Vision, Modeling, and Visualization (2013)
Michael Bronstein, Jean Favre, and Kai Hormann (Eds.)

Poster

Fine-Scale Editing of Continuous Volumes using Adaptive Surfaces

Kai Ruhl[1], Stephan Wenger[1], Dennis Franke[1], Julius Saretzki[1] and Marcus Magnor[1]

[1]TU Braunschweig, Germany

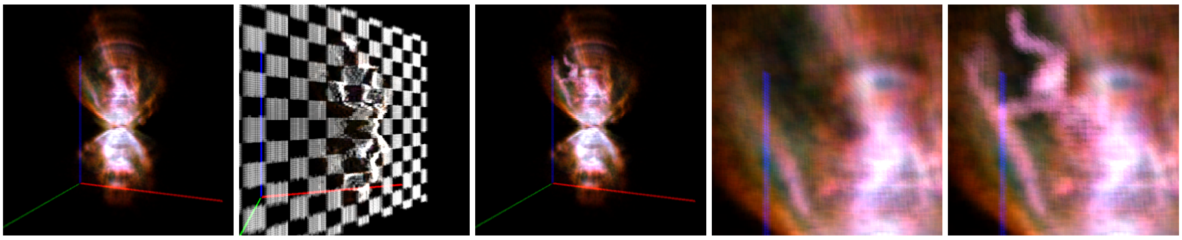

(a) (b) (c) (d) (e)

Figure 1: *Surface painting on Nebula NGC-6302. (a) source volume, (b) adaptive drawing surface with filter kernel size $k_s=1$, (c) adding a pink filament that follows the local density distribution, (d) detail before editing, (e) detail after editing.*

Abstract

Many fields of science such as astronomy and astrophysics require the visualization and editing of smooth, continuous volume data. However, current high-level approaches to volume editing concentrate on segmentable volume data prevalent in medical or engineering contexts, and therefore rely on the presence of well-defined 3D surface layers. Editing arbitrary volumes, on the other hand, is currently only possible using low-level approaches based on the rather unintuitive direct manipulation of axis-aligned slices. In this paper, we present a technique to add or modify fine-scale structures within astronomical nebulae based on adaptive drawing surfaces that enable 2D-image-like editing approaches. Our results look more natural and have been produced in a much shorter time than previously possible with axis-aligned slice editing.

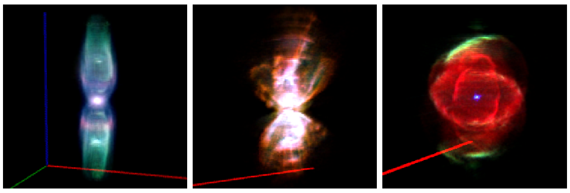
Categories and Subject Descriptors (according to ACM CCS): I.3.5.f [Computer Graphics]: Computational Geometry and Object Modeling—Modeling packages I.4.10.e [Image Processing and Computer Vision]: Image Representation—Volumetric

1. Introduction

In the field of volumetric editing, various approaches exist for separable, semantically tagged data sets such as those present in medical or engineering contexts [BKW08, ZCD*10, HRJNF09, NHY*10]. Editing continuous data sets like those found in astronomy and astrophysics is a harder task, and consequently, domain dependent, specialized approaches are more prevalent [SKW*11]. However, these are not applicable for general continuous volumes. On the other end of the spectrum, generic single-voxel manipulation tools do not guide the user at all in drawing inside the volume. Our goal is to allow semantically-driven, fine-scale painting on smoothly varying continous volumes that do not provide any clear object boundaries.

When reconstructing astronomical nebulae like those shown in Fig. 1 and Fig. 2, only one image (taken from Earth or Hubble) is available, so priors in the form of axial symmetry and smoothness [WAS*12] are required, which can lead to oversmoothing that removes fine structures. An alternative is to model the nebula from scratch using astrophysical expertise [SKW*11] to build diffusion transforms, but again the re-creation of fine structures is difficult.

DOI: 10.2312/PE.VMV.VMV13.221-222



(a) (b) (c)

Figure 2: *Examples of astronomical nebulae which cannot be edited well with traditional modeling tools. All are emissive in three color channels. (a) M2-9, a relatively simply structured cylindric symmetric nebula with a bright core. (b) NGC-6302, a butterfly structured nebula with a multitude of fine filaments. (c) NGC-6543, a nebula with distinct color emission distribution.*

Our baseline for painting single pixels or fine structures is axis-aligned slice editing with 3D brushes of varying size, as proposed by the *3Dish* editor [Ans11]. Since the drawing surface is always planar, painting is not content-aware at all. Upon editing, one frequently wishes for a “snap” behaviour that aligns the drawing surface to existing structures.

