Toward Interactive Scene Walkthroughs from Images

Steven M. Seitz
Microsoft Research
One Microsoft Way
Redmond, WA 98052
sseitz@microsoft.com

Abstract

Interactive walkthrough applications require rendering an observed scene from a continuous range of target viewpoints. Toward this end, a novel approach is introduced that processes a set of input images to produce photorealistic scene reprojections over a wide range of viewpoints. This is achieved by (1) acquiring calibrated input images that are distributed throughout a target range of viewpoints to be modeled, and (2) computing a 3D reconstruction that is consistent in projection with all of the input images. The method avoids image correspondence problems by working in a discretized scene space whose voxels are traversed in a fixed visibility ordering. This strategy takes full account of occlusions and enables reconstructions of panoramic scenes. Promising initial results are presented for a room walkthrough.

1 Introduction

The topic of creating scene "walkthroughs" and "flybys" from images has achieved much recent interest in the research community, due in part to the popularity of commercial products like Apple's Quicktime VR [1]. A primary goal of this effort is to be able to render, or *synthesize*, images of a real scene for a wide range of target viewpoints by processing a set of input images. An interactive visualization of the environment can then be created by synthesizing new views in real time according to the user's changing perspective.

Unlike shape reconstruction, in which complexity tends to increase with the number of input images, the view synthesis problem becomes *simpler* when more images are available. When the input viewpoints finely sample the target range of viewpoints to be synthesized, the problem is solvable by selectively accessing a set of stored images [2]. While this approach is limited to providing only views that were previously captured, it requires little or no processing of the input images. In contrast, creating walkthroughs from a sparse set of input images currently requires highly accurate 3D shape or correspondence maps to be extracted, and is therefore computationally expensive and error-prone. The inherent difficulty of creating accurate walkthroughs can therefore be characterized in terms of the number, or *density* of input views within the target range of synthesized views.

In this paper, we advocate a middle ground between these two extremes in which the input views are separated but well-distributed throughout the target range. The chief advantage of this approach is that the full target range can be accurately synthesized without prohibitively many input images or precise correspondence information. Towards this end we introduce a new voxel-based reconstruction algorithm that possesses a number of key features that make it uniquely well-suited for view synthesis applications such as walkthroughs. First, the space and time complexity of the approach is linear in the number of images so it scales up well for image-intensive applications like walkthroughs. Second, the range of allowable input camera configurations is quite broad and permits *panoramic* walkthroughs, in which the images are acquired from cameras pointing in all directions. Third, the approach models occlusion explicitly and can cope with arbitrary changes in scene visibility without degrading the accuracy of the reconstruction.

Our framework for representing scene appearance, called *voxel coloring*, is introduced in Section 2. Section 2 also presents a novel constraint, called the *ordinal visibility constraint* that greatly simplifies the analysis. Section 3 describes our reconstruction algorithm, and Section 4 presents results of preliminary experiments for the case of reconstructing the interior of a synthetic room scene.

2 Voxel Coloring

The scene reconstruction problem, e.g., stereo, is often cast in terms of solving a 2D image correspondence problem followed by a triangulation step to infer 3D geometry. Rather than analyze the scene reconstruction problem

This research has been supported in part by the National Science Foundation under Grant No. IRI-9530985, and by the Defense Advanced Research Projects Agency and Rome Laboratory, USAF, under Agreement No. F30602-97-1-0138.

in these terms, we find it useful recast the reconstruction problem in terms of a discretized scene space. The *voxel coloring* problem is to assign colors (radiances) to voxels (points) in a 3D volume so as to achieve consistency with a set of basis (input) images. That is, rendering the colored voxels from each basis viewpoint should reproduce the original image as closely as possible. More formally, a 3D scene $\mathcal{S}$ is represented as a set of opaque Lambertian voxels (volume elements), each of which occupies a finite homogeneous scene volume centered at a point $\mathbf{V} \in \mathcal{S}$, and has a fixed color. We assume that the scene is entirely contained within a known, finite bounding volume. The set of all voxels in the bounding volume is referred to as the *voxel space* and denoted with the symbol $\mathcal{V}$. An image is specified by the set $\mathcal{I}$ of all its pixels, each centered at a point $\mathbf{p} \in \mathcal{I}$. For now, assume that pixels are infinitesimally small.

