AN ALGORITHM FOR SUGGESTING DELINEATION PLANES FOR INTERACTIVE SEGMENTATION

...

...
...
...

ABSTRACT

In this paper we present a scheme to reduce the amount of user iterations required to segment an object by delineating on cross-section planes. Starting with an initial segmentation created from a small number of delineated curves, the algorithm progressively analyzes the uncertainty of segmentation with respect to the image features and suggests the "next plane" for delineation that would maximally resolve the uncertain regions. Compared with the few prior art on this problem, we adopt a simpler computational framework that is made up of an RBF-based curve interpolation method, a distance-based uncertainty metric, and a plane determination approach using density-based clustering. We demonstrate using both synthetic and real examples that our method uses less than 50% of the number planes than a random selection scheme to achieve 90% segmentation accuracy.

1. INTRODUCTION

The need for efficient and accurate means for segmenting three-dimensional (3D) medical images continues to rise. While there is significant research into automated methods for segmentation, segmentation in clinical practices still requires heavy intervention from expert users. One of the most popular paradigms for computer-assisted segmentation is plane-based. First, the user (e.g., the clinician) delineates, either manually or using interactive tools, the boundary curves on multiple cross-section planes of the image volume. A 3D surface is then generated automatically to interpolate these planar curves. A main reason for the popularity of this paradigm is that, despite the advance in 3D visualization methods, viewing and outlining are still more effectively done on 2D images, since the user can perceive the entire image at once with no occlusion.

There has been extensive research into methods that assist either in the 2D delineation task (e.g., using Livewire [BM97]) or allow smooth, automatic reconstruction of the 3D surface from the delineated curves (e.g., [LBD*08]). However, another key question in this plane-based segmentation paradigm that has received little attention so far is what plane should be picked to delineate the boundary. The appropriate choice of such planes (and their ordering) can reduce the number of planes, hence the manual effort of delineation, to achieve the same segmentation accuracy. For example, a carefully selection of planes can reconstruct the tripod shape in Figure 2 (top) using only 4 planes, whereas a random selection of planes would require 11 planes to achieve a similar reconstruction accuracy, as shown in Figure 2 (bottom).

If the shape of the object is known *a priori*, The problem of "where to delineate next" can be formulated computationally as finding the plane that maximally reveal the geometric features of the object that have not already been captured by the existing planes [MSM11]. However, the problem becomes ill-posed if the object has an unknown shape (e.g., in the case of segmenting a tumor). We know of only two works addressing the plane selection problem in the context of 3D image segmentation without a shape prior, both developed by Hamarneh and his colleagues [THA10, THA11]. Both methods start by asking the user to delineate on a few arbitrarily selected or user-specified planes. A segmentation is generated using the delineated curves, using either an extended Livewire algorithm [THA10] or a random walk classifier [THA11]. Then the methods analyze the "uncertainty" of segmentation, where the measure of uncertainty differs in these two works. While [THA10] only evaluates the ambiguity of segmentation, [THA11] uses a more complex metric that additionally involves alignment of the surface with strong image edges, the agreement of inside and outside regions to expected intensity distributions, and the smoothness of the surface. The plane that is most "uncertain", defined either as one that best fits a selected set of uncertain planes [THA10] or minimizes the integral of uncertainty metric over the plane [THA11], is found using a gradient-descent search.

In this paper, we propose a new method for suggesting the next plane to delineate. Our method follows a similar flow to the methods by Hamarneh et al. [THA10, THA11]. Starting from the segmentation using a few initial planes, the system alternates between analyzing the segmentation, suggesting a new plane that passes through regions with low confidence, and updating the segmentation. There are three notable differences between our method and the ones in [THA10, THA11]. First, we formulate our segmentation as an interpolation problem and solves it using Radial Basis Functions (RBF). The advantage of our RBF-based segmentation over Livewire or random walks is that the resulting surface minimizes thin-spline energy, a physically based energy that ensures a smooth appearance. As a result, and as the second difference, we no longer need to consider smoothness in our measure of "uncertainty", resulting in a simpler formulation that only involves agreement with strong image edges. Thirdly, we replace the costly gradient-descent search for finding the plane by a simple clustering of uncertain surface regions followed by computing the plane that passes through the largest cluster centers. We demonstrate using both synthetic and anatomical structures that our plan-selection scheme, although computationally simpler than prior art, achieves similar quantitative improvements over random plane selection.

