Volumetric Warping for Voxel Coloring on an Infinite Domain

Gregory G. Slabaugh[1], Thomas Malzbender[2], and W. Bruce Culbertson[2]

[1] School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332
slabaugh@ece.gatech.edu
[2] Client and Media Systems Lab
Hewlett-Packard Laboratories
Palo Alto, CA 94306
{tom_malzbender, bruce_culbertson}@hp.com

Abstract. Starting with a set of calibrated photographs taken of a scene, voxel coloring algorithms reconstruct three-dimensional surface models on a finite spatial domain. In this paper, we present a method that warps the voxel space, so that the domain of the reconstruction extends to an infinite or semi-infinite volume. Doing so enables the reconstruction of objects far away from the cameras, as well as reconstruction of a background environment. New views synthesized using the warped voxel space have improved photo-realism.

1 Introduction

Voxel coloring algorithms [7] [5] [2] reconstruct three-dimensional surfaces using a set of calibrated photographs taken of a scene. When working with such algorithms, one typically defines a *reconstruction volume*, which is a bounding volume containing the scene that is to be reconstructed. Once defined, the reconstruction volume is divided into voxels, forming the voxel space in which the reconstruction will occur. Voxels that are consistent with the photographs are assigned a color, and inconsistent voxels are removed (carved) from the voxel space [7].

These algorithms have been particularly successful in reconstructing small-scale scenes that are restricted to a finite domain. Applying them to large-scale scenes can become challenging, since one must use a large reconstruction volume to contain the scene. Such a large reconstruction volume can consist of an unwieldy number of voxels that becomes prohibitive to process. In addition, it is unnecessary to model far away objects with high resolution voxels. Ideally, one would like a spatially adaptive voxel size that increases away from the cameras.

Furthermore, voxel coloring algorithms are not well suited to capturing the environment (sky, background objects, etc.) of a scene. Typical reconstructions are photo-realistic in the foreground, which is modeled, but empty in the background, which is unmodeled. As a result, synthesized new views can have large

"unknown" regions, as shown in black in Figure 1. For some scenes, such as an outdoor scene, we might like to reconstruct the background as well, yielding a more photo-realistic reconstruction.

(a)   (b)
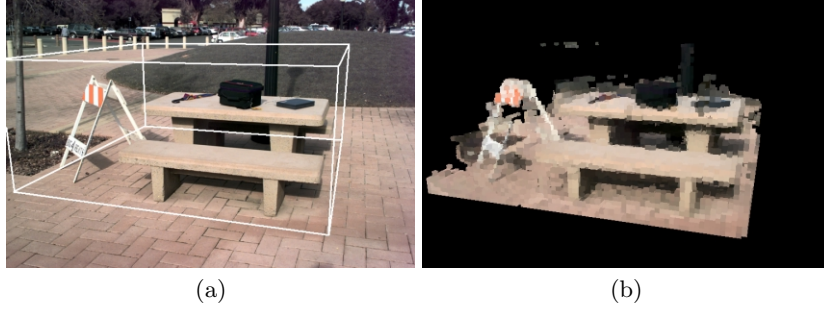
Fig. 1. Unknown regions due to reconstruction on a finite domain. A photograph of our "bench" scene is shown in (a), with the reconstruction volume superimposed. Only voxels within the reconstruction volume are considered in voxel coloring algorithms. The scene contains many objects outside of the reconstruction volume that are not reconstructed, resulting in unknown regions that appear as black in a projection of the reconstruction, shown in (b). The ideas presented in this paper warp the voxel space, so that the reconstruction volume can become infinite, and the background scene and environment can be reconstructed.

To address these issues, we propose a warping of the voxel space so that surfaces farther away from the cameras can be modeled without an excessive number of voxels. In addition, our proposed warping of the voxel space can extend to infinity along any dimension, so that infinite (all of R^3), or semi-infinite (such as a hemisphere with infinite radius) reconstruction volumes can be defined. The latter might best model an outdoor scene. As will be shown in subsequent sections of this paper, we develop a hybrid voxel space consisting of an interior space in which voxels are not warped, and an exterior space in which voxels are warped. The voxels are warped so that the following criteria are met:

1. No warped voxels overlap.
2. No gaps form between warped voxels.
3. The warped reconstruction volume is at least semi-infinite.

