OBJECT MODELING USING SPACE CARVING

Elsayed E. Hemayed and Aly A. Farag

Computer Vision and Image Processing Lab
University of Louisville
Louisville, KY 40292 USA
E-mail: (sayed, farag)@cairo.spd.louisville.edu
URL: http://www.cvip.uofl.edu

ABSTRACT

In this paper we consider the problem of computing the 3D shape of an unknown, arbitrary-shaped object. Starting with an initial surface, larger than the scene, we refine it using object's projections taken at known but arbitrarily distributed viewpoints. These projections are used independently in a recursive manner. Thus, using more projections improves the reconstruction with a slight increase in the computation time. The technique generates an approximate voxelized representation of the object. In this paper, we describe the proposed technique and outline the algorithm. Rendered images of voxel spaces recovered from synthetic and real observation images are shown.

1. INTRODUCTION

We present a new algorithm for volumetric object reconstruction. Specifically, given a set of images of an object taken from arbitrary but known locations, the algorithm builds a 3D model of the object that is consistent with the input images. Like earlier solutions to the same problem, Voxel Coloring [1] and Space Carving [2], our algorithm uses voxels to model the object. Although Voxel Coloring and Space Carving are particularly successful solutions to the object reconstruction problem, our algorithm has advantages over them. Both Voxel Coloring and Space Carving techniques use the entire set of images, from which the voxel is visible, simultaneously to impose the color consistency constraint. Thus, using a long sequence of images can decrease the performance of the algorithm in terms of memory requirement and computation time. Therefore, the length of the sequence is limited by the hardware. At the same time, a long sequence of images with some light variations between images can distort the reconstructed model.

On the other hand, our algorithm uses the images independently (one at a time) in a recursive manner (one after the other). Using a long sequence of images improves the performance of the algorithm with slight increase in the computation time and without requiring additional memory. It does not require the entire set of images to be available before operation but it continues refinement whenever images are available. This enables the reconstruction system to pipeline the data acquisition and the computation processes to speed up the reconstruction process.

In this paper, we first describe earlier solutions to the object reconstruction problem. We compare the merits of those solutions to our own. Next, we discuss the Modified Space Carving technique and describe experiments on real and synthetic image sequences. Computation time and memory requirement are discussed as well.

2. RELATED WORK

Stereo techniques [3] find points in two or more input images that correspond to the same point in the scene (the correspondence problem). Then the depth of the scene point is determined using knowledge of the camera locations and triangulation (the depth estimation problem). Unfortunately, stereo is difficult to apply to images taken from arbitrary viewpoints. This is a two-sided problem. If the input viewpoints are far apart, then corresponding image points are hard to find automatically. On the other hand, if the viewpoints are close together, then small measurement errors result in large errors in the calculated depths. Furthermore, stereo produces a 2D depth map and integrating many such maps into a true 3D model is a challenging problem [4].
Voxel Coloring [1], Space Carving [2] and Generalized Voxel Coloring (GVC) [5] exploit the fact that points on Lambertian surfaces are color-consistent (they have similar colors in all the images from which they are visible.) These methods, including ours, start with an arbitrary number of calibrated images of the scene and a set of voxels that is a superset of the scene. Each voxel is projected into the images from which it is visible. If the voxel projects onto inconsistent colors in several images, it is considered not to be on the surface and, so, it is carved, i.e., declared to be transparent. Otherwise, the voxel is colored, i.e., declared to be opaque and is assigned the color of its projections. These algorithms stop when all the opaque voxels project into consistent colors in the images.

While Space Carving never carves voxels it shouldn’t, it is likely to produce a model that includes some color-inconsistent voxels. This results due to the fact that the color consistency of a voxel is not checked over the entire set of images from which it is visible. This problem is solved in GVC because every voxel in the final constructed model is guaranteed to be color consistent over the entire set of images. However, if the lighting condition of the scene varies from one view to another, then the color consistency check fails in both techniques.

