TREE-METRICS GRAPH CUTS FOR BRAIN MRI SEGMENTATION WITH TREE CUTTING

Ruogu Fang[1], *Yu-hsin Joyce Chen*[2], *Ramin Zabih*[2], *Tsuhan Chen*[1]

[1] Department of Electrical and Computer Engineering, Cornell University, Ithaca, NY. USA
[2] Department of Computer Science, Cornell University, Ithaca, NY. USA

ABSTRACT

We tackle the problem of brain MRI image segmentation using the tree-metric graph cuts (TM) algorithm, a novel image segmentation algorithm, and introduce a "tree-cutting" method to interpret the labeling returned by the TM algorithm as tissue classification for the input brain MRI image.

The approach has three steps: 1) pre-processing, which generates a tree of labels as input to the TM algorithm; 2) a sweep of the TM algorithm, which returns a globally optimal labeling with respect to the tree of labels; 3) post-processing, which involves running the "tree-cutting" method to generate a mapping from labels to tissue classes (GM, WM, CSF), producing a meaningful brain MRI segmentation. The TM algorithm produces a globally optimal labeling on tree metrics in one sweep, unlike conventional methods such as EMS and EM-style geo-cuts, which iterate the expectation maximization algorithm to find hidden patterns and produce only locally optimal labelings. When used with the "tree-cutting" method, the TM algorithm produces brain MRI segmentations that are as good as the Unified Segmentation algorithm used by SPM8, using a much weaker prior. Comparison with the current approaches shows that our method is faster and that our overall segmentation accuracy is better.

Index Terms— brain MRI segmentation, tree-metrics graph cuts, tree cutting, global optimal labeling

1. INTRODUCTION

Computer-aided diagnoses depend on fast, robust algorithms for processing medical images. The advent of magnetic resonance imaging has allowed us to study the brain and diagnose neural diseases as never before; in this light, automatic brain MRI segmentation is crucial for understanding the brain and diseases that affect the brain.

The task of brain MRI segmentation is to segment a large 3D MRI image representing the brain into three types of brain tissue: white matter (WM), gray matter (GM), and cerebrospinal fluid (CSF). Accurate and efficient automatic brain MRI segmentation will provide information critical to diagnosing brain disease in time, reduce the time and labor spent by human experts, and make interpretations of brain MRI images more consistent. However, automating brain MRI segmentation is challenging. Brain MRI images often have non-uniform intensity because there is an MRI bias field: voxels representing grey matter in one region of the image may have different intensities compared to grey matter voxels in a different region of the image. MRI images are also influenced by the partial volume effect; some voxels may have a mixture of several tissue types, making classification difficult. Finally, in order to reduce scan time, many MRI images are intentionally undersampled, resulting in noisy images with aliasing.

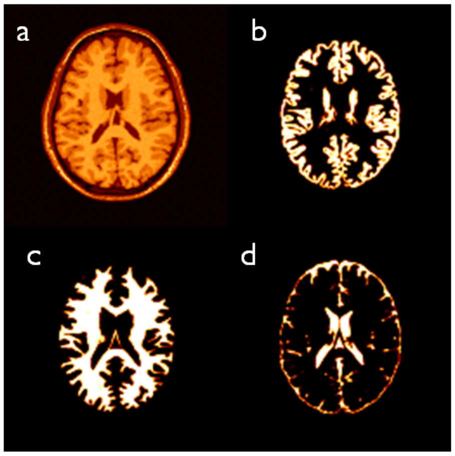

Fig. 1. Brain MRI Segmentation. (a)Brain MRI image (b)Gray Matter (c) White Matter (d) CSF

To address these challenges, we need image-segmentation algorithm that is robust to noise and has provable guarantees on the quality of its outputs. We propose to apply tree-metrics graph cuts (TM) algorithm [1] to segment brain MRI images, and then use our "tree cutting" method to classify the segments as GM, WM or CSF. We will refer to our approach as the TM-TC algorithm, because it applies the tree-metric (TM) algorithm and then the tree-cutting (TC) method.

Our proposed approach's novel aspects include:

- Globally optimal labeling for multiple labels. Conventional approaches such as EMS [2] and EM-style geo-cuts [3] only obtain a locally optimal labeling by using iterations of the expectation-maximization algorithm.
- Low computational cost. The running time of our tree-

978-1-4244-9300-5/10/$26.00 ©2010 IEEE

metric algorithm is $O(\log(k)(g(n)+k))$, for n voxels and k labels (in our case, $n = 4$: GM, WM, CSF and background), where $g(n)$ is the running time of the min-cut algorithm on graph with n nodes.

