

# An Active Volumetric Model for 3D Reconstruction

Xin Liu, Hongxun Yao, Xilin Chen and Wen Gao

School of Computer Science and Technology

Harbin Institute of Technology

Harbin, China

xin\_liu@hit.edu.cn, hxy@vilab.hit.edu.cn, xlchen@ieee.org, wgao@jdl.ac.cn

**Abstract**—In this paper, we present an active volumetric model (AVM) for 3D reconstruction from multiple calibrated images of a scene. The AVM is a physically motivated 3D deformable model which shrinks actively under the influence of multiple simulated forces towards the real scene by throwing away some of its voxels. It provides a computational framework to integrate several constraints in an intelligible way. In the current work, we use three forces derived respectively from the smooth constraint, the compulsory silhouette constraint, and the color consistency constraint. Based on the composition of the 3 forces, our algorithm can significantly restrain holes and floating voxels, which plague voxel coloring algorithms, and produce precise and smooth models. We test our algorithm by experiments based on both synthetic and real data.

**Keywords**—active volumetric model; 3D reconstruction; voxel coloring; snake

## I. INTRODUCTION

Based on color consistency, the voxel coloring algorithm [1–4] can build 3D models from a number of calibrated images with the complex effect of occlusion, parallax, and shading. Since it was first introduced by Seitz [1], the algorithm has attracted the attention of many researchers, and many variations of it have been proposed. Dyer [5] and Slabaugh et al. [6] each provided comprehensive reviews of efforts in this area.

The rationale of the voxel coloring algorithm is very simple. First an initial volume is discretized into small cubes (the voxels). Then those voxels project into significantly variant colors in all images from which they are visible are eliminated. The final result is a solid that can faithfully reproduce the input images. The voxel coloring algorithm, which takes advantage of nearly all information contained in the input images [2], does provide a good solution for the 3D reconstruction problem, but they have at least two inherent drawbacks. First, even enough distinguishable texture exists in input images, some voxels may project into color-consistent pixels by chance, which cause the “floating voxels” in the reconstructed model. Secondly, the color-consistency test can hardly be solved perfectly. Some of the reasons are: quantization error and sensor noise always exist; an image pixel may contain the mixed color of multiple voxels; the computation of the exact projective pixels in each view is impractical and an approximation is often used; the BRDF can

not be modeled easily and a simple assumption is often used. With the dubious measurement of voxel colors and the imperfect BRDF, algorithms have to decide a compromise threshold. A more lenient threshold tends to preserve outliers and fatten the result. A stricter threshold tends to carve voxels excessively. Worse, the wrongly carved voxels may cause the algorithm to draw incorrect conclusions about which images see the remaining uncarved voxels, leading to more incorrect carving and “holes” in the model.

To overcome these problems many approaches have been proposed. G. Slabaugh et al. [7] formulated the voxel reconstruction problem as an energy minimization problem. They add or remove surface voxels until the sum of squared differences between each camera image and corresponding model image is minimized. Broadhurst et al. introduced a probabilistic framework [8]. Instead of making “hard decisions” about voxel existence, they compute a per-voxel probability based on appropriate likelihoods. Isidoro et al. [9] reconstructed a 3D scene by deforming the visual hull surface towards a solution satisfying both photometric and silhouette constraints via a series of 1D epipolar searches at randomly selected surface points. S. Nobuhara et al. [10] also used the deformation of the visual hull surface, but their algorithm works in a snake-like fashion [11]. Mesh vertices move under the simulated forces.

C.H. Esteban et al. [12] combined silhouette and stereo information in a snake framework. Like [10], the snake is a deformable surface model. The texture driven force is computed by a multi-stereo correlation voting approach and a gradient vector flow diffusion. The silhouette driven force is derived from the distance between a vertex and the visual hull surface.

In this paper we present a novel active volumetric model (AVM). It is also inspired by the snakes (a.k.a. active contour models) [11]. But unlike [10,12] we represent the solid by voxels, which exhibits much greater modeling potential and superior advantages to the surface based representation. The model is initiated to be the visual hull and shrinks under the influence of multiple simulated forces. Three forces are derived from the volumetric model, the image silhouettes and the variance of the projective colors in our current work. Based on the composition of the three forces, our algorithm can significantly restrain holes and floating voxels and produce precise and smooth models. Compared with the algorithms proposed in [10], our algorithm is more reliable because it does not have to use the gradient descent approach which may fail.

