Escher's Ants as Metaphor: Topological Marching for the Well-Composed, Genus Zero Crowd

Release 1.00

Nicholas J. Tustison[1], Brian B. Avants[1], Marcelo F. Siqueira[2], and James C. Gee[1]

November 30, 2010

[1]Penn Image Computing and Science Laboratory
University of Pennsylvania
[2]Departamento de Computação e Estatística
Universidade Federal de Mato Grosso do Sul

Abstract

Topological considerations for segmentation results are important for such applications as proper brain segmentation from digital image data. We present an enhancement of the `itk::FastMarchingImageFilter` which allows for topologically constrained evolution of the level set. Identical to the original functionality of the `itk::FastMarchingImageFilter`, the evolution of the level set of a single or multiple genus zero, well-composed seed objects proceeds according to the specified parameters. With our proposed enhancements, the user can either choose to prevent the level set from merging with itself such that the original topology of the initial seed object(s) is not violated (a la the work of [3]) or that no handles are created during the evolution process (a la the work of [7]). However, in contrast to earlier approaches which relied on the concept of the *simple point* implicitly requiring the definition of a user-specified foreground/background connectivity, we use the related, but more restrictive concept of well-composed sets to topologically constrain the evolution of the level set. Utility of our submission is demonstrated on both 2-D and 3-D brain images.

Contents

1 Introduction

In our earlier contribution [9], we outlined the concept of well-composedness for 2-D and 3-D digital images and provided a topological repairing algorithm for 2-D and 3-D binary and multi-labeled images. Well-composedness was initially defined by Latecki et al. [5] who proposed 2-D repairing algorithms to make a binary image well-composed. This was later extended to 3-D and multiple labels in [8]. Well-composedness implies important topological and geometrical properties which simplifies many algorithms such as Marching cubes [6], various thinning algorithms, and Euler characteristic computation.

For $n = \{2,3\}$ an n-D binary digital image is said to be well-composed if and only if the set of points in the pixel boundaries shared by the foreground and background points of the image is a $(n-1)$-D manifold [8]. Intuitively, a well-composed image is one without topological ambiguities. This is made possible by permitting only face-connectedness between foreground voxels and background voxels.

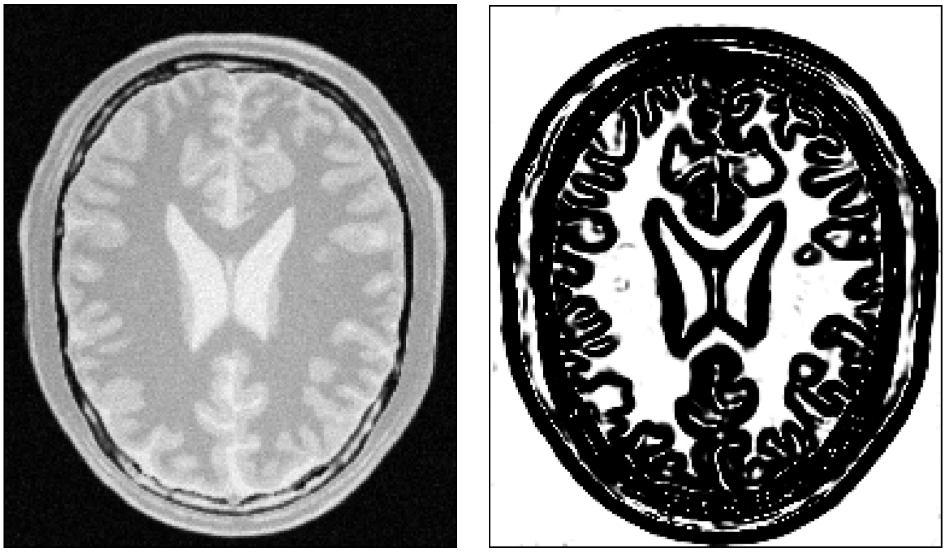
We propose two optional topology checks for the `itk::FastMarchingImageFilter` based on the concept of well-composedness, which are closely related to the concept of *simple points* [1, 2]. Simple points are voxels that can be changed from foreground to background or vice versa without changing the foreground/background topology. The two topology checks discussed in this submission were previously published based on the simple point concept. The first topology check, originally proposed by Han et al. [3], maintains global topology of the initial seed objects during the evolution of the level set. The second, proposed by Segonne [7], prevents the formation of handles and relies upon the concept of *multisimple points* which are points that can be added or removed without creating handles. However, since both algorithms rely upon the simple point for local topology checking, the levels sets produced necessitate a special Marching cubes algorithmic variation to build topologically consistent meshes [3, 7]. By replacing the simple point criterion with the more restrictive well-composed set criterion, the resulting level set results in no such topological ambiguities.

