Image-Based Scene Rendering and Manipulation Research at the University of Wisconsin

Charles R. Dyer[*]
Department of Computer Science
University of Wisconsin
Madison, WI 53706
E-MAIL: dyer@cs.wisc.edu
HOMEPAGE: http://www.cs.wisc.edu/~dyer

Abstract

This report summarizes the research effort at the University of Wisconsin in support of the VSAM Program. Our primary goal is to develop technologies so a user can interactively visualize and virtually modify a 3D environment from a set of images. Current approaches are described for image-based scene rendering, scene manipulation, and appearance modeling.

1 Introduction

The ultimate goal of this project is to develop image-based methods that will enable a user to control the motion of a virtual camera so as to visualize a real 3D environment for applications such as facility monitoring and mission rehearsal. This technology will enable the rapid creation of an interactive visualization capability in which new views are synthesized by adaptively combining or "steering" a set of input images of the environment.

Our approach is image-based in the sense that all input and output about the scene is via images. Since the desired results are images this means that techniques should focus on producing realistic images, not feature space descriptions for classification, 3D model building, or other traditional computer vision goals.

The major challenge of this formulation is to create ways of combining a set of images, obtained from a fixed set of viewpoints, so that a user can interactively survey the real environment by controlling a virtual camera. To create a compelling sense of visual presence, the following user capabilities are key: (1) can interactively change viewpoint with respect to the 3D environment, (2) can render the environment photorealistically at high resolution, frame rate and quantization rate, and (3) can effect virtual changes in the acquired environment.

Our research activities are directed towards accomplishing these three capabilities of interactive virtual camera control, photorealistic rendering, and virtual scene modification operations. This report summarizes current activities related to these issues, emphasizing two new approaches to view synthesis. The first is called **view morphing** and treats primarily the two-input-view case. That is, given a pair of images of a static 3D scene, interpolate in-between views. The second approach treats the general case of synthesizing views over a wide range given an arbitrary number of input views, widely distributed around the environment. This **voxel coloring** approach reconstructs a photometrically-consistent volumetric representation of the scene, which is then used to render new views. The construction of this representation also allows the user to interactively modify the scene through image editing operations, and this capability, called **plenoptic image editing** (joint work with the University of Rochester), is also briefly summarized.

[*] The support of the Defense Advanced Research Projects Agency, and the National Science Foundation under Grant No. IRI-9530985 is gratefully acknowledged.

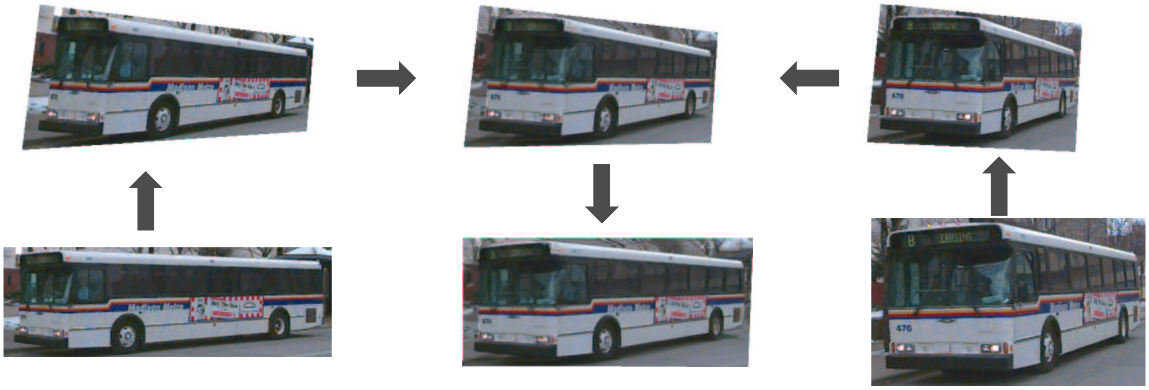

Figure 1: View morphing in three steps. Two original images (bottom left and right) of a bus are first prewarped (top left and right) to make the cameras parallel, and then morphed to create an in-between view (top middle). The desired gaze direction of the morphed view is established with a postwarp operation (bottom middle).

2 View Synthesis from Two Views

Recently, a number of methods have been developed by researchers in computer vision and computer graphics for synthesizing new views from one or more images. Our approach, called **view morphing** [Seitz and Dyer, 1996a, Seitz and Dyer, 1996b, Seitz, 1997, Seitz and Dyer, 1997c], builds on existing image morphing methods for creating compelling image sequences that smoothly transform one image into another. The method synthesizes images corresponding to new viewpoints by performing simple image warping operations. This work extends to perspective projection our earlier results [Seitz and Dyer, 1995].

The fundamental research questions associated with this area are, *when* and *how* can physically-correct new perspective views of a 3D scene be predicted from a set of basis views? With respect to the first question we have shown that when two basis images have the same scene points visible in each, a constraint we call *monotonicity*, this is sufficient to uniquely predict the appearance of the scene for all in-between viewpoints on the line segment connecting the input cameras' optical centers. This is true despite the fact that there may not be sufficient information in the pair of images to uniquely reconstruct the 3D scene. In other words, while any number of distinct scenes could have produced the given input images, the monotonicity constraint guarantees that each of those scenes will project to the same set of in-between images.