Our algorithm can also be thought of as an improvement of the generalized voxel coloring (GVC) algorithm [3], which takes additional constraints into account.

Contributions of this paper include (1) the introduction of the idea of snakes into volumetric approaches which provides a computational framework to integrate several constraints in an intelligible way, (2) the introduction of an erosion model for computing the smooth force directly from the volumetric model, and (3) the introduction of a compulsory silhouette constraint together with the multi-pass procedure to make it more reasonable.

## II. ACTIVE VOLUMETRIC MODEL

The active volumetric model (AVM) is a physically motivated 3D deformable volumetric model which shrinks actively under the influence of multiple simulated forces towards the real scene by throwing away some of its voxels. The model is represented by voxels and initiated to be the visual hull (VH) [6]. Three forces are employed in this paper to evolve the model. The smooth force, which is thought of as an “internal force”, drives the model to be locally continuous. The compulsory silhouette force, which is an “external force” imposed by the image silhouettes, forces the model to be fully silhouette-consistent. Another “external force”, the color force, makes the model faithfully reproduce the input images.

### A. Smooth Force

Smooth is the dominant tendency of nature and thus it is quite reasonable to impose an internal force  $F_i$  to make the model locally continuous. To directly compute  $F_i$  from the volumetric model, we introduce an erosion model. It comes from the observation that natural objects evolve under the erosion of water, wind, etc, tending to loose edges and corners. The more area of surface is exposed, the faster will the part be eaten off. Now think the case that a voxel  $v$  is enveloped by 26 adjacent voxel locations  $l_i$ . We take every voxel  $v_i$  at  $l_i$ , if exists, as a protector that have the same ability to protect  $v$  from being eroded. Thus  $F_i$  can be defined as

$$F_i(v) = NV(v)/26 - \rho, \quad (1)$$

where  $NV(v)$  is the number of vacant locations around  $v$ , and  $\rho$  is a threshold.  $F_i$  helps to smooth the resulting solid, avoid holes, and eliminate floating voxels. Fig. 1 visualizes the effect of the smooth force by showing the evolvement of a cube under the single force  $F_i$  with  $\rho = 0.49$ . In each pass of the evolvement, a voxel is carved away if  $F_i(v) > 0$ .

### B. Compulsory Silhouette Force

We stipulate that a “marginal voxel”, which if is carved away will cause unexpected background pixels in at least one of the input images, can never be carved away. The marginal voxel can be easily determined by the LDI [3] structure. If the  $j$  th LDI at pixel  $p$ ,  $LDI_j(p)$ , records only one voxel  $v$ , then  $v$  is a marginal voxel. Thus the compulsory silhouette force is defined as

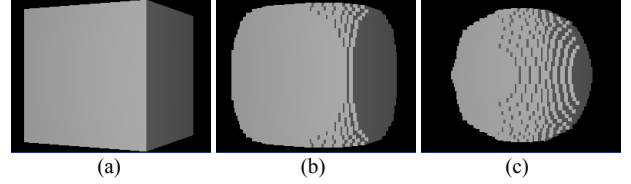


Figure 1. The evolvement of a cube (a) under  $F_i$  after 10 passes (b) and 30 passes (c).

$$F_s(v) = \begin{cases} -\infty, & \text{if } \exists j \text{ and } p \in \text{Prj}_j(v) \ni \overline{\overline{LDI_j(p)}} = 1; \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

It can be known from (2) that the result under the influence of  $F_s$  is relevant to the order of voxel carving. To alleviate this problem, our algorithm executes in multi-pass with the gradually decreased threshold of the color force. By this way, the most “impossible voxels” are carved first, and the “potential voxels” are left to be determined later.  $F_s$  forces the AVM to be fully silhouette-consistent, prevents the edges, angles and thin structures of the scene from being carved away, and keeps the model integrated.