2 Implementation

The only change to the API of the `itk::FastMarchingImageFilter` is the possibility of the user setting the topology check to one of the following three enumerated types: `None` (default—no topology check is performed), `Strict` (similar to [3]), and `NoHandles` (similar to [7]). The well-composedness checks were ported from our earlier contribution and the additional topology check is only about 20 lines. For the second topology check, a connected component image is required to keep track of the number of distinct foreground objects. However, despite the ostensible simplicity, the results have proven quite useful to the work in our lab and we suspect that they might prove useful to others in the ITK community.

For the sample results, we modified the file `FastMarchingImageFilter.cxx` available in the `Examples/` directory. We stripped out the code which constructed the speed image and put specification of the speed image on the command line. We also added the optional topology check demonstrated with the following code snippet.

```
fastMarching->SetTopologyCheck( FastMarchingFilterType::None );
if( argc > 6 && atoi( argv[6] ) == 1 )
  {
  std::cout << "Strict." << std::endl;
```



(a) (b)

Figure 1: (a) The 2D brain image slice and (b) the corresponding speed image.

```
    fastMarching->SetTopologyCheck( FastMarchingFilterType::Strict );
    }
  if( argc > 6 && atoi( argv[6] ) == 2 )
    {
    std::cout << "No handles." << std::endl;
    fastMarching->SetTopologyCheck( FastMarchingFilterType::NoHandles );
    }
```

All the results discussed below are included as tests.

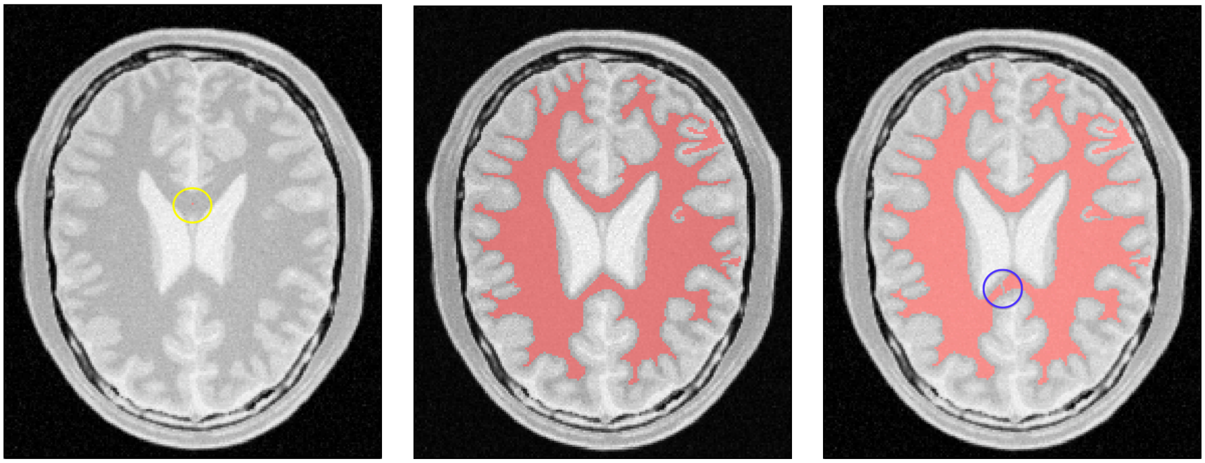
3 Sample Results

3.1 2-D Examples

The first set of results employ the image `BrainProtonDensitySlice.png` (see Figure 1(a)) taken from the `Examples/Data` directory used to demonstrate the `itk::FastMarchingImageFilter` in the ITK guide [4]. Also provided in Figure 1(b) is the corresponding speed image.

Our first 2-D experiment shows the evolution of the level set from a single seed using the speed image given in Figure 1(b) (stopping value = 150). The front of the level set circumambulates in opposing directions around the ventricles. The default behavior provides no constraint on topology so the two fronts merge creating a handle (see Figure 2(b)). This is in contrast to the behavior constrained by both topology checks shown in Figure 2(c) where the fronts do not collide (see the topological break circled in blue).

