Descending Variance Graphs for Segmenting Neurological Structures

George Stetten, Cindy Wong, Vikas Shivaprabhu, Ada Zhang, Samantha Horvath,
Jihang Wang, John Galeotti, Vijay Gorantla, Howard Aizenstein

University of Pittsburgh, Departments of Bioengineering, Psychiatry, and Surgery.
Carnegie Mellon University, Robotics Institute.
Corresponding author: stetten@andrew.cmu.edu

Abstract— **We present a novel and relatively simple method for clustering pixels into homogeneous patches using a directed graph of edges between neighboring pixels. For a 2D image, the mean and variance of image intensity is computed within a circular region centered at each pixel. Each pixel stores its circle's mean and variance, and forms the node in a graph, with possible edges to its 4 immediate neighbors. If at least one of those neighbors has a lower variance than itself, a directed edge is formed, pointing to the neighbor with the lowest variance. Local minima in variance thus form the roots of disjoint trees, representing patches of relative homogeneity. The method works in *n*-dimensions and requires only a single parameter: the radius of the circular (spherical, or hyperspherical) regions used to compute variance around each pixel. Setting the intensity of all pixels within a given patch to the mean at its root pixel significantly reduces image noise while preserving anatomical structure, including location of boundaries. The patches may themselves be clustered using techniques that would be computationally too expensive if applied to the raw pixels. We demonstrate such clustering to identify fascicles in the median nerve in high-resolution 2D ultrasound images, as well as white matter hyperintensities in 3D magnetic resonance images.**

Keywords-component; image analysis, segmentation, graph theory, noise reduction

I. INTRODUCTION

A central challenge in the segmentation of medical images is extracting useful anatomical information from nonhomogeneous structures in the presence of noise. In our recent work, we have developed what we call the *Shells and Spheres* (SaS) framework to enable a statistical approach to segmentation that scales the sample size for intensity to the particular object being segmented. We use the word *sphere* for convenience, since the approach actually works in *n*-dimensions, and by *sphere* we also mean the entire volume, not just a spherical surface. This is analogous to what Attali describes as *balls* for extracting skeletons from predetermined shapes [1]. In the SaS framework, we define an integer *r* as the radius in units of inter-pixel distance, assuming an isotropic image grid. A *shell* of radius *r* centered at pixel location **x** contains any pixel whose distance from **x** rounds to *r*. A sphere of radius *r* is the union of all shells with radii less than or equal to *r*. These definitions extend to *n*-dimensional shells and spheres.

Spheres centered at every voxel may grow or shrink by adding or deleting an outer shell, performing incremental, and thus efficient, computation of mean and variance of pixel intensity of the pixels within a sphere. For spheres that extend beyond the boundaries of the image, only pixels within the image are used to compute mean and variance. Thus, no assumption is made about the value of pixels outside the image. This is an advantage over standard methods such as convolution, which generally require assumptions to be made about the values of pixels outside the image.

We have developed a number of algorithms using the SaS framework and applied them to medical images [2-6], for example, to analyze the shape of the amygdala in MRI images of the brain [3]. These have in general allowed the radius to vary with pixel location and have looked at relationships between the statistics gathered from adjoining spheres on opposite sides of object boundaries. We present here a simpler algorithm based on the same framework, which uses single radius for all spheres and creates a graph structure based on comparisons of variance between spheres centered on neighboring pixels. The single radius of all spheres is the parameter in the initial algorithm.

II. DEFINITION OF ALGORITHM

Given the integer radius *r* of spheres centered at every pixel in the image, mean and variance are computed for intensity within each sphere, and these values assigned to each corresponding central pixel. The spheres centered at neighboring pixels overlap considerably, especially as *r* is permitted to grow.

A comparison is made of the variance at each pixel **x** with that of each of its neighboring pixels (4-connected in 2D, 6-connected in 3D). If any of its neighbors has a lower variance, pixel **x** is made to "point" to the neighbor with the *lowest* variance. If
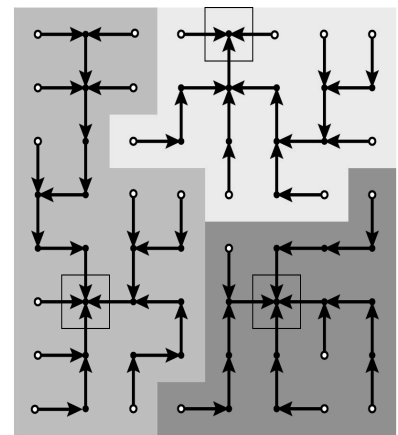
Fig. 1. Directed edges (arrows) between pixels (circles) representing decreasing intensity variance of spheres centered on those pixels. Three disjoint trees form 3 patches (different level of gray) each with its root (square) and sources (open circles).

pixel **x** has a lower variance than any of its neighbors, it does not point to any other pixel.