Given an image pixel $\mathbf{p} \in \mathcal{I}$ and scene $\mathcal{S}$, we refer to the voxel $\mathbf{V} \in \mathcal{S}$ that is visible in $\mathcal{I}$ and projects to p by $\mathbf{V} = \mathcal{S}(\mathbf{p})$. A scene $\mathcal{S}$ is said to be *complete* with respect to a set of images if, for every image $\mathcal{I}$ and every pixel $\mathbf{p} \in \mathcal{I}$, there exists a voxel $\mathbf{V} \in \mathcal{S}$ such that $\mathbf{V} = \mathcal{S}(\mathbf{p})$. A complete scene is said to be *consistent* with a set of images if, for every image $\mathcal{I}$ and every pixel $\mathbf{p} \in \mathcal{I}$,

$$color(\mathbf{p}, \mathcal{I}) = color(\mathcal{S}(\mathbf{p}), \mathcal{S}) \qquad (1)$$

We use the symbol $\aleph$ to denote the set of all consistent scenes. We may now define the voxel coloring problem formally:

> **Voxel Coloring Problem:** Given a set of basis images $\mathcal{I}_0, \ldots, \mathcal{I}_n$ and a voxel space $\mathcal{V}$, determine a subset $\mathcal{S} \subset \mathcal{V}$ and a coloring $color(\mathbf{V}, \mathcal{S})$, such that $\mathcal{S} \in \aleph$.

Observe that our goal is not to obtain the *best* reconstruction, e.g., by using smoothness or regularization criteria, but to simply find a single *consistent* reconstruction. Note that this distinction is critical when the density of input images in the target space is low, because different reconstructions may project to distinct new views. Note, however, that this distinction is *not* critical when the input images represent a good sampling of the target space, since all consistent reconstructions will appear virtually identical throughout the target range.

2.1 The Ordinal Visibility Constraint

Computing voxel colorings poses a significant challenge. Observe that the underlying space is combinatorial: an $N \times N \times N$ grid of voxels, each with M possible color assignments yields 2^{N^3} possible scenes and M^{N^3} possible color assignments. Clearly, a bruce-force search through this space is not feasible. In order to make the problem tractable, we introduce a novel geometric constraint on

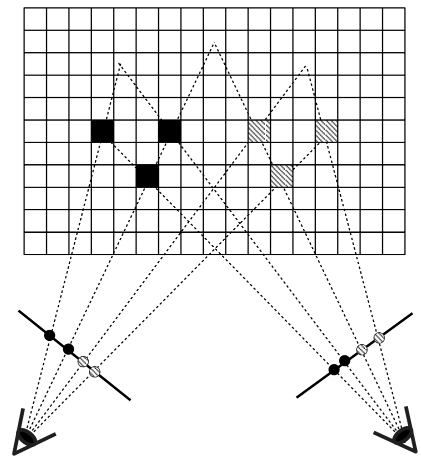


Figure 1: These six points constitute the *voxel coloring* for these two views, representing the consistent scene that is closest to the cameras.

camera placement relative to the scene that simplifies the analysis. This *ordinal visibility constraint* enables the identification of one consistent reconstruction as a limit point of $\aleph$. As such, it can be computed directly, via a single pass though the voxel space.

Let $\mathbf{P}$ and $\mathbf{Q}$ be scene points and $\mathcal{I}$ be an image from a camera centered at $\mathbf{C}$. We say $\mathbf{P}$ *occludes* $\mathbf{Q}$ if $\mathbf{P}$ lies on the line segment $\overline{\mathbf{CQ}}$. We require that the input cameras be positioned so as to satisfy the following constraint:

> **Ordinal visibility constraint:** There exists a norm $\|\cdot\|$ such that for all scene points $\mathbf{P}$ and $\mathbf{Q}$, and input images $\mathcal{I}$, $\mathbf{P}$ occludes $\mathbf{Q}$ in $\mathcal{I}$ only if $\|\mathbf{P}\| < \|\mathbf{Q}\|$.