A voxel coloring algorithm is then executed using the warped reconstruction volume.

The layout of this paper is as follows. First, we explore some related work. Then, we introduce a function that warps the voxel space subject to the criteria enumerated above. Next, we discuss some implementation details that arise when performing a reconstruction in warped space. We then present results that demonstrate the effectiveness of our approach.

2   Related Work

The work presented in this paper is an extension to recent volumetric solutions to the three-dimensional scene reconstruction problem. Seitz and Dyer's [7] voxel coloring technique exploits color correlation of surfaces to find a set of voxels that are consistent with the photographs taken of a scene. Kutulakos and Seitz [5] develop a space carving method that extends voxel coloring to support arbitrary camera placement via a multi-sweep algorithm. Culbertson, Malzbender, and Slabaugh [2] present two generalized voxel coloring (GVC) algorithms, which, like [5] allow for arbitrary camera placement, and in addition use the exact visibility of the scene when determining if a voxel is consistent with the photographs. These three methods, referred to collectively as "voxel coloring algorithms", have been quite successful in reconstructing three-dimensional scenes on a finite spatial domain. In this paper, we extend these three methods in order to reconstruct scenes on an infinite or semi-infinite domain by warping the voxel space used in the reconstruction. Doing so enables the reconstruction of nearby objects, far-away objects, and everything in between.

Saito and Kanade [6], and later Kimura, Saito, and Kanade [4] specify a voxel space using the epipolar geometry relating two [6] or three [4] basis views, for volumetric reconstruction using weakly calibrated cameras. In their approach, a voxel takes on an arbitrary hexahedral shape, a consequence of their projective space. In our approach, we intentionally warp exterior voxels into arbitrarily shaped hexahedra. In [6] and [4], a voxel's size is solely based on its location relative to the cameras that form the basis. In our approach, a voxel's size is instead based on its location in a user-defined voxel space. In [6] and [4], the reconstruction volume is finite, and only foreground surfaces are reconstructed. In contrast, our method warps the voxel space to infinity so that objects far from the cameras can be reconstructed, in addition to foreground surfaces.

In the computer graphics domain, infinite scenes have been modeled and rendered using environment mapping. This method projects the background onto the interior of a sphere or cube that surrounds the foreground scene. Blinn and Newell [1] use such a technique to synthesize reflections of the environment off of shiny foreground surfaces, a procedure also known as reflection mapping. Greene [3] additionally renders the environment map directly to generate views of the background. This approach is quite effective at producing convincing synthetic images. However, since the foreground and background are modeled differently, separate mechanisms must be provided to create and render each. Furthermore, the three-dimensionality of the environment is lost, as the background is represented as a texture-map. Like environment mapping, the techniques described in this paper seek an efficient mechanism to represent the background scene. Our warped volumetric space provides this in a single framework that can more easily accommodate surfaces that appear both in the foreground and background. In addition, we reconstruct the background scene three-dimensionally using computer vision methods.

3   Volumetric warping

The goal of a volumetric warping function is to represent an infinite or semi-infinite volume with a finite number of voxels, while satisfying the requirement that no voxels overlap and no gaps exist between voxels. There are many possible ways to achieve this goal. In this section, we use the term *pre-warped* to refer to the volume before the volumetric warping function is applied.

The volumetric warping method presented here separates the voxel space into an interior space used to model foreground surfaces, and an exterior space used to model background surfaces, as shown in Figure 2 (a). The volumetric warp does not affect the voxels in the interior space, providing backward compatibility with previous voxel coloring algorithms, and allowing reconstruction of objects in the foreground at a fixed voxel resolution.

