MULTI-HYPOTHESIS, VOLUMETRIC RECONSTRUCTION OF 3-D OBJECTS FROM MULTIPLE CALIBRATED CAMERA VIEWS

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
http://www.nt.e-technik.uni-erlangen.de/~eisert/reconst.html

ABSTRACT

In this paper we present a volumetric method for the 3-D reconstruction of real world objects from multiple calibrated camera views. The representation of the objects is fully volume-based and no explicit surface description is needed. The approach is based on multi-hypothesis tests of the voxel model back-projected into the image planes. All camera views are incorporated in the reconstruction process simultaneously and no explicit data fusion is needed. In a first step each voxel of the viewing volume is filled with several color hypotheses originating from different camera views. This leads to an overcomplete representation of the 3-D object and each voxel typically contains multiple hypotheses. In a second step only those hypotheses remain in the voxels which are consistent with all camera views where the voxel is visible. Voxels without a valid hypothesis are considered to be transparent. The methodology of our approach combines the advantages of silhouette-based and image feature-based methods. Experimental results on real and synthetic image data show the excellent visual quality of the voxel-based 3-D reconstruction.

1. INTRODUCTION

There is a tremendous interest from Virtual Reality (VR) and multimedia applications to obtain 3-D computer models of real world objects. One common approach is to take multiple camera views from different positions 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 camera 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 [1, 2] or is supported by additional depth information from range sensors [3]. The second class of algorithms is based on volume intersection, and is often refered to as *shape-from-silhouette* algorithms [4, 5, 6, 7, 8]. 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 combine the advantages of both approaches by using a volumetric representation of the 3-D object and a multi-hypothesis testing of the back-projection of the object surface voxels with the camera views. A similar approach that also combines both advantages has been presented by Seitz and Dyer [9]. However, the algorithm in [9] 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 of the camera 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 volumetric description of the scene.

2. VOLUMETRIC 3-D OBJECT RECONSTRUCTION

In contrast to methods that exploit color information and feature matching to obtain a surface description of the object we use a volumetric description of the scene. All operations during 3-D reconstruction are performed on voxels. In comparison to pixels, voxels are unique from view to view. 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

2.1. 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. 1 shows an example of the initial volume with large voxels for illustration purposes. Typical dimensions are $200 \times 200 \times 200$ voxels.

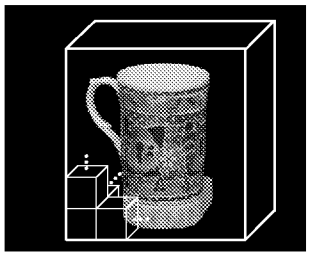

Figure 1: Bounding box of the volume with four voxels for illustration purposes.

2.2. Hypothesis Extraction

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

$$H^k_{lmn} = (R(X_i, Y_i), G(X_i, Y_i), B(X_i, Y_i)) \tag{1}$$

where (X_i, Y_i) is the pixel position of the perspective projection of the voxel center (x_l, y_m, z_n) into the ith camera view. R, G, and B are the three color components. The projection of the voxel center for view i is obtained as

$$\begin{aligned} X_i &= f_x \frac{x_{li}}{z_{ni}} \\ Y_i &= f_y \frac{y_{mi}}{z_{ni}} \end{aligned} \tag{2}$$

with

$$(x_{li}, y_{mi}, z_{ni})^T = \mathbf{R_i}(x_l, y_m, z_n)^T + \mathbf{T_i} \tag{3}$$

with $\mathbf{R_i}$ and $\mathbf{T_i}$ the object rotation and translation in view i with respect to the reference coordinate system. The parameters f_x and f_y describe the camera geometry and the scaling that relates pixel coordinates to world coordinates.

Hypothesis H^k_{lmn} is associated to voxel V_{lmn} if the projection of V_{lmn} into at least one other camera view $j \neq i$ leads to an absolute difference of the color channels less than a predetermined threshold Θ, i.e.,

$$\begin{aligned} e_{ij} = \quad & |R(X_i, Y_i) - R(X_j, Y_j)| + \\ & |G(X_i, Y_i) - G(X_j, Y_j)| + \\ & |B(X_i, Y_i) - B(X_j, Y_j)| < \Theta. \end{aligned} \tag{4}$$

Equation (4) defines the hypothesis criterion for 2 camera views and has to be evaluated for each of $N(N-1)$ pairs (i, j), where N is the number of views. For all combinations of i and j that pass equation (4) a color hypothesis H^k_{lmn} from view i according to (1) 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.

2.3. Consistency Check and Hypothesis Removal

In the previous step we stored multiple hypotheses for each voxel of the working volume. Those hypotheses were extracted from 2 or more consistent views without knowledge of the object's geometry. The non-transparent surface voxels of the overcomplete object representation are now used as an initial geometry estimate for hypothesis removal. We iterate several times over all available views and remove inconsistent voxels until the correct 3-D shape of the object is recovered.

