A System for Image Based Rendering of Walk-throughs

Gaurav Agarwal Dinesh Rathi Prem K. Kalra Subhashis Banerjee

Department of Computer Science and Engineering
Indian Institute of Technology
New Delhi, 110016, India
email: {pkalra,suban}@cse.iitd.ernet.in

Abstract

We present a method for rendering novel views from a set of reference views under the assumption that scene surfaces can be approximated by planar patches. We use a set of sparse correspondences between the reference views to determine homographies through different planar patches using a clustering technique. We also obtain a segmentation of the scene in terms of planar regions visible from at least two views. Such segmentations explicitly resolve the visibility issues in novel view generation. We present results on rendering of realistic walk-through sequences for both indoor and outdoor scenes which demonstrate the applicability of our method.

1. Introduction

Recently, the problem of image based rendering has attracted considerable attention [2, 8, 12, 14, 4, 1], wherein the environmental map for rendering of novel views are maintained in terms of a set of images instead of explicit geometric and photometric models of a scene. In this paper we address the problem of generation of a sequence of novel views of a scene from a set of reference views. We assume that i) the scene surfaces can be approximated by planar patches (a common situation), and ii) every region in a target view is visible from at least two reference views; and develop a complete walk-through generation system starting from a set of sparse correspondences between the reference views. We consider the situation where the scene can have multiple occluding boundaries. We generate an intermediate representation of the scene in terms of planar patches from which enables on-line rendering of novel views in an automatic manner. We also present a simple visibility analysis algorithm to determine which parts of a target novel view are visible in which parts of the reference views. As a consequence, we can explicitly resolve the visibility issues of image based rendering - those of *folds*, where multiple points of the reference images map on to a single point in a target image, and *holes*, where a region occluded in a reference image becomes visible in a target image. Our method is based on segmentation of the scene in terms of homographies between regions in the reference images through the planar patches.

1.1. Previous work

Laveau and Faugeras [8] present a method of representing a 3D scene as a collection of reference images and their pairwise epipolar geometries. They propose a 2D ray-tracing algorithm which start from each point in the target image and locate the corresponding points in the reference images, from which the intensity information can be transferred. In case multiple world points map on to the same pixel in the target image, their method can resolve the ambiguity by depth ordering and can thus account for *folds*. Their method does not require explicit 3D reconstruction but require dense disparity maps and the pairwise epipolar relationships.

McMillan and Bishop [12] (see also [10, 13, 11]) use the “plenoptic function” (a parametrized representation of all visible rays from a camera) originally proposed in [2] to compute aggregate image warps from reference images to target images. In [10, 13] they assume that the projective depth of every point is known and compute the aggregate warp using the projective depth and known camera positions. In [12] they acquire panoramic reference images on cylindrical manifolds, which serve as plenoptic models, and compute the aggregate warp from the cylindrical epipolar geometries and dense angular disparity maps. In all the above methods they follow an order of painting the target images by moving towards the epipoles in the reference images which preserve the correct occlusion-compatible depth order [11]. This “painter’s” ordering result in target images free of *folds*. However all these methods require dense correspondences in some form.

In another significant approach to image based rendering Chen and Williams [3], and, Seitz and Dyer [14] use view interpolation to generate a target view from reference views. Chen and Williams [3] use image flow fields and local neighborhood analysis to reconstruct arbitrary target views with some constraints on gaze angle. Seitz and Dyer [14] show that a target view corresponding to a virtual camera can be generated by linear interpolation of the reference views provided the optical center of the target view lies on the line joining those of the reference views. In both these methods the visibility issues (*folds*) are resolved by z-buffering using disparity values. The view morphing method of Seitz and Dyer [14] has the restriction that the monotonicity of matches must be preserved between corresponding epipolar lines of the two reference views, i.e., the infinite line joining no two visible points P and Q should intersect the base-line between the reference views.

In a recent approach Lhuillier and Quan [9] present an algorithm for joint triangulation of matched regions in reference views and obtaining dense correspondences without using epipolar geometry. They use the joint view triangulation to interpolate a novel view. They do not discuss how they resolve the visibility issues but their results appear to be remarkably good when it comes to handling occlusions.