(a)   (b)
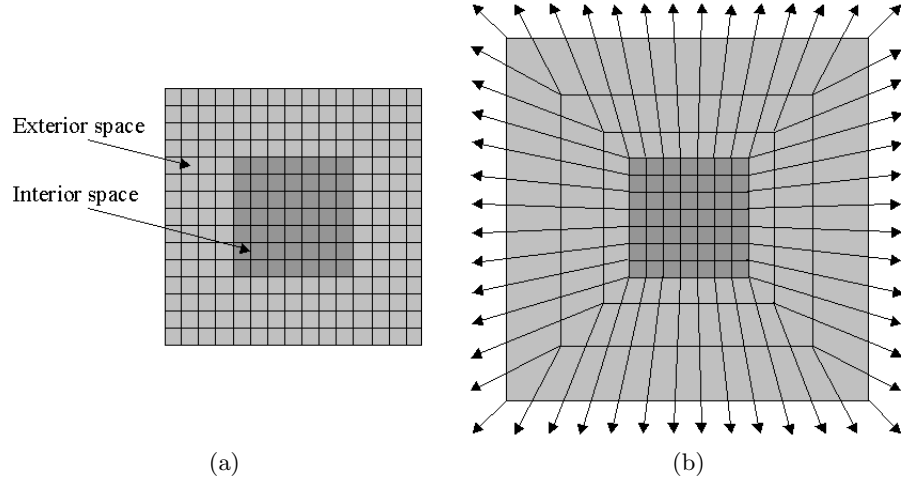
Fig. 2. Pre-warped (a) and warped (b) voxel spaces shown in two dimensions. In (a), the voxel space is divided into two regions; an interior space shown with dark gray voxels, and an exterior space shown with light gray voxels. Both regions consist of voxels of uniform size. The warped voxel space is shown in (b). The warping does not affect the voxels in the interior space, while the voxels in the exterior space increase in size further from the interior space. The outer shell of voxels in (b) are warped to infinity, and are represented with arrows in the figure.

Voxels in the exterior space are warped according to a warping function that changes the size of the voxel based on its distance from the interior space. The further a voxel in the exterior space is located from the interior space, the larger its size, as shown in Figure 2 (b). Voxels on the outer shell of the exterior space have coordinates warped to infinity, and have infinite volume. Note that while

the voxels in the warped space have a variable size, the voxel space still has a regular 3D lattice topology.

To help further limit the class of possible warping functions, we introduce the following desirable property of a warped voxel space:

Constant footprint property: For each image, voxels project to the same number of pixels, independent of depth.

Figure 3 shows an example of a voxel space that satisfies the constant footprint property for two cameras. Assuming perspective projection, a voxel space that satisfies this property has a spatially adaptive voxel size that increases away from the cameras, in a manner perfectly matched with the images. While a useful conceptual construct, the constant footprint property cannot in general be satisfied when more than n cameras are present in R^n space. Thus, for three-dimensional scenes, a voxel space cannot be constructed that satisfies the property for general camera placement when there are more than three cameras. Since reconstruction using three or less cameras is limiting, we instead design our volumetric warping function to approximate the constant footprint property for an arbitrary number of images.
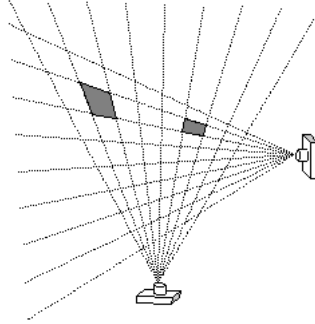
Fig. 3. Example of a 2D voxel space that satisfies the constant footprint property for two images. Notice that the two filled in voxels project to the same number of pixels in the right image, regardless of their respective distance from the camera. Note that this figure is solely used to illustrate the constant footprint property; the warped voxel space developed and used in this paper actually looks like that of Figure 2 (b).

3.1   Frustum Warp

In this subsection, we describe a frustum warp function that is used to warp the exterior space. We develop the equations and figures in two dimensions for simplicity; the idea easily extends to three dimensions.

The frustum warp assumes that both the interior space and the pre-warped exterior space have rectangular shaped outer boundaries, as shown in Figure 4. The pre-warped exterior space is divided into four trapezoidal regions, bounded by (1) lines l connecting the four corners of the interior space to their respective corners of the exterior pre-warped space, (2) the boundary of the interior space, and (3) the boundary of the pre-warped exterior space. We denote these trapezoidal regions as $\pm x$, and $\pm y$, based on the region's relative position to center of the interior space. These regions are also shown in Figure 4.