2. METHOD

The input to our method is a 3D volume (e.g., a MRI or CT scan). As mentioned above, we start by asking the user to delineate boundary curves on a few oblique cross-sections. The user may choose to select these initial planes by hand, or let the system arbitrary pick two orthogonal planes centered in the volume. The method then iterates over creating the segmentation from delineations on existing planes using RBF, assessing the uncertainty over the segmented surface based on distances to image edges, and suggesting a new plane by clustering the high-uncertainty regions. The method flow is illustrated in Figure 1, and the various steps are elaborated next.

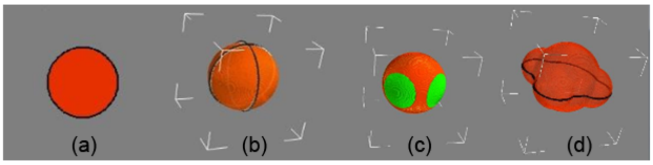

Figure 1: Algorithm Pipeline on a toy example: (a) curve delineated on one of the initial planes. (b) Segmentation using the initial planes (two here). (c) Clusters of segmentation points with high uncertainty. (d) Delineation on the next plane suggested by our algorithm, which passes through the centroids of the three biggest clusters in (c), and the updated segmentation.

2.1. RBF segmentation

Given curves delineated on 2D slices, we want to create a surface that interpolate them, which would segment the volume into foreground and background. We first briefly review the method of RBF for smoothly interpolating scattered data. Given spatial locations (called "centers") x_i for $i = 1, \ldots, n$, each associated with some scalar value f_i. The value at an arbitrary spatial location x can be computed by

$$f(x) = \sum_{i=1}^{n} w_i \phi(\|x - x_i\|)$$

where $\phi(r) = r^2 \ln(r)$ is a radial basis kernel. The weights w_i are computed to satisfy the interpolation property, namely $f(x_i) = f_i$ for all i. The computation of w_i involves solving a linear system with n equations, after which the evaluation of $f(x)$ for any x involves only a simple summation as in the above formula. Such interpolation f is smooth in the sense that it has the minimal integral of squared second derivatives, the so-called thin-plate spline energy.

To apply RBF to our curve interpolation problem, we create a signed volume via interpolation so that the zero iso-surface of the volume interpolates the curves. We start by sampling each curve and creating, for each sample, two RBF centers along the normal of the curve, one on the inside and the other on the outside. The inside center is given a value of 1 and the outside center has value -1. The signed volume is then computed using RBF from the values at these centers.

As mentioned above, the complexity of solving for RBF depends on the number of centers, n. On the other hand, it is often not necessary to use a large number of RBF centers if the shape to be interpolated is inherently smooth. For efficiency, we adapt the RBF center reduction method of [CBC*01], which iteratively adds centers until the interpolation error drops beneath a threshold. Rather than considering centers on all planes at once, our reduction proceeds in two stages that ease the computation. In the first stage, we compute the centers on each 2D plane that are sufficient to interpolate the curve on that plane using a 2D RBF interpolation. Specifically, we start by computing the interpolation for only a tenth of all centers. Let the number of these centers be k. If the interpolation error $\|f_i - f(x_i)\|$ is greater than a threshold for any un-used center x_i, we add in the k centers with the greatest errors and repeat the process. In the second stage, we perform a similar center-addition process in 3D involving centers on all planes, but starting from the reduced set of centers on each plane.

2.2. Assessing uncertainty

While RBF segmentation results a smooth boundary surface that interpolates the input curves (up to a given error threshold), the segmentation may not correctly capture the actual anatomical boundary in the image space in-between the planes. Our goal is to assign an uncertainty score over the segmentation surface that measures deviation from the true boundary. Regions with high uncertainty should then be alerted to the user for further delineation.

We define the uncertainty at a point on the surface simply as its (unsigned) distance to the closest strong image edge. We extract image edges by applying the discrete Laplacian operator to each voxel and taking those voxels (called edge voxels) whose Laplacians are at least 3 standard deviations away from the mean over all voxels. We define a surface voxel as one that has a positive RBF value and a neighboring voxel with a negative RBF value. We then compute the shortest

Euclidean distance from each surface voxel to an edge voxel. This measurement of uncertainty is similar to the boundary term in the formulation of [THA11]. Note that we do not need the smoothness term in [THA11] as we start with a smooth segmentation.

2.3. Finding the next plane

We would like to suggest a plane that covers significant uncertain regions on the surface. The significance would need to take into account both the level of uncertainty and the size of the uncertain region. To this end, we first rule out surface voxels whose uncertainty is lower than the average among all surface voxels. The rest of the surface voxels are considered uncertain. We then find clusters of densely connected uncertain voxels, using a variant of the density-based clustering algorithm DBSCAN [EKJX96]. Starting from an un-clustered uncertain voxel, if it has more than k neighboring uncertain voxels for a given constant k in its 26-neighborhood, we create a cluster for it. A cluster is expanded as long as the density of the cluster is maintained. The clustering method does not require the knowledge of the number of clusters ahead of time, and is not sensitive to the ordering of voxels.