The situation is depicted in Fig. 1, which shows the pixels in an 8× 8 two-dimensional image as small circles (either filled or empty). Arrows are present between neighbors, such that each pixel either points at one of its neighbors, or, in certain cases (within the square boxes) the pixel does not point to any neighbor at all.

Using the terminology of graph theory [7], we have created a graph made up of *nodes* (pixels) connected by *edges* (the arrows between neighboring pixels). Our graph, which we call the *Descending Variance Graph* (DVG), is a *simple directed acyclic graph*. It is *simple* because there are no *loops* (edges between a node and itself), and because there exists at most one edge between any two nodes. A DVG is *directed* because each edge is oriented from one node, pixel **x**, to another node, pixel **y**, as shown by the arrows in Fig. 1. The edge, $e = (\mathbf{x}, \mathbf{y})$ is said to be directed from **x** *to* **y**. (Note that we use bold to denote our nodes, because, in addition to being in a graph, they also inhabit the *n*-dimensional domain of an image.) The node **x** is said to be the *direct predecessor* of **y**, and conversely **y** is the *direct successor* of **x**. If a path consisting of multiple directed edges leads from node **u** to node **v**, then **u** is said to be simply a *predecessor* of **v**, and **v** a *successor* of **u**. The DVG is *acyclic* because no paths exist by which a node may be the successor to itself.

The number of edges reaching a node is the node's *degree*. The DVG is allowed to branch such that a given node can be the *direct successor* of multiple other nodes but the *direct predecessor* of only one other node. This can be formally expressed by saying that a node in the DVG may have an *indegree* of greater than 1,

$$\deg^-(x) = 0, 1, 2, 3, 4...$$ (1)

but an *outdegree* of at most 1.

$$\deg^+(x) = 0, 1$$ (2)

A *sink* is a node with no successors and thus an outdegree $\deg^+(\mathbf{x}) = 0$. A *source* is a node with no predecessors and thus an indegree $\deg^-(\mathbf{x}) = 0$.

Since each node in our DVG can have at most one direct successor, no two directed paths from a node can reconverge at a successor node. Thus a DVG consists of one or more of a particular kind of directed acyclic graphs called a *directed tree*. Since any particular directed tree in the DVG has edges directed towards a single *sink*, it is more specifically called a *rooted tree*, with the *root* node being the only *sink* in that tree. The sink of a given tree in a DVG represents the node with lowest variance for that tree. Since the trees are *disjoint* (their nodes do not overlap) the DVG for the whole image is, itself, called a *forest*.

Looking back at Fig. 1, we see that this particular forest contains 3 disjoint trees, each representing a region or "patch" depicted by a different level of grayscale, and each

with its own unique root (square box). Nodes that are sources (open circles) tend to be at the outsides of the patches, but not always. Also note that a root (in a 2D image) will usually have an indegree of 4, though it is possible for a root to have an indegree of 3 or less.

Once mean and variance have been computed for spheres of a given radius *r* throughout the image, comparisons are made between the variances of neighboring pixels and the edges of the graph formed. Roots are identified by having an outdegree of zero, and a recursive flood-fill operator used to connect a given root to all the nodes of its tree. Except for the initial computation of mean and variance, the algorithm is an O(*n*) operation, where *n* is the number of pixels.

The resulting trees represent relatively homogenous regions. Each root is a local minimum of variance, whose sphere is generally within its own relatively homogenous patch, surrounded by pixels with higher variances that "drain" into it. Each patch is separated from neighboring patches by a ridge in variance, whose spheres overlap boundaries between one relatively homogenous region and another. The system is self-normalizing, with the only parameter being radius *r*.

We next present some preliminary results using two different applications in neurological imaging.

