slanted lenticular sheet is less than that used for vertically oriented sheets. Because the sub-pixels of an LCD panel are rectangular, it is very difficult to have an exact register of sub-pixels and the slanted lenticulae. Van Berkel

has worked on the LCD interlacing process [16]. We are currently using the interlacing process provided to us by StereoGraphics and DDD [1-4]. These interlacing algorithms use a similar approach to van Berkel and we are also developing our own interlacing algorithm following the same method. Figure 5 shows an example of sub-pixel multiplexing required for a lenticular AS display.

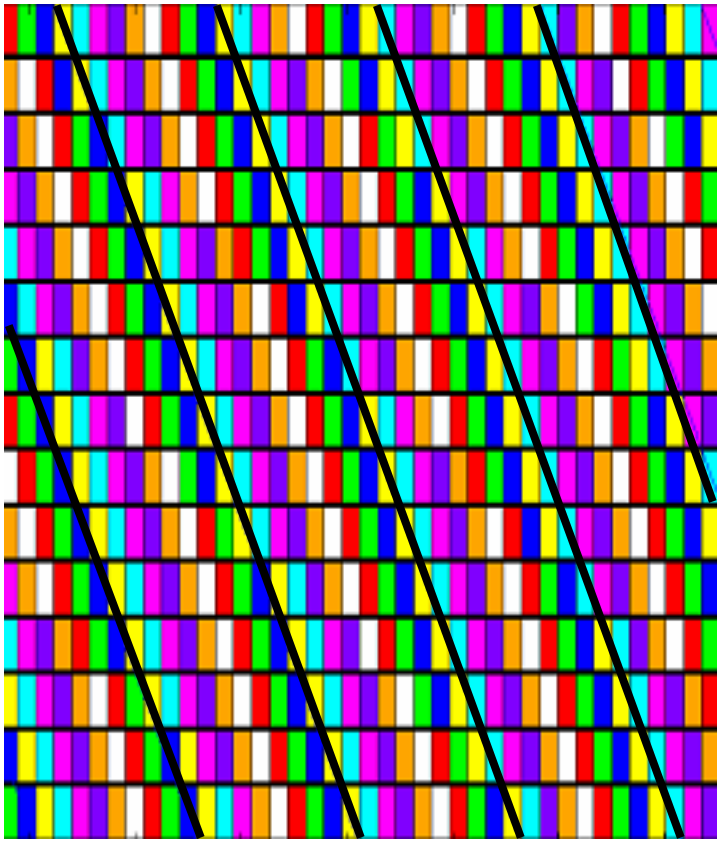

Figure 5. The sub-pixel multiplexing arrangement under several lenticulae of a slanted lenticular sheet. Small rectangles are the red, green and blue sub-pixels of the display. Nine different gray levels represent the interlace patterns for the nine image views as they are mapped on to the display and bold black lines represent the edges of the lenticulae.

2.5. Mathematical Model for Autostereoscopic Interaction

We have created a mathematical model that combines the mathematical descriptions of all the pieces in the system description as shown in Fig. 1. Our mathematical model for interaction is based on [17-19] and relates the 3D image coordinates (X_i, Y_i, Z_i) to interaction space coordinates (X_{in}, Y_{in}, Z_{in}). Because of space limitations it will be described in another publication.

2.6. Comparison to Other Approaches

Our approach is similar to the work done by Multimo3D group. However, we have different advantages and challenges associated with a multi-zone and multi-view display. Using a two-camera configuration, it is possible to reduce dramatically the computation and processing time needed for tracking light sources used as a 3D cursor. We are not

worried about tracking the head for stabilizing the image. Our AS display has nine images inside a viewing zone, therefore it already has a built-in look-around capability. Our interaction algorithm allows a user to interact with images both appearing inside the display and in front of the display. The Multimo3D system allows only interaction with images appearing in front of the display. Since we are using a multi-user AS display, more than one user can see the interaction at the same time and more than one person can interact with the display (by tracking two or more light sources). Connecting two systems via Internet, it is possible to have 3D interaction. For example, one user's drawing can appear and float in space in front of another user at a remote location. The Multimo3D system does not need any handheld objects for help in tracking, therefore gives user more freedom. Our ultimate goal is to have multiple users interact with our AS desktop system at the same time.