Let (x, y) be a pre-warped point in the exterior space, and let (x_w, y_w) be the point after warping. To warp (x, y), we first apply a warping function based on the region in which the point is located. This warping function is applied only to one coordinate of (x, y). For example, suppose that the point is located in the $+x$ region, as depicted in Figure 5. Points in the $+x$ and $-x$ regions are warped using the x-warping function,

$$x_w = x \frac{x_e - x_i}{x_e - |x|},$$
(1)

where x_e is the distance along the x-axis from the center of the interior space to the outer boundary of the exterior space, and x_i is the distance along the x-axis from the center of the interior space to the outer boundary of the interior space, shown in (a) of Figure 5. A quick inspection of this warping equation reveals its behavior. For a point on the boundary of the interior space, $x = x_i$, and thus $x_w = x_i$, so the point does not move. However, points outside of the boundary get warped according to their proximity to the boundary of the exterior space. For a point on the boundary of the exterior space, $x = x_e$, and so $x_w = \infty$.
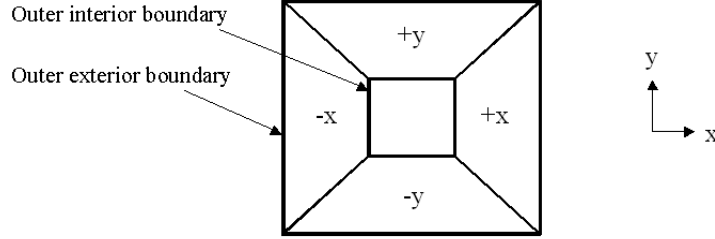
Fig. 4. Boundaries and regions. The outer boundaries of both the interior and exterior space are shown in the figure. The four trapezoidal regions, $\pm x$ and $\pm y$ are also shown.

Continuing with the above example, once x_w is computed, we find the other coordinate y_w by solving a line equation,

$$y_w = y + m(x_w - x),$$
(2)

where m is the slope of the line connecting the point (x, y) with the point a, shown in (b) of Figure 5. Point a is located at the intersection of the line parallel

to the x-axis and running through the center of the interior space, with the nearest line l, as shown in the figure. Note that in general, point a is not equal to the center of the interior space.

(a)

(b)
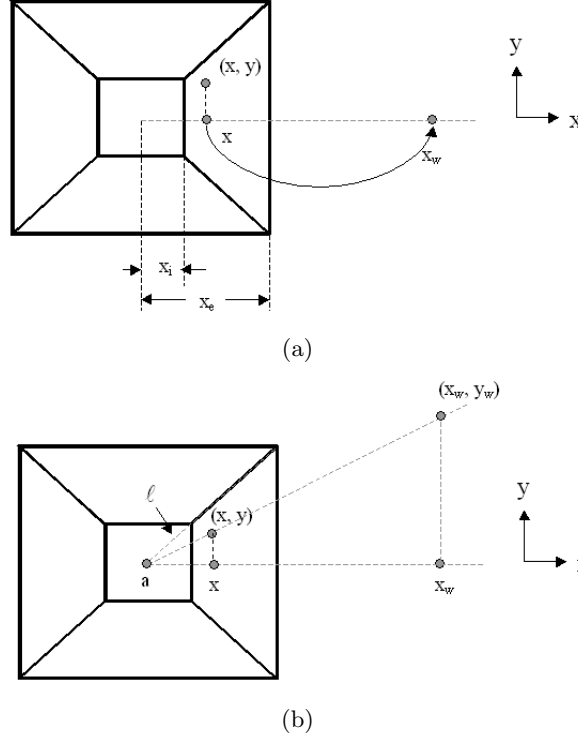
Fig. 5. Finding the warped point. The x-warping function is applied to the x-coordinate of the point (x, y), as the point is located in the $+x$ region. This yields the coordinate x_w, shown in (a). In (b), the other coordinate y_w is found by solving the line equation using the coordinate x_w found in (a).

