# 3-D Shape Reconstruction from Light Fields Using Voxel Back-Projection

Peter Eisert, Eckehard Steinbach, and Bernd Girod

Telecommunications Laboratory, University of Erlangen-Nuremberg
Cauerstrasse 7, 91058 Erlangen, Germany
{eisert,steinb,girod}@nt.e-technik.uni-erlangen.de

## Abstract

We present a method for the 3-D reconstruction of objects from light fields. The estimated shape and texture information can be used for the compression of light fields or a better intermediate view interpolation. The representation of the objects is fully voxel-based and no explicit surface description is required. The approach is based on testing multiple voxel color hypotheses back-projected into the focal plane. Multiple views are incorporated in the reconstruction process simultaneously and no explicit data fusion is needed. The methodology of our approach combines the advantages of silhouette-based and image feature-based methods. Experimental results on light field data show the excellent visual quality of the voxel-based 3-D reconstruction. We also show the quality improvement for intermediate view generation that can be achieved using the estimated 3-D shape information.

## 1 Introduction

There is a tremendous interest from Virtual Reality (VR) and multimedia applications to obtain computer models of real world objects. Caused by the increase of computational power of graphics workstations, more and more sophisticated 3-D geometry and illumination models have been developed for the description of virtual environments. However, for some scenes a photo-realistic impression of the rendered 3-D objects cannot be achieved even with an enormous number of graphics primitives. Trees, hair, fur and objects with fuzzy surfaces are hard to model, as it is the case for complex global illumination conditions.

At Siggraph 1996 the concept of *light fields* was introduced [1], [2] with the aim to overcome the before mentioned limitations. The key idea of light fields is to use multiple images of the scene recorded from different viewing positions for rendering of arbitrary scene views. Due to the use of real camera views also complex object surfaces are rendered correctly. The four-dimensional structure of the light field, however, requires large amounts of memory and disk space which makes coding of light fields essential. Current approaches use vector quantizers [1] or disparity compensation between the planes in the light field in combination with residual coding [3]. A higher gain in coding efficiency can be expected if an approximative geometry model is used for prediction. The same model can also be used for a disparity compensated prediction of new views that are not part of the light field. In contrast to bilinear interpolation that is often used a much better image quality can be achieved.

For both light field coding and view interpolation an approximative geometry model is of advantage. However, we need an algorithm that reconstructs the geometry information automatically. One common approach is to take multiple camera views from different po-

sitions around the object and then to register the information from all views into a complete 3-D description of the object.

In 3-D reconstruction from multiple views we can distinguish two classes of algorithms. The first class computes depth maps from two or more views and then registers the depth maps into a single 3-D surface model. The depth map recovery often relies on sparse or dense matching of image points with subsequent 3-D structure estimation [4, 5] or is supported by additional depth information from range sensors [6]. The second class of algorithms is based on volume intersection, and is often refered to as *shape-from-silhouette* algorithms [7, 8, 9, 10, 11]. The object shape is typically computed as the intersection of the outline cones back-projected from all available views of the object. This requires the reliable extraction of the object contour in all views which restricts the usability to scenes where the object can be easily segmented from the background. Color and feature correspondences are not used in this class of algorithms.

In this work we exploit the advantages of both approaches by using a voxel representation of the 3-D object combined with multi-hypothesis testing of the back-projection of the object surface voxels with the images in the light field. A similar approach for video sequences that also combines both advantages has been presented by Seitz and Dyer [12]. However, the algorithm in [12] introduces constraints on the possible camera setup and therefore restricts the type of scenes that can be reconstructed. The algorithm in this paper does not restrict the viewing positions and allows the reconstruction of arbitrary scenes.

In case we are working with an homogeneous background, the approach shows all the advantages that can be obtained with accurate silhouette description. Moreover, the color of the surface is additionally exploited in a unified framework to estimate the shape also in those regions where the silhouette information is not sufficient. If the background is not homogeneous, the intensity matching takes over and still provides us with a good voxel-based description of the scene.