2. Adaptive Surface Editing

We represent our volume as $V : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ of RGB values $\mathbf{v} = (r, g, b)$ for each position $\mathbf{x} = (x, y, z)$, with dimension $s \times s \times s$. Initially, the drawing surface is an axis-aligned plane $I : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ centered at $\mathbf{x}_I = (\frac{s}{2}, \frac{s}{2}, z_i)$ with varying z_i. Painting on this surface is equivalent to working with [Ans11].

In a first improvement, we parameterize the plane further with angles ϕ_x, ϕ_y to give it arbitrary orientation. Shifting along the plane normal $\vec{n}_z$ supersedes shifts in z_i. Painting on this arbitrarily oriented planar surface I can now at least locally adapt to the nebula form.

However what we really want is a drawing surface I_c that snaps to meaningful nebula structures, such as the one shown in Fig. 1 (b). Since the RGB channels (usually) map to molecular density distributions within V, following color similarities is a valid strategy. Given a user-picked color $\mathbf{c}$, the idea is to minimize an energy:

$$\min_{z(x,y)} \iint \Big(\overbrace{\|\mathbf{c} - V(I(x,y) + z(x,y)\vec{n}_z)\|_1}^{\text{data term}} + \underbrace{\lambda_s \|\nabla z(x,y)\|_2}_{\text{smoothness}} + \underbrace{\lambda_r |z(x,y)|}_{\text{inertia}} \Big) \, \mathrm{d}x \, \mathrm{d}y \tag{1}$$

Here, we seek for each pixel $x, y \in I$ an offset z along the plane normal $\vec{n}_z$. Evaluating only the data term results in a possibly very jagged surface, so we add a smoothness term weighted by λ_s. Since we also want I_c to be near to the previous plane position as long as it does not violate the data term too much, we include an inertia term weighted by (a small) λ_r that tries to keep the adapted surface I_c close to I.

To solve the minimization, we evaluate z pointwise using data and inertia term only by sampling along $V(I + z\vec{n}_z)$, and then filter the result with blur kernels of size k_s. We calculate only one minimization iteration in order to achieve realtime performance. Our z is restricted to be positive, so that the resulting surface I_c always lies in front of the original planar surface I. Depending on k_s which is a user parameter, I_c is more or less jagged or smooth. The nebulae shown in Fig. 2 give good results on $k_s = 2$ for M2-9, $k_s = 1$ for NGC-6302 as in Fig. 1, and $k_s = 1$ for NGC-6543.

From a workflow perspective, a user first positions the fixed plane I inside the volume; then uses the color picker to choose $\mathbf{c}$; and thirdly, optionally adjusts the filter kernel size k_s to determine the final shape of I_c. Painting on this surface is then a matter of simple projection using standard raytracing. A natural-looking result is shown in Fig. 1.

3. Conclusions

We presented an approach to paint fine structures into smoothly varying volumes that do not contain clearly segmentable objects. Our main case was correction of astronomical nebulae reconstructions, but further applications in e.g. medical soft tissue analysis are also possible.

References

[Ans11] ANSTEY C.: http://code.google.com/p/3dish-volumetric-editor/, 2011. 2

[BKW08] BÜRGER K., KRÜGER J., WESTERMANN R.: Direct volume editing. *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (2008), 1388–1395. 1

[HRJNF09] HUFF R., ROSA JR R., NEDEL L., FREITAS C.: Volume sculpting based on geometric tools. *Journal of the Brazilian Computer Society 15*, 2 (2009), 3–18. 1

[NHY*10] NAKAO M., HUNG K., YANO S., YOSHIMURA K., MINATO K.: Adaptive proxy geometry for direct volume manipulation. In *Proceedings of the IEEE Pacific Visualization Symposium (PacificVis)* (2010), pp. 161–168. 1

[SKW*11] STEFFEN W., KONING N., WENGER S., MORISSET C., MAGNOR M.: Shape: A 3D modeling tool for astrophysics. *IEEE Transactions on Visualization and Computer Graphics (TVCG) 17*, 4 (Apr. 2011), 454–465. 1

[WAS*12] WENGER S., AMENT M., STEFFEN W., KONING N., WEISKOPF D., MAGNOR M.: Interactive visualization and simulation of astronomical nebulae. *Computing in Science & Engineering 14*, 3 (May 2012), 78–87. Editorial article. 1

[ZCD*10] ZHAO Y., CLAPWORTHY G., DONG F., ZHANG X., CHEN W., VICECONTI M.: iVolBrush: an intelligent interactive tool for efficient volume selection and editing. In *Proceedings of the First International Conference on Intelligent Interactive Technologies and Multimedia* (2010), pp. 231–235. 1