As shown above, the exterior space is divided into four trapezoidal regions for the two-dimensional case. In three dimensions, this generalizes to six frustum-shaped regions, $\pm x$, $\pm y$, $\pm z$; hence the term *frustum warp*. There are three warping functions, namely the x-warping function as given above, and y- and z-warping functions,

$$y_w = y\frac{y_e - y_i}{y_e - |y|} \tag{3}$$

$$z_w = z\frac{z_e - z_i}{z_e - |z|}, \tag{4}$$

In general, the procedure to warp a point in the pre-warped exterior space is as follows.

1. Determine in which frustum-shaped region the point is located.
2. Apply the appropriate warping function to one of the coordinates. If the point is the in $\pm x$ region, apply the x-warping function, if the point is in the $\pm y$ region, apply the y-warping function, and if the point is the $\pm z$ region, apply the z-warping function.
3. Find the other two coordinates by solving line equations using the warped coordinate.

After reconstruction, we intend the model to be viewed from near or within the interior space. For such viewpoints, voxels will project to approximately the same footprint in each image.

3.2   Other Warping Functions

The frustum warp presented above is not the only possible warp. Any warp that does not move the outer boundary of the interior space, and warps the outer boundary of the pre-warped exterior space to infinity, while satisfying the criteria that no gaps form between voxels, and that no voxels overlap, is valid. Furthermore, it is desirable to choose a warping function that approximates the constant footprint property for the cameras used in the reconstruction as well as the camera placements during new view synthesis. An example of an alternative warping function is one that warps radially with distance from the center of the reconstruction volume.

4   Implementation Issues

Reconstructing a scene using a warped reconstruction volume poses some new challenges, described in this section.

4.1   Cameras Inside Volume

Perhaps the most difficult challenge is that of having the cameras embedded inside the reconstruction volume. Typically, when one uses a standard voxel coloring algorithm, the cameras used to take the photographs of the scene are placed outside of the reconstruction volume, so that at least two cameras have visibility of each voxel. The photo-consistency measure used in voxel coloring algorithms, qualitatively, determines if all the cameras that can see a voxel agree on its color. This photo-consistency is poorly defined when a voxel is visible from only one camera.

Since the warped reconstruction volume can occupy all space, cameras get embedded inside the voxel space, as shown in (a) of Figure 6. Our reconstruction algorithm initially assumes that all voxels are opaque. Therefore, camera views are obscured, and the cameras cannot work together to carve the volume. This

poses a problem, since to be properly defined, the photo-consistency measure requires that at least two cameras have visibility of a voxel. Consequently, the voxel coloring algorithm cannot proceed, and terminates without removing any voxels from the volume.

To address this issue, we must remove (pre-carve) a section of the voxel space so that initially, each surface voxel is observed by at least two cameras, validating the photo-consistency measure, as shown in (b) of Figure 6. There are a variety of possible methods to achieve this result. A generic method is to have a user identify regions of the voxel space to pre-carve. Obviously, the pre-carved regions must only consist of empty space, i.e. not contain any scene surfaces to be reconstructed. While effective, this method precludes a fully automatic reconstruction. Alternatively, one can pre-carve the volume using a heuristic. For example, if appropriate, one could require that the cameras have visibility of the boundary between the interior space and the exterior space. Other heuristics are possible. Once the pre-carving is complete, we execute a standard voxel coloring algorithm using the warped voxel space.