# 2 Light Fields

A light field [1] is a new image-based rendering technique that uses multiple views of a scene from different viewing positions. The key idea is to capture the light coming from a scene in a large array. The lighting condition of a scene can be characterized by the five-dimensional *plenoptic function* that describes the flow of light (radiance) at every point in the three-dimensional space (3 degrees of freedom) in any direction (2 degrees of freedom). Assuming a bounded object and a transparent surrounding medium the radiance does not change along a line of sight. This reduces the five-dimensional space to a four-dimensional one. The light field is now a sampled representation of this four dimensional space. Usually the rays specifying the direction of the light flow are parameterized by the coordinates of the intersection of the line with two parallel planes. For the two-dimensional case this is shown in Fig. 1. The camera plane

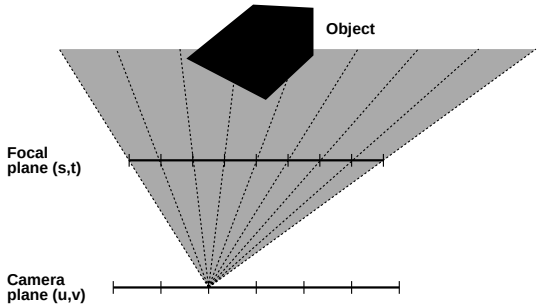


Figure 1: Light field geometry.

has the coordinates $(u, v)$, the focal plane the coordinates $(s, t)$. Such a structure can be achieved by putting a camera at different positions $(u, v)$ on the camera plane with the optical axis of the camera being perpendicular to the planes. The focal plane can then be interpreted as the image plane of the camera with $(s, t)$ being the pixel positions. The quadruple $(u, v, s, t)$ contains the color information of pixel $(s, t)$ at the viewing position

$(u, v)$. To obtain a complete scene description that allows arbitrary viewing directions 6 pairs of planes can be used each being aligned to one side of a cube. Of course, other camera positions and directions can also be used for the acquisition of the light field, if the resulting 4-D space is resampled to match the light field representation.

## 2.1 Basic Geometry and Coordinate Systems

The basic geometry of the projection planes, the objects and their coordinate systems are shown in Fig. 2. All object points in the 3-D

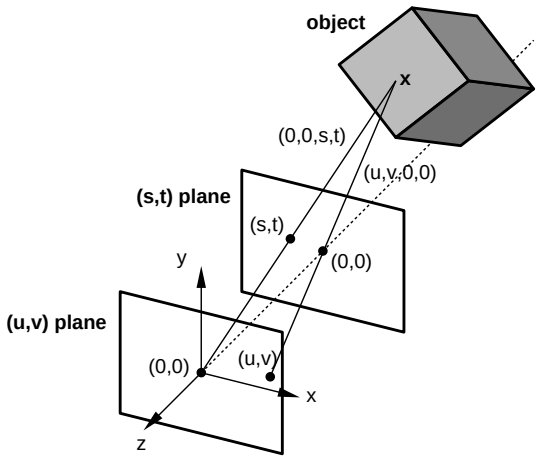


Figure 2: Coordinate systems.

space $\mathbf{x} = [x\ y\ z]^T$ are referenced relative to the world coordinate system that is located in the middle of the camera plane. Both camera and focal plane are parallel to the (x,y) plane in the world coordinate system. The distance between the planes corresponds to the focal length f. With this representation, the light field coordinates $(u, v)$ and $(s, t)$ correspond to the 3-D coordinates $(u, v, 0)$ and $(s, t, -f)$ respectively. The object point $\mathbf{x}$ is then projected into the point $(s, t)$ on the focal plane according to the following equation:

$$\begin{aligned} s_{(u,v)} &= u - f\frac{x}{z} \\ t_{(u,v)} &= v - f\frac{y}{z} \end{aligned} \quad (1)$$

where $u$, $v$, describe the projection center of the particular view in the $(u, v)$ plane. Due to the pixel nature of the light field images the world coordinates of the focal plane $(s, t)$ have to be transformed to pixel coordinates $(S, T)$ using

$$\begin{aligned} S &= s_x s \\ T &= s_y t \end{aligned} \quad (2)$$

with $s_x$ and $s_y$ the horizontal and vertical scaling factors that transform world into pixel coordinates.