All the above methods require dense correspondences between the features in the reference views for rendering the novel views. Such correspondences are hard to obtain automatically, especially when the reference images differ in scale and illumination due to forward zooming motion (common in walk-throughs) or large baseline separations. Providing dense correspondence information manually is a tedious process and often require simplifying assumptions (like planarity, as in our case) for interpolation. In most of the above methods occlusion is handled either by a painting order or by z-buffering. While the painter's methods may result in unnecessary re-painting of target image regions, z-buffering is known to be memory inefficient. Laveau and Faugeras [8] explicitly compute the correct depth ordering in case multiple world points project on to the same image point on to the target view, but they need dense correspondences to do so.

In contrast, we automatically compute homographies between planar scene patches from a small set of seed correspondences (typically 15 to 20) which can either be provided manually or may even be computed automatically. The homographies through the planar patches provide the implicit interpolation necessary for rendering the novel views. Further, we explicitly compute a segmentation of the scene in terms of visibility from the reference views. As a consequence, we can explicitly resolve the visibility issues and no re-painting or z-buffering is necessary.

The rest of the paper is organized as follows. In Section 2 we introduce the notations and state the problem. In Section 3 we present our clustering algorithm for detecting homographies through all planar scene patches. In Section 4 we present our method for joint view segmentation for each planar patch. In Section 5 we present our algorithm for rendering a novel view. In Section 6 we present results on walk-through generation and finally, in Section 7 we conclude the paper.

2. System overview

We consider the problem of rendering target views along a specified walk-through path from a set of reference views V_r, $r = 1, \ldots, n$. We assume uncalibrated cameras, and, without loss of generality, set the camera matrix corresponding to the first camera to $\mathbf{P}_1 = [\mathbf{I} \mid \mathbf{0}]$. The camera matrices corresponding to the other reference views can then be chosen as $\mathbf{P}_r = [\mathbf{M}_r \mid \mathbf{e}_r]$ for $r = 2, \ldots, n$ [7]. Here $\mathbf{e}_r$ is the epipole in V_r with respect to the first camera and $\mathbf{M}_r = [\mathbf{e}_r]_\times \mathbf{F}_r$, where $\mathbf{F}_r$ is the fundamental matrix between the first and the r^{th} view. The epipolar geometry and the camera matrices are computed in the standard way using the 8-point algorithm from a small number of seed correspondences [7].

In the off-line stage we compute a panoramic representation for each planar patch in the scene. The panoramic representation is computed using a segmentation of jointly visible parts of each plane from pair-wise analysis of reference images. The jointly visible parts are determined using

1. Computing homographies through planar patches using a clustering technique, and

2. A subsequent segmentation of planar regions visible in two reference views.

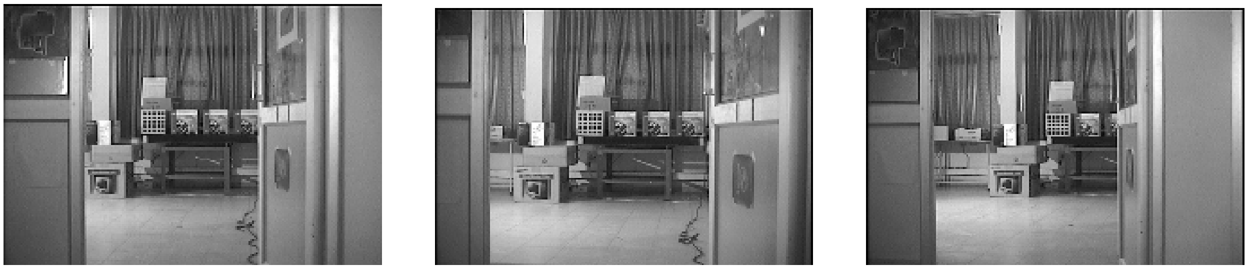
In the on-line stage of rendering of walk-throughs we perform a visibility analysis which based on the panoramic representations and the epipolar geometry.

It is often convenient to specify the path of the walk-through (the path followed by the camera centers of the novel views) in Euclidean terms. For this purpose, we assume that the camera centers of the reference views lie on a plane (a horizontal plane) and provide a 2D projective transformation between the projective representations of the camera centers and the rough Euclidean positions. The projective representations of the target novel views along the walk-through path are then approximated as convex combinations of the cameras corresponding to the reference views [14].

