Czech Technical University, Prague, Czech Republic
The Publication of the Center for Machine Perception

Rendering Real-World Objects Using View Interpolation

Tomáš Werner Roger David Hersch Václav Hlaváč
{werner,hlavac}@vision.felk.cvut.cz, Roger.Hersch@di.epfl.ch

September 23, 1996

Reference

[1] Tomáš Werner, Roger David Hersch, and Václav Hlaváč. Rendering real-world objects without 3-d model. In Václav Hlaváč and Radim Šára, editors, *Proceedings of the 6th International Conference on Computer Analysis of Images and Patterns, CAIP'95*, pages 146–153, Praha, Czech Republic, 1995. Springer Verlag.

This publication can be obtained via anonymous ftp from
ftp://cmp.felk.cvut.cz/pub/cvl/ARTICLES/WERNER/CAIP96.ps.Z

Czech Technical University, Faculty of Electrical Engineering
Department of Control Engineering, Center for Machine Perception, Computer Vision Laboratory
121 35 Prague 2, Karlovo náměstí 13, Czech Republic, FAX +42 2 290159, phone +42 2 24357465
http://cmp.felk.cvut.cz

1 Introduction

Methods which are able to capture a real object and render it from an arbitrary different viewpoint usually use a 3-D model of the object. The bottleneck of these methods is the 3-D model reconstruction. It is a non-trivial problem which often fails for objects of complex shapes. In our work, we show that, provided that only rendering is required, the model reconstruction can be avoided and thus much more complex objects can be handled. Moreover, faster access to a view can be achieved than by rendering the 3-D model.

Our approach involves the representation of the object by a *set of primary views*[1], rather than by its 3-D model. Any view not contained in this set is obtained as a combination of a small number of the closest primary views. We will show that it can be calculated as a *linear combination* of a subset of the views under an assumption of their proximity. We will also suggest how to solve the visibility problem in the new view.

Unlike many other approaches utilizing active methods with an auxiliary source of a structured light for acquiring correspondence, our approach allows the use of a *passive method*. View interpolation is less sensitive to errors in correspondence caused by ambiguity in areas of constant intensity than the 3-D model reconstruction (these errors are inherent to passive matching algorithms). We use the intensity-based passive algorithm [1], which we have further improved.

The presented work is part of a project called *Interview*[2], aiming to develop a prototype of a device which is able to capture and render complex real-world objects by view interpolation. The rendering algorithm should be implemented on a multiprocessor system. We expect to achieve a frame generation rate sufficient for real-time moving of the objects on the screen. This paper overviews the theoretical background along with the description of the preliminary experiments with the interpolated view synthesis, indicating that our approach is robust and feasible.

2 Previous Work

As far as we know, there are very few papers on using view interpolation for accessing an image of a 3-D object from an arbitrary viewpoint. The most relevant is the recent paper [10], in which the authors propose to represent a 3-D scene by a collection of corresponding *reference views* and the fundamental matrices.

The work of Ullman and Basri [15] is inspiring, as it describes an approach to 3-D object recognition using a set of views on an object instead of its model. It is shown that any instance of the object can be expressed as a linear combination of these views assuming orthogonal projection. A similar and even earlier work is presented in [7].

Faugeras and Robert [3] deal with a prediction of coordinates, tangents and curvatures in a view on the basis of knowing other views. The cameras need to be only weakly calibrated.

[1] We use the term *primary views* not to confuse it with *characteristic views*, which represents the sets of structurally similar views on the viewing sphere [7].

[2] The project is supported (1) by the Peripheral Devices Laboratory, École Polytechnique Fédérale de Lausanne, Switzerland, (2) by the grant No. 102/93/0954 of the Grant Agency of the Czech Republic, (3) by the grant of the Czech Technical University No. 38196.

In the work of Gudmundsson and Randen [6], the view interpolation is used to speed up the generation of ray-traced sequences of objects rotating in 3-D space. They achieve speed-up factors of 10–15 compared to the full rendering.

Skerjanc and Liu in [14, 13] want to use intermediate views to achieve an effect of 3-D TV (the viewpoint changes according to the position of the viewer). Only the two extreme views are to be transmitted and the interpolated ones obtained by interpolation. In [16], an object is scanned along a pre-defined path, the correspondence is found, several views are selected, and an arbitrary interpolated view is constructed.

