TITLE: **USING STEREOSCOPIC IMAGING FOR
VISUALIZATION APPLICATIONS**

AUTHOR(S): Stephen J. Adelson

SUBMITTED TO: IS&T/SPIE Conference on Electronic Imaging Science and Technology
February 6-10, 1994
San Jose, California

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive royalty-free license to publish or reproduce the published form of this contribution or to allow others to do so, for U.S. Government purposes.

The Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy.

Los Alamos

Los Alamos National Laboratory
Los Alamos New Mexico 87545

MASTER

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Using stereoscopic imaging for visualization applications

Stephen J. Adelson

MS M986, Los Alamos National Laboratory
Los Alamos, NM 87544

e-mail: adelson@gardener.lanl.gov

ABSTRACT

The purpose of scientific visualization is to simplify the analysis of numerical data by rendering the information as an image. Even when the image is familiar, as in the case of terrain data, preconceptions about what the image *should* look like and deceptive image artifacts can create misconceptions about what information is actually contained in the scene. One way of aiding the development of unambiguous visualizations is to add stereoscopic depth to the image.

Despite the recent proliferation of affordable stereoscopic viewing equipment, few researchers are at this time taking advantage of stereo in their visualizations. It is generally perceived that the rendering time will have to be doubled in order to generate the pair, and so stereoscopic viewing is sacrificed in the name of expedient rendering. We show that this perception is often invalid. The second half of a stereoscopic image can be generated from the first half for a fraction of the computational cost of complete rendering, usually no more than 50% of the cost and in many cases as little as 5%. Using the techniques presented here, the benefits of stereoscopy can be added to existing visualization systems for only a small cost over current single-frame rendering methods.

1. INTRODUCTION

Scientific visualization is now used in many fields of study, and the visualizations represent data from vastly different sources, including simulations, magnetic resonance (MRI) and computed tomography (CT) medical images, sonar and radar images, mathematical models and equations, and satellite telemetry. Rendering methods range from traditional polygon-based scan conversion to more exotic point-based and inverse-mapped voxel rendering. The one attribute shared by all visualization applications is that they are using computer graphics to display and gain insight into numerical datasets which are as large as several gigabytes in size for a single time step.

In the majority of cases, the data has some three-dimensional structure which is not always intuitively obvious, as in atmospheric phenomena, water currents, or molecular models. This structure is important to comprehend in order to properly analyze the image. Even when there is experience with equivalent real-world objects, as in an architectural model or terrain overview, artifacts of texture and shading in the image can lead to misinterpretation of the data. Stereoscopic imaging overcomes these defects by providing unambiguous depth cues.

The last decade has seen an explosion of affordable stereo viewing apparatus, and in the last five years, an emergence of methods for efficiently creating stereoscopic images. In this paper we investigate the use of these methods in scientific visualization. We first address the question, why rely on stereoscopic imaging for depth information when other methods are available and have been used effectively? Next, some examples of stereoscopic scientific visualization are examined. Finally, we discuss the methods for creating stereoscopic views and explain several algorithms for creating stereoscopic views in less computational time than two completely rendered images.

2. WHY STEREO?

Depth in images must be communicated to the viewer in some way. Sometimes it can be perceived through object interposition. More directly, viewing depth is transmitted through perspective, shading, and motion cues. Visualizations have used all three of these cues independently and concurrently with various degrees of success, but all have defects.

2.1 Depth using perspective

Perspective gives the illusion of depth by rendering more distant objects smaller than near objects. It is generally of little use in scientific visualization. First, a viewer must have some idea of the relative size of objects to properly understand perspective

images. No such size generalizations exist in many scientific visualizations. Furthermore, visualizations are often generated using orthogonal projections, mostly for speed, in which distant objects do *not* decrease in size. If the image is generated using vectors or particles, there is no size information at all. We may conclude that perspective will rarely be of use in visualization images, except as confirmation of depth when used simultaneously with other cues.

2.2 Depth using shading

The shading of objects can reveal their relative positions, but again only with solid objects. Even then, circumstances can arise which make shading ineffective as a depth cue. In one common shading technique, depth shading (also known as luminance), positions more distant from the viewer are shaded in progressively darker tones, equivalent to having a single light source at the viewing position. Depth shading is simple and quick to implement, but it limits the number of available colors; otherwise, one artifact of the image might be mistaken for a more distant portion of another. Also, depth shading can make examination of certain image portions more difficult, since the cue requires relatively dimmer shading in distant areas.

Using a more complex shading model, it is possible to view the image with any number of light sources from many positions. While this often provides more authentic looking images, realistic shading requires more time to implement than depth shading and it is still possible because of the particular view of the data for images to be misleading.

Sometimes color in visualizations is not used to represent depth at all. The image may be, for example, a rendering of the ozone in the atmosphere or ocean currents rendered as isosurfaces, particles, or vectors. Color in these cases is often used to present qualities of the data, such as speed, pressure, concentration, or type of particle. In these cases, some other method must be used to transmit structure.

It should be noted that stereoscopic views from television cameras have been judged a necessary part of many telepresence operations.¹⁻³ If television images, using the ultimate in familiar scenery and "realistic shading", require stereoscopic views, it would certainly seem that computer generated images would require them as well.