3. Determining homographies through planar patches using clustering

Images of world points lying on a plane are related to the corresponding image points in a second view by a 2D (planar) homography [5, 7]. Writing the first camera as $\mathbf{P} = [\mathbf{I} \mid \mathbf{0}]$ and a second camera as $\mathbf{P}' = [\mathbf{M} \mid \mathbf{e}']$, the homography induced by plane defined by $\pi^T\mathbf{X} = 0$ with $\pi = (\mathbf{v}^T, 1)^T$ is given as

$$\mathbf{H} = \mathbf{M} - \mathbf{e}'\mathbf{v}^T \qquad (1)$$



$(View1)$ $(View2)$ $(View3)$

Figure 1: Three reference views

and is completely determined by correspondences of four points on the plane. For the four point homography to be a plane induced homography the consistency condition $\mathbf{He} = \mathbf{e}'$ must be satisfied.

In what follows we briefly describe our algorithm for determining homographies through different planar patches using a clustering technique. We use the Harris corner detector [6] to detect corners in the two images. Suppose the corner detector returns a set S_m of m corners in the first image. From the above set we select a small subset S_n of n of reliable seed matches such that they are uniformly distributed over the images. For reference images obtained from widely disparate view points it may be necessary to hand pick these seed correspondences. We use RANSAC [7] on this small set of point correspondences to determine homographies through different planar patches as follows:

Algorithm:

1. Select 4 point correspondences from the set S_n of n correspondences randomly such that the image distances of the four points are less than a preset threshold (we give a chance for the four points to be chosen from the same region).

2. Compute the homography between the two images using these 4 points. Verify that $\mathbf{He} = \mathbf{e}'$ within a tolerance; otherwise select a different set of four points.

3. Transfer the set S_m of detected corners in the first image to the second using the homography computed in the above step. Determine a subset $S_k \subset S_m$ of size k corners which fall within a distance threshold of corners in the second image and also satisfy the epipolar constraint $\mathbf{x}'^T\mathbf{Fx} = 0$.

4. If the number k is greater than some threshold T, then re-compute the homography using all the k corners by least-squares (include the newly found correspondences). Otherwise, repeat the above steps with a new choice of the 4 points. If the number k remains less than T after N trials, use the largest consensus set to compute the homography. We also record the region of support of the homography in the two images as the convex cover of the set S_k of k points.

5. Remove the correspondences that have been accounted for from both sets S_m and S_n. If the remaining correspondences (out of n) are above a threshold (minimum number of correspondences required per plane), repeat the steps above to find another homography through a new planar patch.

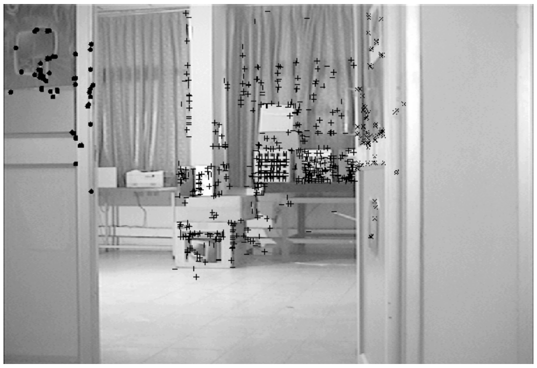


Figure 2: Corner matches between views 1 and 3 corresponding to the three planar patches. The matches corresponding to each patch are shown with different symbols.

Let the homographies found using the above procedure be $\mathbf{H}_i, i = 1, \ldots, l$ and their regions of support be S_i. Each of these homographies represent distinct planar patches in the scene. In Figure 1 we show three reference images used to test our novel view generation scheme. In Figure 2 we show the matches (out of S_m) in reference views 1 and 3 projected on view 3 through the homographies corresponding to three planes. We have cropped the floor because no reliable seed correspondences could be found in this region. The above algorithm correctly determines the homographies through the two doors and the back plane. The entire scene between the doors and above the floor get represented by a single homography because of their small relative depth separation compared to the distances from the cameras. In the next section we present our scheme for segmentation of the scene regions visible in both the reference views using the homographies computed above.

4. Segmentation of planar regions visible in two reference views

The segmentation algorithm is described as follows:
Algorithm:

For each of the homographies $\mathbf{H}_i, i = 1, \ldots, l$ computed above do the following:

1. Select in the first image a region somewhat larger than its region of support S_i computed above. In our experiments we have enlarged each region S_i by 10 pixels in all directions.

2. Warp the intensity information of the enlarged regions (selected above) towards the second image using $\mathbf{H}_i$ and compute a difference (color) image. Segmentation of the dark regions in the difference image gives the final region of support for the homography in both views. We obtain the final segmentation by region-growing. In cases where the occluding boundaries are known to be straight lines, the region boundaries are refined by edge detection.

In Figure 3 we show the difference images computed after warping the enlarged regions of support for each homography in the first image towards the third. These are the common regions visible in both the reference views.