For each view we determine the currently visible voxels and compare all associated hypotheses with the corresponding pixel color at the pixel position in (2). The similarity measure is again the absolute difference of the color components in (4). If the error in (4) exceeds the threshold Θ 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 one voxel 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 volumetric description of our 3-D object. The color values associated with the resulting non-transparent voxels which are on the object surface can now be used for rendering.

2.4. Visible Surface Determination

The hypothesis check and subsequent hypothesis removal for each view i requires the determination of the visible surface voxels from the current view of the volume. The algorithm used for visible voxel determination works as follows.

We first determine the plane of the volume bounding box that is facing the camera for the current view. This is achieved by rotating the optical axis $\mathcal{OA} = (0, 0, -1)^T$ of the camera of view i according to the object pose

$$\mathcal{OA}_i = \mathbf{R_i}^{-1} \mathcal{OA}. \tag{5}$$

The largest scalar product of the transformed optical axis $\mathcal{OA}_i$ and the 6 surface normals of the bounding box $(0, 0, 1)$, $(0, 0, -1)$, $(0, 1, 0)$, $(0, -1, 0)$, $(1, 0, 0)$, and $(-1, 0, 0)$ determines the required permutation of the loop indices (l, m, n). In order to determine if the loop indices have to be evaluated in decreasing or increasing order we transform the 8 corners of the volume into the camera coordinate system of view i and compute the distance d^k of these points (x^k_c, y^k_c, z^k_c), $k = 1..8$ to the camera projection center $(0, 0, 0)$

$$d^k = \sqrt{(x^k_c)^2 + (y^k_c)^2 + (z^k_c)^2}\,. \tag{6}$$

The corner corresponding to the smallest distance determines the direction how to access the voxels for each index l, m, and n. This ensures that we access voxels in the order of increasing depth from this camera view. From an implementation point of view this simply results in exchanging the loop indices l, m, n when stepping through the volume. Using the resulting voxel ordering we store for each pixel in the camera image the index of the first voxel that is projected into that pixel. This voxel has the smallest depth of all voxels projected onto that pixel in the current view.

2.5. Rendering of Arbitrary Views

Once the volumetric description of the object is determined we can render views from arbitrary viewing positions. For that purpose

we transform the volume according to equation (3). The pixels in the virtual views are set using the projection formulae in (2). 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. EXPERIMENTAL RESULTS

The first sequence is an 11 view sequence of a cup with homogeneous background recorded with a video camera. The original frames 0, 3, 6, and 9 are shown in Fig. 2. From this sequence we

Figure 2: Original frames 0, 3, 6, and 9 of the *cup* sequence.

reconstructed the 3-D volume data set for camera positions that were estimated using camera calibration. The recovered volumetric description of the object is shown in Fig. 3 in terms of the corresponding depth map obtained from the surface voxels for the same views as in Fig. 2. The projection of the recovered 3-D model into

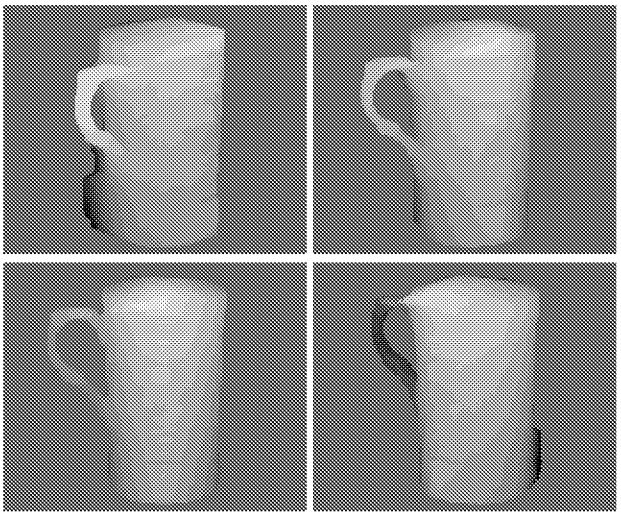

Figure 3: Depth map of the reconstructed 3-D model for the same viewing positions as in Fig. 2.

a virtual image plane is shown in Fig. 4. Fig. 5 shows a zoomed

Figure 4: Rendered 3-D model for the same viewing positions as in Fig. 2.

version of the original and reconstructed image for view 0 of the cup sequence. It can be seen that the rendered 3-D model has about the same sharpness as in the original view.

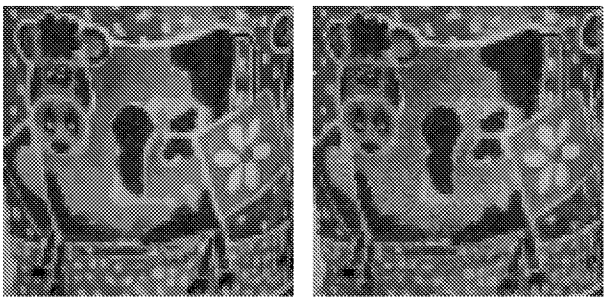

Figure 5: Magnified frame 0 original (left) and rendered (right). The rendered model shows the same sharpness as the original frame.

