Fast Stereo Volume Rendering

Taosong He and Arie Kaufman

Department of Computer Science
State University of New York at Stony Brook
Stony Brook, NY 11794-4400
{taosong, ari}@cs.sunysb.edu

Abstract

We present new volume rendering techniques for efficiently generating high quality stereoscopic images and propose criteria to evaluate stereo volume rendering algorithms. Specifically, we present fast stereo volume ray casting algorithms using segment composition and linearly-interpolated re-projection. A fast stereo shear-warp volume rendering algorithm is also presented and discussed.

1. Introduction

Stereoscopic rendering is paramount for a variety of applications such as virtual reality and scientific visualization. A stereoscopic pair of images can solely provide unambiguous depth information, or enhance the depth discrimination provided by other depth cues (e.g., [7, 5, 17]) Intuitively, if the time for rendering a single image is t, a stereo pair of images can be generated in $2t$ by rendering the left-eye image (in short, left-image) and the right-eye image (in short, right-image) separately. Fast stereo rendering, on the other hand, utilizes the coherence between the left-image and the right-image, and generates a stereo pair in one pass. Significant rendering time, therefore, can be saved. Most of the existing fast stereo rendering algorithms are based on a standard rendering scheme, such as ray tracing [1]. They usually can be divided into the following three steps:
(1) Select an appropriate viewpoint position (e.g., left-eye, right-eye, or the middle-eye, that is, the middle point between the left-eye and the right-eye.)
(2) Apply the chosen standard rendering scheme for the selected viewpoint.
(3) Re-project the results or intermediate results of step 2 onto the right image or left-image or both according to certain coordinate transformations.

Direct volume rendering is a key technology for the visualization of large sampled, simulated, or synthesized 3D datasets from scientific, engineering, and medical applications. It approximates how the volume data generates, scatters, or occludes light [8]. Unlike surface-based rendering, where only the properties on the object surfaces are displayed, volume rendering presents information about both the surfaces and the inner structures of the model. Therefore, it has been widely used to visualize models with no tangible surfaces, such as clouds and fog. However, volume rendering is notoriously slow because of the large amount of data to be processed and the lack of hardware support. A fast stereo volume rendering algorithm is therefore critical to many applications. Adelson and Hansen [2] have proposed the first fast stereo volume rendering algorithm. Basing their approach on the standard volume ray casting algorithm [10], they follow the three steps of fast stereo rendering described above. First, the viewpoint is placed at the left-eye position; then, conventional ray casting is applied for the generation of the left-image. At the same time, sampling points along the left-eye rays (for short, left-rays) are appropriately chosen to re-project along the right-eye ray (for short, right-ray) direction onto the right-image. An efficient viewing geometry has been developed to reduce the operations needed for re-projection to a minimum. Using this method, the right-image can be generated in a small fraction of the time of the left-image. However, the sampling points along the left-rays normally are not re-projected onto the integer grids of the right-image. The solution used by the algorithm is to round the re-projected samples to their nearest integer neighbor on the right-image. As a result, errors are introduced.

In this paper, we propose a new *segment composition* scheme to further accelerate the stereo volume ray casting. Basically, since the angle between the left-rays and the right-rays is small, it is much faster to re-project several sampling points simultaneously. Although perspective projection is very important for stereo rendering, it is more difficult to implement than parallel projection for fast stereo rendering since the relation between the left-image and the right-image is more complicated. In this paper we focus on parallel projection, which has been used by Adelson and Hansen and in many visualization applications. The rest of the paper is organized as follows. We propose in Section 2 criteria for evaluating the performance of a stereo rendering algorithm. In particular, we consider that balanced image quality between the left-image and the right-image is an important criterion. We then present in Section 3 the segment composition scheme and apply it to standard ray casting. We introduce in Section 4 a linearly-interpolated re-projection method to increase image accuracy of stereo rendering. Existing

Proceedings of the 7th IEEE Visualization Conference (VIS'96)
1070-2385/96 $17.00 © 1996 IEEE

volume ray casting acceleration methods can then exploited and applied to the stereo pair generation. In Section 5 we present a fast stereo shear-warp volume rendering algorithm, which has lower image quality compared to the standard ray casting. The testing results of our algorithms are summarized in Section 6.

2. Stereo Volume Rendering Criteria

Generally, to measure the performance of a fast stereo rendering algorithm, we propose the following three criteria:
(1) Minimum total time for generating both the left-image and the right-image.
(2) High image quality.
(3) Balanced image quality between the left-image and the right-image.
The first and the second criteria are obvious, and are addressed in Sections 3 and 4. The third criterion states that image quality should be balanced between the stereo pair. To avoid the discomfort caused by an unbalanced stereo pair, it is better to generate both the left-image and the right-image in medium accuracy than to generate one of them in high accuracy and one of them in low accuracy. Here image quality is judged based on the result of the corresponding single-eye rendering algorithm, and can be calculated using methods such as mean square error. For example, in [2], since the re-projected position of a left-eye sampling point is simply rounded to the nearest pixel on the right-image, the right-image is of lower accuracy than the left-image. To generate balanced images, the idea of re-projection can be intuitively extended by placing the viewpoint in the middle between the left and the right eyes, and re-projecting the sampling points on the *middle-eye* rays to both the left-image and the right-image. Since the angle between the middle-point rays and the left-rays is half of the angle between the left-rays and the right-rays, balanced image quality is achieved. Unfortunately, depending on the order of the rays cast (left to right scanline order or vice versa), one of the stereo pair images must be composed in a back-to-front order, which prevents the use of early ray termination. As described in Section 5, this problem can be solved by applying the stereo shear-warp volume rendering scheme.