We do the above analysis pair-wise between all reference views. Finally for each planar patch visible from at least two views, we create a panoramic representation [15] P_i by registering the intensity images using the homographies found by each pair-wise analysis. We create the representation on the image plane in which it occupies the largest area. Each panoramic representation P_i represents the union of all regions of a planar patch that are visible from at least two reference views. Note that this panoramic representation is a conceptual device and need not be explicitly rendered. It is merely a data structure threading the pair-wise common segments corresponding to a planar patch through the corresponding homographies. In Figure 4 we show the panoramic representation of the back plane created from all the reference views (3 of which are shown above) for illustration.

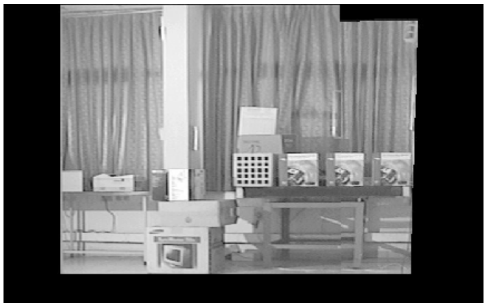


Figure 4: The back plane panorama

5. Visibility analysis for rendering of a novel view

In order to render a novel view with a known camera matrix, we first need to establish the homographies between the novel view image plane and the panoramic representations P_i of each planar patch. Since all camera matrices and the pair-wise epipolar geometries are known, these homographies can be computed in any of the two ways.

1. For each planar patch P_i compute the explicit representation of the plane π from the reference views using Eqn. 1. Once π is known, use the epipolar geometry between the patch P_i and the target view to compute the homography using Eqn. 1 again.

2. Transfer at least four points (usually several more) for every planar patch from two reference views to the target view using the *trifocal tensor* [7], and use least-squares to compute the homography between the panoramic representation and the target view image plane.

Both the methods give good results in practice. Let the homography from the novel view to the i^{th} panoramic patch be $\mathbf{H}_i$.

We also create a representation of all detected planar patches on any one of the reference images, say V_p, and estimate the homographies induced by each planar patch between V_p and the target view. Let these homographies be $\mathbf{G}_i$. We also estimate the epipolar geometry between V_p and the target view. The rendering algorithm can then be described as follows:
Algorithm:

1. For each pixel $\mathbf{x}$ in the target image compute the transfer $\mathbf{H}_i\mathbf{x}$ to the panoramic representation of each planar patch using the respective homographies $\mathbf{H}_i$ and determine whether the transferred point lies within the panoramic image segment.

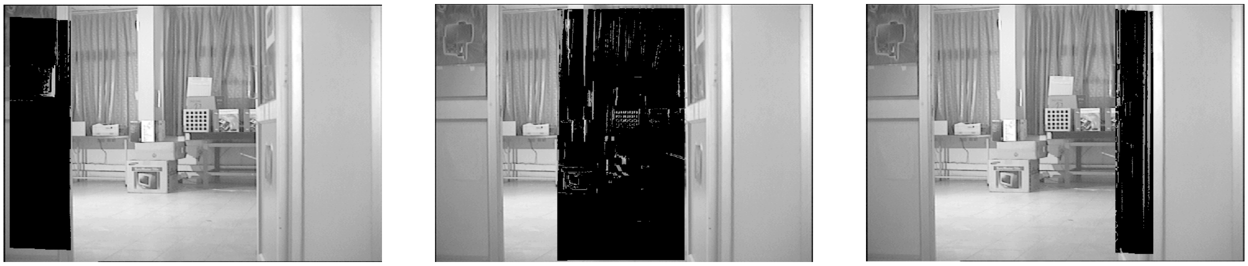


Figure 3: The joint view segmentations projected in view 3

2. If $\mathbf{H}_i\mathbf{x}$ lies within the corresponding panoramic image segment for only one value of i, then transfer the color information from $\mathbf{H}_i\mathbf{x}$ to $\mathbf{x}$. In such a case there is no ambiguity and the ray back-projected through $\mathbf{x}$ in the target view intersects only one finite plane of common visibility.

3. Suppose $\mathbf{H}_i\mathbf{x}$ lies within the corresponding panoramic image segment for more than one value of i. This indicates that the ray back-projected through $\mathbf{x}$ intersects more than one plane in the visible domain (see Figure 5). Then, for each of these planar patches, transfer $\mathbf{x}$ to the reference image V_p using the homographies $\mathbf{G}_i$ corresponding to these patches.