We call such a norm *occlusion-compatible*. For some camera configurations, it is not possible to define an occlusion-compatible norm. However, a norm *does* exist for a broad range of practical configurations. For instance, suppose the cameras are distributed on a plane and the scene is entirely below that plane. For every such viewpoint, the relative visibility of any two scene points depends entirely on which point is closer to the plane, so we may define $\|\cdot\|$ to be distance to the plane. More generally, the ordinal visibility constraint is satisfied whenever **no scene point is contained within the convex hull $\mathcal{C}$ of the camera centers**. Here we use the occlusion-compatible norm $\|\mathbf{P}\|_{\mathcal{C}}$, defined to be the Euclidean distance from $\mathbf{P}$ to $\mathcal{C}$. This generalization enables "panoramic" configurations of outward-facing cameras, as in Fig. 3. For convenience, $\mathcal{C}$ is referred to as the *camera volume*.

2.2 Defining a Voxel Coloring

let $\mathcal{I}_0, \ldots, \mathcal{I}_n$ be a set of images for which the ordinal visibility constraint is satisfied. For a given image point

$\mathbf{p} \in \mathcal{I}_j$ define $\mathbf{V_p}$ to be the voxel in $\{\mathcal{S}(\mathbf{p}) \mid \mathcal{S} \in \aleph\}$ that is closest to the camera volume. We denote the collection of these points $\overline{\mathcal{S}}$:

$$\overline{\mathcal{S}} = \{\mathbf{V_p} \mid \mathbf{p} \in \mathcal{I}_i,\ 0 \leq i \leq n\}$$

It is easily shown that $\overline{\mathcal{S}}$ is a consistent scene. Note that $\overline{\mathcal{S}}$ is complete, since it contains a voxel corresponding to each pixel in the basis images. To show that it is consistent, for each $\mathbf{V} \in \overline{\mathcal{S}}$, choose $\mathbf{p} \in \mathcal{I}_i$, $0 \leq i \leq n$, such that $\mathbf{V} = \overline{\mathcal{S}}(\mathbf{p})$. Define

$$color(\mathbf{V}, \overline{\mathcal{S}}) := color(\mathbf{p}, \mathcal{I}_i) \qquad (2)$$

To show that this coloring is well defined, suppose $\mathbf{p} \in \mathcal{I}_i$ and $\mathbf{q} \in \mathcal{I}_j$ are two points such that $\overline{\mathcal{S}}(\mathbf{p}) = \mathbf{V} = \overline{\mathcal{S}}(\mathbf{q})$. Let $\mathcal{S}$ be a consistent scene such that $\mathbf{V} \in \mathcal{S}$. By the definition of $\overline{\mathcal{S}}$, it follows that $\mathcal{S}(\mathbf{p}) = \mathbf{V} = \mathcal{S}(\mathbf{q})$. Hence, by Eq. (1),

$$color(\mathbf{p}, \mathcal{I}_i) = color(\mathbf{V}, \mathcal{S}) = color(\mathbf{q}, \mathcal{I}_j)$$

Therefore Eq. (2) is a well-defined voxel coloring and is consistent with the basis images.

Fig. 1 shows $\overline{\mathcal{S}}$ for one pair of images. These six voxels have a unique color interpretation, constant in every consistent scene. They also comprise the closest consistent scene to the cameras in the following sense—every point in each consistent scene is either contained in $\overline{\mathcal{S}}$ or is occluded by points in $\overline{\mathcal{S}}$. An interesting consequence of this distance bias is that neighboring image pixels of the same color produce cusps in $\overline{\mathcal{S}}$, i.e., protrusions toward the camera volume. This phenomenon is clearly evident in Fig. 1, where the black and gray points form two separate cusps. Also, observe that $\overline{\mathcal{S}}$ is not a minimal reconstruction; removing the two closest points in Fig. 1 still leaves a consistent scene.