3. Segment Composition

As mentioned above, the main idea of the fast stereo volume rendering proposed in [2] is to re-project the sampling points of the left-rays onto the right-image. By placing the origin at the left center of projection, the re-projected pixel on the right-image can be found with two multiplications, two additions, and one round operation.

One of the main problems of the above approach is that re-projection and composition are performed on the right-image for each sampling point on the left-rays. To further accelerate the generation of the right-image, we develop a segment composition scheme. The basic idea is to re-project and composite several sampling points simultaneously. Before we discuss the scheme, we first define the timesaving V for the right-image generation as:

$$V = 1 - \frac{T_{l+r} - T_l}{T_l} \tag{1}$$

where T_{l+r} is the total time for the rendering of a stereo pair, T_l is the time for the rendering of the left-image, and T_r is the time for the rendering of the right-image. Usually, the angle between the left-rays and the right-rays, ϕ, is small. For example, Hodges [6] has recommended that ϕ should be smaller than 1.5 degrees. As a result, a series of sampling points along a single left-ray could be re-projected to the same right-eye pixel. This effect is illustrated in Figure 1, where ϕ has been greatly exaggerated for the legibility of the figure.

Mathematically, we assume that a sampling point (x_p, y_p, z_p) is projected to (x_{sl}, y_{sl}) of the left-image and (x_{sr}, y_{sr}) of the right-image, and the distance between the left and the right centers of projection is e. If we assign

$$c = x_p \cos\phi - e \cos\phi/2 \tag{2}$$

then for a left-ray, *Ray*, cast from (x_{sl}, y_{sl}), c is a constant, and

$$x_{sr} = round\ (c + z_p \sin\phi). \tag{3}$$

Assume that

$$c + z_p \sin\phi = n \tag{4}$$

where n is an integer, then for all the sampling points on *Ray* in the interval:

$$[(x_p, y_p, z_p - 0.5/\sin\phi), (x_p, y_p, z_p + 0.5/\sin\phi)] \tag{5}$$

we have

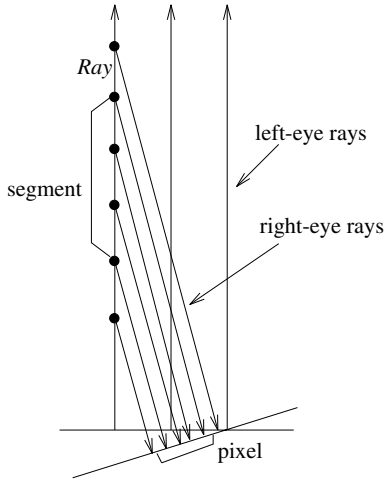


Figure 1: *Segment re-reprojection.*

$$x_{sr} = n. \tag{6}$$

Equation 6 states that instead of re-projecting the sampling points on *Ray* one by one, we can re-project segments of *Ray* with length $l = 1/\sin\phi$ each, and composite the color and opacity of the segments onto a pixel on the right-image. The composition of segments is straightforward, and we present the formulas using the simple over operations [14]. To simplify the calculations, the translucency $t = 1 - \alpha$ is used in our implementation instead of opacity α. Suppose that the color and the accumulated translucency are C_{in} and t_{in}, respectively, at the entry point of a certain segment, and t_{out} and C_{out}, respectively, at the exit point, the translucency t_{seg} and color C_{seg} of this segment can be calculated as:

$$t_{seg} = \frac{t_{out}}{t_{in}} \tag{7}$$

$$C_{seg} = \frac{C_{out} - C_{in}}{t_{in}(1 - t_{seg})} \tag{8}$$

and the composition on the re-projected pixel is:

$$C_{pixel} = C_{pixel} + C_{seg}\ t_{pixel}\ (1 - t_{seg}) \tag{9}$$

$$t_{pixel} = t_{pixel}\ t_{seg}. \tag{10}$$

The main advantage of this segment composition scheme is that the re-projection and composition time on the right-image can be greatly reduced. Since composition time could take most of T_r, the timesaving V is increased. Mathematically, assuming that the sampling distance on the image plane is d_{img}, and the average sampling distance along the left-rays is d_{ray}, the number of composition operations needed using segment composition is about $(d_{ray}\sin\phi)/d_{img}\ 100\%$ of the number of composition operations needed for re-projecting each sample point as in [2]. For example, when $d_{img} = d_{ray}$, and even when $\phi = 5^o$, which is three times larger than recommended, the composition operations needed for segment by segment re-projection is only about 8.7% of that needed for point by point re-projection. Of course, the actual timesaving depends on the relationship between d_{ray} and d_{img} and the operations used for composition. It also depends on the translucency of the specific dataset, because of the utilization of early ray termination.