(a)   (b)
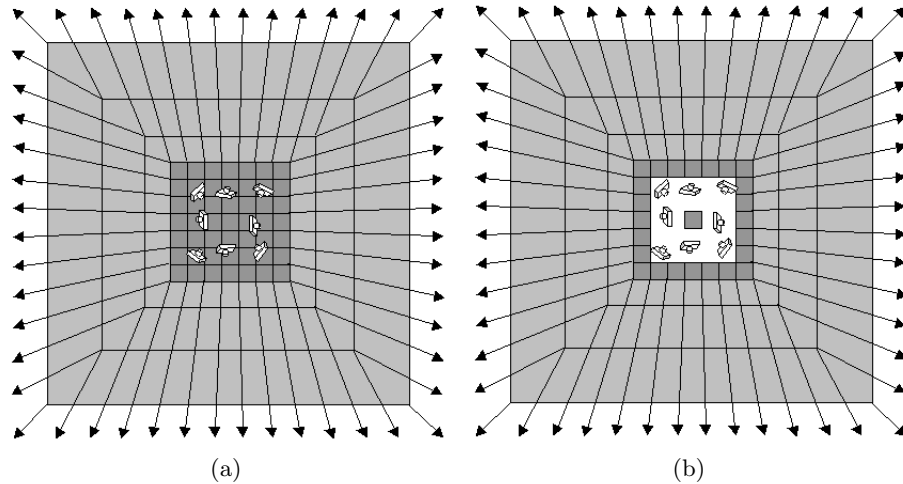
Fig. 6. Pre-carving operation. Reconstruction in the warped space causes the cameras to be embedded in the voxel space, as shown in (a). For many camera placements, it would be impossible to carve any voxels, since no voxel is visible to more than one camera. We execute a pre-carving step in (b) so that cameras can work together to carve the volume.

4.2 Preventing Visible Holes in the Outer Shell

Due to errors in camera calibration, image noise, inaccurate color threshold etc., voxel coloring sometimes removes voxels that should remain in the volume. Thus,

it is possible that voxels on the outer shell of the voxel space will be deemed inconsistent. Removing such voxels can result in unknown black regions similar to those in Figure 1 during new view synthesis, as no voxel would project onto the camera for some pixels in the image plane. Since one cannot see beyond infinity, we do not carve voxels on the outer shell of the voxel space, independent of the photo-consistency measure.

5   Results

We have modified the GVC and GVC-LDI algorithms [2] to utilize the warped voxel space. We created a synthetic data set, called "marbles", consisting of twelve 320 x 240 images of five small texture mapped spheres inside a much larger sphere textured with a rainbow-like image. We reconstructed the scene using a voxel space that consisted of 48 x 48 x 48 voxels, of which the inner 32 x 32 x 32 were in the interior space and unwarped. The voxel space was set up so that the five small texture mapped spheres were reconstructed in the interior space, while the larger sphere, making up the background, was reconstructed in the exterior warped space. Sample images from the data set are shown in (a) and (b) of Figure 7. A reconstruction was performed using the warped voxel space. The reconstruction was projected to the viewpoints of (a) and (b), yielding (c) and (d). Note that the background environment was reconstructed using our warped voxel space.

Next, we took a series of ten panoramic (360 degree field of view) photographs of a quadrangle at Stanford University, using a Panoscan[1] digital camera. These photographs had resolution of about 2502 x 884 pixels. One photograph from the set is shown in Figure 8 (a). We have found that when reconstructing an environment, it is preferable to use large field of view images, as objects far from the cameras are visible in many photographs. This achieves a sufficient sampling of the scene with fewer photographs. A voxel space of resolution 300 x 300 x 200 voxels, of which the inner 200 x 200 x 100 were interior voxels, was pre-carved manually by removing part of the voxel space that containing the cameras. Then, the GVC algorithm was used to reconstruct the scene. Figure 8 (b) shows the reconstructed model reprojected to the same viewpoint as in (a). Note that objects far away from the cameras, such as many of the buildings and trees, have been accurately reconstructed. New synthesized views are shown in (c) and (d) of the figure.

Despite the successes of this reconstruction, it is not perfect. The sky is very far away from the cameras (for practical purposes, at infinity), and should therefore be represented with voxels on the outer shell of the voxel space. However, since the sky is nearly textureless, cusping [7] occurs, resulting in inaccurate computed geometry, apparent in an animated sequence of new views of the reconstruction. Reconstruction of outdoor scenes is challenging, as surfaces often do not satisfy the Lambertian assumption. To compensate, we used a higher consistency threshold [7], also resulting in some inaccurate geometry. On the whole,

---

[1] www.panoscan.com

though, the reconstruction is reasonably accurate and produces convincing new views[2].