Since we want to reduce the global uncertainty as much as we can with the next user step, we seek a plane that will pass through the clusters with the greatest size. In addition to the largest cluster, we will consider (at most) two other clusters if they are at least half the size of the largest cluster. If we have all three clusters, the plane is defined to pass through the three cluster centroids (see Figure 1 (d)). Otherwise, we perform the Principal Component Analysis (PCA) of the one (or two) largest cluster and define the plane through the centroid of all cluster voxels in the direction spanned by the first two major PCA axes.

3. RESULTS

We start by demonstrating our method on a tripod-shaped 3D object in Figure 2. The synthetic input is created as a binary volume where voxels within the shape (represented by a mesh) have values 1 and those outside the shape have values 0. The top row shows the first few slices suggested by the algorithm. Observe that these planes quickly navigate to cover the interesting part of the shape, and the model is well captured after just 4 slices.

As in previous works [THA10, THA11], we perform a baseline comparison with a randomly selected sequence of planes through the center of the volume for delineation. As shown in the bottom row of Figure 2, which renders the segmentation after delineation on 3, 7 and 11 randomly selected planes, it takes much more effort to achieve a similar segmentation quality than using our algorithmically suggested planes.

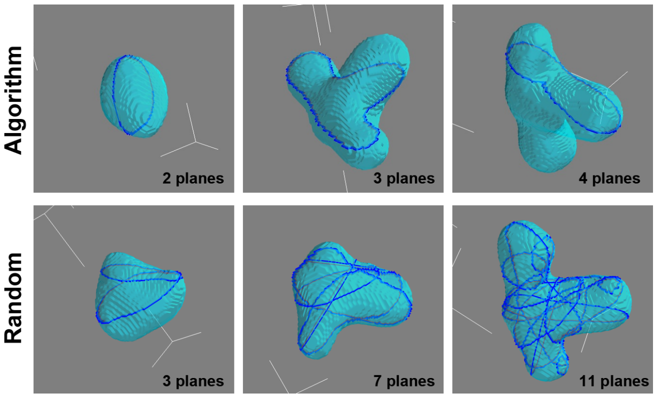

Figure 2: Segmenting a synthetic tripod shape using planes suggested by the algorithm (top) and selected at random (bottom), showing the intermediate segmentations after the noted number of planes.

To test our method on more complicated and real-world shapes, we created synthetic binary volumes (similar to the tripod example above) using three segmented anatomical models from BodyParts3D, including a right calcaneus (of dimension 104x93x128), L1 vertabrea (of dimension 114x93x128) and left hepatic vein (of dimension 156x96x135). For quantitative evaluation, we measure the accuracy of segmentation compared with the ground truth (the original anatomical models) using the well-known Dice coefficient, which is expressed as the ratio $\frac{2\|C\|}{\|A\|+\|B\|}$ where A, B, C are the sets of voxels interior to our segmentation, interior to the ground truth segmentation, and shared by the interior of both segmentations. We plotted the segmentation accuracy for different number of suggested planes using our approach and using random plane selection for these examples in Figure 3. Observe that the use of our algorithm achieves a reduction of at least 50% of the number of delineation planes that are needed to reach a 90% segmentation accuracy. Such reduction is particularly apparent for objects with more irregular shape, such as the Vertebra.

Finally, we evaluated our algorithm on two real biomedical data set, a MRI scan of the brain (of dimension 98x113x171) where the ventricle is of interest, and a CT scan of the humeral bone (of dimension 120x110x155). The ground truth segmentation was created for each scan by expert marking. As shown in the plots of Figure 4, our algorithm achieves a similar reduction rate on the number of delineation planes needed to achieve 90% segmentation accuracy.

All our experiments were performed on an Intel Core i7 machine with 8 cores and 16GB RAM. It took on average 12 seconds for the algorithm to select the next plane for the biomedical data sets. Note that this is roughly twice as fast as the state-of-art method [THA11]. The speed-up is attributed to multiple factors, including our RBF-based segmentation using reduced centers, our simpler formulation of the uncertainty field, and the direct plane selection mechanism based on clustering.