3 A Voxel Coloring Algorithm

We now describe how to compute $\overline{\mathcal{S}}$ via a single pass through a discretized scene volume, by exploiting the ordinal visibility constraint. This constraint limits the possible basis view configurations, but the benefit is that visibility relationships are greatly simplified. In particular, it becomes possible to partition the scene into a series of voxel *layers* that obey a monotonic visibility relationship: for *every* input image, voxels only occlude other voxels that are in subsequent layers. Consequently, visibility relationships are resolved by traversing voxels one layer at a time.

3.1 Layered Scene Decomposition

To formalize this idea, we define the following partition of 3D space into voxel layers of uniform distance from the camera volume:

$$\mathcal{V}_d = \{V \mid \|V\| = d\} \qquad (3)$$

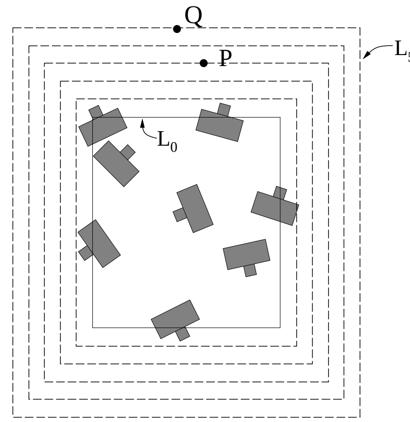


Figure 2: Layered scene traversal. Voxels can be partitioned into a series of layers of increasing distance from the camera volume.

$$\mathcal{V} = \bigcup_{i=1}^{r} \mathcal{V}_{d_i} \qquad (4)$$

where $d_1, \ldots, d_r$ is an increasing sequence of numbers and $\|\cdot\|$ is an occlusion-compatible norm.

For the sake of illustration, consider a set of views positioned along a line facing a two-dimensional scene. Choosing $\|\cdot\|$ to be orthogonal distance to the line gives rise to a series of parallel linear layers that move away from the cameras. Notice that for any two voxels $\mathbf{P}$ and $\mathbf{Q}$, $\mathbf{P}$ can occlude $\mathbf{Q}$ from a basis viewpoint only if $\mathbf{Q}$ is in a higher layer than $\mathbf{P}$. The simplification of visibility relationships for this special case of colinear views was previously noted by Katayama et al. [3].

The linear case is easily generalized for any set of cameras satisfying the ordinal visibility constraint. Fig. 2 shows a layer partition for the case of outward-facing cameras. This type of camera geometry is useful for acquiring *panoramic* scene visualizations, as in [4, 5]. One valid set of layers corresponds to a series of rectangles radiating outward from the camera volume. Layer 0 is the axis-aligned bounding box $\mathcal{B}$ of the camera centers and the subsequent layers are determined by uniformly expanding the box one unit at a time. This set of layers corresponds to a norm given by the L_∞ distance to $\mathcal{B}$.

Decomposing a 3D scene into layers can be done in the same manner. In the 3D case the layers become surfaces that expand outward from the camera volume. An especially useful layering strategy is the 3D analog of Fig. 2, in which each layer is an axis-aligned cube. The advantage of this choice of layers is that layers are computed and traversed very efficiently.

3.2 Voxel Consistency

To compensate for the effects of image quantization and noise, suppose now that the images are discretized on a grid of finite non-overlapping pixels. If a voxel $\mathbf{V}$ is not fully occluded in image $\mathcal{I}_j$, its projection overlaps a nonempty set of image pixels, π_j. Without noise or quantization effects, a consistent voxel should project to a set of pixels with equal color values. In the presence of these effects, we evaluate the correlation $\lambda_{\mathbf{V}}$ of the pixel colors to measure the likelihood of voxel consistency. Let s be the standard deviation and m the cardinality of $\bigcup_{j=0}^{n} \pi_j$. One possible approach is to threshold the color space error:

$$\lambda_{\mathbf{V}} = s \tag{5}$$

Alternatively, a statistical measure of voxel consistency can be used. In particular, suppose the sensor error (accuracy of irradiance measurement) is normally distributed[1] with standard deviation σ_0. The consistency of a voxel can be estimated using the likelihood ratio test, distributed as χ^2 with $n-1$ degrees of freedom [6]:

$$\lambda_{\mathbf{V}} = \frac{(m-1)s^2}{\sigma_0^2} \tag{6}$$

If σ_0 is unknown, it can be estimated by imaging a homogeneous surface and computing the standard deviation s_0 of m' image pixels. In this case, Eq. (6) should be replaced with

$$\lambda_{\mathbf{V}} = \frac{s^2}{s_0^2} \tag{7}$$

which has an F distribution with $m-1$ and $m'-1$ degrees of freedom.