For a more sophisticated example, we initialize our level set with multiple genus zero, well-composed seed objects shown in Figure 3(a) where the `m_TrialPoints` are determined using the `itk::LabelContourImageFilter`. Using a stopping value of 150, evolution proceeds while maintaining a constant topology. After evolution and recoloring using `itk::RelabelComponentImageFilter`, one can see that the topology of the original level set remains the same. In contrast, without topology constraints, the level set is allowed to merge thus violating the topology of the original configuration.



(a) (b) (c)

Figure 2: (a) Starting from a single voxel seed (circled in yellow), the level set evolves (stopping value = 150) according to the speed image. The default behavior, results in given in (b), does not prevent the two fronts from merging. However, both topology checks result in non-merging fronts since that would violate the genus zero topology of the original seed.

3.2 3-D Examples

We introduce the 3-D capabilities of our submission with a simple toroidal speed image (Figure 4(a)) initialized with four genus zero, well-composed seed blobs given in Figure 4(a). After evolution and user-selection of the strict topological check, topology is maintained and the four 3-D objects are visualized in Figure 4.

Finally, we provide a more sophisticated 3-D example. We construct a simple speed image from a crude 3-D segmentation of the white matter of a brain represented by the axial slice shown in Figure 5(a). We initialize with 7 genus zero, well-composed seeds (four of which are shown in the axial slice of Figure 5(a)). A superior/frontal perspective of all 7 seed objects is given in Figure 5(d). Using the `Strict` topological constraint and relabeling the connected components after evolution, we visualize the final segmentation in which the 7 objects remain genus zero and well-composed ((b) and (e)). In contrast, using the `NoHandles` option of the resulting 7 objects shown in (e) and the single object shown in (f) was verified by converting to a VTK mesh using the standard Marching cubes algorithm.

References

[1] G. Bertrand. Simple points, topological numbers and geodesic neighborhoods in cubic grids. *Pattern Recognit. Lett.*, 15(10):1003–1011, 1994. 1

[2] G. Bertrand. A boolean characterization of three-dimensional simple points. *Pattern Recognit. Lett.*, 17:115–124, 1996. 1

[3] Xiao Han, Chenyang Xu, and J.L. Prince. A topology preserving level set method for geometric deformable models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(6):755–768, 2003. (document), 1, 2, 3