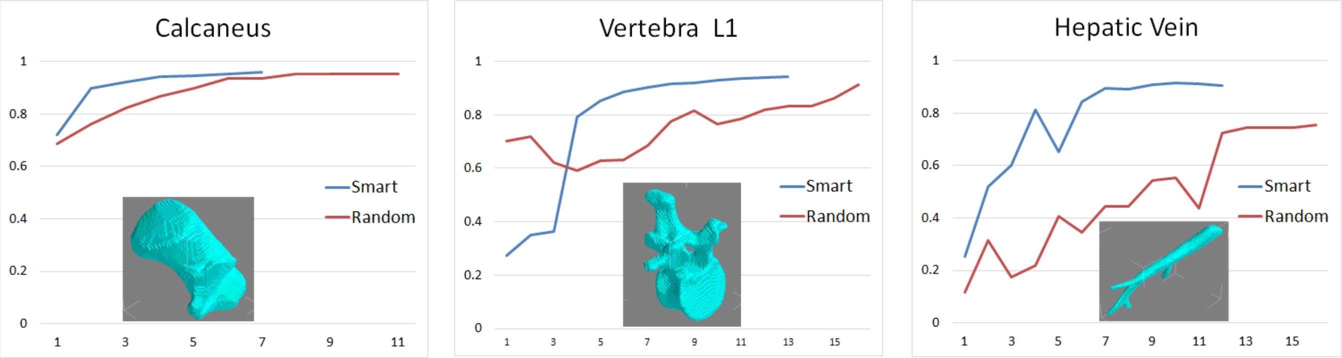

Figure 3: Segmentation accuracy as a function of number of planes produced by the algorithm (blue) and at random (red). X axis is the number of planes; Y axis is Dice coefficient.

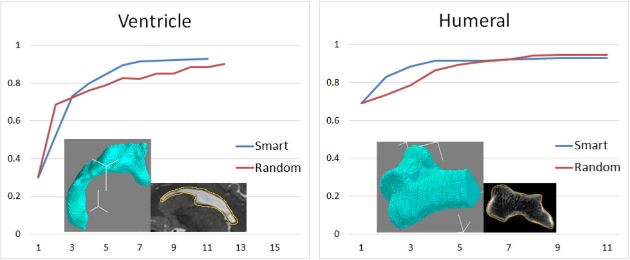

Figure 4: Segmentation accuracy on real-world datasets as a function of number of planes produced by the algorithm (blue) and at random (red). X axis is the number of planes; Y axis is Dice coefficient. Each plot also shows a delineated slice and the ground truth segmentation.

4. CONCLUSION AND DISCUSSION

We reported a novel algorithm for identifying the next plane for user to delineate in the plane-based interactive segmentation paradigm. While being both mathematically and computationally simpler than prior art on this problem, we demonstrated that the method achieves similar reduction with prior art in the number of planes needed to achieve a reasonable segmentation accuracy over random plane selection.

As a future work, we would like to conduct user studies to further evaluate the benefit of using our algorithmically selected planes versus expert-picked planes. We would also like to further improve the efficiency of the algorithm by exploring GPU-based signed distance generation for computing the uncertainty field and further optimizing the RBF interpolation method.

5. REFERENCES

[BM97] Barrett W. A., Mortensen E. N.: Interactive live-wire boundary extraction. *Medical Image Analysis 1* (1997), 331–341.

[CBC*01] Carr J. C., Beatson R. K., Cherrie J. B., Mitchell T. J., Fright W. R., McCallum B. C., Evans T. R.: Reconstruction and representation of 3D objects with radial basis functions. *Proc. of SIGGRAPH* (2001), 67–76.

[EKJX96] Ester M., Kriegel H.-p., Jörg S., Xu X.: A density-based algorithm for discovering clusters in large spatial databases with noise. 226–231.

[LBD*08] Liu L., Bajaj C., Deasy J., Low D. A., Ju T.: Surface reconstruction from non-parallel curve networks. *Comput. Graph. Forum 27*, 2 (2008), 155–163.

[MSM11] McCrae J., Singh K., Mitra N. J.: Slices: a shape-proxy based on planar sections. *ACM Trans. Graph. 30*, 6 (Dec. 2011), 168:1–168:12.

[THA10] Top A., Hamarneh G., Abugharbieh R.: Spotlight: automated confidence-based user guidance for increasing efficiency in interactive 3d image segmentation. In *Proceedings of the 13th international conference on Medical image computing and computer-assisted intervention* (2010), MICCAI'10, pp. 204–213.

[THA11] Top A., Hamarneh G., Abugharbieh R.: Active learning for interactive 3d image segmentation. In *Proceedings of the 14th international conference on Medical image computing and computer-assisted intervention* (2011), MICCAI'11, pp. 603–610.