Early ray termination [11] is a commonly used volume rendering acceleration method. It states that when the accumulated opacity along a certain ray that emanated from the eye-point reaches a certain threshold (e.g., 1), the ray traversal can be terminated. Yet for the fast stereo volume ray casting as described above, if a left-ray is terminated early, the area behind the termination point becomes unknown. To guarantee the correct composition on the right-image, right-eye re-projection of any sampling point should not go across this area in the first pass. However, the disadvantage of this approach is that left-eye sampling points behind the termination points that could contribute to the right-image are ignored in the first pass, and a second pass is thus needed.

In our approach, the left-ray is not terminated when the opacity threshold is reached. Instead, we first continue to traverse the sampling points inside the same segment, then jump to the entry point of the next segment which corresponds to the next non-opaque right-eye pixel, and re-project the sampling points inside that segment. The same process is applied to all the segments that correspond to non-opaque right-eye pixels and are behind the early termination point. As a result, the corresponding right-ray can also be terminated early. Note that after the opacity threshold is reached, the sampling points behind the left-ray termination point are re-projected one by one, not segment by segment, onto the right-image. This process is illustrated in Figure 2. Unlike [2], in our algorithm the right-image is generated solely from the left-eye sampling points. Both time and space are saved by omitting the overhead of re-examining and re-casting the unfinished right-rays, and by not keeping the last projection position for each right-eye pixel.

4. Linearly-interpolated Re-projection

A very important criteria for stereo volume rendering is image quality. To increase the image quality of the right-image, we propose a linearly-interpolated re-projection scheme. In the simple re-projection method discussed in [2] and in Section 3, a right-ray is the zero-order interpolation between the corresponding left-rays. This is

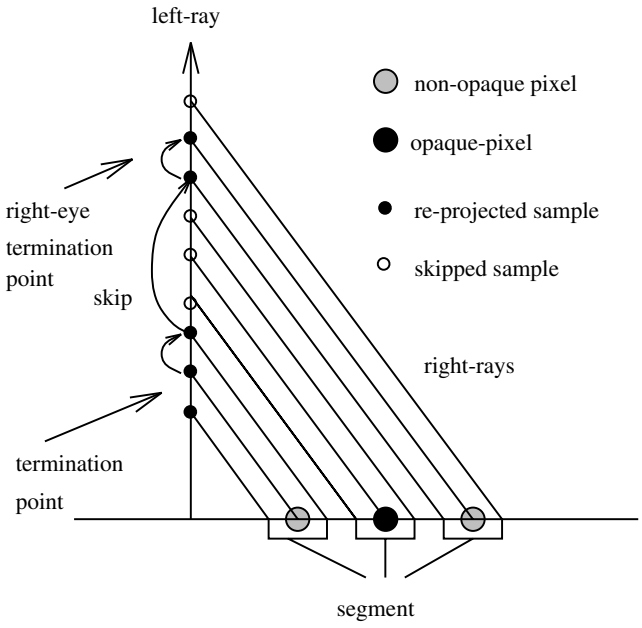


Figure 2: *Early ray termination.*

illustrated in Figure 3(a), where segment S_r on a right ray is equivalent to segment S_l on the neighboring left-ray. To increase image quality, linear-interpolation can be applied. One of the difficulties, though, is that since composition is not a linear process, sampling values for the right ray must be calculated, as illustrated in Figure 3(b).

To derive the value of the sampling points along S_{r1}, both the sampling points on S_{l1} and S_{l2} are needed. In other words, if a scanline ray traversal order is adopted, sampling results of the previous rays must be saved. Similar to Section 2, we notice that each sampling point on a left-ray affects one sampling point on each of the two neighboring right-rays. Similar to that for segment composition, each left-ray can be divided into several segments with length $d_{img}/\sin\phi$, and all the sampling points in a segment affect the same right-eye pixels. The linear-interpolation algorithm is thus similar to the simple re-projection, where the right to left scanline order is adopted. Specifically, for each scanline:

(1) Process the right-most left-ray:
For the *nth* sampling point, save the color, opacity, affected right-eye pixel x coordinate, and the weight in the *nth* element of a 1D array *Ray*.

(2) For each remaining left-ray from right to left:
(a) For the *nth* sampling point on the ray, generate the right-eye sample values as:
$R\_Sample_{(c,\alpha)} = Sample_{(c,\alpha)} \times (1 - Ray[n].weight)$
$+Ray[n].(c,\alpha) \times Ray[n].weight$.
(b) Re-project the right-eye sampling points onto $Ray[n].pixel$.
(c) $Ray[n].weight = 1 - Ray[n].weight$
$Ray[n].pixel = Ray[n].pixel - 1$
$Ray[n].(c,\alpha) = Sample_{c,\alpha}$

To apply early ray termination, we apply an idea similar to the one presented in Section 3. When the accumulated opacity of the left-ray is above a certain threshold, it is not terminated. The remaining sampling points on the ray are checked segment by segment, but re-projected point by point. If both the right-eye pixels affected by a left-ray segment are opaque, the segment is skipped.