1.1. Brain MRI segmentation as metric labeling

The brain MRI segmentation problem can be interpreted as an instance of metric labeling; specifically an instance of image denoising. In image denoising, the observations are labels; we want to pick a new label (gray-scale intensity) for each of our voxels. Each of the new values should be similar to the observed one in the original MRI image, and the values of nearby voxels should be similar.

Let the brain 3D MRI image be represented as an undirected weighted graph $G = (V, E)$, where vertices V correspond to voxels and E are edges between neighboring voxels. Let L be the set of labels, and let $f : V \rightarrow L$ be a labeling. Furthermore, let $c(v, f(v)) = w_v d(o(v), f(v))$ (assignment cost function) be the cost for givin label $f(v)$ to object v, where $o(v)$ is the observed label and $d(a, b)$ is a distance on L. The goal is to find a labeling f that minimizes the cost function

$$Q(f) = \sum_{v \in V} w_v d(o(v), f(v)) + \sum_{(u,v) \in E} w_{uv} d(f(u), f(v))$$

We refer to the first summation in $Q(f)$ as the "data term" and the second summation in $Q(f)$ as the "prior term" (or "smoothness term"). As our prior, we want objects connected by an edge in E to have similar labels.

The image denoising problem is NP-hard for arbitrary distance $d(a, b)$. However, if the distance d is actually a "tree-metric", the problem is very tractable; under these circumstances, the TM algorithm returns a globally optimal labeling f and can handle large numbers of labels. The tree-metric d can be represented as a tree of labels, such that the distance between any pair of labels is the length of the path between them in the tree.

2. METHOD

In order to apply the tree-metric graph cuts (TM) algorithm to brain MRI segmentation, we perform pre-processing to create suitable inputs to the TM algorithm, and post-processing on the output segmentation returned by the TM algorithm a to create a meaningful brain segmentation, classifying the tissue at each voxel.

The TM algorithm takes 3 things as input: a brain MRI image, a tree of labels, and a smoothness parameter $\lambda \geq 0$. We generate a tree of labels with agglomerative clustering based on observed gray level intensity.

We apply the TM algorithm to the brain MRI image and the tree of labels, which returns a globally optimal labeling of the brain MRI with respect to the tree of labels. We use a process called "tree cutting" to compute the mapping of labels to tissue classes, completing our brain segmentation.

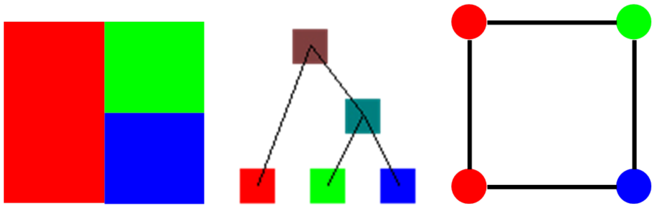

Fig. 2. Example of tree generation. Left: Input synthetic image with three colors. Middle: Generated binary tree from observation. Right: Underlying graph.

Each stage of our approach (TM-TC algorithm)— tree generation, sweep, and pruning— is detailed in this section as follows.

2.1. Tree Generation

We use agglomerative hierarchical clustering to create a tree of labels. For example, in Figure 2, we apply agglomerative hierarchical clustering on the colors at each pixel to generate tree T based on Ward's variance criteria [4]. For brain MRI segmentation, we perform agglomerative hierarchical clustering to create a tree of labels with 256 leaves.

In agglomerative hierarchical clustering, closest pairs of clusters (of labels) in the feature space are repeatedly merged, such that the distance between two clusters C and D reflects the increase in variance of the feature vectors in $C \cup D$ relative to the variance in C and D.

In practice, we implement agglomerative hierarchical clustering in phases. At each phase, we use k-nearest neighbors (measured in Euclidean distance) to compute candidate clusters (using the nearest neighbor data structure [5]) and merge them. If the number of clusters is very large, we use approximate nearest neighbors in our computation instead. When the maximum variance becomes greater than two times of the minimum variance, we stop merging clusters.

2.2. Tree-Metric Algorithm Sweep

Now that we've generated a tree of labels representing the tree-metric distance function d, we apply the TM algorithm to, the 3D brain MRI image, the tree of labels, and the smoothness parameter $\lambda \geq 0$. The TM algorithm will minimize the cost function for a distance d and labeling f:

$$Q(f) = \sum_{v \in V} d(o(v), f(v)) + \lambda \sum_{(u,v) \in E} d(f(u), f(v))$$

The TM algorithm can compute the globally optimal labeling f for the cost function $Q(f)$ in $O(\log(k)(g(n)+k))$ time for n voxels and k labels, where $g(n)$ is the running time of the min-cut algorithm on graph with n nodes. The $\log(k)$

term in the running time assumes that the tree of labels is balanced; [1] mentions ways binarize and balance any arbitrary tree.