3 View Interpolation

3.1 Our Approach

Let us outline how to render a 3-D object from an arbitrary viewpoint without having its model. A set of views covering the whole visible surface of the object is captured. These views can be accessed directly; all we need to do is display them. What is demanded are the intermediate views. If information of primary views correspondence is available, it is possible to obtain any interpolated view using a subset of the primary views close to it. This allows us to avoid the difficult fusion of the views that is necessary for the 3-D model reconstruction.

To succeed, the following problems must be solved:

1. How to determine the position and intensity of a point in the interpolated view if positions and intensities of corresponding points in the primary views are known?

2. How to determine the visibility of points in the interpolated view?

3. How to find the possibly smallest set of necessary primary views?

4. How to find correspondences between primary views?

In the following sections, we give answers to some of these questions, namely to 1, 2 and 4. The problem of the choice of the smallest and still sufficient set of the primary views is non-trivial. The more restricted fundamental problem of how to choose a set of so-called *characteristic views* (i.e., the minimum set of views in which all points of a given surface are visible) still remains unsolved for general non-convex objects. In our case the primary views set would have to fulfill additional requirements, e.g., a reasonable sampling frequency for all points on the surface. We do not deal with question 3 in this paper.

3.2 Position and Intensity of the Interpolated Point

Let us assume n corresponding points in n primary images, i.e., $2n$-tuple $[\mathbf{x}_1, I_1, \ldots, \mathbf{x}_n, I_n]$, where $\mathbf{x}_i$ are image coordinates of i-th point and I_i, its intensity. The i-th point is a projection of some point $\mathbf{X}$ on the object's surface in the i-th view with view parameters[3]

[3]We consider ideal perspective projection acting in image formation. *View parameters* do not mean a complete set of camera parameters, but rather a point in a variety of allowed views. E.g., if an object is viewed by a camera moving on a spherical surface and looking to the center of this sphere, the view parameters are three angles.

$\mathbf{p}_i$. The goal is to obtain the coordinates $\mathbf{x}$ and intensity I of the projection of $\mathbf{X}$ in a interpolated view with parameters $\mathbf{p}$.

If all the views are calibrated, the following system of equations is usually to be solved:

$$\begin{aligned}\mathbf{x}_i &= f(\mathbf{X}, \mathbf{p}_i),\ i = 1, \dots, n \\ \mathbf{x} &= f(\mathbf{X}, \mathbf{p}),\end{aligned} \tag{1}$$

where f is a function assigning its projection to a point in 3-D space (f and $\mathbf{p}_i$ are known from the calibration). If $n = 2$ the system has a unique solution, if $n \geq 2$ the solution can be obtained by a least squares fit. The case of $n = 2$ is equal to finding all three coordinates of $\mathbf{X}$ by triangulization, followed by the projection to the virtual camera coordinate system.

We show that this approach can be simplified. Let us express the parameters of the interpolated view $\mathbf{p}$ as a linear combination of parameters of n primary views, $\mathbf{p} = \sum \alpha_i \mathbf{p}_i$. If $\sum \alpha_i = 1$ and $\mathbf{p}_i$ are close enough to each other, it holds that:

$$f(\mathbf{X}, \sum_{i=1}^{n} \alpha_i \mathbf{p}_i) \approx \sum_{i=1}^{n} \alpha_i f(\mathbf{X}, \mathbf{p}_i), \tag{2}$$

i.e.,

$$\mathbf{x} = \sum_{i=1}^{n} \alpha_i \mathbf{x}_i. \tag{3}$$

The proof can be made by substituting $\mathbf{p}_i = \mathbf{q}_0 + \Delta\mathbf{q}_i$, $\mathbf{p} = \mathbf{q}_0 + \sum \alpha_i \Delta\mathbf{q}_i$, and by expanding the function $\mathbf{g}(\mathbf{p}) = f(\mathbf{X}, \mathbf{p})$ to its Taylor series along the point $\mathbf{q}_0$, neglecting the terms higher than first order.

The same procedure can be used for interpolation of image intensities (or values of color components). If we formally replace f with reflectance (or color) model, we obtain $I = \sum \alpha_i I_i$.