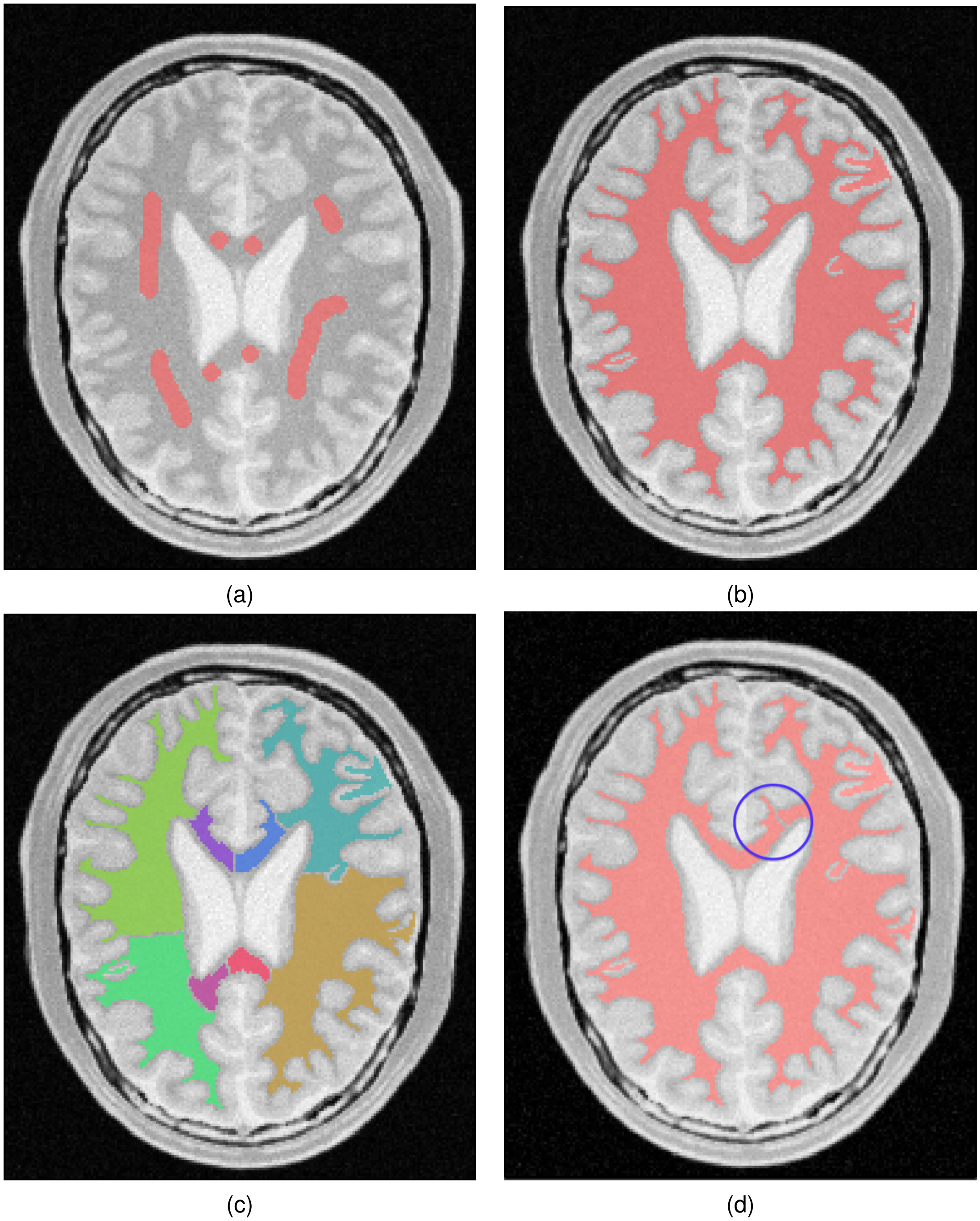


Figure 3: (a) Level set initialization with multiple genus zero, well-composed seed objects. (b) Default output where no topology checks occur. (c) Resulting image after using the 'Strict' topology check of [3] where we have colored the different connected components. (d) Using the topology check of [7], even though we start with eight well-composed objects, they merge until such merging would cause a handle to form. Note the topological break circled in blue preventing a handle from forming.

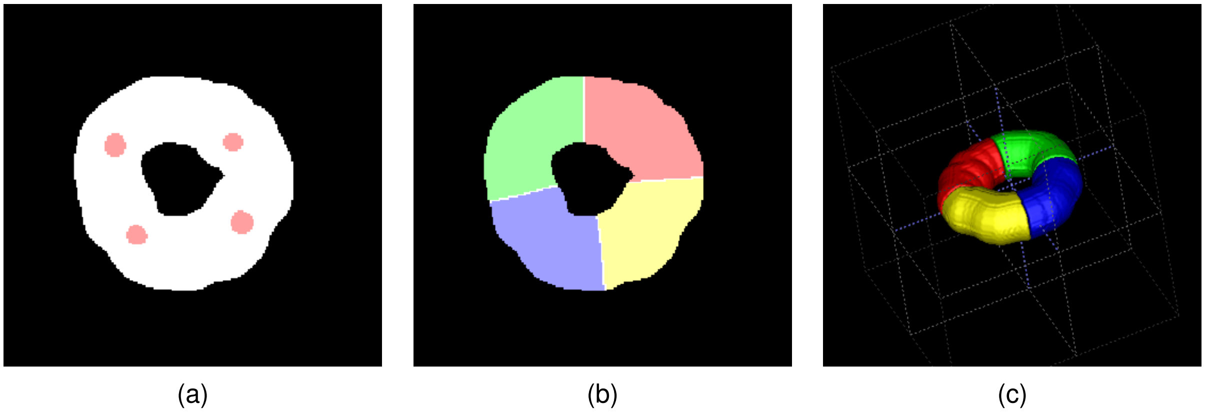


Figure 4: Simple 3-D toroid example. (a) Initial level set of four genus zero, well-composed seed objects. (b) After termination, four genus zero, well-composed objects remain. (c) Mesh output of the results.