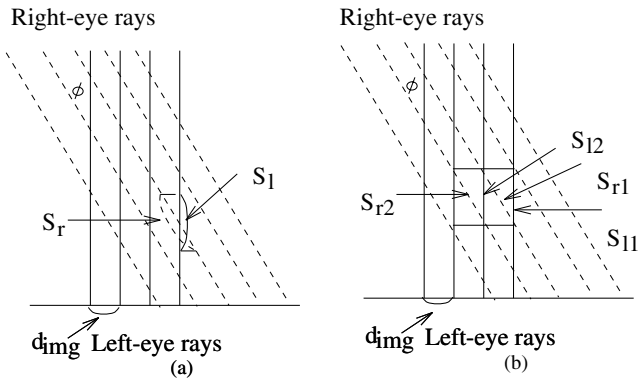


Figure 3: *Zero order vs. linear interpolation.*

Although the linearly-interpolated re-projection is more accurate than the zero-order interpolation, it is still not equivalent to the standard rendering image. The reason is that in the standard ray casting, interpolation is performed within the neighboring voxels, not between neighboring rays. The commonly used tri-linear interpolation is a high order interpolation along the non-main-axis direction, and the viewing direction is generally non-orthogonal. Nevertheless, the linearly-interpolated re-projection scheme is more accurate, and taking advantage of the re-projection pattern as described above, it is still much faster than separately rendering the left-eye and right-eye images.

5. Stereo Shear-warp Volume Rendering

In general, a volume ray casting algorithm can be divided into three main steps, [10] traversing and sampling along the ray, shading the sampling point, and compositing the color. For stereo volume ray casting, there is generally a tradeoff between speed and the achieved image quality. Re-projecting the left-eye sampling points achieves very high timesaving for right-image generation because it skips the traversing/sampling and shading steps, and shortens the color composition step by using segment composition. The linear interpolation approach simplifies the traversing/sampling step, and skips the shading step by directly interpolating the color and opacity from the left-eye sampling points.

The standard ray casting algorithms used for stereo rendering cast rays in such a *ray-by-ray* order. For example, both [2] and the segment composition described above follow the scanline order for casting rays, and process the rays one by one. In Section 2, we have discussed the difficulties of generating balanced stereo images through re-projection when applying such a ray-by-ray order. Recently a volume rendering scheme based on *slice-by-slice* traversing order has been developed (e.g. [15, 4, 9, 13],). The basic idea of slice-by-slice rendering is to first shear the original slices so that the viewing direction is parallel to a main axis, then traverse consecutive data slices perpendicular to this main axis in scanline order. Among such algorithms, shear-warp volume rendering proposed by Lacroute and Levoy [9] further takes advantage of the data coherence, and achieves high performance. It is a hybrid of image-order ray casting and object-order projection algorithms, and it takes advantage of both. In this section, we focus on the application of shear-warp to stereo rendering. We first discuss briefly the shear-warp volume rendering scheme, and then propose a stereo shear-warp algorithm.

5.1. Shear-warp rendering

For parallel projection, the main steps of shear-warp volume rendering are:

(1) Preprocessing the dataset, including pre-computing normals, opacities, and a shading lookup table.
(2) Transforming the volume so that the viewing direction is parallel to a main axis. Processing slices perpendicular to this main axis in a front-to-back order. Jumping opaque pixels, and skipping empty voxels by marching along the intermediate image scanlines and the voxels simultaneously.
(3) Shading, re-sampling, and compositing the voxels that are not skipped.
(4) Warping the intermediate image onto the final image.
The reasons that shear-warp rendering achieves fast speed include performing pre-computation, accessing the memory consecutively, directly warping the low resolution image into the high resolution image, bi-linear interpolating slices instead of tri-linear interpolating the volume, and adopting view-dependent sampling rate.

On the other hand, the performance of shear-warp is closely related to the property of the dataset and the opacity transfer function. More specifically, since image quality is decided by the volume resolution through a 2D warp, it does not improve with the increase of the image resolution. Unfortunately, even with a very smooth transfer function, this 2D warp can not generate a high quality image. The complete proof of this claim is out of the scope of this paper, and here we only outline the general steps. We first accept the basic assumption of volume rendering that the input signal is band-limited and that the original signal is properly sampled. In other words, the continuous signal represented by the volume can theoretically be perfectly reconstructed. We then assume that simple density accumulation, instead of complex non-linear composition, is used for image rendering. Therefore, the Fourier slice projection theorem [12] can be applied. The theorem states that the projection of the 3D data volume in a certain view direction can be obtained by extracting a 2D slice perpendicular to that view direction out of the 3D Fourier spectrum, and inverse Fourier transforming it. Suppose that the resolution of the original volume is N^3; the resolution of the slice extracted from the spectrum can be $(\sqrt{3}N)^2$ when the viewing direction is $\pi/4$ to one of the main plane. Thus, the highest frequency of the projected image can be $\sqrt{3}$ times the highest frequency in the volume. As a result, the image resolution should be at least $\sqrt{3}$ of the volume resolution. When the complex non-linear composition is incorporated, the required image resolution should be even higher. Another place where inaccuracy is introduced for shear-warp is the view-dependent sampling distance. For parallel rendering of uniform grids, the sampling distance along $\pi/4$ viewing direction is $\sqrt{3}$ of that along $\pi/2$ viewing direction. Again, we can prove that the sampling distance along $\pi/4$ viewing direction should be at least as high as that along $\pi/2$ viewing direction.

5.2. Stereo Shear-warp

There are several methods,based on the shear-warp approach, of generating both the left-image and the right-image in one pass by trading speed for quality. Conventional shear-warp can be first applied for the rendering of the left-image; then sampling points are appropriately re-projected onto the intermediate right-image. Special attention must be paid to using early ray termination for the right-image. A more accurate approach is to linearly interpolate the left-eye sampling points for the generation of the right-image. In this section, we describe how to apply the segment composition scheme for stereo shear-warp. In the discussion, we assume that both the left-rays and the right-rays are perpendicular to the same volume slices in the sheared object space. When this condition is not satisfied, several solutions are proposed.

