Automatic Subcortical Segmentation Using a Contextual Model

Jonathan H. Morra[1], Zhuowen Tu[1], Liana G. Apostolova[1,2], Amity E. Green[1,2], Arthur W. Toga[1], and Paul M. Thompson[1]
[1]Laboratory of NeuroImaging and [2]Dept. of Neurology, UCLA School of Medicine

Center for Computational Biology (CCB)

ABSTRACT

Automatically segmenting subcortical structures in brain images has the potential to greatly accelerate drug trials and population studies of disease. Here we propose an automatic subcortical segmentation algorithm using the auto context model (ACM). Unlike many segmentation algorithms that separately compute a shape prior and an image appearance model, we develop a framework based on machine learning to learn a unified appearance and context model. We trained our algorithm to segment the hippocampus and tested it on 83 brain MRIs (of 35 Alzheimer's disease patients, 22 with mild cognitive impairment, and 26 normal healthy controls). Using standard distance and overlap metrics, the auto context model method significantly outperformed simpler learning-based algorithms (using AdaBoost alone) and the FreeSurfer system. In tests on a public domain dataset designed to validate segmentation, our new algorithm also greatly improved upon a recently-proposed hybrid discriminative/generative approach, which was among the top three that performed comparably in a recent head-to-head competition.

Introduction and Methods

- We want to solve $Y^* = \arg\max p(Y|X) = \arg\max p(X|Y)p(Y)$
- Previous subcortical algorithms focus either on strong shape models, $p(Y)$ (such as atlas models) or appearance information, $p(X|Y)$, (image intensities), such as snakes
- Can we formulate a model based on both simultaneously, so we instead solve directly for the posterior $p(Y|X)$?

- Define $\mathbf{P}^{(0)}(i)$ as the posterior distribution at voxel i
- Define N_i as a general neighborhood around voxel i
- We then want to solve the following equation by iterating n

$$p^{(n)}(y_i \mid X(N_i), \mathbf{P}^{(n-1)}(N_i)) \rightarrow p(y_i \mid X) = \int p(y_i, y_{-i} \mid X) dy_{-i}$$

Given a set of training images together with their label maps, $S = \{(Y_j, X_j), j = 1..m\}$: For each image X_j, construct probability maps $\mathbf{P}_j^{(0)}$, with a distribution (possibly uniform) on all the labels. For $t = 1, ..., T$:

- Make a training set $S_t = \{(y_{ji}, X_j(N_i), \mathbf{P}_j^{(t-1)}(N_i)), j = 1..m, i = 1..n\}$.
- Train a classifier on both image and context features extracted from $X_j(N_i)$ and $\mathbf{P}_j^{(t-1)}(N_i)$ respectively.
- Use the trained classifier to compute new classification maps $\mathbf{P}_j^{(t)}(i)$ for each training image X_j.

The algorithm outputs a sequence of trained classifiers for $p^{(n)}(y_i|X(N_i), \mathbf{P}^{(n-1)}(N_i))$

- We can prove that the error is decreasing with each iteration of ACM

$$\varepsilon_{t-1} = -\sum_i \log \mathbf{P}^{(t-1)}(i)(y_i) \quad \varepsilon_t = -\sum_i \log p^{(t)}(y_i \mid X_i, \mathbf{P}^{(t-1)}(i))$$

- If the algorithm chooses no features based on the previous posterior, then we get $\varepsilon_t = \varepsilon_{t-1}$
- However any effective classifier (such as AdaBoost) will choose features that minimize a convex error function
- Therefore, $\varepsilon_t \leq \varepsilon_{t-1}$

Segmentation Definitions

- $Precision = \frac{A \cap B}{B}$
- $Recall = \frac{A \cap B}{A}$
- $RelativeOverlap = \frac{A \cap B}{A \cup B}$
- $SimilarityIndex = \frac{A \cap B}{\frac{A+B}{2}}$
- $Mean = avg_{a \in A}(min_{b \in B}(d(a,b)))$
- $H_1 = max_{a \in A}(min_{b \in B}(d(a,b)))$
- $H_2 = max_{b \in B}(min_{a \in A}(d(b,a)))$
- $Hausdorff = \frac{H_1 + H_2}{2}$