3.3 A Single-Pass Algorithm

In order to evaluate the consistency of a voxel, we must first compute π_j, the set of pixels that overlap $\mathbf{V}$'s projection in $\mathcal{I}_j$. Neglecting occlusions, it is straightforward to compute a voxel's image projection, or *footprint*, based on voxel shape and the known camera configuration. Accounting for occlusions is more difficult, however, and we must take care to include only the images and pixel positions from which $\mathbf{V}$ should be *visible*. This difficulty is resolved by using the ordinal visibility constraint to visit voxels in an occlusion-compatible order and *marking* pixels as they are accounted for.

Initially, all pixels are unmarked. When a voxel is visited, π_j is defined to be the set of *unmarked* pixels that overlap $\mathbf{V}$'s footprint in $\mathcal{I}_j$. When a voxel is evaluated and found to be consistent, all m pixels in π_j are marked. Because of the occlusion-compatible order of voxel evaluation, this strategy is sufficient to ensure that π_j contains only the pixels from which each voxel is visible, i.e., $\overline{\mathcal{S}}(\mathbf{p}) = \mathbf{V}$ for each $\mathbf{p} \in \pi_j$. Note that by assumption voxels within a layer do not occlude each other. Therefore, the pixel marking phase can be delayed until after all the voxels in a layer are evaluated.

The complete voxel coloring algorithm can now be presented as follows:

```
S̄ = ∅
/* Iterate through the layers */
for i = 1,...,r do
   /* Iterate through voxels in the layer */
   for every V ∈ V_{d_i} do
      /* Project the voxel to each image */
      for j = 0,...,n do
         compute footprint ρ of V in I_j
         π_j = {p ∈ ρ | p unmarked}
      end for j
      /* Evaluate voxel consistency */
      compute λ_V
      if m > 0 and λ_V < thresh then
         /* Color the voxel */
         S̄ = S̄ ∪ {V}
         /* Remember image pixels to mark */
         π = π ∪ ⋃_{j=0}^{n} π_j
      end if
   end for V
   mark pixels in π
end for
```

The threshold, $thresh$, corresponds to the maximum allowable correlation error. An overly conservative (small) value of $thresh$ results in an accurate but incomplete reconstruction. On the other hand, a large threshold yields a more complete reconstruction, but one that includes some erroneous voxels. Instead of thresholding correlation error, it is possible to optimize for model *completeness*. In particular, a completeness threshold $tcomp$ may be chosen that specifies the minimum allowable percentage of image pixels left unmarked. For instance, $tcomp = 75\%$ requires that at least three quarters of the (non-background) image pixels correspond to the projection of a colored voxel.

Given $tcomp$, we seek the minimum value of $thresh$ that yields a voxel coloring achieving this completeness threshold. Since completeness increases monotonically with $thresh$, it is sufficient to run the single-pass algorithm for a succession of increasing values of $thresh$, stopping when $tcomp$ is achieved.

[1] Here we make the simplifying assumption that σ_0 does not vary as a function of image intensity.

3.4 Discussion

The voxel coloring algorithm visits each of the N^3 voxels exactly once and projects it into every image. Therefore, the time complexity of voxel coloring is: $O(N^3 n)$, where n is the number of images. To determine the space complexity, observe that evaluating one voxel does not require access to or comparison with other voxels. Consequently, voxels need not be stored in main memory during the algorithm; the voxels making up the voxel coloring will simply be output one at a time. Only the images and one-bit mark masks need to be allocated. The fact that the space and time complexities of voxel coloring are linear in the number of images is essential so that large numbers of images can be processed at once.