3. RESULTS

Our current system tracks the motion of a flashlight cursor running at 15 fps and draws an image with 640x480 resolution in real-time. The current main functions of the system are move, draw and save.

The move function translates an object's location in stereo space coordinate (X_i, Y_i, Z_i). Figure 6 shows the picture of a cube moving in AS display space.

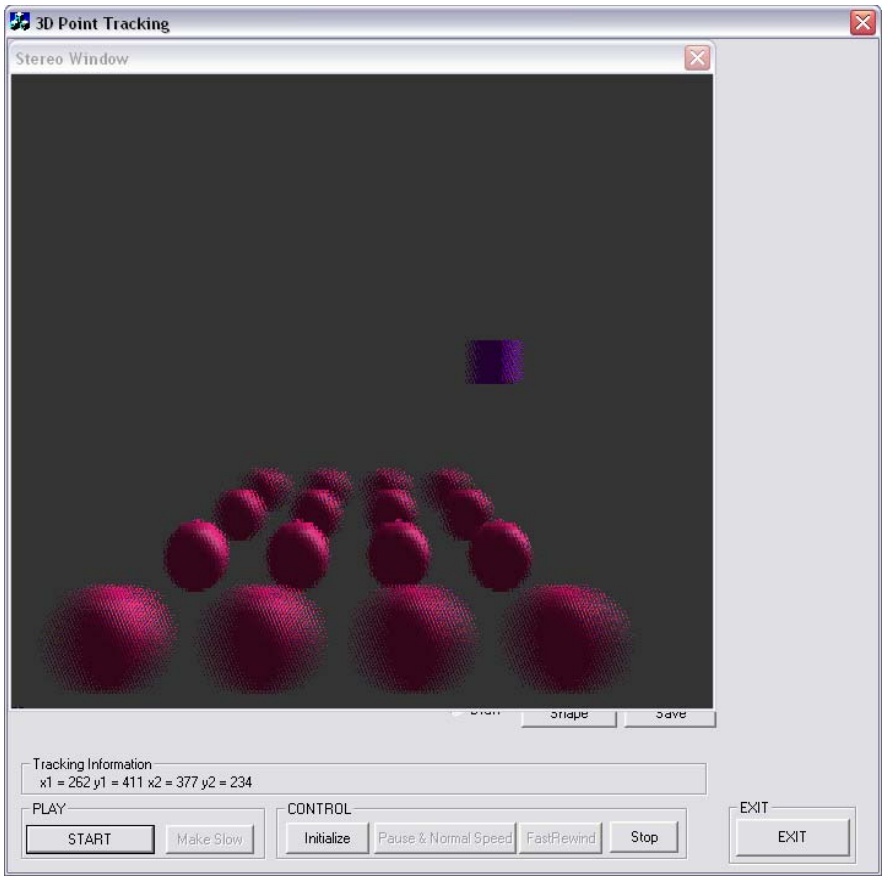

Figure 6. Demonstration of move function. In this screen shot of a monoscopic display, a 4x4 array of spheres serve as depth level indicators in the stereo window. The single cube moves in stereo space according to the user commands. The fuzziness of the displayed spheres comes from the interlacing algorithm. When viewed with a lenticular display the row with the largest spheres appears in front of the display surface, the second row of spheres appears at the display surface and the rows with the smallest spheres appear behind the display surface.

The draw function creates a solid object. In order to save processing time we implemented it as a wire-frame drawing tool. The user first creates a wire-frame object made of dots. When the user finishes drawing the wire-frame object he can stop the tracking and make the drawing a solid object by using a basic shape such as a sphere as shown in Fig. 7.