Figure 4 illustrates the relationship between the viewing direction and the volume coordinate system. Assume that the left-ray direction is (α_1, β_1) and the right-ray direction is (α_2, β_2). Note that the translation between the left-image and the right-image origins does not affect the discussion below. The shear factor for the *nth* slice, starting from the nearest slice most perpendicular to the viewing direction, can therefore be calculated as :

$$\begin{cases} V_{left} = -ndz(\dfrac{\cos\alpha_1 \cos\beta_1}{\sin\beta_1}, \dfrac{\sin\alpha_1 \cos\beta_1}{\sin\beta_1}, 1) \\ V_{right} = -ndz(\dfrac{\cos\alpha_2 \cos\beta_2}{\sin\beta_2}, \dfrac{\sin\alpha_2 \cos\beta_2}{\sin\beta_2}, 1) \end{cases}. \tag{11}$$

Since the angle between the left-ray and the right-ray is ϕ,

$$\frac{V_{left} \bullet V_{right}}{|V_{left}||V_{right}|} = \cos\phi \tag{12}$$

yielding:

$$\cos\beta_1 \cos\beta_2 \cos(\alpha_1 - \alpha_2) + \sin\beta_1 \sin\beta_2 = \cos phi. \tag{13}$$

Therefore,

$$\alpha_1 - \alpha_2 \le \phi \quad \text{and} \quad \beta_1 - \beta_2 < \phi. \tag{14}$$

Without loss of generality, we assume that $\beta_1 \ge \beta_2$ and

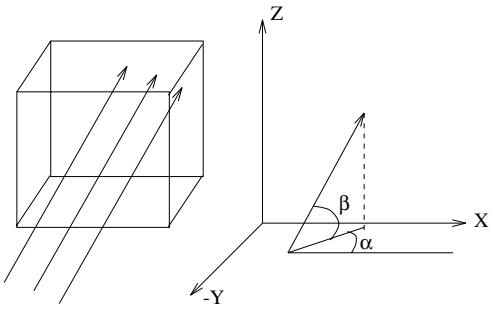


Figure 4: *Relation between viewing rays and volume coordinate system.*

$\beta_1 \geq \pi/4$. Then as in Section 3, we use the zero-order interpolation on the right-image. For a point $v_{x,y,z}$ on a left-ray in the volume coordinate system,

$$(x + ndz\,\frac{\cos\alpha_1 \cos\beta_1}{\sin\beta_1},\ y + ndz\,\frac{\sin\alpha_1 \cos\beta_1}{\sin\beta_1},\ z + ndz). \quad (15)$$

The projection of $v_{x,y,z}$ onto the left-image is:

$$P_l = (x, y) \quad (16)$$

and the projection of $v_{x,y,z}$ onto the right-image is:

$$P_r = (x + ndz(\frac{\cos\alpha_1 \cos\beta_1}{\sin\beta_1} - \frac{\cos\alpha_2 \cos\beta_2}{\sin\beta_2}),$$

$$y + ndz(\frac{\sin\alpha_1 \cos\beta_1}{\sin\beta_1} - \frac{\sin\alpha_2 \cos\beta_2}{\sin\beta_2})). \quad (17)$$

In other words, the offset between the left-eye projection and the right-eye projection is the same for all the sampling points on the same slice. Let:

$$\begin{cases} C_x = \dfrac{\cos\alpha_1 \cos\beta_1}{\sin\beta_1} - \dfrac{\cos\alpha_2 \cos\beta_2}{\sin\beta_2} \\ C_y = \dfrac{\sin\alpha_1 \cos\beta_1}{\sin\beta_1} - \dfrac{\sin\alpha_2 \cos\beta_2}{\sin\beta_2} \end{cases}. \quad (18)$$

since $a_1 - a_2 \leq \phi$:

$$C_x^2 + C_y^2 \leq (\frac{\sin(\beta_1 - \beta_2)}{\sin\beta_1 \sin\beta_2})^2 + 2(\frac{\cos(\beta_1 - \beta_2) - \cos\phi}{\sin\beta_1 \sin\beta_2}) \quad (19)$$

Let

$$\phi_1 = \pi/4 - (\beta_1 - \beta_2) \quad (20)$$

since $\beta_1 \geq \pi/4$,

$$C_x^2 + C_y^2 \leq (\frac{\sin(\pi/4 - \phi_1}{\sin\pi/4 \sin\phi_1)})^2 + 2(\frac{\cos(\pi/4 - \phi_1) - \cos\phi}{\sin\pi/4 \sin\phi_1)}$$

$$= \frac{1}{\sin^2\phi_2} + 2 - (\frac{4\cos\phi}{\sqrt{2}})\frac{1}{\sin\phi_2}. \quad (21)$$