In the previous experiment the object was captured in front of a homogeneous background where the silhouette information is implicitly exploited. In a second experiment we applied the algorithm to a synthetic sequence of 14 views of a video cassette where the background was not homogeneous. We turned the video cassette but the background remained constant. Fig. 6 shows frames 0 and 3 of the sequence. For this scene the segmentation of the video cassette from the background is not trivial and therefore no explicit silhouette information is exploited. Because the motions of the object and the background do not coincide we reconstruct only those parts of the image that move in the same way as our voxel data set. This leads to an implicit segmentation of objects with different motion parameter sets. The recovered volumetric description of the object is shown in Fig. 7 in terms of the corresponding depth map obtained from the surface voxels for the same views as in Fig. 6. The projection of the recovered 3-D model into a virtual image plane is shown in Fig. 8. The 3-D models can be rendered from arbitrary viewing positions. Fig. 9 shows examples

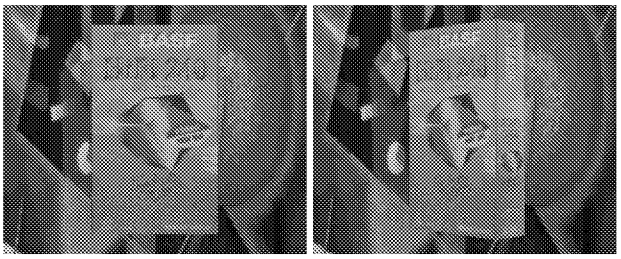

Figure 6: Original frames 0 and 3 of the *cassette* sequence.

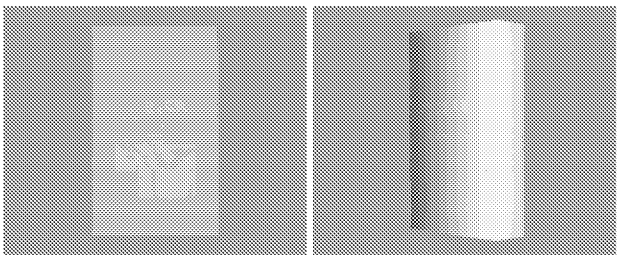

Figure 7: Depth map of the reconstructed 3-D model for the same viewing positions as in Fig. 6.

of views that are not included in the original sequences. Table 1 summarizes the simulation parameters selected for the two reconstruction examples. During one iteration in the last row of Table 1 all views are processed once. It can be seen from the table that the initial number of hypotheses is much higher than the number of finally recovered non-transparent surface voxels.

4. CONCLUSION

In this paper we presented a voxel-based approach for the 3-D reconstruction of real world objects from multiple calibrated camera views. The algorithm first extracts a set of hypotheses for 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 camera views. 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 initial set of camera views.

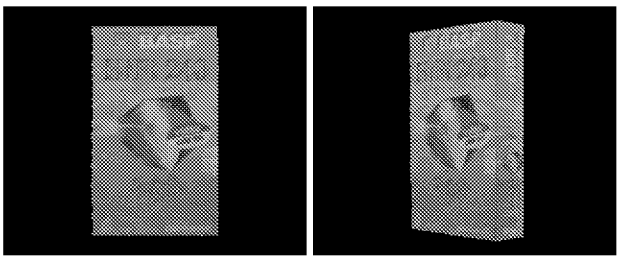

Figure 8: Rendered 3-D model for the same viewing positions as in Fig. 6.

	cup sequence	cassette sequence
image resolution	352×288	352×288
number of images	11	14
voxel resolution	$240 \times 240 \times 140$	$180 \times 256 \times 60$
initial number of hypotheses	$3.62 \cdot 10^7$	$2.26 \cdot 10^7$
final number of visible voxels	$6.8 \cdot 10^4$	$1.02 \cdot 10^5$
number of iterations	15	24

Table 1: Experimental results: image and volume resolution, number of hypotheses and number of iterations.

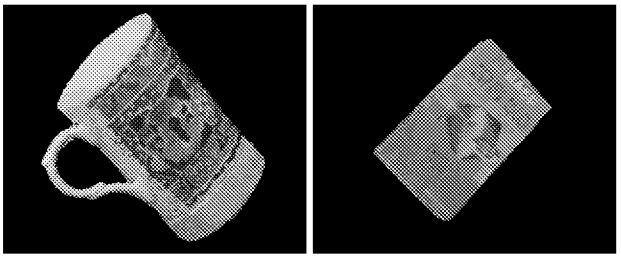

Figure 9: Reconstructed images for viewing positions that are not part of the camera sequence.

5. REFERENCES

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

[2] 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.

[3] 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.

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

[5] 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.

[6] 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.

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

[8] 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.

[9] 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.