The numbers α_i can be obtained by solving the equations:

$$\begin{aligned}\sum_{i=1}^{n} \alpha_i \mathbf{p}_i &= \mathbf{p},\ i = 1, \dots, n \\ \sum_{i=1}^{n} \alpha_i &= 1\end{aligned} \tag{4}$$

The (4) has a solution for $n \geq dim(\mathbf{p}) + 1$. It implies that we need at least $dim(\mathbf{p}) + 1$ primary views to determine the interpolated one, without any camera calibration.

It is reasonable to call the construction of a new view an *interpolation*, if for all i holds $0 \leq \alpha_i \leq 1$, and an *extrapolation*, if the converse is true. We expect the error due to the linearization to be smaller for the interpolation.

3.3 Visibility of interpolated points

Let us make a restriction to opaque objects (we will not deal with transparent objects in this paper). The interpolated view can be constructed as a set of combinations $\sum \alpha_i f(\mathbf{X}_k, \mathbf{p}_i)$ of corresponding projections of the object's surface points $\mathbf{X}_k$ from a given subset of the whole primary view set. Let us denote p_i and p the projections of a constructed point $\mathbf{X}_k$ in views with parameters $\mathbf{p}_i$ and $\mathbf{p}$, respectively. Let us now distinguish six cases that can arise for the visibility of $\mathbf{X}_k$ in particular views (Figure 1) and discuss the way they influence the construction of the interpolated view:

1. All p_i are visible, p is visible.
   The interpolated point p can be constructed from the given subset and it is visible. This case should occur for as many interpolated points as possible.

2. All p_i are visible, p is invisible.
   The point $\mathbf{X}_k$ is occluded in the interpolated view, but interpolated point p can be constructed from the given subset. This case occurs if for some other n-tuple $f(\mathbf{X}_l, \mathbf{p}_i), l \neq k$, the equality $\sum \alpha_i f(\mathbf{X}_l, \mathbf{p}_i) = \sum \alpha_i f(\mathbf{X}_k, \mathbf{p}_i)$ holds[4] (and it can be also detected like this). It is necessary to decide which of the two points $\mathbf{X}_k$, $\mathbf{X}_l$ is visible. This decision can be made using their disparity, because disparity is a descending function of depth. The suitable data structure for the implementation is a z-buffer — the buffered value can be the disparity of the interpolated points, referred to a chosen primary view.

3. Some p_i are visible and some invisible, p is visible.
   The point $\mathbf{X}_k$ is occluded in one of the primary views. Therefore the n-tuple $f(\mathbf{X}_k, \mathbf{p}_i)$ does not exist, if a real device was used to capture it, and the interpolation at the point p cannot be computed. The areas where these situations occur remain empty[5] in the constructed view. The missing points can be obtained by an extrapolation of a different (e.g. neighboring) subset of primary views, provided that they are not occluded in them.

4. Some p_i are visible and some invisible, p is invisible.
   The point $\mathbf{X}_k$ is not visible from the interpolated view and it is not constructed.

5. All p_i are invisible, p is visible.
   The point $\mathbf{X}_k$ is visible in the interpolated view but occluded in all primary views in the subset. If it is occluded in all other primary views (which is likely to occur if p is constructed by view interpolation), there is no information about it. This case is undesirable and it should be prevented by choosing the optimal set of primary views. It can be detected the same way as case 2.

6. All p_i are invisible, p is invisible.
   This is the same case as 4.

We conclude that during the construction of the interpolated view, it is necessary (and possible) to detect the cases 2, 3 and 5. The correct handling of the cases 2 and 3 can be ensured, while the case 5 must be prevented in advance by the proper choice of the set of primary views.

[4]This occurs if and only if the so-called ordering constraint is violated in some pair of the primary views. Therefore, a matching algorithm utilizing this constraint cannot be used for acquiring correspondence. Unfortunately, the majority of them does so.

[5]These areas must be distinguished from the holes caused by unequal sampling frequencies in the primary and interpolated views.