Clearly, the right of Inequation 22 achieves maximum value when $\phi_2 = \pi/4 - \phi$. If we assume that $\phi = 5^o$, three times bigger than recommended, then,

$$C_x^2 + C_y^2 \leq 0.03676944. \quad (22)$$

In other words,

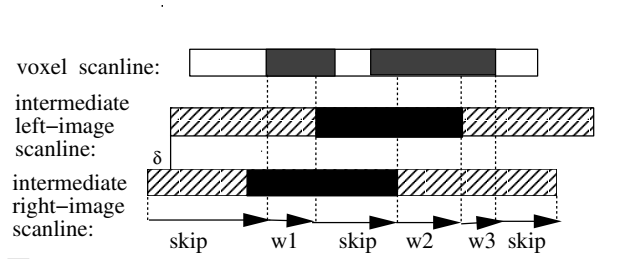
$$C_x \leq 0.19175359 \ \text{ and } \ C_y \leq 0.19175359. \quad (23)$$

This states that on the average, we can re-project the left-eye composition result of at least five slices onto the right-image when the image sampling distance is equivalent to the volume sampling distance. Since the actual ϕ is usually much smaller, a higher acceleration rate can be achieved through segment composition.

The key idea of stereo shear-warp is thus to apply the conventional shear-warp for the left-image, but re-project the composition result of several left-eye slices to the right-image. To assure the correct early ray termination, a left-eye intermediate image scanline and the corresponding right-eye intermediate image scanline are marched simultaneously with the voxel scanline. Only when a pixel on both of the scanlines is opaque is the corresponding voxel skipped. This process is illustrated in Figure 5. Note that in Figure 5, there are three different ways to process a non-transparent voxel. The first is simply to composite on the left-eye pixel when the corresponding right-eye pixel is opaque. The second is to composite with the right-eye pixel when the left-eye pixel is opaque and the left-ray is terminated earlyi. The third way is to composite on the left-eye pixel, but only re-project and composite on the right-eye pixel when the integer right-eye re-projection offsets (δ_x, δ_y) are changed. Since C_x and C_y in Equation 18 are constants, these changes can be detected by two additions and comparisons for each slice.

Compared to the ray-by-ray order stereo ray casting, stereo shear-warp further simplifies right-image calculations because of the constant offsets for each slice, as illustrated in Figure 5. Balanced images can be generated by placing the viewpoint in the middle-eye, applying the conventional shear-warp algorithm, and re-projecting the middle-eye results onto both the left-image and the right-image. Since the slices are processed in a front-to-back order, early ray termination for both the left-rays and the right-rays can be applied. By using this middle-eye algorithm, the angle between the middle-ray and both the left-ray and the right-ray is reduced to half the angle between the left-ray and the right-ray. As a result, the length of a segment on the middle-ray is doubled.

The algorithm discussed above works only when both the left-rays and the right-rays are parallel to the same main axis in the sheared object space. When this condition is not satisfied, a simple solution is to separately render the left-image and the right-image under such situations. However, since the angle ϕ between the left-eye and right-eye viewing direction is very small, we can assume that



w1 : normal marching for left−eye image
w2: re−projecting to right−eye image pixel point by point
w3: re−projecting to right− pixel using segment composition
δ : integer offsets in x and y directions

Figure 5: *Scanline marching for stereo shear-warp.*

both the left-rays and the right-rays are perpendicular to the same volume slice, and continue to apply the above algorithm with little sacrifice of the sampling distance. The maximum enlargement of the sampling distance is 108% at the worst case when ϕ is 5^o. When the middle-eye algorithm is used, the enlargement is even smaller. It can be proved that by following an appropriate scanline order, correct slice composition can still be achieved.

6. Results

We have implemented the segment composition scheme as outlined in Section 3, together with the stereo shear-warp algorithm. All the experiments were conducted on a Silicon Graphics High Impact, equipped with one 250MHZ R4400 processor and 128MB of RAM.

The implementation of segment composition is based on the high quality volume ray casting of the public domain *VolVis* system, [3] developed at Stony Brook. Figure 6(top) presents the rendering of a negative potential of a high potential iron protein using the segment composition scheme. The resolution of the dataset is $66 \times 66 \times 66$; the image size is 256×256. We set the angle between the left-eye and the right-eye viewing directions to be 1^o, and the maximal voxel opacity in the data to be 0.1. The total rendering time for the stereo pair is 8.61 sec. For comparison, Figure 6(bottom) presents the stereo pair generated by rendering the datasets separately from the left-eye and the right-eye view points using standard ray casting. The rendering time is 8.31 sec for the left-image, and 8.52 sec for the right-image. The timesaving V is therefore 97%.

The performance of segment composition is closely related to the translucency of the datasets. Generally, if a dataset is mostly opaque, T_r will be smaller because of the early ray termination. For example, in Figure 6 no left rays are terminated early. If we increase the maximal voxel opacity in the data to be 0.5, 27% of the left-rays are early terminated. The total rendering time for both the left-image and right-image using segment composition is 7.98 sec. The time for separately rendering left-image and right-image is 7.20 and 7.31 sec, respectively, and T_r is 89%.

Our implementation of the stereo shear-warp volume rendering algorithm is based on the public domain VolPack volume renderer package. [9] For comparison, we use the human head dataset enclosed in the package. Figure 7 presents the shear-warp rendering of the MRI scan of a human head. The resolution of the dataset is $128 \times 128 \times 84$. The maximal voxel opacity has been set to be 1.0, and the opacity threshold has been set to be 0.95. For rendering, the volume has been pre-classified and shaded, which maximizes the rendering speed. The image size is 256×256, and the angle between the left-eye and the right-eye viewing direction is one degree.