2.3 Depth using motion

Motion cues are the depth information perceived in the relative motion of objects. From real world experience, we are all familiar with the effect of observing scenery in motion; near objects appear to move faster than those farther away. In scientific visualization, motion cues are most commonly provided by continuous motion on a sphere surrounding the data, and less often by switching between a limited number of positions.

Motion cues are a very effective depth cue. However, in many cases it is not possible to generate real-time images, and the views must be pre-calculated. If the rendering time for an image is several hours on a supercomputer, rendering multiple images for a full circle about the data is increasing the rendering time to several days. More importantly, motion cues are only effective while the image remains in motion. It is impossible to stop and examine a portion of the image and retain the depth information.

It is also possible to alternate between two views (horizontally or vertically) at about 10 Hz and perceive depth information in the images. Some success has been reported using this technique in medical applications.^{4,5} When other depth cues are available, though, a comparison finds that users perform better with the other methods and find alternating pairs objectionable.⁶

2.4 Depth using stereoscopic viewing

Stereoscopic viewing provides another alternative, although it can be used in combination with other cues. Stereo pairs provide a very strong depth cue in isolation (as tested using random dot stereograms⁷) as well as with other cues to support it. It is also the method judged by users to be "most natural" and most preferred in isolation over other cues.^{6, 8-11} Even in depth discrimination tasks where motion cues were found to be more effective than stereoscopic vision, stereo is nonetheless judged to be a significant performance enhancement.^{12,13} And, as mentioned in section 2.2, there are tasks which are impossible to perform without stereo images.

Admittedly, stereoscopic viewing does have disadvantages. About 10% of those with two normal eyes cannot perceive stereoscopic depth¹⁴, and of course those with only a single normal eye are excluded as well. Also, unless the viewer cares to free-view the two halves of the stereo pair (fuse the two images on the computer screen by crossing the eyes), stereoscopic viewing requires specialized equipment. This may require the user to wear glasses or the use of screen-based equipment such as a parallax barrier, lenticular screen, or moving slits.^{15,16}

Stereo pairs are generated from a virtual viewing position. If head-tracking is available, images can always be correctly projected, but only for the person being tracked. Also, viewing the image away from its virtual generation point will warp the image, although relative depth will be preserved. Fortunately, a slight movement off the point results in little warping, and relative structure is usually sufficient for analysis.

The longest running debate about the use of stereoscopic imaging is its disassociation of accommodation and convergence. In normal vision, the eyes accommodate (focus on a point in space) and the same time that they converge on that same point in space.^{14,17} In stereoscopic imaging, the eyes are accommodating to the plane of the viewing apparatus and converging on objects before and behind the plane. Accommodation and convergence act together in a learned ratio in natural vision, and some users report fatigue and headaches before they learn to disassociate the two cues. One unscientific 19th century study even claimed that absolute depth judgment will be in error to a degree by which accommodation and convergence differed from natural vision.¹⁸

Various methods have been suggested for correcting this discrepancy, including special optics with eye tracking, true volumetric displays, projecting to optical infinity, and guidelines for minimizing the use of stereo.¹⁹⁻²⁴ However, association of the two muscle groups can usually be painlessly unlearned, and there is evidence that depth in stereo is achieved not through accommodation and convergence but through the perception of "blur".²⁵ Accommodation and convergence disassociation, while occasionally a problem with new viewers, is not ordinarily a concern for those who wish to use stereoscopic imaging on a continuing basis. If the concession is allowed that a minority of the population will be unable to view the images stereoscopically, then stereoscopy provides significant advantages over monoscopic images.

3. SOME EXAMPLES OF STEREOSCOPIC SCIENTIFIC VISUALIZATION

If one allows the definition of scientific visualization to include any electronic machine-generated images, then the first use of stereoscopic imaging in this context may have been at the University of Minnesota in 1949.²⁶ Professor Otto H. Schmitt used a modified stereoscope to solve three variable mathematical problems and to view three voltages of an electrocardiogram as three-dimensional images. Mention was also made about its possible use as three-dimensional radar, as well as this interesting description of Professor Schmitt's device: "The observer can produce the effect of walking around the object, climbing up over it, crawling under it or plunging into it." Arguably, this was the first virtual reality system.

Stereoscopes have been used with photographs of microscopic images since at least 1887.²⁷ Primitive stereo viewing devices were used with computer generated images rather extensively beginning in the late 1960's, primarily with crystal structures and images of static molecules.²⁸⁻³³ In 1981, Kirby used stereo pairs for analyzing energy plots and other physics data, and suggested that other non-physical datasets like his energy plots could be more effectively studied using stereoscopic images.³⁴ Stereo images have been used to display phenomena which is normally invisible to the eye, such as atmospheric data^{35,36}, and to model the familiar, such as terrain databases³⁷.