Because our label set for brain MRI images contains 256 labels representing gray level intensities, the TM algorithm returns a labeling f with 256 labels, vastly greater than the 4 labels (GM, WM, CSF and background) we want to label the brain with.

2.3. Tree Cutting

The labeling returned by TM algorithm does not directly correspond to the tissue classes in the brain. Therefore we need to map the gray-scale labels to 4 labels: GM, WM, CSF or background. We create this mapping with the "tree-cutting" method. To reduce 256 labels to 4 labels, we "cut" our binary tree of labels at depth $\log_2(4) = 2$; for each node at depth 2 (where the root node is depth 0), their child subtrees now map to the same label (as their ancestor node at depth 2). By cutting the tree of labels at depth 2, we are left with 7 labels; 1 label at the root (depth 0), 2 labels at depth 1, and 4 labels at depth 2. All of the nodes deeper than depth 2 map to one of the labels at depth 2. Finally, map each of the labels shallower than depth 2 (the root, nodes at depth 1) to one of the 4 labels at depth 2 with the closest Euclidean distance.

After the "tree-cutting" procedure, we are left with only the 4 labels at depth 2. It is now easy to assign each of the 4 labels as either GM, WM, CSF or background.

3. RESULTS

We validated our approach on both simulated and clinical brain MRI dataset and compare the performance of our algorithm to that of EMS and EM styled geo-cuts.

3.1. Simulated Brain Database

The first dataset is simulated brain database from Brain Web [6], which contains 20 normal anatomical models of realistic MRI data volumes produced by an MRI simulator.

Our baseline algorithms were the widely used EMS algorithm and EM-style geo-cuts. Both of them are state-of-art brain tissue segmentation methods, yet both of them depend on iterations to approximate optimal segmentation of the brain tissue. Figure 3 shows the comparison of segmentation results of brain MRI using EMS, EM-style geo-cuts and our approach. After finding the absolute number of voxels classified as WM, GM, and CSF, the normalized confusion matrix was created in order to show the accuracy of our results on a percentage basis.

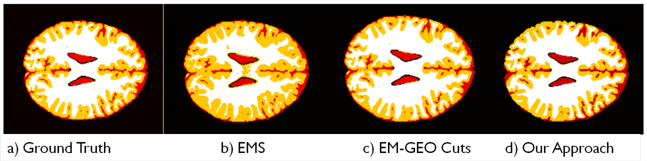

Fig. 3. Segmentation results on simulated brain dataset.a)Original bias-corrected brain MRI data b)EMS c)EM-style geo-cuts d)Our approach (Color: white matter - white, gray matter - gray, CSF - deep gray)

Table 1. Classification accuracy for brain tissue using EMS, EM-style geo-cuts and TMN

Method	All classes	WM	GM	CSF
EMS	85.65%	61.67%	99.58%	95.31%
Geo-cuts	96.43%	99.83%	93.00%	98.60%
Ours	97.21%	99.80%	96.52%	92.39%

We can see that the segmentation of our approach (Fig. 3 d) are smoother: for example, white matter segmentation has less noise than EMS and EM-style geo-cuts, and fine curvatures of gray matter are better preserved. By comparing the overall segmentation accuracy on three brain tissues (excluding non-brain materials), the differences are significant: **EMS**: 85.64%, **EM-style Geo-cuts**: 96.43%, **Our Approach**: 97.21%. More importantly, the running time of EMS and EM-style geo-cuts can go up to hours for entire 3D brain MRI data, due to its iterative nature, while our approach needs only minutes to complete the segmentation. A statistical t-test shows that our approach is better than EMS and EM-style geo-cuts with 99% confidence. The quantitative comparison of segmentation accuracy for three types of brain tissues and the overall accuray are shown in Table 1.

Furthermore, we compare the performance of our approach with popular state of the art brain segmentation software SPM8, which uses "Unified Segmentation" by Ashburner and Frist [7]. The "Unified segmentation" algorithm unifies segmentation and registration into a single probabilistic framework, using atlases for probability maps; unlike TM algorithm, unified segmentation uses a very strong prior and uses iterative computations to converge to a solution. In Table 2, we demonstrate that TM algorithm performs at least as well as unified segmentation under the dice metric on BrainWeb data with 3% simulated noise.

The Dice metrics we use in the evaluation is

$$\text{Dice metric} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}}$$

TM algorithm runs in one minute on one core of a 2.16GHz Intel Core Duo machine with 2GB of RAM, while unified segmentation runs in 21 minutes.