Figure 7(top) presents the result of the stereo shear-warp rendering. The total rendering time is $188ms$. For comparison, Figure 7(bottom) presents the stereo pair generated by rendering the datasets separately from the left-eye and the right-eye viewpoints using conventional shear-warp. The rendering time is $131ms$ for the left-image and $144ms$ for the right-image, respectively. The timesaving S_r is therefore 60%.

One of the reasons that T_r is low is that our stereo shear-warp does not save time on 2D warping. For large datasets, the 2D warping time can usually be ignored. However, for small datasets such as the MRI head in this example, warping can take a significant portion of the time. The 2D warping time for the left-image and the right-image in the Figure 7(top) is $41ms$ and $38ms$, respectively; and that for the left-eye image and right-image in Figure 7(bottom) is $43ms$ and $43ms$, respectively. If we exclude the time for warping, the timesaving V is 79%. This warping-excluded timesaving is generally irrelevant to the resoultion of the datasets.

Another reason that T_r is relatively low is that the transfer function has been assigned so that the volume is mostly opaque. If we set the maximal voxel opacity to be 0.2, the total time for rendering using stereo shear-warp is $429ms$, while the warping time is $41ms$ and $44ms$, respectively. The time for separately rendering the left-image and right-image is $340ms$ and $354ms$, respectively, while the warping time is $42ms$ and $42ms$, respectively. The V is 75%, and the warping-excluded V is 85%.

In all the experiments performed, many parameters are precomputed. For example, generally the most time-consuming step of volume rendering, on-the-fly shading, has been replaced by pre-shading and table lookup on-the-fly. The result is the speedup of volume rendering. On the other hand, timesaving V for the right-eye image is much higher if these volume rendering parameters are calculated on-the-fly. The reason is that the shaded left-eye sampling points are directly re-projected to generate the right-eye image.

7. Conclusions and Future Work

In this paper, we have presented new volume ray casting techniques for generating high quality stereoscopic images by using segment composition and linearly-interpolated re-projection. Stereo volume rendering criteria are proposed to evaluate the fast stereo rendering algorithms. Existing volume rendering accelerators are discussed, and specifically, a new stereo shear-warp volume rendering has been designed. The efficiency of our algorithms are mathematically proved. Our experiments have presented a from 79% to 97% saving for the generation of the second image in a stereo pair. Compared to the previous algorithm, much higher acceleration rates have been achieved.

Perspective projection is inherent to stereo rendering. The idea of re-projecting the sampling points of the left-rays onto the right-image can be directly applied to perspective

projection. Segment composition is still valid, but the calculation becomes more complicated. For perspective shear-warp, each volume slice has different shear and scale factors for the left-image and the right-image, and a more complicated scanline marching scheme must be designed. However, the basic algorithm is still similar to that of Figure 5. Nevertheless, the timesaving V for perspective projection is expected to be lower than that for parallel projection. Further research has to be conducted in this area.

Fast stereo volume rendering with high quality for both images is still an open problem. The main reason is that the complicated color composition requires paticular distribution of the sampling points. For example, the rendering time of ray casting is basically decided by the number of sampling points. Unfortunately, if N sampling points are needed to generate a single-eye image, it is impossible to find $N + o$ sampling points satisfying the uniform distribution requirement on both the left and right rays, where o is a small number compared to N. The solution we are working on is to design new algorithms specifically for stereo volume rendering. We are also designing algorithms to apply the fast stereo rendering to recursive volumetric ray tracing. [16] Another research direction is to apply the same idea of fast stereo volume rendering to fast animation.

8. Acknowledgments

This work has been partially supported by the National Science Foundation under grants CCR-9205047 and MIP-9527694 and by the Department of Energy under the PICS grant. The high potential iron protein in Figure 6 is courtesy of Scripps Clinic, La Jolla, CA, and the MRI head in Figure 7 is included in VolPack.