### C. Color Force

As most voxel coloring algorithms do, we assume the Lambertian reflection model. Color consistency provides the most important force to deform the AVM. Many proposed measurements for color consistency [13] can be used. Here we adopt the standard deviation for its simplicity and good performance. Let  $t \in \{R, G, B\}$  stand for the color channel, and  $s^t(v)$  for the standard deviation of the projective color intensities in all images from which  $v$  is visible. The color force is defined as

$$F_c(v) = \max_t (s^t(v)) - \sigma, \quad (3)$$

where  $\sigma$  is a threshold.

### D. Multi-pass Evolvement Algorithm

The active model, initially being the VH, continuously evolves (shrinks) under the composite force

$$F(v) = \alpha F_i(v) + F_s(v) + F_c(v), \quad (4)$$

where  $\alpha$  is the weight of the smooth force. If  $F(v) > 0$ ,  $v$  is carved away.

Each force in formula (4) promotes the voxel carving if positive, and prevents it if negative. Specifically  $F_i$  tends to carve extrusive voxels and protect concave ones;  $F_s$  compulsively prevents the carving of marginal voxels;  $F_c$  promotes the carving of voxels that project to significantly variant colors and prevents the carving of color-consistent voxels. To alleviate the indeterminacy brought by  $F_s$ , we execute the evolvement algorithm in multi-pass with the gradually decreased threshold  $\sigma$ . The pseudocode is shown in Fig. 2, in which  $\bar{c}$  stands for the average of the projective

```

initialize  $v.state$  of every voxel  $v$  by shape from
silhouettes;
initialize SVL; render SVL to LDIs;
for each decreased  $\sigma$  {
  copy SVL to CVSVL;
  while (CVSVL is not empty) {
    remove a voxel  $v$  from CVSVL; compute  $F(v)$ ;
    if ( $F(v) > 0$ ) {
       $v.state = EXTERIOR$ ; remove  $v$  from SVL;
      for each image  $i$ 
        for each pixel  $p \in Prj_i(v)$  {
          if ( $v$  is head of  $LDI_i(p)$ )
            add next voxel on  $LDI_i(p)$  to CVSVL
        }
      if it is not in CVSVL;
      remove  $v$  from  $LDI_i(p)$ ;
    }
    for every voxel  $v'$  sharing a face with  $v$ 
      if ( $v'.state == INTERIOR$ ) {
        add  $v'$  to SVL;  $v'.state = SURFACE$ ;
        for each image  $i$ 
          for every pixel  $p \in Prj_i(v')$ 
            insert  $v'$  to  $LDI_i(p)$ ;
        add  $v'$  to CVSVL;
      }
    }
  }
  else  $v.color = \bar{c}$ ;
}
}

```

Figure 2. Pseudo-code for the AVM algorithm.

colors of  $v$  in all images from which  $v$  is visible, SVL for the surface voxel list, and CVSVL for the changed visibility SVL.

### III. EXPERIMENTAL RESULTS

We present the experimental results based on both synthetic images and real photos as well as a comparison between the AVM algorithm and the GVC-LDI algorithm [3] for the synthetic case. The algorithms ran on a desktop computer equipped with an Intel Pentium IV 3.2GHz CPU and 1G byte main memory. Input images were rendered (for the synthetic) or resized (for the real) to  $300 \times 199$  pixels.

We first reconstructed a sofa model from 18 synthetic images with both AVM and GVC-LDI algorithms. The volume was initially represented by  $128^3$  voxels. The VH, which served as the initial model of both algorithms, is shown in Fig. 3 (b). A marching cubes algorithm [14] was used to extract a triangular mesh model of the VH and improve the renderings, but it also made the mesh model thinner than the voxel model. For the AVM algorithm, we set  $\alpha = 50$ ,  $\rho = 13/26$ , and  $\sigma = \{95, 75, 60, 50, 45\}$ . The total running time was 73.33s. Fig. 3 (c) shows the intermediate results after the first 2 passes with