III. RESULTS IN 2D WITH ULTRASOUND

Nowhere is automated analysis more problematic than in ultrasound, where high noise and variability in image intensity make reliable segmentation difficult. We are currently using high-frequency ultrasound to monitor regeneration after hand transplant surgery and peripheral nerve injury, and to determine the efficacy of experimental drug therapies [8]. Accurate identification of changes in
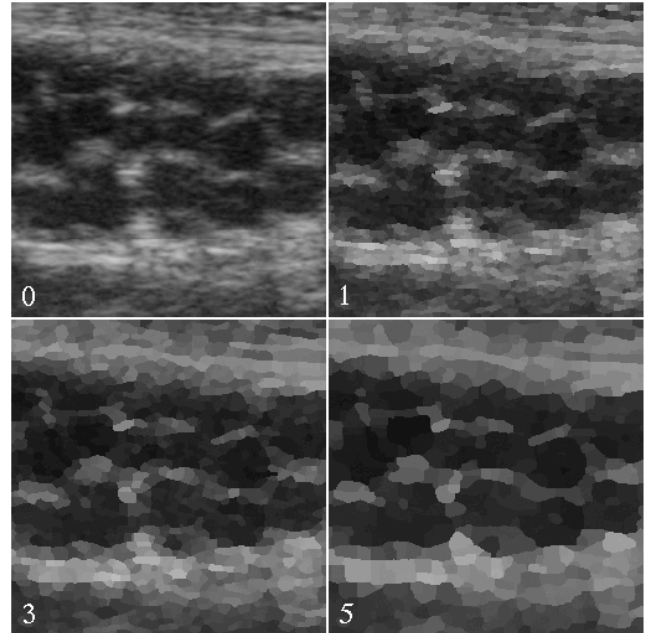

Fig. 2 Descending variance graphs applied at various radii (in pixels) to an ultrasound image of the median nerve.

nerve anatomy such as edema, myelin debris, or fascicular/axonal changes could help objectively diagnose nerve injury or monitor nerve regeneration after trauma or surgery.

Cross-sectional ultrasound scans of a human medial nerve were performed on a normal volunteer (author) using a VisualSonics Vevo 2100 scanner (with consent for human use) operating at 50MHz. Penetration depth was approximately 5mm with 30μm resolution. Nerve fascicles were easily identified in the cross-sectional scans. A 256× 256 pixel 8-bit grayscale image was used as input to the algorithm.

The results are shown in Fig. 2. The original ultrasound image is labeled 0. The images labeled 1, 3, and 5, show the results of applying the DVG algorithm with variances computed using spheres of those respective radii. In each of these three images, the mean intensity of all pixels within a given patch was set to the mean of its root pixel. The operation preserves anatomical structure, including edge location, while significantly reducing noise.

Since a goal of creating the patches is to permit further analysis to proceed with fewer elements than raw pixels, we examined the number of patches as a function of radius for the ultrasound image and for an image containing pure

Number of Patches vs. Radius
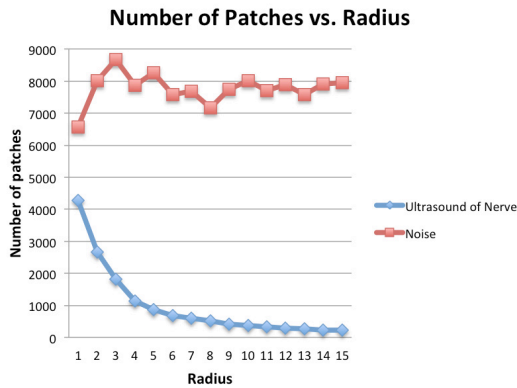
Fig. 3 The number of patches in an ultrasound image of a nerve drops monotonically with the radius of the spheres used to compute variance. This is not true for an image with only noise.

Gaussian noise (mean 138, standard deviation 10). Both were 256 × 256 pixel 8-bit grayscale images, containing 65536 pixels. At a radius of 1, both images yielded approximately 10% as many patches as pixels (see Fig. 3). But whereas the ultrasound image showed a continuing monotonic reduction in the number of patches with increasing radius, the image of pure noise showed no such
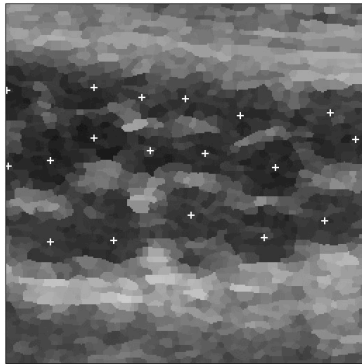

Fig. 4 Fascicles identified by roots of trees consisting of clustered DVG patches.

reduction. Evidently the patches are organizing anatomical content.