References

1. S. Adelson and L. Hodges, "Stereoscopic Ray Tracing," *The Visual Computer* **10**(3) pp. 127-144 (December 1993).
2. S. J. Adelson and C. D. Hansen, "Fast Stereoscopic Images with Ray-Traced Volume Rendering," *1994 Symposium on Volume Visualization*, pp. 3-9 (October 1994).
3. R. Avila, T. He, L. Hong, A. Kaufman, H. Pfister, C. Silva, L. Sobierajski, and S. Wang, "VolVis: A Diversified Volume Visualization System," *IEEE Visualization'94 Proceedings*, pp. 31-38 (October 1994).
4. G. Cameron and P. Undrill, "Rendering Volumtetric Medical Image Data on a SIMD-architecture Computer," *Proceedings of the Third Eurographics Workshop on Rendering*, pp. 135-145 (Bristol, UK).
5. L. Hodges and S. McWhor, "Stereoscopic Display for Design Visualization," *Image Communication* **4**(1) pp. 3-13 (November 1991).
6. L. Hodges, "Time-Multiplexed Stereoscopic Computer Graphics," *IEEE Computer Graphics and Application* **12**(2) pp. 20-30 (March 1992).
7. J. Hsu, C. F. Babbs, D. M. Chelberg, Z. Pizlo, and E. J. Delp, "A Study of the Effecttiveness of Stereo Imaging Truth or Dare: Is Stereo Viewing Really Better?," *SPIE Proceedings 2177a: Stereoscopic Displays and Applications V*, pp. 211-222 (February 1994).
8. W. Krueger, "The Applications of Transport Theory to Visualization of 3D Scalar Data Fields.," *Computer in Physics*, pp. 397-406 (July/August 1991).
9. P. Lacroute and M. Levoy, "Fast Volume Rendering Using a Shear-Warp Factorization of the Viewing Transformation," *Computer Graphics Proceedings, Annual Conference Series, ACM SIGGRAPH*, pp. 451-458 (July 1994).
10. M. Levoy, "Display of Surfaces from Volume Data," *IEEE Computer Graphics and Applications* **8**(5) pp. 29-37 (May 1988).
11. M. Levoy, "A Hybrid Ray Tracer for Rendering Polygons and Volume Data," *IEEE Computer Graphics and Applications* **10**(3) pp. 33-40 (March 1990).
12. T. Malzbender, "Fourier Volume Rendering," *ACM Transactions on Graphics* **12**(3) pp. 233-250 (July 1993).
13. H. Pfister and A. Kaufman, "Cube -- A Scalable Architecture for Real-Time Volume Rendering," *ACM Volume Visualization Symposium*, (October 1996).
14. T. Porter and T. Duff, "Compositing Digital Images," *Computer Graphics (SIGGRAPH'84 Proceedings)* **18**(3) pp. 253-259 (July, 1984).
15. P. Schroder and G. Stoll, "Data Parallel Volume Rendering as Line Drawing," *Proceedings of the 1992 Workshop on Volume Visualization*, pp. 25-32 (October 1992).
16. L. Sobierajski and A. E. Kaufman, "Volumetric Ray Tracing," *1994 Symposium on Volume Visulization*, pp. 11-18 (October 1994).
17. Y. Yeh and L. Silverstein, "Visual Performance with Monoscopic and Stereoscopic Presentation of Identical Three-Dimensional Visual Tasks," *1990 SID International Symposium Digest of Technical Papers*, pp. 359-362 (May 1990).

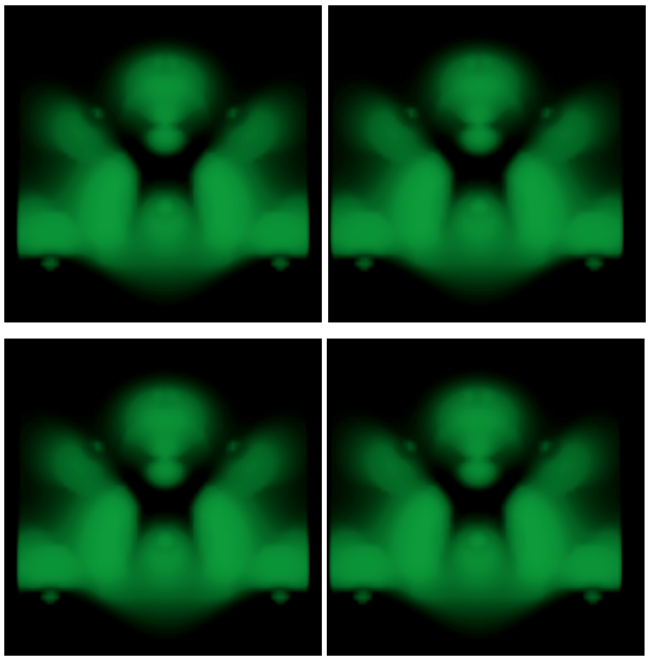


Figure 6: *Fast stereo rendering of a negative potential of a high potential iron protein. (top) a stereo pair generated using segment composition, (bottom) the same stereo pair generated using the standard ray casting.*

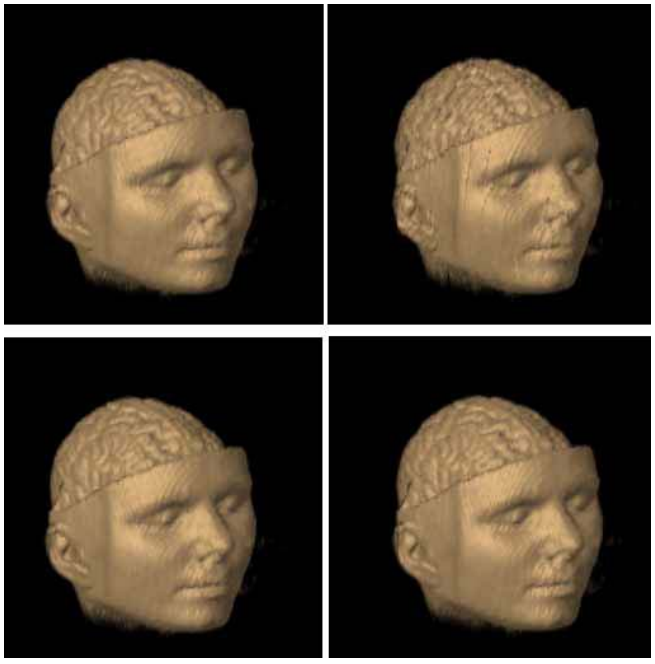


Figure 7: *Fast stereo shear-warp rendering of an MRI head. (top) a stereo pair generated using fast stereo shear-warp volume rendering, (bottom) the same stereo pair generated using the standard shear-warp rendering.*