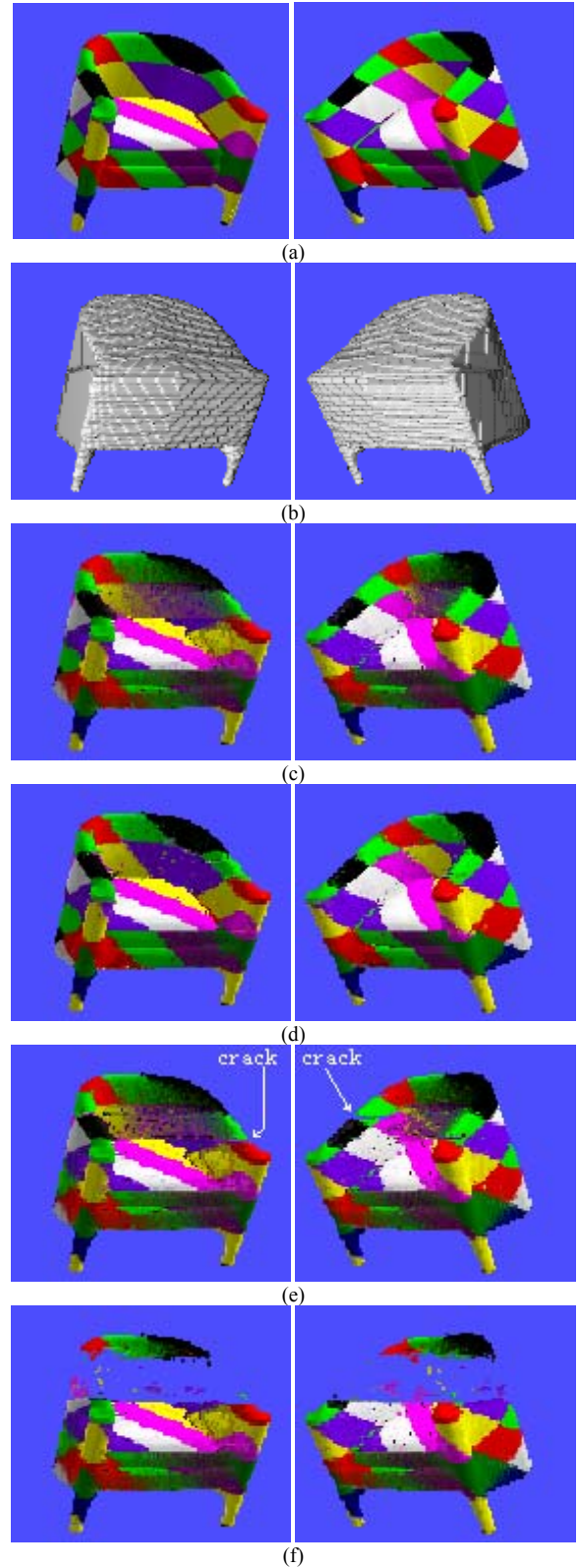


Figure 3: Images rendered from viewpoints that are different from any of the input images. (a) is rendered from the original model; (b) from the VH; (c) and (d) from the reconstructions of the AVM algorithm; (e) and (f) from the reconstructions of the GVC-LDI algorithm.

$\sigma = \{95, 75\}$ ; Fig. 3 (d) shows the final results. When the

algorithm proceeded, outliers were thrown away, the texture became clearer and clearer and the model approached the real geometry of the object. It can be seen from Fig. 3 (d), the final reconstructed model is precise and smooth. No obvious hole appears, and the floating voxels are significantly restrained. Because the computation of the color force was still face with the drawbacks of the voxel coloring algorithm as is mentioned in Section 1, some noise remained. For the GVC-LDI algorithm, we set the threshold for the standard deviation to 80, which is approximately the best value. The running time was 43.13s. The results are shown in Fig. 3 (e). As can be seen from Fig. 3 (e), the results are far from satisfactory. The texture is seriously blur, which reveals that a lot of outliers exist in the model; floating voxels are obvious; cracks (or holes) appear in the armrests. An attempt to reduce outliers by a tighter threshold caused severe cracks. Fig. 3 (f) shows the results obtained with the threshold for the standard deviation set to 75.