# 3 Voxel-based 3-D Object Reconstruction

In contrast to methods that explicitly match features to obtain a surface description of the object we use a voxel-based description of the scene. All operations during 3-D reconstruction are performed on voxels. Other than the mapping from pixels to voxels, the mapping from voxels to pixels is straightforward. Therefore, we avoid the search for corresponding points and the fusion of several incomplete depth estimates. Our proposed algorithm proceeds in three steps:

- volume initialization
- hypothesis extraction for all voxels from all available camera views
- consistency check and hypothesis testing over all views and hypothesis removal

## 3.1 Volume Initialization

The first step is to define a volume in the reference coordinate system that encloses the 3-D object to be reconstructed. The volume extensions are determined from the calibrated camera parameters and its surface represents a conservative bounding box of the object. The volume is discretized in all three dimensions leading to an array of voxels with associated color, where the position of each voxel in the 3-D space is defined by its indices $(l, m, n)$. Initially, all voxels are transparent. Fig. 3 shows an example of the initial volume with large voxels for illustration purposes. Typical dimensions are $200 \times 200 \times 200$ voxels.

Figure 3: Bounding box of the volume with four voxels for illustration purposes.

## 3.2 Hypothesis Extraction

During the hypothesis extraction step a set of color hypotheses is assigned to each voxel of the predefined volume. The $k$th hypothesis $H^k_{lmn}$ for a voxel $V_{lmn}$ with voxel index $(l, m, n)$ is

$$H^k_{lmn} = (R_{uv}(S,T), G_{uv}(S,T), B_{uv}(S,T)), \tag{3}$$

where $R_{uv}$, $G_{uv}$, and $B_{uv}$ are the three color components and $(S, T)$ is the pixel position of the perspective projection of the voxel center $(x_l, y_m, z_n)$ into view $(u, v)$.

Hypothesis $H^k_{lmn}$ is associated to voxel $V_{lmn}$ if the projection of $V_{lmn}$ into at least one light field view $(u', v') \neq (u, v)$ leads to a normalized difference of the color channels less than a preset threshold $\Theta$, i.e.,

$$\begin{aligned} &\left| \frac{R_{uv}(S,T)}{N_{uv}(S,T)} - \frac{R_{u'v'}(S',T')}{N_{u'v'}(S',T')} \right| + \\ &\left| \frac{G_{uv}(S,T)}{N_{uv}(S,T)} - \frac{G_{u'v'}(S',T')}{N_{u'v'}(S',T')} \right| + \\ &\left| \frac{B_{uv}(S,T)}{N_{uv}(S,T)} - \frac{B_{u'v'}(S',T')}{N_{u'v'}(S',T')} \right| < \Theta \end{aligned} \tag{4}$$

with

$$N_{uv}(S,T) = R_{uv}(S,T) + G_{uv}(S,T) + B_{uv}(S,T) \; . \tag{5}$$

The color in (4) is normalized in order to be able to deal with illumination changes during the image acquisition process. Equation (4) defines the hypothesis criterion for 2 light field views $(u, v)$ and $(u', v')$ and has to be evaluated for each of $N(N-1)$ pairs, where N is the total number of available light field views. For all combinations of $(u, v)$ and $(u', v')$ that pass test (4) a color hypothesis $H^k_{lmn}$ from view $(u, v)$ according to (3) is stored. Please note that the voxel need not be visible in all views due to occlusions and that it might not be visible in any view at all if it is inside the object. At this stage of the algorithm we do not know the geometry of the object and cannot decide whether a voxel is visible or not. We therefore have to remove those hypotheses of the overcomplete set that do not correspond to the correct color of the object's surface in the following processing stage.