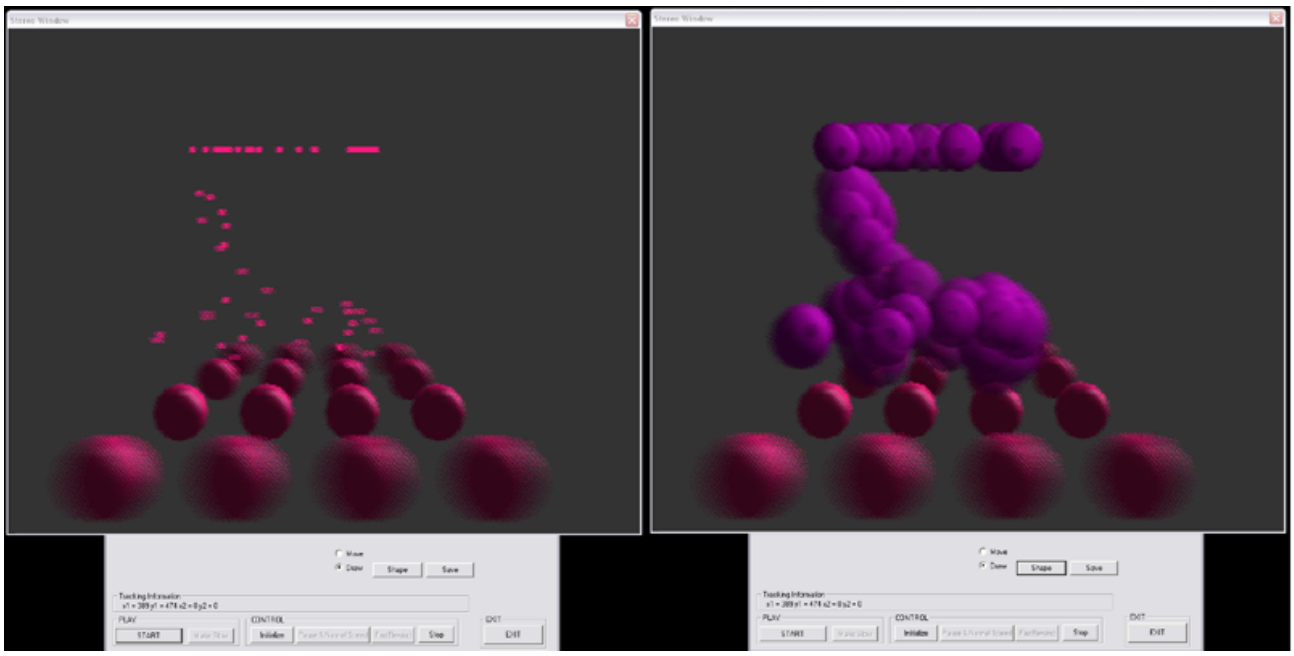

Figure 7. Demonstration of draw function. The figure on the left shows the wire-frame version of the object drawn by user. When the user finishes the wire-frame version he can connect the dots by using a basic three-dimensional figure such as a sphere.

We save the figures as text files. Instead of saving the whole image we save the 3D coordinates the program uses for creating and transforming an object. The file size for an image such as the one in Fig. 7. is just a few KBs.

We also implemented a rotation tool that can rotate the object by clicking buttons on the desktop using a mouse. In the future the rotation and other manipulation functions will be moved to stereo space.

4. CONCLUSIONS

We have implemented a system for 3D human-computer interaction based on a multi-view autostereoscopic display. The system currently tracks a user held light source and uses it like a pen to create and manipulate three-dimensional objects in stereo space.

In the future we are planning extensive user testing. We will investigate how our system changes the human implementation of tasks using the display, the effectiveness of the user interface and the effects of accommodation and vergence conflict for in-register implementation.

This system can be improved with the addition of 3D binaural or 5.1-channel surround sound. Research shows that 3D audio can enhance the stereoscopic experience [20, 21]. We are building our software in a modular structure so that adding new parts is easy to do. With additional programming we can modify this software for use in virtual sculpting and remote virtual interaction. Later our system can be extended to using hand gesturing instead of a cursor. Because of the increase in the speed of USB 2.0 it is also possible to use USB-based cheaper web-cams instead of FireWire cameras to track the user input as in [22]. We view this work as a foundation toward studying the larger problem of 3D operating system and 3D computers.