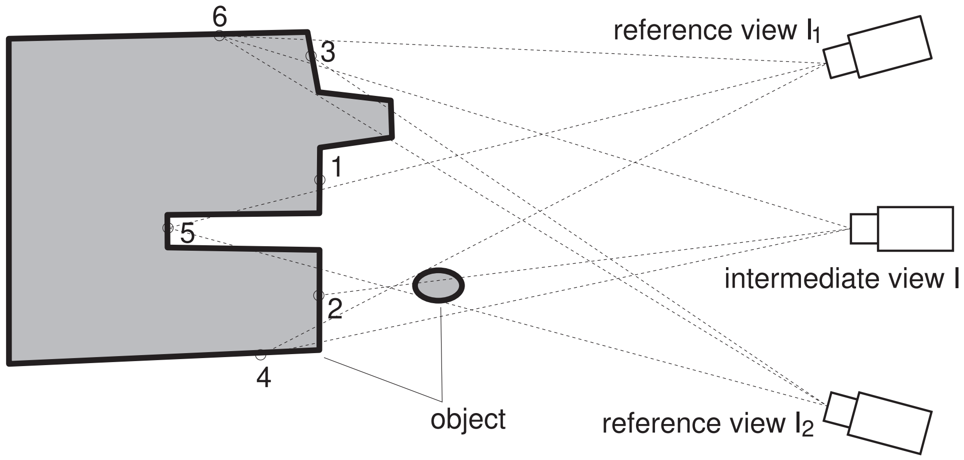

Figure 1: The six cases which can arise for the visibility of points in the interpolated view. The numbers correspond to the notation in text.

3.4 Correspondence Acquisition

We need to find correspondence for the subsets of the primary views from which the interpolated views will be constructed. In other words, we are looking for all n-tuples $[\mathbf{x}_1, \ldots, \mathbf{x}_n]$ of pixels coordinates (each from a different view) so that every n-tuple contains projections of a single point on the object's surface, up to errors caused by discretization and noise. This *correspondence problem* is difficult mainly due to: (1) the change of geometric and photometric properties with the viewpoint, (2) the presence of noise, (3) discretization, and (4) the lack of information in areas of constant intensity.

Rendering by view combination brings an important advantage in comparison to the approaches using a 3-D model: low sensitivity to correspondence errors in areas of (almost) constant intensity. The areas in which these errors show up in the interpolated view will also have a constant intensity, so the image will *look* the same. Even if the visibility of the interpolated pixel is inferred from disparities (which is quite a rare situation), only minor artifacts can be expected. This insensitivity allows us to use passive methods for acquiring correspondence, having the potential to capture both indoor and outdoor objects, and the possibility to provide directly dense correspondence.

3.4.1 Matching Algorithm

There are two alternatives for determining correspondence of n views: (1) n-ocular stereo, and (2) composition of pairwise correspondences[6]. We chose the alternative (2).

The algorithm we use is inspired by the binocular stereo matching algorithm [1]. It provides dense correspondence, as it is intensity-based. It is similar to the more known algorithm [11], but rather than intervals between edges, the raw pixel intensities are matched. Epipolar, unicity, and ordering constraints are utilized. For each pair of corresponding epipolars, such a pair of non-descending transformation functions is sought, that

[6]The accumulation error limits the number of compositions. This error arises by accumulating errors in composed pairwise correspondences, always present due to noise.

transforms the intensity functions so that the sum of costs of either matches or occlusions is minimized. This optimization problem is solved by dynamic programming. The cost of a match and occlusion is derived — the problem being formulated as a Bayesian sensor fusion. The advantage of the algorithm is that it directly produces dense correspondence.

We have observed that the algorithm yields more accurate results if the maximized quantity is a sum of correlations of windows having the matched pixels as their centers, rather than costs of matches of the pixels alone. Thus, we have improved the original algorithm by combining it with a correlation method.

3.4.2 Calibration

As the epipolar constraint is utilized, the two views must be *weakly calibrated.* Our calibration is based on the notion of fundamental matrix [3, 2].

The candidates for a match to a pixel $\mathbf{m}_1$ in the view 1 are located on a line $\mathbf{l}_2$ in the view 2 (called epipolar line), rather than in the whole image 2, as is the case without the constraint[7]. It holds that:

$$\mathbf{l}_2 = \mathbf{F}_{12}\mathbf{m}_1 \tag{5}$$

(for the views switched, the same is true with $\mathbf{F}_{12}^T$ instead of $\mathbf{F}_{12}$). $\mathbf{F}_{12}$ is the fundamental matrix. It completely determines the epipolar geometry and if it is known, the camera pair is said to be weakly calibrated. It is singular and determined up to a scale factor.