## 3.3 Consistency Check and Hypotheses Removal

In the previous step we stored multiple hypotheses for each voxel of the working volume. Those hypotheses were extracted from two or more consistent views without knowledge of the object's geometry. We now iterate over all available views and check if the extracted hypotheses are consistent with all available views. The corresponding algorithmic steps are illustrated in Fig. 4. Starting

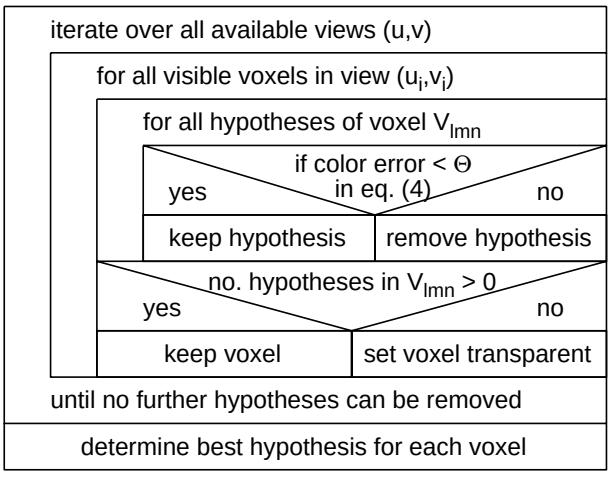


Figure 4: Algorithmic steps of the consistency check and hypotheses removal.

with the initial and overcomplete voxel volume, for each view $(u, v)$ the currently visible voxels are determined. We compare all associated hypotheses with the corresponding pixel

color at the pixel position in (1). The similarity measure is again the normalized difference of the color components in (4). If the error in (4) exceeds the threshold $\Theta$ for this view we remove the corresponding hypotheses from the voxel. This is now possible because we are looking at the outmost voxels that cannot be occluded by other voxels and must therefore be visible. If all hypotheses of one voxel are removed, the voxel is set to be transparent and the visible surface for the next view moves towards the interior of the volume. This implies that during the first iteration only voxels on the surface of our volume can be removed. We therefore iterate multiple times over all available views until no more hypotheses are removed and the number of transparent voxel converges. The remaining non-transparent voxels constitute the 3-D description of our object. Since we still have multiple color hypotheses for each surface voxel we have to choose one for the voxel color. Here, we use the color hypothesis that has the least absolute error in the color components summed over all views where the voxel is visible. The color values associated with the resulting non-transparent voxels which are on the object surface can now be used for rendering.

### 3.4 Visible Surface Determination

The hypothesis check and subsequent hypothesis removal for each view $(u, v)$ requires the determination of the visible surface voxels from the current view of the volume. If the light field consists of 6 pairs of planes there are exactly 6 possible ways to index the voxels in their visibility order. For a particular $(u, v)$ plane this order is in increasing distance along the axis that is perpendicular to the $(u, v)$ plane.

### 3.5 Rendering of Arbitrary Views

Once the voxel description of the object is determined we can render views from arbitrary viewing positions which are not necessarily part of the $(u, v)$ plane. For that purpose we shift the volume according to the desired position $(u*, v*)$ and render all visible voxels into the virtual viewing plane. The pixels in the virtual views are set using the projection formulae in (1). A simple z-buffer ensures that only visible voxels are rendered when stepping through the volume. The depth map for the view can be taken directly from the z-buffer.