6   Conclusion

In this paper we have proposed extensions to voxel coloring that permit reconstruction of a scene using a warped voxel space, in an effort to comprehensively reconstruct objects both near and far away from the cameras used to photograph the scene. We have presented a frustum warp function, which describes a method to warp the voxel space to model infinite volumes while maintaining the requirements that no voxels overlap and no gaps form between the warped voxels. We have presented results showing the ability of this approach to reconstruct a background environment, in addition to a foreground scene.
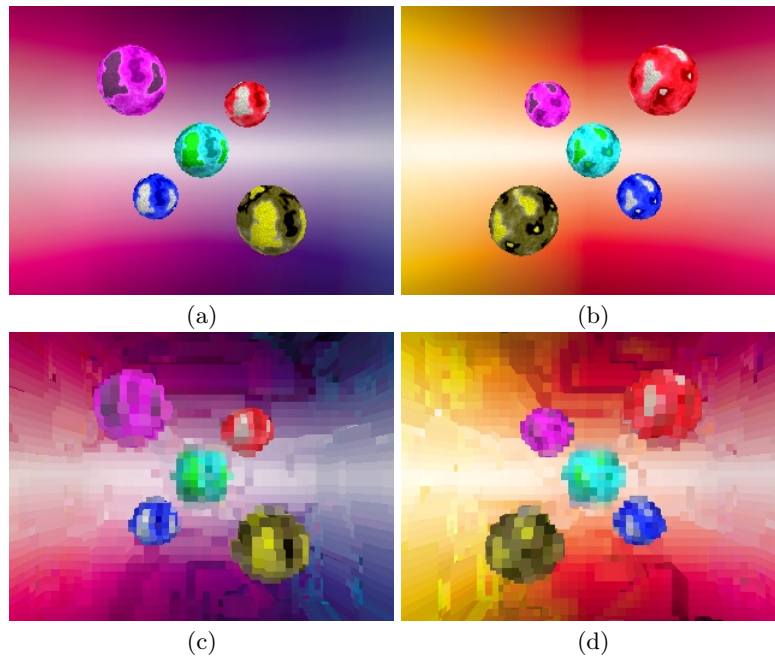
Fig. 7. Original images of the marbles data set are shown in (a) and (b), and a reconstruction projected to the same viewpoints of (a) and (b) is shown in (c) and (d), respectively.

---

[2] An animation showing new synthesized views of our Stanford scene is available online at www.ece.gatech.edu/users/slabaugh/projects/warp.

(a)

(b)

(c)

(d)

Fig. 8. Results for the Stanford scene. One of the ten panoramic photographs is shown in (a). The reconstructed model, projected to the same viewpoint as that of (a) is shown in (b). New synthesized panoramic views are shown in (c) and (d).

7   Future Work

Since voxels can warp to points infinitely far from the camera centers, using z-values (such as in a z-buffer) to establish depth order can be problematic due to a computer's finite precision. We are interested in exploring alternate methods, such as painter's algorithms, to determine depth order of voxels during reconstruction and rendering.

Acknowledgments

References

1. J. Blinn and M. Newell, "Texture and Reflection on Computer Generated Images," *Communications of ACM*, Vol. 19, No. 10, pp. 542 - 547, Oct. 1976.
2. W. B. Culbertson, T. Malzbender, and G. Slabaugh, "Generalized Voxel Coloring," *Vision Algorithms: Theory and Practice*, Springer-Verlag Lecture Notes in Computer Science, pp. 100 - 115, 1999.
3. N. Greene, "Environment Mapping and Other Applications of World Projections," *IEEE Computer Graphics and Applications*, pp. 21 - 29, Nov. 1986.
4. M. Kimura, H. Satio, and T. Kanade, "3D Voxel Construction Based on Epipolar Geometry," *Proc. ICIP*, pp. 135 - 139, 1999.
5. K. Kutulakos and S. Seitz, "A Theory of Shape by Space Carving," *Proc. ICCV*, pp. 307 - 314, 1999.
6. H. Saito and T. Kanade, "Shape Reconstruction in Projective Grid Space from Large Number of Images," *Proc. CVPR*, pp. 49 - 54, 1999.
7. S. Seitz and C. Dyer, "Photorealistic Scene Reconstruction by Voxel Coloring," *International Journal of Computer Vision*, Vol. 35, No. 2, pp. 151-173, Feb. 1999.