4. Now, since all these points on V_p are corresponding points of $\mathbf{x}$ through different planar homographies $\mathbf{G}_i$, they must lie on an epipolar line in V_p. Clearly, the correspondence closest to the epipole occludes all others (Figure 5). Determine the plane for this correspondence and transfer the intensity information from the corresponding position of the panoramic representation of this planar patch on to $\mathbf{x}$. Note that even when the epipole is a point at infinity, the relative ordering for occlusion holds.

In case $\mathbf{x}$ doesn't correspond to any pixel in any of the panoramic representations of the planar patches there will be a *hole* at $\mathbf{x}$. Note that this would indicate one or both of the following cases:

1. The assumption that every region in the target image is visible from at least two reference views is falsified, or,

2. The segmentation of jointly visible regions (described above) is incorrect.

In either of the above cases manual intervention will be necessary to modify the segmentation.

Note that the rendering algorithm explicitly accounts for resolution of occlusions, and no re-painting or z-buffering is required. Consequently, there can be no *folds*.

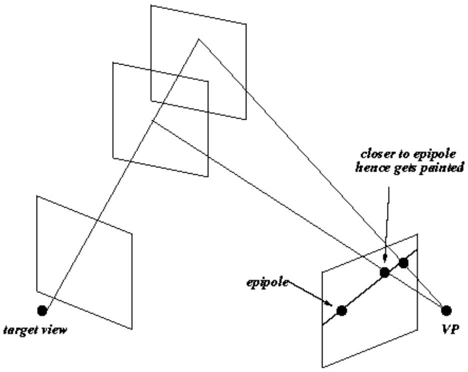


Figure 5: Occlusion handling

6. Results

In Figure 6 we show twelve views of the results of generation of a walk-through along a zig-zag path towards the door (please see the movie at http://www.cse.iitd.ernet.in/∼suban/demo/labscene.mpg). In this example, no manual intervention other that hand picking of approximately 25 initial correspondences was necessary. In Figure 7 we show the rendering of a sequence of novel views with the two end views as reference images (these reference images have been obtained from the web-site http://www.inrialpes.fr/movi/people/Lhuillie/demo3.html; see also [9]).

7. Conclusion

We have engineered a complete system for image based rendering of scenes with planar patches starting from a small set of point correspondences using standard and well tested techniques from computer vision. We exploit ”spatial coherence” where a scene in the reference image(s) is subdivided into planar patches that are mapped onto the target image through homographies. The main features of the sys-