### 3.6 Interpolation of New Light Field Views

In simple light field rendering intermediate views which are not part of the original light field are typically interpolated using adjacent samples of the light field data. The interpolation of these samples is normally accomplished via bi-linear interpolation of corresponding intensity values [1]. The result becomes blurry if the depth of the object does not coincide with the distance of the $(s, t)$ plane to the $(u, v)$ plane. Considerably better results can be achieved if information about the object shape is available [2]. This is the case in our approach where we have a voxel-based description of the object. The interpolation can now be performed considering the varying disparity as a function of the object depth.

## 4 Experimental Results

The first light field is a synthetic light field of a plane called the *Airplane* light field. Fig. 5 shows the $(s, t)$ plane for 64 different $(u, v)$ positions all lying on the same $(u, v)$ plane. Fig. 6 shows the reconstruction result when using the algorithm described in this paper. The quality of the reconstructed shape of the object can be judged via examination of the corresponding depth maps for the two views in Fig. 6. These depth maps are reproduced in Fig. 7.

The next experiment shows the excellent quality of intermediate light field view generation. The view to be generated is exactly in

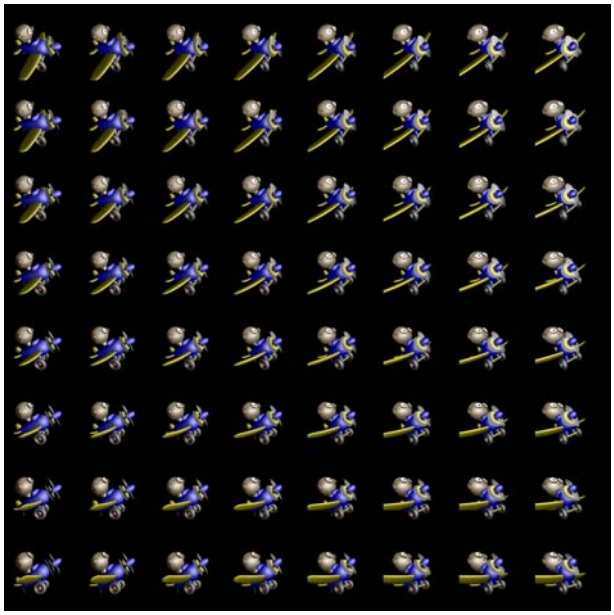

Figure 5: 64 light field images of the *Airplane* light field.

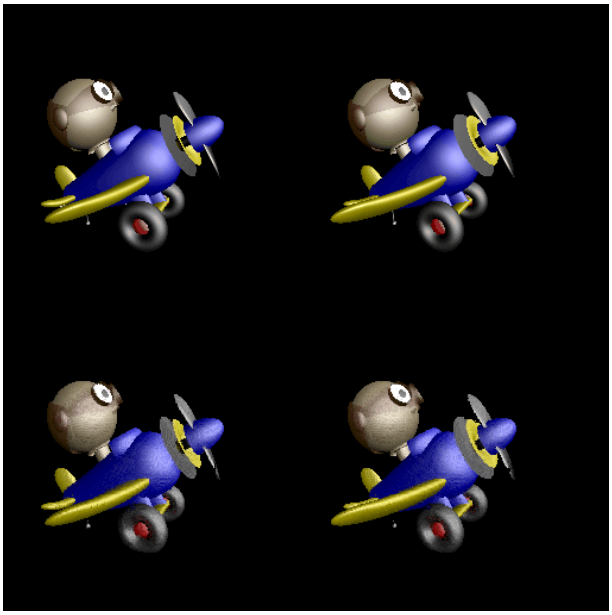

Figure 6: **Top:** Two original images from the *Airplane* light field. **Bottom:** The two corresponding views rendered from the reconstructed voxel volume.