The fundamental matrix can be determined from a set $\{[\mathbf{m}_{1i}, \mathbf{m}_{2i}]\}, i = 1, \ldots, N$, of non-coplanar corresponding pairs. If there is a pixel in the view 2 corresponding with $\mathbf{m}_1$, it is $\mathbf{m}_2^T\mathbf{l}_2 = 0$, hence, using eq. (5), $\mathbf{m}_2^T\mathbf{F}_{12}\mathbf{m}_1 = 0$ (Longuet-Higgins equation). It can be rewritten to:

$$\mathbf{M}\mathbf{f} = 0, \tag{6}$$

where $\mathbf{M}$ is a $9 \times N$ matrix containing coordinates of the corresponding pairs, and $\mathbf{f}$ is a 1×9 vector of $\mathbf{F}_{12}$ elements. In order to eliminate the scale factor, we introduce the constraint $\mathbf{f}^T\mathbf{f} = 1$. If the system of equations (6) is overdetermined, we obtain $\mathbf{f}$ as a solution to the optimization problem:

$$\min_{\mathbf{f}^T\mathbf{f}=1} \{(\mathbf{M}\mathbf{f})^T\mathbf{M}\mathbf{f}\}. \tag{7}$$

It can be easily shown that the solution is the norm eigenvector of the matrix $\mathbf{M}^T\mathbf{M}$ associated with the eigenvalue that has the smallest absolute value.

The matching algorithm uses *rectified images*, in which epipolars are parallel to scanlines. The rectified images can be obtained as projective transformations of the original ones which transform the epipolars to scanlines. After matching, the coordinates of the matched pixels are transformed by the inverse transformations.

4 Experiments

We conducted several experiments with the synthesis of interpolated views. We used real objects of quite complex shapes, for which the 3-D model reconstruction would have been

[7]Points $\mathbf{m}$ and lines $\mathbf{l}$ are assumed in homogeneous coordinates.

difficult. For simplicity, we allowed objects only to rotate around a vertical axis, the view parameters thus having one component (azimuth angle).

Our experimental setup consisted of a camera, a calibrated rotating table and a calibration grid. The corresponding pixel pairs for weak calibration were obtained by detecting line intersections in the calibration grid. The non-coplanarity of these calibration points was achieved by rotating the grid. After the calibration, the objects were placed on the table, and the views changed by rotating the table.