Many applications of stereoscopic imaging in scientific visualization appear in virtual environments; interactive, immersive landscapes where the illusion is partially maintained by stereoscopy. Among these are a virtual wind tunnel for visualizing flowfields³⁸, simulation environments for virtual surgery^{39,40}, virtual ultrasound used in constructing a three-dimensional model of a fetus from ultrasonograms⁴¹, and an interactive system used with MRI data of a beating human heart⁴². Another system by Ribarsky uses interactive stereo images to analyze atomic stresses and the actions of certain chain molecules as lubricants.⁴³ This application combines color, shape, and structure via stereo to represent the various forces on the atoms. However, aside these and a few similar exceptions, stereoscopic viewing is not typically used in modern scientific visualization, despite the exponential growth of three-dimensional datasets.

4. STEREOSCOPIC IMAGING GEOMETRIES

In general, stereo pairs are generated using two perspective projections, one for each eye. It is not uncommon in scientific visualization for parallel projection to be used, primarily because it is somewhat faster to implement but also because certain direct volume rendering methods cannot easily adapt to perspective views. Stereo pairs can be generated using parallel projection, when certain restrictions are placed on the rendering parameters. We will first look at creating stereoscopic pairs with perspective projections.

4.1 Stereoscopic images with perspective projection

A standard monoscopic perspective projection have a center of projection at the origin and a projection plane parallel to the x-y plane at $z = d$. For stereoscopic images, two differently projected views of the image are required. Each of the observer's eyes is presented with a view of the scene from different (usually horizontally displaced) positions. The most efficient derivation uses two viewing positions, one at the origin and the second displaced to the right⁴⁴, but other methods are possible^{20, 46-48}.

Assume a projection plane located at $z = d$ and two centers of projection: one for the left-eye view (LCoP) located at $(0, 0, 0)$ and another for the right-eye view (RCoP) located at $(e, 0, 0)$ as shown in figure 1. A point $P = (x_p, y_p, z_p)$ projected to LCoP has projection plane coordinates $P_{sl} = (x_{sl}, y_{sl})$ where

$$x_{sl} = x_p d / z_p - e/2 \quad (1)$$

$$y_{sl} = y_p d / z_p. \quad (2)$$

The same point projected to RCoP has projection plane coordinates $P_{sr} = (x_{sr}, y_{sr})$ where

$$x_{sr} = (x_p + e) d / z_p + e/2 \quad (3)$$

$$y_{sr} = y_p d / z_p. \quad (4)$$

Figure 1: Stereoscopic Perspective Projection

If the d / z_p term is retained between the x and y calculations and $e/2$ is stored as a constant, both pairs can be calculated with three additions and six multiplications. However, the projected y values (equations (2) and (4)) are identical, so both pairs can actually be calculated with three additions and four multiplications. Furthermore, the projected points will share normals (since they both originate from the same spatial position) and non-eyepoint dependent coloring information, so this information need be calculated only once.

Given that one view (suppose the left) can be rendered in a point-sampled manner, we can subtract equation (1) from equation (3) to find the right-eye view x coordinate from the left-eye view:

$$x_{sr} = x_{sl} + e - e d / z_p. \quad (5)$$

Both e and $e d$ are constants, so the second x position can be found with two additions and one multiplication. Using equation (5) to find the new x position has been dubbed *reprojection*.

Love demonstrated that stereo pairs could be generated by a series of translations and shears on Z.⁴⁷ Although, as published, the equations are off by a constant factor d , the corrected formulae (given centers of projection at $(\pm e/2, 0, 0)$) are:

$$\begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ S & 0 & 1 & 1/d \\ T & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} x + S z + T \\ y \\ z/d \\ 0 \end{bmatrix}$$

where $S = \pm e / 2d$ (the shear to put the viewpoints on the z axis) and $T = -S Z_w$, Z_w the z value of the projection plane after the viewing transformation. The projected position is then:

$$x_s = [x + S z + T] d / z \quad (6)$$

$$y_s = y d / z \quad (7)$$

Multiplying and rearranging the terms of equation (6) yields the equations (1) and (3).⁴⁴ This method is mathematically equivalent to the translation and projection method above, and involves the extra calculations to generate S and T . This shear method is best thought of as a conveniently expressed matrix which can be combined with the viewing transformation matrix in cases where there are no operations to be done in transformed space (such as clipping, shading, or back-face removal).

Harrison extended Love's work⁵⁰, and suggested that the right-eye view x position could be determined by moving the projected points to the second half of the pair using a difference equation. The difference equation for the above geometry is:

$$x_{sr} = x_{sl} - 2S d - 2T d / z_p \quad (8)$$

Replacement of S and T by their factors and rearrangement yields(5).

4.1.1 Choosing values for e and d

When rendering computer-generated stereoscopic images, the level of apparent depth is determined by the *horizontal parallax*, the separation of the two projected coordinates. There are therefore only a limited number of depth levels which can be displayed, dependent upon the value of viewpoint separation, e , and the device resolution. It is important to choose values for rendering constants e and d so that the range of the data contains as many depth levels as possible. e is physically limited by the interocular distance measured in device units (approximately 2.5 inches), but in practice viewers have difficulty fusing images separated by that great of a distance. The comprehensive study by Yeh⁵¹ suggested that e be limited to 27 minutes of arc for crossed disparity (objects appearing before the projection plane, right-eye projection to the left of the left-eye projection, figure 2) and 24 minutes of arc for uncrossed disparity (objects appearing behind the projection plane, right-eye projection to the right of the left-eye projection). These limits are equal to $e = -0.008D$ and $e = 0.007D$, respectively, where D is the physical distance from the viewer to the screen, scaled to device coordinates. Hodges supported a somewhat more lenient value of 1.5 degrees for both crossed and uncrossed disparity, so that $e = \pm 0.028D$.⁴⁸ The value chosen depends in part on the viewers' experience with stereoscopy; experienced viewers can generally fuse greater disparity. However, a study of the literature reveals that a value of 1.5 degrees is in no way unreasonable even for novice users.