the middle of four views from the grid reproduced in Fig. 5. On the left hand side of Fig. 8 we show the traditional bilinear interpolation without knowledge of scene depth. The right hand side of Fig. 8 shows the interpolation result when using the 3-D information that has been extracted from the light field using the methodology described in this paper. The interpolation is performed using disparity compensated pixel positions in the 4 surrounding light field views. With bilinear interpolation the interpolated image is unacceptably blurred, while with disparity compensation it exhibits only very minor artifacts. The third experiment uses 24 views of the *Cactus* light field. 4 original light field views are shown in Fig. 9. Each view is taken from a different $(u, v)$ plane. Since the light source was moving with the camera during acquisition of the light field, strong illumination effects are present in the sequence. There is, e.g., strong shading at both sides of the flowerpot. Since the similarity criterion in (4) uses normalized colors, those effects do not diminish the quality of the reconstructed views as can be seen in Fig. 10, where the corresponding views rendered from the reconstructed voxel volume are reproduced. The reconstructed voxel-based description is of high quality and the corresponding depth maps are reproduced in Fig. 11.

# 5 Conclusion

In this paper we presented a voxel-based approach for the 3-D reconstruction of objects from light field images. The algorithm is fully voxel-based and therefore does **not** require to establish image point correspondences. The algorithm first extracts a set of hypotheses for

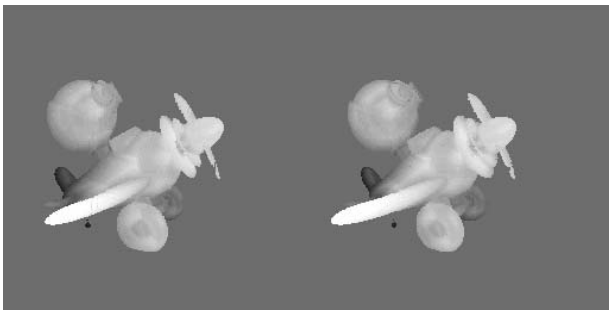

Figure 7: Depth map of the reconstructed 3-D model for the same viewing positions as in Fig. 6.

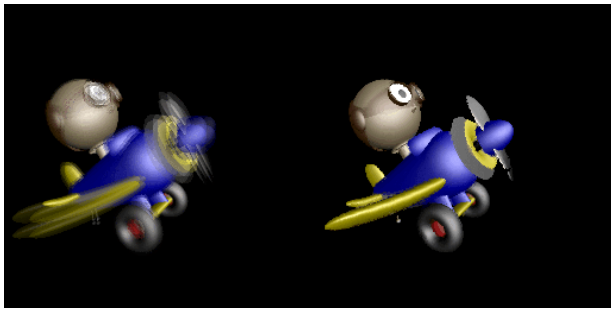

Figure 8: **Left:** a new view generated from the light field using standard bilinear interpolation in the $(u, v)$ plane. **Right:** Disparity compensated interpolated view with reconstructed approximative geometry.

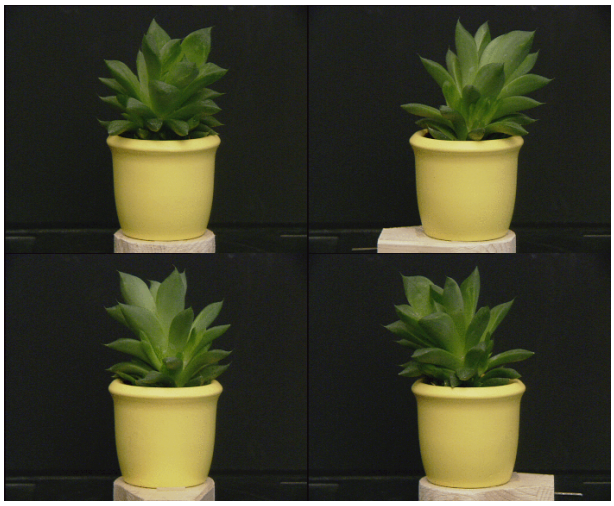

Figure 9: 4 light field views of a cactus.

each voxel in the scene and then exploits the back-projection of the visible surface voxels to remove all hypotheses which are not consistent with the individual light field views. In this way, it combines the advantages of silhouette-based and surface-based 3-D reconstruction techniques. The description of the object is a set of voxels with associated color values that can be used for either surface extraction or the production of new intermediate views which were not available in the light field. The usage of normalized voxel colors leads to an illumination insensitive reconstruction that provides excellent reconstruction quality even for moving light sources.