We then modeled a toy bear from real photos with the AVM algorithm. 18 images were captured by a fixed camera and a turntable. A Nikon D70 camera with a Nikkor 50mm lens was used. Both the camera and the turntable were calibrated imperfectly. The scene was illuminated by a common fluorescent lamp above the camera. The initial model contained  $100 \times 150 \times 100$  voxels. We set  $\alpha = 50$ ,  $\rho = 13/26$ , and  $\sigma = \{110, 90, 75, 65, 60\}$  to compensate for the illumination variance relative to the object and the errors of the Lambertian assumption. The running time was 33.69s. As is shown in Fig. 4 (b), the results are impressive. The surface is precise and smooth. Holes and floating voxels are well restrained. The handle and some other parts of the basket were carved away by the shape from silhouette algorithm because of the imprecise calibration.

#### IV. CONCLUSIONS

In this paper we have presented an active volumetric model for 3D reconstruction from a set of calibrated images. The model shrinks under the influence of simulated forces. It provides a framework to integrate several constraints in an intelligible way. Based on the smooth force, the compulsory silhouette force and the color force, our algorithm can significantly restrain holes and floating voxels and produce precise and smooth models.

#### REFERENCES

[1] S. Seitz and C. Dyer, "Photorealistic scene reconstruction by voxel coloring," Proc. of CVPR, pp. 1067–1073, June 1997.  
 [2] K.N. Kutulakos and S.M. Seitz, "What do N photographs tell us about 3D shape?," TR680, Computer Science Dept. U. Rochester, January 1998.  
 [3] W.B. Culbertson, T. Malzbender, and G. Slabaugh, "Generalized voxel coloring," Proc. of the ICCV Workshop, Vision Algorithms Theory and Practice, Springer-Verlag Lecture Notes in Computer Science 1883, pp. 100–115, September 1999.



(a)



(b)

Figure 4: Reconstruction from real images. (a) One of the 18 input images. (b) Renderings of the reconstructed model from new viewpoints.

[4] K.N. Kutulakos and S.M. Seitz, "A theory of shape by space carving," IJCV, Vol. 38, No. 3, pp. 199–218, July 2000.  
 [5] C.R. Dyer, "Volumetric scene reconstruction from multiple views," Foundations of Image Understanding, LS Davis, ed., Kluwer, Boston, pp. 469–489, 2001.  
 [6] G. Slabaugh, B. Culbertson, and T. Malzbender, "A survey of methods for volumetric scene reconstruction from photographs," Proc. International Workshop on Volume Graphics, pp. 81–100, 2001.  
 [7] G. Slabaugh, B. Culbertson, T. Malzbender, and R. Schufer, "Improved voxel coloring via volumetric optimization," Technical Report 3, Center for Signal and Image Processing, Georgia Institute of Technology, 2000.  
 [8] A. Broadhurst, T.W. Drummond, and R. Cipolla, "A probabilistic framework for space carving," Proc. of ICCV, pp. 388–393, 2001.  
 [9] J. Isidoro and S. Sclaroff, "Stochastic refinement of the visual hull to satisfy photometric and silhouette consistency constraints," Proc. of ICCV, pp. 1335–1342, 2003.  
 [10] S. Nobuhara and T. Matsuyama, "Dynamic 3D shape from multi-viewpoint images using deformable mesh model," Proc. of 3rd International Symposium on Image and Signal Processing and Analysis, pp.192–197, 2003.  
 [11] M. Kass, A. Witkin, and D. Terzopoulos. "Snakes: active contour models," IJCV, 1 (4), pp. 321–331, 1988.  
 [12] C.H. Esteban and F. Schmitt, "Silhouette and stereo fusion for 3D object modeling," Proc. of the Fourth International Conference on 3-D Digital Imaging and Modeling (3DIM'03), pp. 46–53, 2003.  
 [13] G.G. Slabaugh, W.B. Culbertson, T. Malzbender, M.R. Steven, and R.W. Schafer, "Methods for volumetric reconstruction of visual scenes," IJCV, 37(3), pp. 179–199, 2004.  
 [14] W.E. Lorensen and H.E. Cline, "Marching cubes: a high resolution 3D surface construction algorithm," Computer Graphics, Volume 21, Number 4, pp. 163–169, July 1987.