Figure 2: Uncrossed and crossed disparity

Whatever formula is used to generate e , let us call the maximum value e_{max} , and the minimum value, e_{min} . If the entire image is known to exist on the opposite side the projection plane from the viewer, e_{min} may be set to the maximum e for crossed disparity. If this is not the case, e_{min} must be set to 0; otherwise, there is a risk that objects which appear to be in front of the projection plane will be obscured by the edge of the virtual window behind it.

Let us also call the maximum depth of the data after viewing transformation $z_{\max}$, and the minimum depth, $z_{\min}$. If a series of stereoscopic images is being generated, e and d should not change, so $z_{\max}$ and $z_{\min}$ must be valid across all frames. For a point at $z_{\max}$ a horizontal parallax of $e_{\max}$ is desired, and for a point at $z_{\min}$, $e_{\min}$ is desired. Equation (5) can be rearranged to yield:

$$x_{sr} - x_{sl} = \text{horizontal parallax} = e - e d/z_p. \quad (9)$$

Substituting,

$$\begin{aligned} e_{\max} &= e - e d/z_{\max} \\ e_{\min} &= e - e d/z_{\min}; \\ e &= (e_{\max} z_{\max} - e_{\min} z_{\min})/(z_{\max} - z_{\min}) \end{aligned} \quad (10)$$

$$d = z_{\max} - e_{\max} z_{\max}/e. \quad (11)$$

If $e_{\min}$ is equal to 0, d simplifies to $z_{\min}$.

Note that units of e and d are in *device* coordinates. If e or d are required for other reasons, such as some of the algorithms outlined in section 5, they must first be scaled back to object space coordinates.

4.2 Stereoscopic images with parallel projection

Hodges showed how a common method for generating perspective stereo pairs, rotation of the camera from the two view points, caused image artifacts in stereo pairs.⁴⁹ However, they can be used with parallel projections under certain conditions.

Given a single viewing position at the origin, the two views can be created by rotating the image about a point $P = (0, 0, R)$ a distance of $\pm\Phi/2$, where Φ is the viewing angle. A more efficient method displaces and rotates the data so that the LCoP is at the origin and the RCoP is at the point $(e \cos(\Phi/2), 0, e \sin(\Phi/2))$, figure 3.⁴⁴ The x coordinates, then are:

$$x_{sl} = x_p \quad (12)$$

$$x_{sr} = x_p \cos(\Phi) + z_p \sin(\Phi) - e \cos(\Phi/2) \quad (13)$$

$$y_{sl}, y_{sr} = y_p \quad (14)$$

Figure 3: Viewing Angle Φ for stereo pairs with parallel projections

Only two additions and two multiplication is necessary to calculate both pairs of projected coordinates. The final (constant) subtraction may be replaced by a hardware pan of the final image. The shift and rotation of the data is combined with the viewing transformation matrix, so no additional calculations are needed to transform the data. The reprojection equation in this case is actually *more* complex than the full right-eye view equation (13).

Since Φ is small, we can assume the simplifications that $\sin(\Phi/2) = \Phi/2$ and $\cos(\Phi/2) = 1$. The right-eye x coordinate is then $x_p + z_p \Phi - e \Phi/2$, which is a z shear followed by a translation. The constant $(e \Phi/2)$ term can be removed and the entire image panned by that amount.

In parallel projected stereoscopic images, unlike those generated with perspective projection, horizontal parallax is unbounded. To keep horizontal parallax reasonable, the z values of the data and values of the rendering constants e and Φ must be kept within a specific range.

The bounds of e, $e_{\max}$ and $e_{\min}$, were calculated by multiplying some constant by the viewing distance. If those constants are K1 and K2, then the range of z needed to give the correct parallax bounds is:

$$D (1 - K1/[2 \sin(\Phi/2)]) \leq z \leq D (1 - K2/[2 \sin(\Phi/2)]).^{44} \quad (15)$$

The range of z is maximized as $\sin(\Phi/2)$ approaches 0, but $\sin(\Phi/2)$ must be greater than K1/2 in order to keep all z values greater than zero. D may be fixed at any comfortable viewing distance and the value of e can be found in terms of D and Φ :

$$e = 2D \tan(\Phi/2) \quad (16)$$

The data will probably require a scale and a shift in order to fit into the derived z range. These operations can be added to the viewing transformation matrix so that no additional transformational operations are required. Note that when the displaced rotated geometry is used, the scale and shift must be added between the original viewing transformation and the data rotation and displacement.