Figure 10: The corresponding cactus views rendered from the reconstructed voxel volume.

# References


[1] M. Levoy and P. Hanrahan, "Light Field Rendering," *Proc. Siggraph '96*, pp. 31-42, August 1996.

[2] S. J. Gortler, R. Grzeszczuk, R. Szeliski and M. F. Cohen, "The Lumigraph," *Proc. Siggraph '96*, pp. 43-54, August 1996.

[3] M. Magnor and B. Girod, "Hierarchical Coding of Light Fields with Disparity Maps," *ICIP '99*, Kobe, Japan, October 1999.

[4] P. Beardsley, P. Torr, and A. Zisserman, "3D Model Acquisition from Extended Image Sequences," *Proc. ECCV '96*, pp. 683-695, Cambridge, UK, 1996.

[5] M. Pollefeys, R. Koch, M. Vergauwen, and L. Van Gool, "Flexible Acquisition of 3D Structure from Motion," *Tenth IMDSP Workshop 1998*, pp. 195-198, Austria, 1998.

[6] B. C. Vemuri, J. K. Aggarwal, "3-D Model Construction from Multiple Views Using Range and Intensity Data," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 435-437, Miami Beach, 1986.

[7] E. Boyer, "Object models from contour sequences," *Proc. European Conference on Computer Vision (ECCV)*, pp. 109-118, 1996.

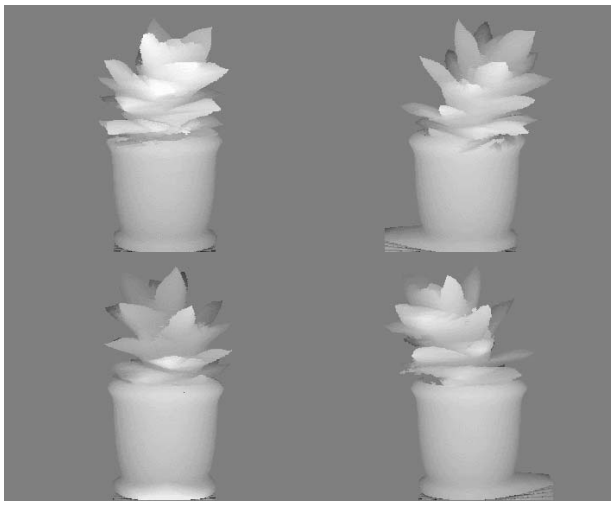

Figure 11: Depth maps rendered from the reconstructed voxel-based description of the cactus.


[8] S. Sullivan and J. Ponce, "Automatic model construction, pose estimation, and object recognition from photographs using triangular splines," *Proc. International Conference on Computer Vision (ICCV)*, 1998.

[9] W. Niem, J. Wingbermühle, "Automatic Reconstruction of 3D Objects Using a Mobile Monoscopic Camera", *Proceedings of the International Conference on Recent Advances in 3D Imaging and Modelling*, Ottawa, Canada, May 1997.

[10] R. Szeliski, "Rapid octree construction from image sequences," *CVGIP 93*, pp. 23-32, July 1993.

[11] A. W. Fitzgibbon, G. Cross, and A. Zisserman, "Automatic 3D Model Construction for Turn-Table Sequences," *Proc. ECCV 98 Workshop on 3D Structure from Multiple Images of Large-Scale Environments*, Freiburg, 6-7th June 1998.

[12] S. M. Seitz and C. R. Dyer, "Photorealistic Scene Reconstruction by Voxel Coloring," *Proc. Computer Vision and Pattern Recognition (CVPR '97)*, pp. 1067-1073, Puerto Rico, 1997.