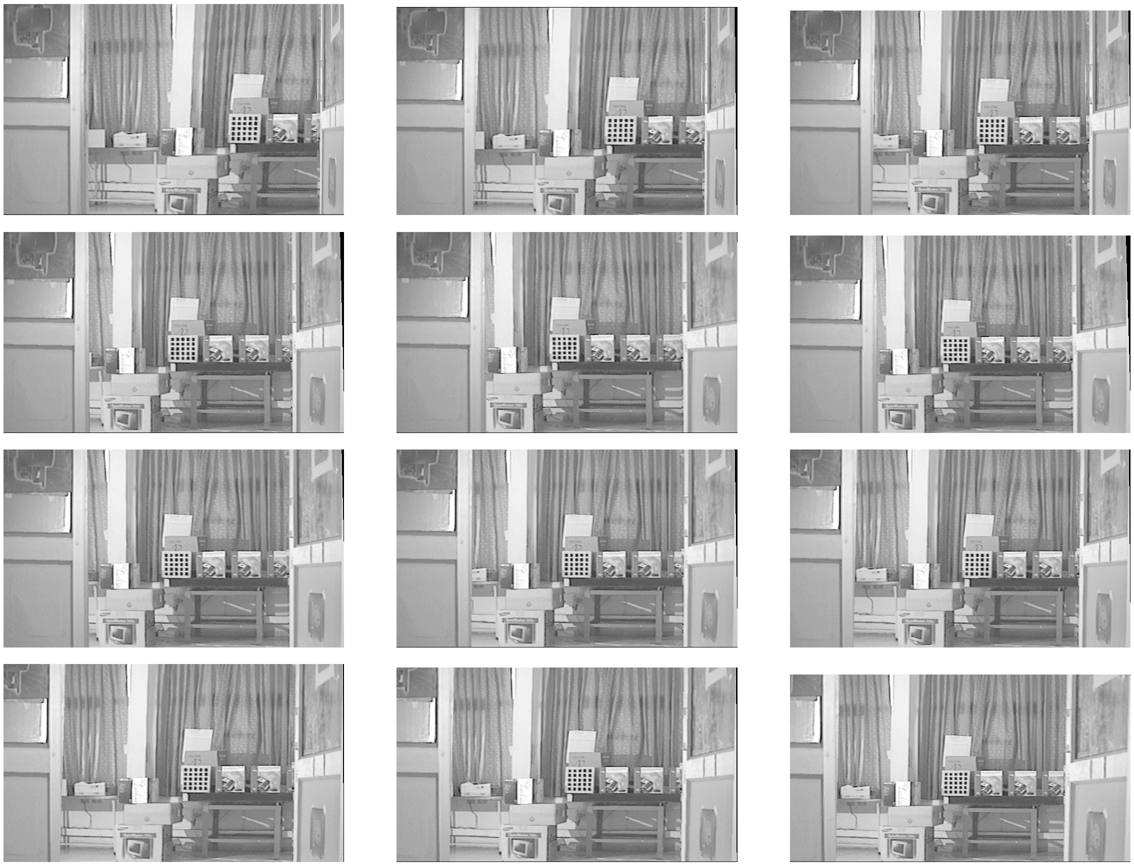


Figure 6: Some views of the rendered walk-through sequence

tem are i) only a small set of initial correspondences are required, ii) visibility is resolved explicitly, and no re-drawing or z-buffering is necessary, and iii) the method is largely automated, though it may require some manual intervention during the off-line segmentation stage. The method can easily be extended to deal with panoramic reference views [15]. The results demonstrate the robustness of the method.

References

[1] *Image-Based Modeling, Rendering, and Lighting,* SIGGRAPH Course 39, *SIGGRAPH*, 1999.

[2] E. H. Adelson and J. R. Bergen, "The Plenoptic function and the Elements of Early Vision," *Computational Models of Visual Processing,* Chapter 1, Edited by Michael Landy and J. Anthony Movshon. The MIT Press, Cambridge, Mass. 1991.

[3] S. E. Chen and L. Williams, "View Interpolation for Image Synthesis," *SIGGRAPH*, 1994.

[4] P. E. Debevec, C. J. Taylor and J. Malik, "Modeling and Rendering Architecture from Photographs," *SIGGRAPH*, 1996.

[5] O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*, The MIT Press, 1996.

[6] C. Harris and M. Stevens, "A Combined Corner and Edge Detector," *Proc.* 4^{th} *Alvey Vision Conference*, pp. 153-158, 1988.

[7] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.

[8] S. Laveau and O. D. Faugeras, "3-D Scene Representation as a Collection of Images and Fundamental Matrices," INRIA, Technical Report No. 2205, 1994.

[9] M. Lhuillier and L. Quan, "Image Interpolation by Joint View Triangulation," *CVPR*, 1999.

[10] L. McMillan, *An Image-Based Approach to Three Dimensional Computer Graphics*, Ph.D thesis, University of North Carolina at Chapel Hill, 1997 (Technical Report TR97-013).

[11] L. McMillan, "Computing Visibility Without Depth," *UNC Technical Report*, TR95-047, University of North Carolina, 1995.

[12] L. McMillan and G. Bishop, “Plenoptic Modeling: An Image Based rendering System,” *SIGGRAPH*, 1995.

[13] L. McMillan and G. Bishop, “Shape as a Perturbation to Projective Mapping,” *UNC Technical Report*, TR95-046, University of North Carolina, 1995.

[14] S. M. Seitz and C. R. Dyer, “View Morphing,” *SIGGRAPH*, 1996.

[15] R. Szeliski, “Image Mosaicing for Tele-Reality Applications,” *DEC and Cambridge Research Lab. Technical Report,*, CRL 94/2, 1994.

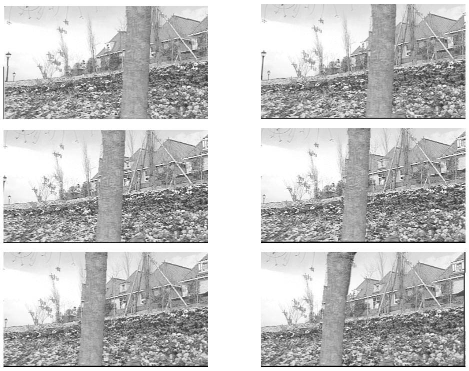


Figure 7: Rendering of an outdoor scene from the two end reference views