3.2. Clinical Brain MRI Data

We also tested our brain MRI segmentation algorithm on real clinical brain MRI data with corrected bias field. The dataset is T1-weighted brain MRI with 256 × 256 pixels matrix. The dataset was acquired from University of San Fracisco Medical

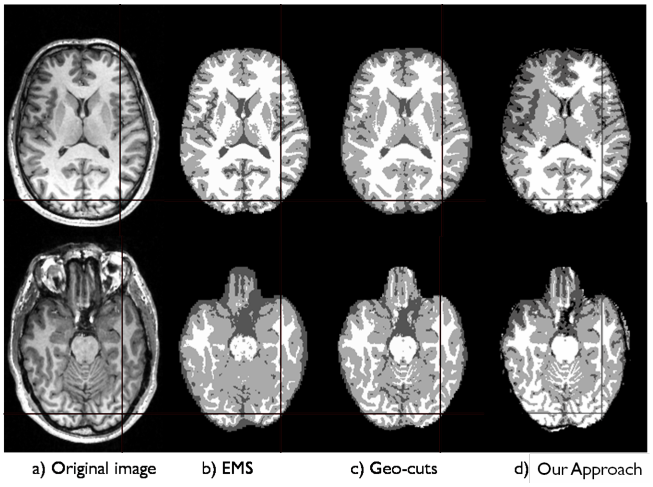

Fig. 4. Segmentation results on clinical brain dataset.a)Original bias-corrected brain MRI data b)EMS c)EM-style geo-cuts d)Our approach (Color: white matter - white, gray matter - gray, CSF - deep gray)

School. The dataset has more ambiguity than simulated data due to instrumental noise and sampling inaccuracies. Figure 4 shows the comparison of segmentation results on real brain MRI data, which clearly shows the improvement from our algorithm on the segmentation of fine structures of white matter and CSF in the central part of the brain, where both EMS and geo-cuts fails to represent the complete white matter structures.

4. CONCLUSION

In this paper we presented a novel approach to apply graph cuts algorithm with tree metrics using pruning to the medical image segmentation problem and compare our results with the state-of-art methods. Our approach, which is efficient in computation and globally optimized in the tree-metrics labeling, performs better than the conventional approaches in terms of the overall accuracy, and avoids the iterated method which is computational intense.

There are several interesting directions for future work on this problem. To learn the proper structure of the tree, for instance, the optimal degree of the tree and the clustering criterion, will improve the segmentation result for arbitrary number of segments in the output. Moreover, since the computational time is mostly spent on the agglomerative clustering stage, where ANN is applied to find the nearest neighbors, we could explore better methods to generate the tree with higher efficiency.

Table 2. Compare performance in terms of classification accurcy under dice metric of TM algorithm and SPM8's Unified Segmentation on BrainWeb data with 3% simulated noise.

(a) Tree-metric algorithm.

Dice metric	T1	T2	PD
Grey Matter	0.957	0.902	0.893
White Matter	0.967	0.922	0.940
Whole Brain	0.980	0.954	0.943

(b) Unified Segmentation full model.

Dice metric	T1	T2	PD
Grey Matter	0.932	0.883	0.872
White Matter	0.961	0.916	0.923
Whole Brain	0.977	0.967	0.957

5. REFERENCES

[1] Pedro F. Felzenszwalb, Gyula Pap, va Tardos, and Ramin Zabih, "Globally optimal pixel labeling algorithms for tree metrics," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[2] Koen Van Leemput, Frederik Maes, Dirk Vandermeulen, and Paul Suetens, "Automated model-based tissue classification of mr images of the brain," 1999.

[3] Nikos Paragios, Rachid Deriche, and Sophia Antipolis Cedex, "Geodesic active regions for supervised texture segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 1999, pp. 926–932.

[4] Jr. Ward, "Hierarchical grouping to optimize an objective function.," *Journal of the American Statistical Association*, vol. 58, pp. 236–244, 1963.

[5] Sunil Arya, David M. Mount, Nathan S. Netanyahu, Ruth Silverman, and Angela Y. Wu, "An optimal algorithm for approximate nearest neighbor searching in fixed dimensions," in *ACM-SIAM SYMPOSIUM ON DISCRETE ALGORITHMS*, 1994, pp. 573–582.

[6] Chris A. Cocosco, Vasken Kollokian, Remi K.-S. Kwan, and Alan C. Evans, "Brainweb: Online interface to a 3d mri simulated brain database," *NeuroImage*, vol. 5, pp. 425, 1997.

[7] John Ashburner and Karl J. Friston, "Unified segmentation.," *NeuroImage*, vol. 26, no. 3, pp. 839–851, July 2005.