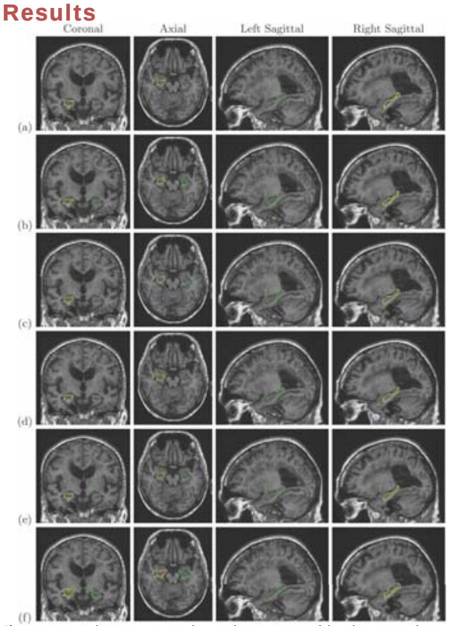
We defined a variety of error metrics using the following definitions: A, the manually segmented region of interest (ROI), B, the automatically segmented ROI, and $d(a,b)$, the Euclidean distance between points a and b. Note that we are using asymmetrized Hausdorff distance.

Hippocampus Results

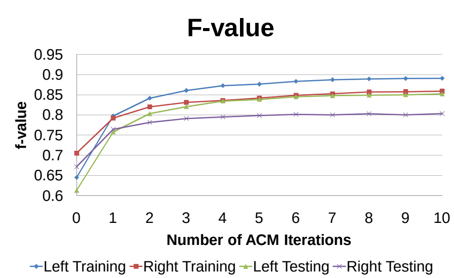
	Prec.	Rec.	R.O.	S.I.	Haus.	Mean
L Train	0.914	0.868	0.802	0.890	2.96	0.00204
R Train	0.883	0.836	0.751	0.857	3.85	0.00331
L Test	0.860	0.845	0.739	0.849	3.68	0.00411
R Test	0.857	0.750	0.656	0.785	4.61	0.00370
L Free	0.587	0.878	0.543	0.700	5.44	0.432
R Free	0.588	0.917	0.558	0.713	5.04	0.271

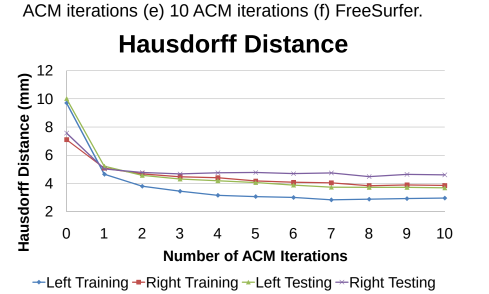
	Prec.	Rec.	R.O.	S.I.	Haus.	Mean
L Train	28.90%	49.35%	74.95%	42.57%	-69.53%	-83.97%
R Train	10.92%	35.89%	42.82%	25.54%	-45.82%	-48.07%
L Test	36.08%	42.12%	73.85%	43.22%	-63.27%	-80.85%
R Test	13.81%	27.14%	34.66%	20.90%	-39.13%	-52.75%

Precision (Prec.), recall (Rec.), relative overlap (R.O.), similarity index (S.I.), Hausdorff distance (Haus.; in mm), and mean distance (in mm) are reported for the left (L) and right (R) training set, testing set, and FreeSurfer (Free). The bottom table reports the percent change from 0 ACM iterations to 10 ACM iterations.



Hippocampal segmentations improve with the number of ACM iterations. (a) ground truth manual segmentation by an expert (b) 0 ACM iterations (c) 1 ACM Iteration (d) 4 ACM iterations (e) 10 ACM iterations (f) FreeSurfer.





Caudate Results

Caudate segmentation based on the Grand Challenge segmentation workshop from last year's MICCAI

	OE	Score	VD	Score	AD	Score	RMSD	Score	MD	Score	Total
No ACM	40.84	74.21	-23.93	58.30	1.22	57.89	2.53	57.45	17.33	50.62	59.17
With ACM	33.34	78.26	-10.60	76.03	0.97	75.51	1.52	77.31	13.67	59.78	73.38

Volumetric overlap error (%, OE), relative absolute volume difference (%, VD), average symmetric surface distance (mm, AD), RMS symmetric surface distance (mm, RMSD), and maximum symmetric surface distance (mm, MD) are reported along with a "score" (relative to other competitors), and a total score (also relative to other competitors).

Grant support for this work was provided by the National Institute for Biomedical Imaging and Bioengineering, the National Center for Research Resources, National Institute on Aging, the National Library of Medicine, and the National Institute for Child Health and Development (EB01651, RR019771, HD050735, AG016570, LM05639 to P.M.T.) and by the National Institute of Health Grant U54 RR021813 (UCLA Center for Computational Biology). L.G.A. was also supported by NIA K23 AG026803 (jointly sponsored by NIA, AFAR, The John A. Hartford Foundation, the Atlantic Philanthropies, the Starr Foundation and an anonymous donor) and NIA P50 AG16570.