In much the same spirit, recent work on voxelized reconstruction by Bonet and Viola [6], Saito and Kanade [7] and Cross et al. [8]. Modified Space Carving is an attempt to release some of the limitations of all the above approaches. In particular, our modified Space Carving technique does not rely on the color content of the images. Therefore, the Lambertian surface assumption is not needed and the lighting condition does not affect the performance of the algorithm. Second, it uses the images independently and in a recursive manner. Therefore, the algorithm does not consume large memory and can be efficiently applied to a large sequence of images. Third, it does not attempt to solve the correspondence problem. Therefore, it can handle arbitrary and widely dispersed image viewpoints.

3. MODIFIED SPACE CARVING

Three-dimensional (3D) object reconstruction is the reverse process of 2D object’s projection (image formation). This is not a one-to-one relationship since one can find different object representations that generate the same 2D projection. For example, the projections of a cube and a cylinder with similar heights and widths look the same from a viewpoint perpendicular to their medial axes. This ambiguity can be minimized by using different projections of the object in the reconstruction process.

The relationship between 3D object reconstruction and 2D object projection can be represented mathematically as follows: Let $L$ be a 3D model of the actual object. The 2D projections of this object are represented by $l_i$, where $i$ denotes the viewpoint index. The projection process is denoted by the function $\text{project}(\cdot)$ in Eq. 1, where $V_i$ is the camera parameters at viewpoint $i$.

$$l_i = \text{project}(L, V_i) \quad (1)$$

The reconstruction process is denoted by the function $\text{reconstruct}(\cdot)$ in Eq. 2, where $\bar{L}$ is an approximation of the model $L$. Theoretically, $L$ converges to $\bar{L}$ as $N$ reaches to $\infty$.

$$\bar{L} = \text{reconstruct}(l_1, l_2, l_3, \ldots l_N) \quad (2)$$

The complexity of Eq. 2 is simplified by a recursive formula as follows: First select a 3D model, $K_0$, such that $K_0 \supset L$. Second define a function $\text{refine}(\cdot)$ in Eq. 3. This function uses the different 2D projections to recursively refine the model $K_i$ in order to converge to $L$.

$$K_{i+1} = \text{refine}(K_i, l_i) \quad (3)$$

In the proposed algorithm, we select $K_0$ to be a box surrounding the object, i.e., a bounding volume. In order to easily refine the subsequent model $K_i$, we represent it as a set of voxels. Each voxel is represented by its center of mass and assigned a flag to indicate if it exists (opaque) or not (transparent). All voxels have the same height, width and depth. The refinement process is applied to the model $K_i$ whenever there is a new 2D projection $l_i$ for the object to be reconstructed. The idea of the refinement process is to eliminate (carve) the voxels of $K_i$ that violate the projection $l_i$. Therefore, we start each refinement iteration by segmenting the object from the background. Then, we project each opaque voxels in $K_i$, using the camera parameters $V_i$, to the projection $l_i$. We carve all voxels that project to the background and keep the others. In general, $K_i \rightarrow L$ as $i \rightarrow \infty$. However, a good approximation of $L$ can be obtained after few number of iterations. Having more projections improves the final reconstruction.

3.1. System Description

The accuracy of the reconstructed models relies on the number and location of object projections used in the reconstruction/refinement process. Since it is difficult and expensive to set a large number of cameras around an object to be reconstructed, other design schemes are considered. The idea is to create a system that is
capable of getting projections of the object from different viewpoints. At the same time, the system should keep track of the camera parameters at each projection. One approach is to have a single camera mounted to a calibrated robotic arm that can rotate around the stationary object in order to get projections from different viewpoints. The other approach is to have a single stationary camera and a moving object. For simplicity, we use the second approach where the camera does not change its location while the object does. We put the object on a calibrated rotating table. In order to create different viewpoints, we change the rotation angle of the table, see Fig. 1. The main advantage of this system is its simplicity; it is also inexpensive. The pitfall of this system is the possibility of losing some details in complex objects. A solution for this problem is to add another camera to the system and direct it to where the missing details might be. As Fig. 1 shows, a single stationary camera captures an image of the object at each rotation angle $\theta_t$. A segmentation process is applied to the image to separate the object from the background. A stationary homogeneous background can be used to automatically segment the image. At each angle $\theta_t$, we update the camera parameters to simulate a virtual camera with the same intrinsic parameters as the actual one, but its optical axis makes an angle $\theta_t$ with the actual optical axis. The actual camera is calibrated using Robert calibration technique [9] assuming a pinhole camera model. In this case, a point $\mathbf{M}$ in the 3D model is projected to a point $\mathbf{m}$ in the 2D projection using the following equation:

$$
\mathbf{m} = \mathbf{P} \mathbf{M} = \mathbf{A} \mathbf{D} \mathbf{M}
$$

where $\mathbf{P}$, a 3x4 matrix, is the perspective projection matrix, $\mathbf{A}$ is a 3x4 matrix that contains the intrinsic parameters and $\mathbf{D}$ is a 4x4 matrix that contains the extrinsic parameters, rotation and translation. Since the camera intrinsic parameters do not change from one view to another then the matrix $\mathbf{A}$ remains the same with any angle $\theta_t$. Matrix $\mathbf{D}$ is modified using the following equation:

$$
\mathbf{D}_i = \mathbf{T}(\mathbf{t}) \mathbf{D} \mathbf{R}(-\theta_i) \mathbf{T}(-\mathbf{t})
$$

where $\mathbf{D}_i$ is the extrinsic parameters matrix at rotation angle $\theta_i$, $\mathbf{T}(\mathbf{t})$ is a 4x4 translation matrix and $\mathbf{R}(-\theta_i)$ is a 4x4 rotation matrix. To simplify the updating process, we choose the rotation axis, $\mathbf{t}$, to be parallel to one of the primary axes (e.g., the Z-axis). Therefore, the rotation is performed around the Z-axis. The new perspective projection is given by the following equation:

$$
\mathbf{P}_i = \mathbf{A} \mathbf{T}(\mathbf{t}) \mathbf{D} \mathbf{R}(-\theta_i) \mathbf{T}(-\mathbf{t})
$$

where $\mathbf{P}_i$ is the perspective projection matrix at rotation angle $\theta_i$. Using $\mathbf{P}_i$ and Eq. 4, all opaque voxels in the current 3D model are projected into the segmented image taken at angle $\theta_t$. A voxel is carved if it projects to the background. Other voxels that project to the object region are kept for further refinement. The process of space carving is repeatedly applied to successively improve the approximations of the reconstruction.

4. RESULTS

Performance of the proposed algorithm was examined on a variety of real and synthetic data. In the first experiment, a set of synthetic images were generated using 3D Studio, a commercial graphics package. Thirty six 320x240 images were generated of a scene containing a duck. These images were generated at 10° rotation angle increments in order to get a complete view of the object. The initial volume contains 35x35x35 voxels and the final volume has 3923 voxels. Due to the space limit of the paper, we show only the results of the real data.

In the second experiment, eight images of the Egyptian Writer Statue were acquired at 45° rotation angle increments. The initial volume contains 70x70x70 voxels and the final volume has 6169 voxels. The results are displayed in Fig. 2.

In the third experiment, 36 images of the Barney toy were acquired at 10° rotation angle increments. The initial volume contains 70x70x70 voxels and the final has 11608 voxels. The results are displayed in Fig. 3.

Since the Modified Space Carving technique uses the object projections independently and in a recursive manner, the memory requirement does not depend on the number of projections. It depends mainly on
the size of the volume being refined. For example, a 100x100x100 volume size requires a memory of 12 MByte to store the center coordinate \((X,Y,Z)\) of each voxel as a 4-byte floating number. This figure represents the initial and maximum memory requirements for such high resolution. It will decrease with further refinements of the volume since more voxels will be carved away. Similarly, the computation time depends mainly on the volume size. It slightly increases when using longer sequence of images. For example, the computation time (running on an SGI - INDIGO2 machine with MIPS R4400 and 128 MB Memory) of our first experiment (35x35x35 volume size and 36 images) was 5 seconds. In experiment 2 (70x70x70 volume size and 8 images) the time was 8 seconds. In experiment 3 (70x70x70 volume size and 36 images) the time was 15 seconds. These figures of computation time are much lower than Kutulakos and Seitz Space Carving technique [2].

The Modified Space Carving is capable of generating a quick good approximate representation of objects without consuming large space of memory. At the same time, it can generate a highly accurate model by using a higher resolution volume and a long sequence of images in a reasonable amount of time with moderate machine resources.

5. REFERENCES