To answer the "how" question we have developed extensions to image morphing that correctly handle 3D projective camera and scene transformations. View morphing works by prewarping two input images, computing an image-morph (image warp and cross-dissolve) between the prewarped images, and then postwarping each in-between image produced by the morph. The prewarping step corresponds to changing the orientations of the two input views, but not the camera positions. The images are projected onto a common image plane that is parallel to the line between the two cameras' optical centers. From this special parallel camera configuration, we have shown that image morphing (i.e., linear image interpolation) produces physically-correct perspective views. These views correspond to positioning a virtual camera on the line between the original cameras, and orienting the camera parallel to the prewarped views. The postwarping step warps the morphed image so as to change the orientation of the virtual camera. This sequence of steps is illustrated in Figure 1.

The major contributions to view synthesis that are achieved by view morphing include:

- Ability to synthesize image sequences corresponding to linear camera motion between two basis images' known or unknown camera positions (with the orientation of the camera along that path specifiable by the user)

- Can compute smooth transitions between any two images, regardless of source or content, producing simultaneous transitions in viewpoint, shape and color; consequently, the approach can perform both rigid and non-rigid transformations, and use a variety of types of input from photographs to drawings

- Does not require knowledge of 3D shape nor does it need calibrated cameras

- When a generic visibility assumption holds, which we call monotonicity, view morphing guarantees a unique, physically-correct solution for all viewpoints on the line between the optical centers of two input cameras

- When visibility changes occur between basis views, the monotonicity assumption is violated, but image quality degrades only locally and can be minimized by using different feature correspondences

- When a stronger version of monotonicity holds for a set of basis views, new views can be synthesized for all viewpoints within the convex hull of the input cameras' optical centers

- Efficient implementation of the algorithm is possible because many steps are 1D scanline operations

2.1 Evaluation Plan

Research results related to view morphing will be demonstrated and evaluated by both theoretical analysis and experimental testing of prototype systems. With respect to theoretical properties of interest, view morphing is currently limited in that it cannot directly cope with significant changes in visibility, is difficult to use for synthesizing a large range of views of a scene from many basis images, and can require solving the correspondence problem for views that are far apart. These issues will be investigated further. In addition, we intend to continue the development of our view morphing system implementation in order to decrease the processing time, require less user interaction, and improve the realism of the synthesized views. Experimental evaluation using VSAM-related data sets will be performed.

3 View Synthesis from Many Views

To synthesize new views from arbitrary camera viewpoints given a set of basis images is a difficult unsolved problem. One important requirement is the ability to integrate information from images containing significant differences in the parts of the scene that are visible. Second, since the desired results are photorealistic new views, methods must be "dense" so as to render images containing accurate texture and color information at every pixel, not a sparse set of feature descriptions. A third requirement is scalability—the capability for combining an arbitrary number of basis views, with corresponding improvements in the quality of the synthesized views.

With these requirements in mind, we are developing a new approach, called **voxel coloring** [Seitz and Dyer, 1997a, Seitz and Dyer, 1997b], that reconstructs the "color" (radiance) at points in an unknown scene. In our initial study we assume a static scene containing Lambertian surfaces under fixed illumination so the radiance from a scene point can be described simply by a scalar value, which we call *color*.

Coping with images with large changes in visibility means we must solve a difficult correspondence problem between images that are very different in appearance. Rather than use traditional approaches such as stereo, we use a scene-based approach. That is, we discretize 3D scene space into a set of voxels that are traversed and colored in a special order. The advantage of this is that simple voxel projection determines corresponding image pixels. The main disadvantage is that this requires precise camera calibration to achieve the necessary accuracy.

We have shown that certain voxels have an invariant color, constant across all possible interpretations of the scene that are consistent with the basis images. This leads to a volu-

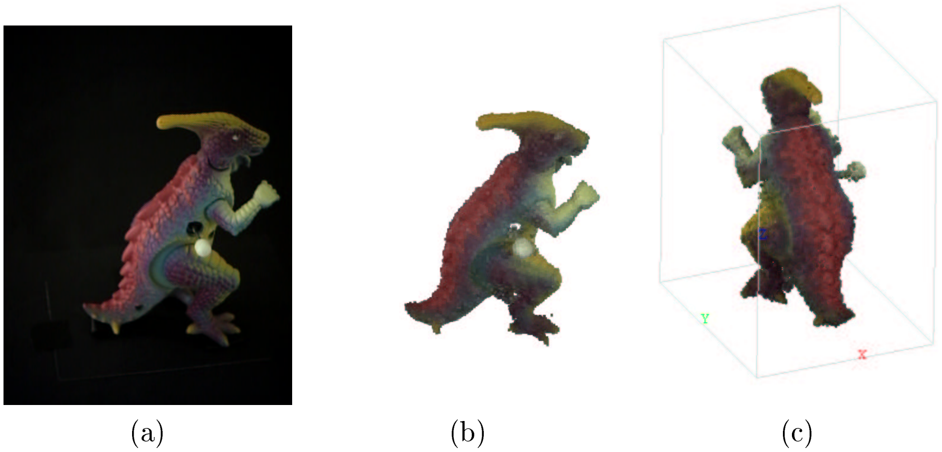

Figure 2: Reconstruction of a dinosaur toy. (a) One of 21 original images taken from slightly above the toy while it was rotated 360°. (b-c) Two views rendered from the reconstruction.