We next clustered DVG patches to identify fascicles in the ultrasound image. For each root, we identified all other roots within a radius of 18 pixels. This parameter was chosen to match the general size of a fascicle. We calculated the variance of the intensities of these patches and connected neighboring patches with directed edges, as we had done previously with pixels. The roots of these trees are shown in Fig. 4. Only those below a certain threshold of intensity are shown, clearly identifying the centers of nerve fascicles. Admittedly, we have introduced two new parameters in this step: expected fascicle radius and intensity. However, we consider this acceptable, since some prior information is required to identify particular structures.

IV.   RESULTS IN 3D WITH MRI

To demonstrate the formation of DVG patches in 3D, we applied the algorithm described in section II to MRI images of the brain, to identify periventricular white matter hyperintensities. White matter hyperintensities (WMHs) are seen on T2-weighted MR images in up to 50% of individuals over the age of 65 [9]. These hyperintense lesions are associated with underlying ischemic demyelination and gliosis [10], and have been associated with a number of prevalent age-related conditions including cognitive impairment, depression, and mobility impairment [11]. Image processing methods for quantifying and localizing WMHs would facilitate their further development as useful biomarkers.
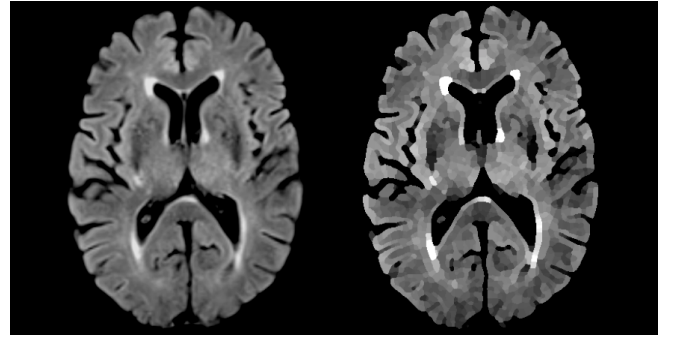
Fig. 5 Original MRI image **(left)**, DVG patches **(right)**.
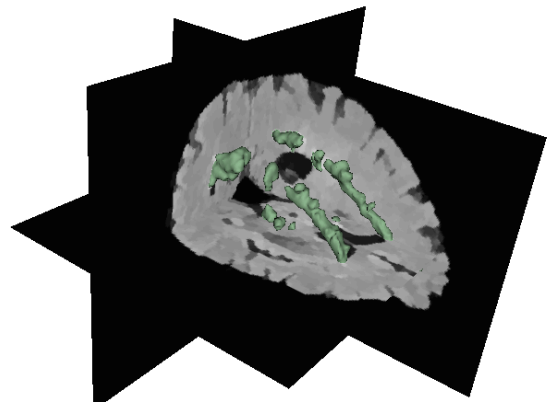
Fig. 6 Patches surface-rendered in 3D over MRI slices.

Figure 5 (left) shows a T2 fluid-attenuated inversion recovery (FLAIR) image of a patient with periventricular WMHs. Images were collected on a 3T Siemens Trio TIM scanner. The 3D version of the DVG algorithm was applied with a radius of 1, and a slice at the same location through the resulting 3D patches is shown in Fig. 5 (right). A threshold for intensity was applied and the patches (which are themselves 3D regions) were overlaid using surface rendering on orthogonal slices of a corresponding T1 image (Fig. 6). Rendering in this manner naturally clusters neighboring patches into larger 3D regions, by showing only the outer surface of their union.