The algorithm differs from stereo and tracking techniques in that it does not perform window-based image correlation during the reconstruction process. Correspondences are found during the course of scene traversal by voxel projection. A disadvantage of this searchless strategy is that it requires very precise camera calibration to achieve the triangulation accuracy of stereo methods. The effects of calibration and quantization errors are most significant at high-frequency regions such as image edges. Preserving high-frequency image content requires a higher voxel sampling rate because of Nyquist considerations. However, smaller voxels result in fewer pixels being integrated in the correlation step and therefore are more sensitive to calibration errors. Accuracy and run-time also depend on the voxel resolution, a parameter that can be set by the user or determined automatically to match the pixel resolution, calibration accuracy, and computational resources.

Importantly, the voxel coloring approach reconstructs only one of the potentially numerous scenes consistent with the input images. Consequently, it is susceptible to aperture problems caused by image regions of near-uniform color. These regions cause cusps in the reconstruction (see Fig. 1), since voxel coloring yields the reconstruction closest to the camera volume. This is a bias, just like smoothness is a bias in stereo methods, but one that guarantees a consistent reconstruction even with severe occlusions.

4 Experimental Results

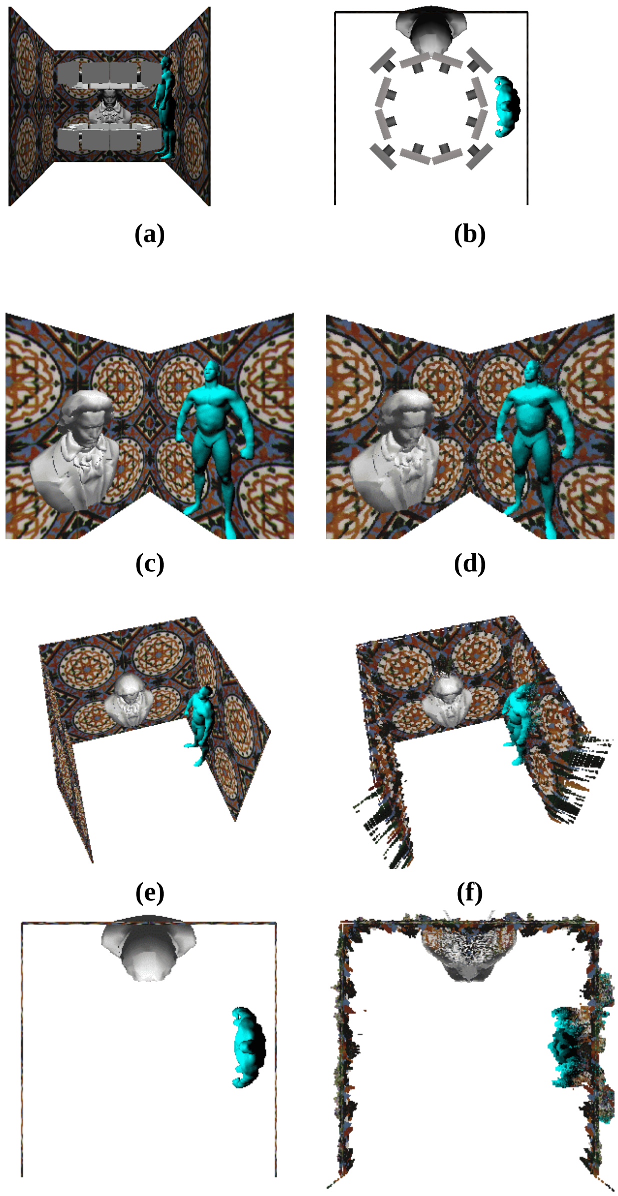
In order to evaluate the performance of the voxel coloring algorithm for panoramic scenes, basis images were generated by placing several cameras in a synthetic room scene. The room consisted of three texture-mapped walls and two shaded figures. The figures, a bust of Beethoven and a scanned Cyberware model of a human figure, were illuminated diffusely from a downward-oriented light source at infinity. 24 cameras were placed at different positions and orientations *inside* the room, as shown in Fig. 3.