metric voxel coloring algorithm that labels the color-invariant scene voxels based on their projected correlation with the input images. Correlation consistency will work only if we can determine when a voxel in fact corresponds to the projected pixel in an image, or whether the pixel corresponds to a different (closer) scene point, which occludes the current voxel. To solve this visibility problem, we introduce a geometric constraint on the input camera positions that enables a single visibility ordering of the voxels to hold for every input viewpoint. This is a relatively weak constraint in that it allows significant freedom in the placement of the input cameras, but it enables the visibility problem to be solved by simply traversing and labeling the voxels in increasing distance from the input cameras. Furthermore, the method is independent of scene complexity.

Putting this all together, the voxel coloring algorithm works as follows. The scene is initialized to a volume of voxels. These voxels are traversed layer-by-layer, where a layer contains all voxels that are equidistant from the cameras' convex hull. The layer closest to the cameras is visited first, and so on until the layer of voxels that is farthest from the cameras is considered. A voxel is processed by projecting it into each basis image and determining how well its corresponding image pixels' colors are correlated. If the correlation is above a threshold, the voxel is added to the reconstructed shape and labeled with the color of its pixels.

The final result is a dense, volumetric reconstruction, with associated color information, of scene surface points that is guaranteed to be consistent with all the basis images, regardless of visibility changes and scene concavities. Using this reconstruction, the scene can be rendered from any view by projecting the voxels in the desired direction. Figure 2 shows two views rendered using the reconstruction produced by a set of 21 input images.

The major contributions to view synthesis that are achieved by voxel coloring include:

- Reconstructs a dense description of scene surface points and associated color (radiance) values, which can be used for rendering arbitrary new views

- Uses color invariance to ensure that all voxels reconstructed are consistent with all of the basis images

- Allows input views containing large visibility differences by operating in scene (voxel) space and using a weak camera position constraint

- Permits widely separated input camera positions

- Scales up directly to an arbitrary number of basis views, with processing time linear in the number of input images

The reconstruction produced by voxel coloring can also be used for virtually *modifying* the reconstructed scene by image editing operations such as image painting, scissoring, and morphing. We call this **plenoptic image editing** [Seitz and Kutulakos, 1997] because the user can edit any one image and those changes are propagated automatically, in a physically-consistent way, to all other images as if the 3D environment had itself been modified. This allows a user to visualize how edits to an object via one image will affect the object's appearance from other viewpoints. The key component in realizing these operations is the reconstruction produced by voxel coloring. While preliminary results are encouraging, there are many open research problems that need to be addressed to make this approach more effective.

3.1 Evaluation Plan

Research results related to voxel coloring will be demonstrated and evaluated by both theoretical analysis and experimental testing of prototype systems. Improved methods are needed for handling large numbers of images, for example from a video stream, which are uncalibrated. Extensions are also needed to handle non-Lambertian scenes and dynamic scenes. The plenoptic image editing framework needs to be explored further to determine the types of scene modification operations that would be useful for VSAM applications. We plan to continue developing our system implementation so that the photorealistic quality, processing speed, scalability to large numbers of basis views, and large range of feasible output views are experimentally demonstrable. Also, can the method produce smooth and natural scene visualizations corresponding to a moving camera? Synthesized view quality assessment will be performed if data sets with ground truth are available.

References

[Seitz and Dyer, 1995] S. M. Seitz and C. R. Dyer. Physically-valid view synthesis by image interpolation. In *Proc. IEEE Workshop on Representations of Visual Scenes*, pages 18–25, 1995.

[Seitz and Dyer, 1996a] S. M. Seitz and C. R. Dyer. Toward image-based scene representation using view morphing. In *Proc. 13th Int. Conf. on Pattern Recognition, Vol. I*, pages 84–89, 1996.

[Seitz and Dyer, 1996b] S. M. Seitz and C. R. Dyer. View morphing. In *Proc. SIGGRAPH 96*, pages 21–30, 1996.

[Seitz and Dyer, 1997a] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. In *Proc. Computer Vision and Pattern Recognition Conf.*, 1997. To appear.

[Seitz and Dyer, 1997b] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. In these Proceedings.

[Seitz and Dyer, 1997c] S. M. Seitz and C. R. Dyer. View Morphing: Uniquely predicting scene appearance from basis images. In these Proceedings.

[Seitz and Kutulakos, 1997] S. M. Seitz and K. N. Kutulakos. Plenoptic image editing. Technical Report 647, Computer Science Department, University of Rochester, Rochester, NY, January 1997.

[Seitz, 1997] S. M. Seitz. Bringing photographs to life with view morphing. In *Proc. Imagina 97*, pages 153–158, 1997.