V. DISCUSSION

Our method has some similarities to other graph-based methods for image segmentation, which generally treat the entire image as a graph, with each pixel as a node connected to its neighbors by edges. Felzenszwalb and Huttenlocher's graph-based segmentation [12] begins with each pixel in its own sub-graph, and then joins sub-graphs if the edges between them have a low enough weight (i.e. the pixels they connect are similar). Edges are examined in order of weight, so that "easy" connections are made first, leaving the more ambiguous decisions for later in the algorithm. The normalized cuts method [13] begins with the entire image as a single fully connected, un-segmented graph. The edge strength is determined as a function of the intensity difference between the two pixels it connects, compared to nearby pixels. The cut-cost of each edge is computed from inter- and intra-sub-graph similarities, along with the edge strength. Segmentation is accomplished by computing the minimum cut (the set of edges with the smallest combined weight that will split a graph when removed). Our method may also be considered a form of the so-called *superpixel* [14], in which over-segmentation based on local similarities preserves most of the structure while reducing the dimensionality of the overall image (and reducing noise).

VI. CONCLUSION

The contribution of our work, we believe, is to provide a simple and rapid method to reduce the noise while preserving meaningful structure in *n*-dimensional images. The initial step requires only a single parameter: the radius of the spheres used to compute mean and variance of pixel intensity. The technique is inherently *n*-dimensional. We have demonstrated it in neurological images in 2D and 3D. We are currently further exploring ways to cluster these patches with guidance of priors using higher-level graphs as well as other techniques.

ACKNOWLEDGMENT

REFERENCES

1. Attali, D.: Computing and Simplifying 2D and 3D Continuous Skeletons, Computer Vision and Image Understanding, Vol. 67, No. 3, pp. 261–273. (1997)
2. Cois, A., Rockot, K., Galeotti, J., Tamburo, R., Gottlieb, D., Mayer, J., Powell, A., Sacks, M., Stetten, G.: Automated segmentation of the right heart using an optimized shells and spheres algorithm, International Society of Biomedical Imaging, Arlington VA. (2007)
3. Tamburo, R., Siegle, G., Stetten, G., Cois, C., Reynolds, C., Aizenstein, H.: Amygdala Morphometry in Late-Life Depression, special issue on Neuroimaging in Geriatric Psychiatry, in International Journal of Geriatric Psychiatry, Vol. 24, pp. 837–846. (2009)
4. Stetten, G., Horvath, S., Galeotti, J., Shukla, G., Chapman, B.: Image segmentation using the student's t-test on adjacent spherical populations of pixels. SPIE Medical Imaging, paper #7623-125. (2010)
5. Cois, A., Galeotti, J., Tamburo, T., Sacks, M., and Stetten, G.: Shells and Spheres: An *n*-Dimensional Framework for Medial-Based Image Segmentation, International Journal of Biomedical Imaging, vol. 2010, Article ID 980872. (2010)
6. V. Shivaprabhu, Automated method for n-dimensional shape detection based on medial image features, Masters Thesis, Department of Bioengineering, University of Pittsburgh. (2010)
7. West, D.B.: Introduction to Graph Theory, 2nd ed. Prentice Hall, Upper Saddle River, NJ. (2001)
8. Tuffaha, S., Quigley, M., Ng, T., Gorantla, V.S., Shores, J.T., Pulikkottil, B., Shestak, C., Brandacher, G., Lee, W.P.: The effect of chondroitinase on nerve regeneration following composite tissue allotransplantation. J Hand Surg Am., 36(9):1447-52 (2011)
9. Longstreth Jr, W. T., Manolio, T., Arnold, A., Burke, G., Bryan, N., Jungreis, C., Enright, P., et al.: Clinical Correlates of White Matter Findings on Cranial Magnetic Resonance Imaging of 3301 Elderly People: The Cardiovascular Health Study. *Stroke*, *27*(August), 1274–1282, (1996)
10. Fazekas, F., Kleinert, R., Offenbacher, H., Schmidt, R., Kleinert, G., Payer, F., Radner, H., et al. Pathologic correlates of incidental MRI white matter signal hyperintensities. Neurology, 43(9), 1683–1683. doi:10.1212/WNL.43.9.1683 (1993)
11. Ovbiagele, B., & Saver, J. L. Cerebral white matter hyperintensities on MRI: Current concepts and therapeutic implications. Cerebrovascular Diseases, (2-3). doi:10.1159/000093235 (2006)
12. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient graph based image segmentation, Int. J. Comput. Vision, vol. 59, pp.167-181. (2004)
13. Shi, J., Malik, J.: Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell., vol. 22, no. 8, pp.888-905. (2000)
14. Ren, X.; Malik, J.: Learning a classification model for segmentation. Ninth IEEE International Conference on Computer Vision, vol., no., pp.10-17 vol.1, 13-16. (2003)