5. EFFICIENT GENERATION OF STEREOSCOPIC IMAGES

In the systems which have used stereoscopic imaging, almost all have generated the two requisite views from scratch, doubling the necessary work. (The one exception of which we are aware is Papathomas' atmospheric phenomena program³⁶). A better rendering strategy is to take advantage of the stereoscopic viewing geometry, specifically that projections between the two views will occur on the same scan-line and that one x position can be quickly found from the other. In this section, we examine polygon-based rendering, point-sampled rendering, volume rendering, and a method for quickly generating approximate images when exact images are not necessary.

5.1 Polygon-based rendering techniques

Many of the traditional rendering algorithms are based on scan-conversion of planar polygons. For this reason, many visualization programs either use polygonal models or produce polygonal models as an intermediate step before using a traditional renderer to generate the image. In this section we look at polygon rendering algorithms modified for stereoscopic imaging, using one detailed example and an overview of several others.^{45,48,52}

A common preprocessing step to polygon rendering is back-face removal, the elimination of those polygons which cannot be seen because of their orientation to the viewer. Given the equation of the plane of the polygon, $Ax + By + Cz + D$, the viewing position can be evaluated. A negative value indicates a back-facing polygon. If (A, B, C) is a normalized vector, back-faces can be found simultaneously using the following pseudo-code:

```
tmp = Axc + Byc + Czc + D           // (xc, yc, zc) is the LCoP, RCoP = LCoP + (xadd, yadd, zadd)
if (tmp > 0)
    polygon is a front-face for at least the left-eye view
else if (tmp < e)                        // tmp will vary by no more than ±e between views.
    polygon is back-face for both views
else if (tmp > -Axadd - Byadd - Czadd)
    polygon is a front-face for the right eye view
```

Since most visible polygons (93% or more) are a front-face or a back-face for both centers of projection⁴⁵, almost all will be identified by the first two ifs. A slightly more complex version will identify sides that are specifically front-faces or back-faces for each view, but usually we are more interested in identifying front-faces for either view so that transformations will be performed only on the necessary polygons.

Recall from equations (2) and (4) that the projected y values of a polygon vertices will be identical. This means that projected polygons will have the same height, but differing widths. We rely on this fact for several optimizations. Note that there will be the same number of vertical steps in projected non-horizontal edges. Gouraud shading interpolates colors along edges, and then along scan-lines.⁵³ These interpolated edge colors need only be calculated once. Phong shading interpolates normals in the same way, and calculates the colors at each pixel.⁵⁴ Again, the colors (and normals) along edges will only need to be calculated once for both views. Polygons will be active (intersecting the current scan-line) in the same regions in both views, so only one data-structure will be necessary to track the active polygons.

Line clipping can be 50% more efficient since the clipping for the top and bottom window edges will be the same. Polygon clipping is less efficient because the polygon is reshaped to follow the window contours when clipped, but those polygons needing to be partially clipped can finish operations 10% faster than two passes of a traditional clipping algorithm.

5.2 Point-sampled rendering techniques

Some surfaces are not polygonal, but rather are based on some function and rendered using rays. Even when the surfaces are composed of polygons, they are sometimes rendered in a point-sample method such as is found in ray-tracing. Simple forms of volume rendering (those with solid surfaces and no translucency) and more complex methods like dividing cubes⁵⁵ also produce point samples of the data. We can make the following observations about such sampled images⁵⁶⁻⁵⁸:

Using a reprojection equation (either (5) or (16), depending on the projection method), projected point samples can be moved to the second half of the pair. This point has the same diffuse color, normal, 3D spatial location, and it is shadowed (or not) by the same object. Eye-point dependent shading still has to be calculated twice, but the normal and intersection are already available. Likewise, reflective and refractive rays must be calculated twice, but again the ray direction can be easily calculated since normals and intersections are available. Of course, most visualizations do not consider reflection and refraction or even shadowing, so these phenomena are not generally a concern.

Two problems that occur when reprojection are the *overlapped pixel problem* and the *bad pixel problem*.¹ In the overlapped pixel problem, more than one value reprojects into the same location in the second view. It can be shown, however, that processing the pixels in a left to right order across the scan-line guarantees that the last reprojection is the correct one.

The bad pixel problem can appear when the second of two consecutive pixels (when processed left to right) has a greater depth than the first. Reprojection will shift the second pixel farther than the first, creating a gap between them. This gap corresponds to an area of the image which can be seen by the right eye but not by the left. It is possible that pixels processed previously had reprojected to a point in the middle of the gap, and may remain in the final image, even though they possibly should be hidden. There is no way to know if such pixels are correct, so we err on the side of caution and clear the values of intervening pixels when a gap appears.

All empty pixels remaining after reprojection are filled using more point samples. The total number of empty pixels, including bad pixels forced to emptiness, generally comprises 7 - 9% of the image. 90+% of the samples in the second half of the pair are reused directly from the first half (minus any eye-point dependent phenomena).

Experiments have shown that the second half of the stereo pair can be generated for less than 10% of the effort of the first half in images without reflection or refraction, and that those images including this phenomena generally have at least a 50% increase in rendering speed. An analysis shows that when speed is measured by the number of rays cast, reprojection will always be faster than two passes of a standard algorithm, regardless of the complexity of the image.

5.3 Stereoscopic volume rendering

Volume rendering in stereo poses special problems when a translucent medium is involved. When ray casting is used, rays accumulate opacity until they exit the volume or accrue opacity of 1.0. However, each of the samples in the original ray has a different depth and, hence, reprojects to a different point in the other view.