ACKNOWLEDGMENTS

The authors would like to thank to Isaac Cohen and Sung Lee from the USC Computer Science Department for helpful discussions about hand tracking research and FireWire cameras. We also thank Julien Flack from DDD for helpful discussions about software available for lenticular AS displays.

REFERENCES

1. StereoGraphics Corporation, www.stereographics.com
2. StereoGraphics, "The SynthaGram Handbook", http//www.stereographics.com/products/synthagram/The%20SynthaGram%20Handbook%20v71.pdf
3. L. Lipton, M. Feldman, "A new autostereoscopic display technology The SynthaGram™", *Proceedings of SPIE Vol. 4660, Stereoscopic Displays and Virtual Reality Systems IX*, 2002.
4. DDD Group Plc., www.ddd.com
5. Sharp Systems of America, http://www.sharp3d.com/
6. T. DeWitt, "Visual Music Searching for an Aesthetic", *Leonardo*, Vol. 20, No. 2, pp. 15-122, 1987.
7. J. Liu, S. Pastoor, K. Seifert, J. Hurtienne, "Three dimensional PC toward novel forms of human-computer interaction", *Three-Dimensional Video and Display Devices and Systems SPIE CR76*, 5-8 Nov. 2000 Boston, MA USA.
8. Heinrich-Hertz-Institute, mUltimo3D group website http://imwww.hhi.de/blick/
9. C. van Berkel, "Touchless Display Interaction", *SID 02 DIGEST*, 2002.
10. InterSense, Inc., "Tracking products website", http//www.isense.com/products/
11. J. Brederson, M. Ikits, C. Johnson, C. Hansen and J. Hollerbach, "The Visual Haptic Workbench", *Proceedings of the Fifth PHANToM Users Group Workshop*, 2000.
12. H. Kakeya, "Autostereoscopic 3-D Workbench", http//www2.crl.go.jp/jt/jt321/kake/auto3d.html, Multimodal Communication Section, Communications Research Laboratory, MPHPT, Japan, retrieved on April 3, 2003.
13. H. Kakeya, Y. Arakawa**,** "Autostereoscopic display with real-image virtual screen and light filters", *Proceedings of SPIE Vol. 4660, Stereoscopic Displays and Virtual Reality Systems IX*, 2002.
14. S. Yoshida, S. Miyazaki, T. Hoshino, T. Ozeki, J. Hasegawa, T. Yasuda, S. Yokoi, "A technique for precise depth representation in stereoscopic display" *The Visual Computer*, Vol. 17, No. 1, 2001.
15. C. F. Neveu, L. W. Stark, "The Virtual Lens", *Presence*, Vol. 7, No. 4, pp 370-381 August 1998.
16. C. van Berkel, "Image Preparation for 3D-LCD", *Proceedings of SPIE Vol. 3639 Stereoscopic Displays and Virtual Reality Systems VI*, 1999.
17. A. Woods, T. Docherty, R. Koch, "Image Distortions in Stereoscopic Video Systems", *Stereoscopic Displays and Applications IV, Proceedings of the SPIE*, Volume 1915, San Jose, CA, Feb. 1993.
18. C. W. Smith, "3-D or not 3-D?" *New Scientist*, Vol. 102 No. 1407, pages 40-44, 26 April 1984.
19. T. Okoshi, "Three-Dimensional Imaging Techniques", Academic Press, New York, 1976.
20. J. Freeman, J. Lessiter, "Hear There & Everywhere The Effects of Multi-channel Audio on Presence", *ICAD 2001 - The Seventh International Conference on Audio Display*, Helsinki University of Technology, July 29th - August 1st 2001.
21. J. A. Rupkalvis, "Human considerations in stereoscopic displays", *Proceedings of SPIE Vol. 4297, Stereoscopic Displays and Virtual Reality Systems VIII*, 2001.
22. M. Andiel, S. Hentschke, T. Elle, E. Fuchs, "Eye-Tracking for Autostereoscopic Displays using Web Cams", *Proceedings of SPIE Vol. 4660, Stereoscopic Displays and Virtual Reality Systems IX*, 2002.