The geometry of this scene and the camera configuration would pose significant problems for previous image-based reconstruction methods. In particular, the room interior is highly concave, making accurate reconstruction by volume intersection or other contour-based methods impractical. Furthermore, the numerous cameras and large amount of occlusion would create difficulty for most stereo approaches. Notable exceptions include panorama-based stereo approaches [4, 5] that are well-suited for room reconstructions. However, these methods require that a panoramic image be constructed for each camera location prior to the stereo matching step, a requirement that is avoided by the voxel coloring approach. This requirement does not enable camera configurations such as the one shown in Fig. 3.

Fig. 3 compares the original and reconstructed models of the room from new viewpoints. The reconstruction contained 320,000 voxels and required 45 minutes to compute on a 250 MHz Silicon Graphics Indigo2. The voxel coloring reproduced images from the room interior extremely accurately, as shown in (d). A pixel correlation error threshold of 2.4% was used to account for image quantization. As a result of these errors, some fine details were lost, e.g., in the face of the Beethoven bust. The overhead views (f) and (h) more clearly show some discrepancies between the original and reconstructed models. For instance, the reconstructed walls are not perfectly planar, as some points lie just off the surface. This point drift effect is most noticeable in regions where the texture is locally homogeneous, indicating that texture information is important for accurate reconstruction. The quality of the overhead view shown in (f) is especially commendable, given that the viewpoint is very far away from the input views. The extreme overhead view (h) is worse than that of (f) but clearly shows that the overall shape of the scene was very well captured by reconstruction.

5 Conclusions

This paper presented a new scene reconstruction technique that is ideally suited for view synthesis applications like environment walkthroughs and flybys. It has the following key advantages: first, the space and time complexity of the approach is linear in the number of images so it scales up well for image-intensive applications like walkthroughs. Second, it enables *panoramic* visualizations, in which images are acquired from cameras pointing in any number of directions. Third, the approach models occlusion explicitly and can cope with arbitrary changes in scene visibility without degrading the accuracy of the reconstruction. Initial results show that the technique provides very accurate synthesized views when applied to synthetic imagery. Results on images of small objects [7] also indicate that the approach is also very effective when applied to real imagery. Evaluating the approach with real images of larger-scale environments is a topic of current investigation.



(a) **(b)** **(c)** **(d)** **(e)** **(f)** **(g)** **(h)**

Figure 3: Panoramic room scene reconstruction. The input camera positions and scene are shown together, from a frontal (a), and overhead (b) perspective. (c-h): renderings of the true scene are shown at left and the reconstructed scene at right. (c) and (d): an input viewpoint, from *inside* the room. (e) and (f): a new viewpoint from above the room. (g) and (h): an extreme overhead view. Fidelity is best near the input viewpoints and degrades smoothly as the camera moves further away.

References

[1] S. E. Chen, "Quicktime VR — An image-based approach to virtual environment navigation," in *Proc. SIGGRAPH 95*, pp. 29–38, 1995.

[2] A. Lippman, "Movie-maps: an application of the optical videodisc to computer graphics," in *Proc. SIGGRAPH 80*, pp. 32–42, 1980.

[3] A. Katayama, K. Tanaka, T. Oshino, and H. Tamura, "A viewpoint dependent stereoscopic display using interpolation of multi-viewpoint images," in *Proc. SPIE Vol. 2409A*, pp. 21–30, 1995.

[4] L. McMillan and G. Bishop, "Plenoptic modeling," in *Proc. SIGGRAPH 95*, pp. 39–46, 1995.

[5] S. B. Kang and R. Szeliski, "3-D scene data recovery using omnidirectional multibaseline stereo," in *Proc. Computer Vision and Pattern Recognition Conf.*, pp. 364–370, 1996.

[6] J. E. Freund, *Mathematical Statistics*. Englewood Cliffs, NJ: Prentice Hall, 1992.

[7] S. M. Seitz and C. R. Dyer, "Photorealistic scene reconstruction by voxel coloring," in *Proc. Computer Vision and Pattern Recognition Conf.*, pp. 1067–1073, 1997.