Fortunately, there is no bad or overlapped pixel problem in a translucent medium. If the rays from the left-eye view are cast right to left across scan-lines, samples will accumulate in the right-eye view positions in a front to back order. Some problems may arise when objects are small in relation to the ray separation; samples in the right view are not accumulated because the left view ray hits a small object and stops. An article is currently in preparation which deals with this and similar problems.

Other volume rendering techniques can use reprojection less directly. As mentioned above, point sample methods can be treated like rays. For example, splatting involves taking a voxel and projecting it to an area of the screen like a snowball.⁵⁹ Samples accrue to the pixels which are struck by the voxel snowball. The pixels affected in the second view can be found by reprojection, which will be a constant shift for any given snowball. Other voxel projecting methods can use similar strategies.

Experiments are currently underway to quantify the error of these methods, but it appears to be on the same order as the polygonal and point-sample methods.

5.4 Quick and dirty stereoscopic images

There are times when exact stereoscopic images are not required; only a general sense of the structure is needed. Also, when the second half of a stereo pair will take a significant quantity of time to completely render yet the first half is complete, a rough stereoscopic view could be desired as an approximation of the fully rendered image. Under such conditions, Quick and Dirty (QnD) stereoscopic images are desirable. QnD stereo relies on the fact that in most images 90+% of the pixels are visible in both halves of the stereoscopic pair. Using reprojection, accounting for the overlapped and bad pixel problems, and filling the gaps using heuristics, an approximate stereoscopic pair can be generated at least 50% faster than even an efficient stereoscopic algorithm. Since the second half of the pair can be generated in constant time, the savings approaches 100% as the rendering time of the true image increases.

A similar technique was suggested by Harrison.⁵⁰ However, his method did not address the bad and overlapped pixel problems, nor did it attempt to fill the resulting gaps in the image. These empty gaps can cause a condition known as *binocular rivalry*, which occurs when one eye sees a different image from the other, as opposed a different view of the same image.^{60,61} Binocular rivalry results in the visual channels vying for dominance, intermittent switching of the primary image, and possible loss of the stereo effect.

QnD stereo works best with images which have no reflection or refraction, since two different reflections or refractions are required; these cannot be generated without more calculations, significantly reducing the speed of QnD. The one exception is volume rendering with a translucent medium. In such a case, there is no refraction as through ordinary transparent medium. For QnD volume rendering, we recognize that most translucent medium renderings usually have a single solid surface which is large in relation to ray separation. Therefore, all possible samples are accumulated to the right-eye view without regard to any small blocking objects. The effectiveness of this technique is currently under study.

6. CONCLUSIONS

With the growing availability of low-cost stereoscopic imaging technology, it has become feasible for nearly all researchers to use stereo pairs in scientific visualization. Other depth cues used in computer graphics, interposition of data, perspective, shading, and motion cues, cannot always be effectively used in visualizations. Even when they are available, difficulties exist which can make analysis difficult. Stereoscopy, in contrast, provides unambiguous depth cueing both with and without other confirming cues.

Quite often, a stereo pair can be rendered in much less than twice the rendering time of a single image, in some cases for only 5% additional effort. If an approximate stereoscopic pair are acceptable, the second half of the pair can be generated for a small, constant cost. Stereoscopic imaging can be a great boon to scientific visualization. It significantly adds to the analysis of a given image, providing a sense of structure to still images as well as enhanced depth to moving images.

7. REFERENCES

1. RE Cole, JO Merritt, S Fore and P Lester, "Remote Manipulation Tasks Impossible Without Stereo TV," SPIE Proceedings 1256: Stereoscopic Displays and Applications, pp. 255-265, February 1990.
2. EH Spain, "Stereo Advantage for a Peg-in-hole Task Using a Force-Feedback Manipulator," SPIE Proceedings 1256: Stereoscopic Displays and Applications, pp. 244-254, February 1990.
3. JO Merritt, "Perceptual Training with Cues for Hazard Detection in Off-Road Driving," SPIE Proceedings 1457: Stereoscopic Displays and Applications II, pp. 133-138, February 1991.