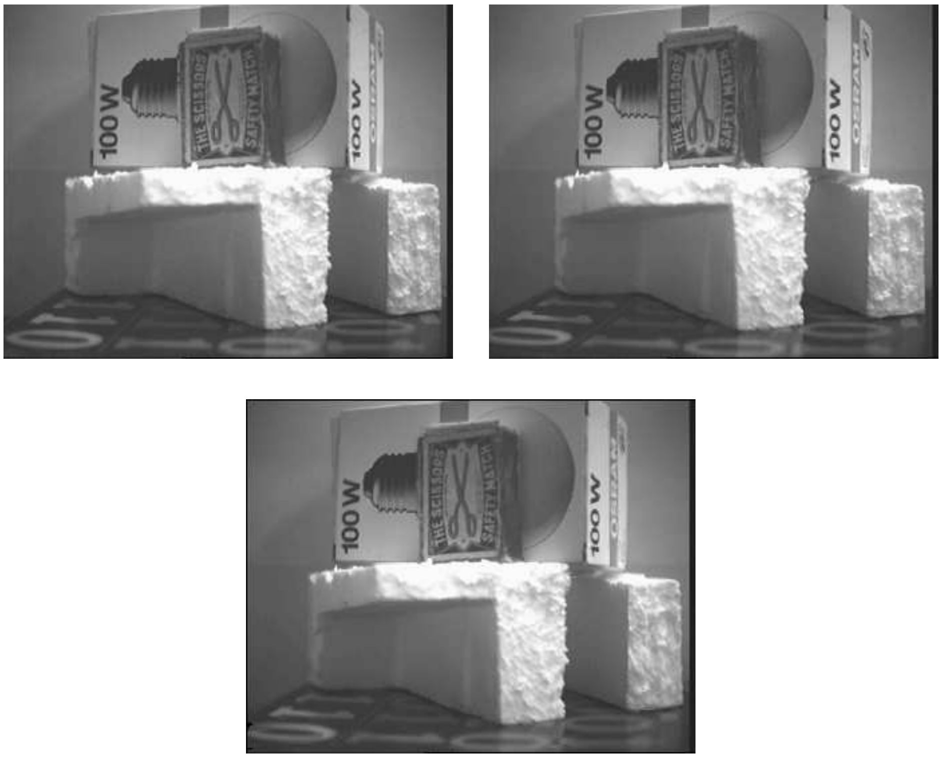

Figure 2: Two primary views of the first object (consisting of a book, a piece of styrofoam, a box, and a matches box), $\mathbf{p}_1 = [0°], \mathbf{p}_2 = [5°]$ (top) and the interpolated view $\mathbf{p}_1 = [2.5°]$, $\alpha_1 = \alpha_2 = 0.5$ (bottom).

As $dim(\mathbf{p}) = 1$, at least two real images are needed for calculating interpolated views. We captured two primary views of an object and matched them (after rectifying) using the described algorithm. Then we calculated interpolated views by view interpolation.

Fig. 2 shows one interpolated view of the first object. The difference between azimuth angles was 5°. Occluded areas, as well as holes due to unequal sampling frequencies were filled in by linear interpolation of intensities on epipolars, so the results of the section 3.3 were not utilized. Even then, no artifacts in the interpolated image are visible.

Fig. 3 shows a more complex object. The difference of the azimuth angles is 10°. As large areas of the object's surface are occluded, pixels in or near these areas were matched incorrectly. While this would cause rough errors in the 3-D model, the interpolated image is quite acceptable.

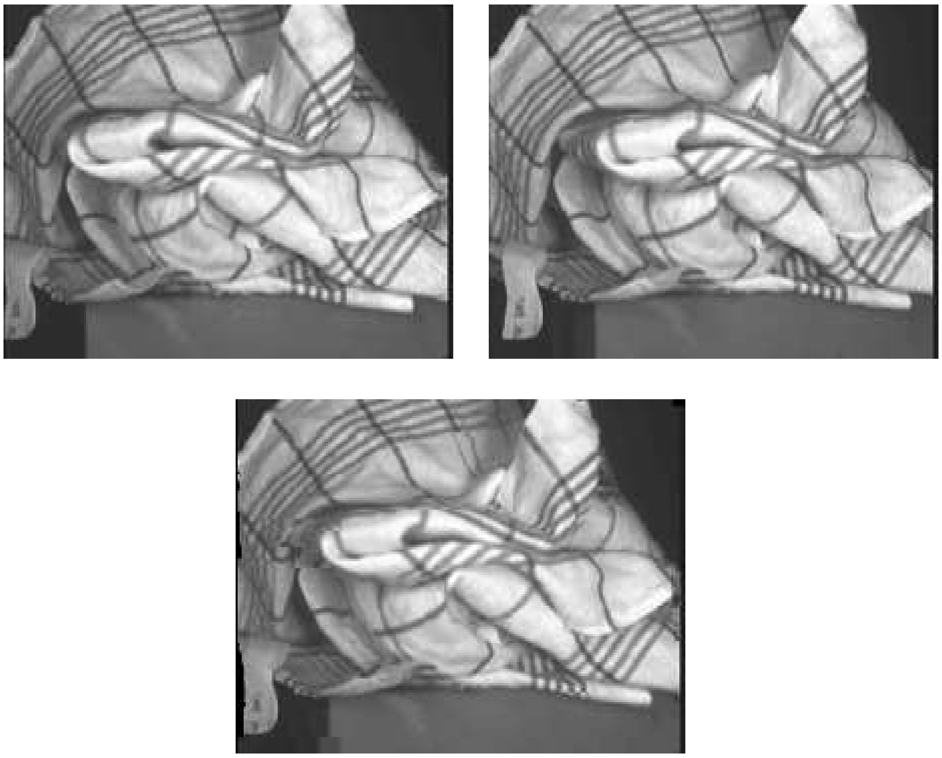

Figure 3: Two primary views of the second object (a linen towel), $\mathbf{p}_1 = [0°], \mathbf{p}_2 = [10°]$ (top) and the interpolated view $\mathbf{p}_1 = [5°]$, $\alpha_1 = \alpha_2 = 0.5$ (bottom).