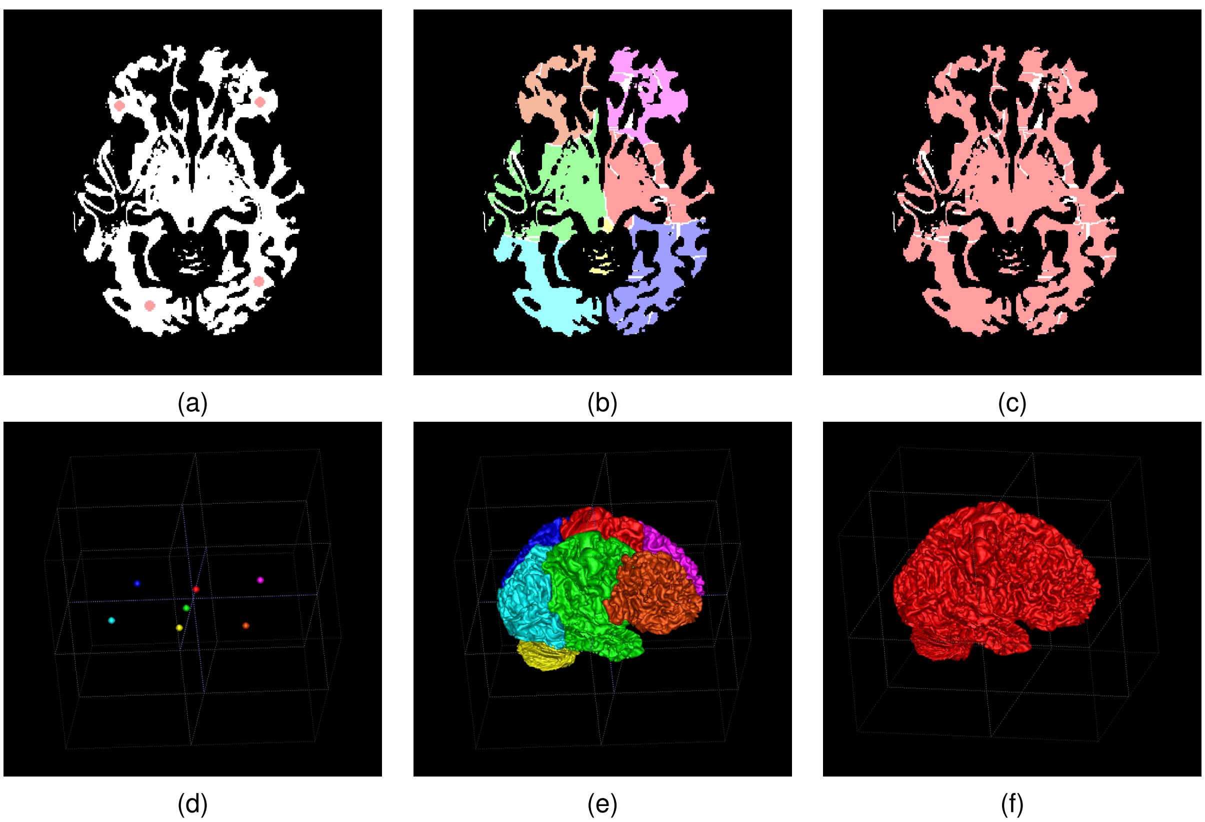


Figure 5: A more sophisticated white matter example. (a) and (c) Initialization with 7 genus zero, well-composed objects. (b) and (e) Strict topology results in 7 genus 0 objects. (c) and (f) No handles topology results in a single genus 0 object.

[4] Luis Ibanez, Will Schroeder, Lydia Ng, and Josh Cates. *The ITK Software Guide*. Kitware, Inc., Albany, NY, 2 edition, November 2005. 3.1

[5] Longin Latecki, Ulrich Eckhardt, and Azriel Rosenfeld. Well-composed sets. *Computer Vision and Image Understanding*, 61:70–83, 1995. 1

[6] William E. Lorenson and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *Computer Graphics*, 21(4):163–169, 1987. 1

[7] Florent Segonne. Active contours under topology control–genus preserving level sets. *Int. J. Comput. Vision*, 79(2):107–117, 2008. (document), 1, 2, 3

[8] Marcelo Siqueira, Longin Jan Latecki, Jean Gallier, Nicholas Tustison, and James Gee. Topological repairing of 3d digital images. *Journal of Mathematical Imaging and Vision*, 30(3):249–274, March 2008. 1

[9] Nicholas J. Tustison, Marcelo Siqueira, and James C. Gee. Well-composedness and the topological repairing of 2-D and 3-D digital images. *Insight Journal*, page published online, 2007. 1