4. ER Jones, L Cathey and AP McLaurin, "VISIDEP: Visual Image Depth Enhancement by Parallax Induction," SPIE Proceedings 437: Advances in Display Technology IV, pp. 16-19, 1984.
5. AP McLaurin, ER Jones and JL Mason, Jr., "Three-dimensional Endoscopy Through Alternating Frame Technology," SPIE Proceedings 1256: Stereoscopic Displays and Applications, pp. 307-311, February 1990.
6. SJ Adelson, J Allen, AN Badre, LF Hodges and A Lawrence, "Performance Comparison of Multiple Image Depth and Shape Cues," International Journal of Human Computer Interaction, Vol. 5, 4, pp. 347-360, November 1993.
7. B Julesz. Foundations of Cyclopean Perception, University of Chicago Press, Chicago, 1971.
8. WF Reinhart, RJ Beaton and HL Snyder, "Comparison of Depth Cues for Relative Depth Judgments," SPIE Proceedings 1256: Stereoscopic Displays and Applications, pp. 12-21, February 1990.
9. Y Yeh and LD Silverstein, "Visual Performance with Monoscopic and Stereoscopic Presentation of Identical Three-Dimensional Visual Tasks," SID Digest XXI, pp. 359-362, May 1990.
10. WF Reinhart, "Depth Cueing for Visual Search and Cursor Positioning," SPIE Proceedings 1457: Stereoscopic Displays and Applications II, pp. 221-232, February 1991.
11. LF Hodges and SM McWhorter, "Stereoscopic Display for Design Visualization," Image Communication, Vol. 4, 1, pp. 3-13, November 1991.
12. RL Sollenberger and P Milgram, "A Comparative Study of Rotational and Stereoscopic Computer Graphics Depth Cues," Proceedings of the Human Factors Society 35th Annual Meeting, pp. 1452-1456, October 1991.
13. C Ware, K Arthur and KS Booth, "Fish Tank Virtual Reality," INTERCHI '93, pp. 37-42, April 1993.
14. GF Poggio and T Poggio, "The Analysis of Stereopsis," Annual Review of Neuroscience, Vol. 7, pp. 379-412, 1984.
15. LF Hodges, S Love, and DF McAllister, "Hard copy for true three-dimensional images," Information Display, Vol. 3, 4, pp. 18-22, May 1987.
16. LF Hodges and DF McAllister, "True three-dimensional CRT-based displays," Information Display, Vol. 3, 8, pp. 12-15, 25, September 1987.
17. D Brewster, "On the Knowledge of Distance Given by Binocular Vision," Transactions of the Royal Society of Edinburgh, Vol. 15, pp. 121-135, 1844.
18. WLC Stevens, "The Theory of the Stereoscope," The Journal of the Franklin Institute, Vol. 114, 4, pp. 279-287, October 1882.
19. ET Ferguson, "Some Notes on Stereoscopic Display and an Isochrome Anaglyph CRT," 9th Annual Symposium on Information Display, pp. 201-207, 1968.
20. J Baker, "Generating Images for a Time-Multiplexed Stereoscopic Computer Graphics System," SPIE Proceedings 761: True 3D Imaging Techniques and Display Technologies, pp. 44-52, February 1987.
21. T Yamazaki, K Kamijo and S Fukuzumi, "Quantitative Evaluation of Visual Fatigue Encountered in Viewing Stereoscopic 3D Display - Near Point Distance and Visual Evoked Potential Study," Japan Display '89, Proceedings of the 9th Annual International Display Research Conference, pp. 606-609, 1989.
22. AM Busquets, RV Parrish and SP Williams, "Depth-Viewing-Volume Increase By Collimation of Stereo 3D Displays," IEEE Southeastcon '90, paper #90-260, 1990.

23. T Miyashita and T Uchida, "Cause of Fatigue and Its Improvement in Stereoscopic Displays," Proceedings of the SID, Vol. 31, 3, pp. 249-254, 1990.
24. SP Williams and RV Parrish, "New Computational Control Techniques and Increased Understanding for Stereo 3D Displays," SPIE Proceedings 1256: Stereoscopic Displays and Applications, pp. 73-82, February 1990.
25. L Kaufman. Sight and Mind: An Introduction to Visual Perception, Oxford Press, New York, 1974.
26. "Radar in Three Dimensions," Popular Mechanics, Vol. 92, 10, pp. 110-111, October 1949.
27. CF Himes, "The Stereoscope and Its Applications," The Journal of the Franklin Institute, Vol. 123, 5-6, pp. 398-408, 425-441, May and June 1887.
28. FM Richards, "The Matching of Physical Models to Three-Dimensional Electron Density Maps: A Simple Optical Device," Journal of Molecular Biology, Vol. 37, 1, pp. 225-230, October 1968.
29. CD Barry, RA Ellis, SM Graesser, and GR Marshall, "CHEMAST: A Computer Program for Modeling Molecular Structures," Proceedings of the 1971 IFIP, pp. 1552-1558, 1971.
30. WJ Perkins, EA Piper, FG Tattam and JC White, "Interactive Stereoscopic Computer Displays for Biomedical Research," Computers and Biomedical Research, Vol. 4, 3, pp. 249-261, 1971.
31. SD Stellman, "Three-Dimensional Interactive Graphics in X-Ray Crystallographic Analysis," Computers and Graphics, Vol. 2, 3, pp. 279-288, September 1975.
32. HL Monaco, JL Crawford and WN Lipscomb, "Three-Dimensional Structures of Aspartate Carbamoyltransferase from Escherichia coli and of its Complex with Cytidine Triphosphate," Proceedings of the National Academy of Sciences USA, Vol. 75, 11, pp. 5276-5280, November 1978.
33. JA Roese and LE McCleary, "Stereoscopic Computer Graphics for Simulation and Modeling," Computer Graphics, Vol. 13, 2, pp. 41-47, August 1979.
34. GH Kirby and A Rixon, "Computer Graphics and Stereoscopy for Three-Dimensional Data," Computer Physics Communications, Vol. 21, 3, pp. 287-291, January 1981.
35. M desJardins and AF Hasler, "Stereoscopic Displays of Atmospheric Model Data," Computer Graphics, Vol. 14, 3, pp. 134-139, July 1980.
36. TV Papathomas, JA Schiavone and B Julesz, "Stereo Animation for Very Large Data Bases: Case Study - Meteorology," IEEE Computer Graphics and Applications, Vol. 7, 9, pp. 18-27, September 1987.
37. H Veron, DA Southard, JR Leger, and JL Conway, "Stereoscopic Displays for Terrain Database Visualization," SPIE Proceedings 1256: Stereoscopic Displays and Applications, pp. 124-135, February 1990.
38. S Bryson and C Levit, "The Virtual Wind Tunnel," IEEE Computer Graphics and Applications, Vol. 12, 4, pp. 25-34, July 1992.
39. S Pieper, S Delp, J Rosen and S Fisher, "A Virtual Environment System for Simulation of Leg Surgery," SPIE Proceedings 1457: Stereoscopic Displays and Applications II, pp. 188-197, February 1991.
40. JR Merrill, "Surgery on the Cutting Edge," Virtual Reality World, Vol. 1, 2/3, pp. 34-38, November / December 1993.
41. M Bajura, H Fuchs and R Ohbuchi, "Merging Virtual Objects with the Real World -- Seeing Ultrasound Imagery within the Patient," Computer Graphics, Vol. 26, 2, pp. 203-210, July 1992.