5 Conclusion

This paper is an attempt to show that a 3-D model is not the only possible representation of a 3-D object, provided that only rendering is required. We have suggested an alternative representation by a relatively sparse set of captured views. The missing views are calculated using information about their mutual correspondence. Furthermore, we have dealt with construction of these new views. We have shown that a interpolated view can be expressed as a linear combination of the primary ones, assuming the proximity of the views, and how the visibility of points in the interpolated view can be determined. We have not yet solved the important question of the choice of the optimal primary view set.

We have proven the possibility to construct interpolated views for quite complex objects. The presented results indicate the feasibility of our approach.

In the future, we plan to make an analysis of errors due to linear combination of views, to improve the matching algorithm and to find at least a partial solution to the problem of optimal primary view set. Finally, we intend to implement the rendering algorithm on a multiprocessor-multidisk system [5].

References

[1] I. J. Cox, S. Hingorani, B. M. Maggs, and S. B. Rao. Stereo without disparity gradient smoothing: a Bayesian sensor fusion solution. In *British Machine Vision*

Conference, pages 337–346, Berlin, 1992. Springer-Verlag.

[2] O. Faugeras. *Three-Dimensional Computer Vision: A Geometric Viewpoint.* The MIT Press, 1993.

[3] O. Faugeras and L. Robert. What can two images tell us about a third one? In *European Conference on Computer Vision*, pages 485–492, 1994. Also the technical report 1993, available on ftp.inria.fr.

[4] O. D. Faugeras, Q. T. Luong, and S. J. Maybank. Camera self-calibration: Theory and experiments. In *European Conference on Computer Vision*, pages 321–333, 1992.

[5] B. A. Gennart, B. Krummenacher, L. Landron, and R. D. Hersch. Gigaview parallel image server performance analysis. In *World Transputer Congress'94*, pages 120–135. IOS Press, September 1994.

[6] B. Gudmundsson and M. Randen. Fast generation of volume projection sequences. Linkoping University, Dept. of Electrical Eng., Sweden, 1991.

[7] I. Chakravarty and H. Freeman. Characteristic views as a basis for three-dimensional object recognition. In *Proc. of the Society for Photo-Optical Instrumentation Engineers Conference on Robot Vision*, pages 37–45, 1982.

[8] A. Koschan. Methodic evaluation of stereo algorithms. In R. Klette and W. G. Kropatsch, editors, *Theoretical Foundations of Computer Vision.* Akademie Verlag, 1992.

[9] A. Koschan. What is new in computational stereo since 1989: A survey on current stereo papers. Technical Report 93–22, Technische Universität Berlin, 1993.

[10] S. Laveau and O. Faugeras. 3-D scene representation as a collection of images. In *Proc. of 12th International Conf. on Pattern Recognition, Jerusalem, Israel*, pages 689–691, October 9–13 1994. Also the technical report, available on ftp.inria.fr.

[11] Y. Ohta and T. Kanade. Stereo by intra- and inter-scanline search using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(2):139–154, Mar 1985.

[12] A. Shashua. On geometric and algebraic aspects of 3D affine and projective structures from perspective 2D views. Technical Report AI Memo No. 1405, C.B.C.L. Paper No. 78, Massachusetts Institute of Technology, Artifical Intelligence Laboratory, July 1993.

[13] R. Skerjanc and J. Liu. Computation of intermediate views for 3DTV. In R. Klette and W. G. Kropatsch, editors, *Theoretical Foundations of Computer Vision.* Akademie Verlag, 1992.

[14] R. Skerjanc and J. Liu. A three camera approach for calculating disparity and synthesizing intermediate pictures. *Signal Processing: Image Communication*, 4(1):55–64, 1991.

[15] S. Ullman and R. Basri. Recognition by linear combinations of models. Memo 1152, MIT Artificial Intelligence Laboratory, August 1989.

[16] A. Zakhor and F. Lari. 3d camera motion estimation with applications to video compression and scene reconstruction. In *SPIE Symposium on Electronic Imaging*, pages 1–14, Dept. of Electrical Eng. and Computer Sciences, University of California, Berkeley CA94720, 1993.