42. Y Jean, LF Hodges and R Pettigrew, "A Method for Interactive Manipulation and animation of Volumetric Data," Georgia Tech Technical Report GIT-GVU-91-32, Georgia Institute of Technology, 1991.
43. W Ribarsky, LF Hodges, R Minsk and M Bruchez, "Visual Representation of Discrete Multivariate Data," Georgia Tech Technical Report GIT-GVU-92-04, Georgia Institute of Technology, February 1992.
44. SJ Adelson, "Stereoscopic projections: parallel viewing vectors, rotations, and shears," Los Alamos National Laboratory Technical Report, in press.
45. SJ Adelson, "Efficient Generation of Stereoscopic Images and Applications to Animation," Ph.D. Thesis, Georgia Institute of Technology, Atlanta, Georgia, September 1993.
46. DJ Sandin, E Sandor, WT Cunnally, M Resch, TA DeFanti and MD Brown, "Computer-Generated Barrier-Strip Autostereogram," SPIE Proceedings 1083: Non-Holographic True 3D Display Technology, pp. 65-75, February 1989.
47. S Love, "Nonholographic, Autostereoscopic, Nonplanar Display of Computer Generated Images," Ph.D. Thesis, North Carolina State University, Raleigh, North Carolina, 1990.
48. LF Hodges, "Time-Multiplexed Stereoscopic Computer Graphics," IEEE Computer Graphics and Applications, Vol. 12, 2, pp. 20-30, March 1992.
49. LF Hodges and DF McAllister, "Computing Stereoscopic Views," Stereo Computer Graphics and Other True 3D Technologies, ed. DF McAllister, pp. 71-89, Princeton University Press, Princeton, 1993.
50. L Harrison and DF McAllister, "Implementation Issues in Interactive Stereo Systems," Stereo Computer Graphics and Other True 3D Technologies, ed. DF McAllister, pp. 119-151, Princeton University Press, Princeton, 1993.
51. Y Yeh and L Silverstein, "Limits of Fusion and Depth Judgment in Stereoscopic Color Displays," Human Factors, Vol. 32, 1, pp. 45-60, February 1990.
52. SJ Adelson, JB Bentley, IS Chong, LF Hodges, and J Winograd, "Simultaneous Generation of Stereoscopic Views," Computer Graphics Forum, Vol. 10, 1, pp. 3-10, March 1991.
53. H Gouraud, "Continuous Shading of Curved Surfaces," IEEE Transactions on Computers, Vol. C-20, 6, pp. 623-629, June 1971.
54. P Bui-Tuong, "Illumination for Computer-Generated Images," Communications of the ACM, Vol. 18, 6, pp. 331-317, June 1975.
55. HE Cline, WE Lorensen, S Luoke, CR Crawford and BC Teeter, "Two Algorithms for the Three-Dimensional Reconstruction of Tomograms," Medical Physics, Vol. 15, 3, 320-327, May / June 1988.
56. R Devarajan and DF McAllister, "Stereoscopic Ray Tracing of Implicitly Defined Functions," Proceedings 1457: Stereoscopic Displays and Applications II, pp. 37-48, February 1991.
57. SJ Adelson and LF Hodges, "Visible Surface Ray-Tracing of Stereoscopic Images." Proceedings of the 30th Annual Southeast Conference, pp. 148-157, April 1992.
58. SJ Adelson and LF Hodges, "Stereoscopic Ray-Tracing." To appear in The Visual Computer.
59. L Westover, "Footprint Evaluation for Volume Rendering," Computer Graphics Vol. 24, 4, pp. 367-376, August 1990.
60. WJM Levelt. On Binocular Rivalry, Mouton, The Hague, 1968.
61. PD Heber and DF McAllister, "Color Quantization Aspects in Stereopsis," SPIE Proceedings 1457: Stereoscopic Displays and Applications II, pp. 233-241, February 1991.

DATE

FILMED

3 / 8 / 94

END