Ridge flow models for image segmentation

David Eberly and Stephen M. Pizer

University of North Carolina, Department of Computer Science
Chapel Hill, North Carolina 27599–3175

ABSTRACT

In this paper we introduce a new algorithm for segmentation of medical images of any dimension. The segmentation is based on geometric methods and multiscale analysis. A sequence of increasingly blurred images is created by Gaussian blurring. Each blurred image is segmented by locating its ridges, decomposing the ridges into curvilinear segments and assigning a unique label to each, and constructing a region for each ridge segment based on a flow model which uses vector fields naturally associated with the ridge finding. The regions from the initial image are leaf nodes in a tree. The regions from the blurred images are interior nodes of the tree. Arcs of the tree are constructed based on how regions at one scale merge via blurring into regions at the next scale. Objects in the image are represented by unions and differences of subtrees of the full tree. The tree is used as input to a visualization program which allows the user to interactively explore the hierarchy and define objects. Some results are provided for a 3–dimensional magnetic resonance image of a head.

1. INTRODUCTION

Segmentation of images into meaningful regions has been an important area in medical image analysis. Rapid, accurate, automated identification and measurement of anatomical objects in medical images, such as organs and tumors, is a desirable goal. Three typical applications requiring segmentation are display from regions of interest, measurement of object information, and atlas identification. A 3–dimensional image contains a large amount of data, so rendering of the complete data set can be time consuming. Selecting regions of interest will reduce the amount of data to process and will reduce the rendering time. Also, objects that the user wants to display may be obscured by other irrelevant objects in the data. To display a single unobscured object, the selection of a region of interest containing only the object becomes an important issue. A good segmentation algorithm should produce a region just slightly larger than the object. Display of an object gives a good qualitative description, but quantitative measurements might also be required, for example, the volume of a tumor. Current clinical tools allow radiologists to "hand–segment" objects in images by drawing contours around the objects of interest on a slice–by–slice basis. Volume estimates can be made by calculating the enclosed area on each slice and then adding up the areas. This process is slow and tedious. A good segmentation algorithm should allow the radiologists to make more accurate measurements with far less cost of user time. Finally, if the objects in an image are precisely located and represented by an abstract structure such as a tree or graph, then the objects can be identified by matching their representations against an atlas of representations stored as a data base of previously segmented objects. The resulting naming can be fed back to allow a model to improve the segmentation. It can allow object–specific measurements to be made, including deviation from normality.

Automated segmentation processes are not as adept as humans at recognizing objects. The chances are small that the segmentation process will correctly associate with each object a single primitive region. Realizing this limitation, a good design goal is to create many small primitive regions and attempt to semantically link them into a hierarchy. The semantic rules can be based on such constructs as pyramids[1,2], the difference–of–low–pass transform[3], tree matching[4], or multiscale methods[5,6,7,8] where the scale space is constructed via Gaussian blurring. A goal of this paper is to show how such hierarchies can be used for interactive object definition. The hierarchies also have the potential for object recognition using tree matching algorithms.

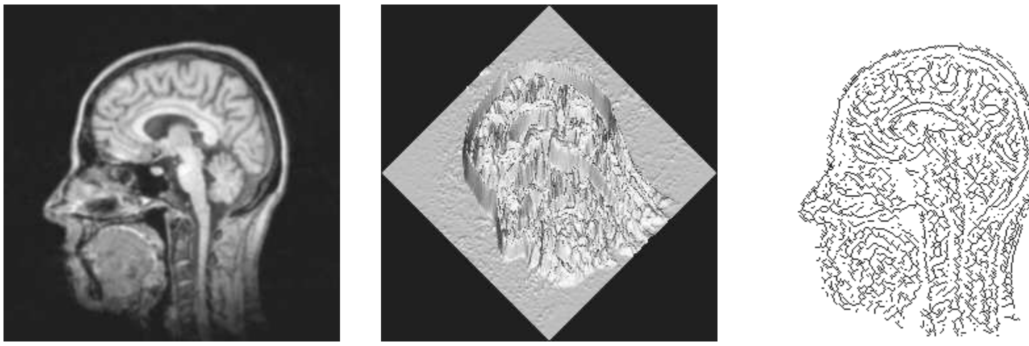
In Section 2 we present a general method for segmenting medical images which is based on the height ridge definition[9]. Ridges of an image appear to be associated with regions of perceptual importance. The constructed hierarchies are based on annihilation of ridges through increasing scale. The leaf nodes of the hierarchy represent small, primitive regions of an image. These regions are constructed by using a ridge flow model. The ridges are segmented into curvilinear segments. A ridge segment is assigned a region of points, each point being the initial value for a flow line in the image plane such that the flow terminates at the ridge segment. The flow process is a generalization of the reverse watershed method, but in our case the flow line directions are determined not by the gradient of intensity, but by the gradient of a function which is in some sense natural to the ridge construction.

The interior nodes of the hierarchy relate the primitive regions in a way that reflects the natural object structure of the image. Each level of nodes in the hierarchy represents primitive regions of an image which has been blurred to a selected scale using variable conductance diffusion[10,11,12]. The idea is that small regions at one scale are blurred into a single larger region at a larger scale. Such small regions will be linked to a parent node at the next level in the hierarchy. As the scale increases without bound, the blurring process yields a constant image. Thus, at a large scale, all nodes will eventually have a single ancestor–the root of the tree. Anatomical objects are obtained as unions of subtrees of the hierarchy. A detailed description of the hierarchy construction is given in Section 3.

We have implemented the algorithms for any dimension $n \geq 1$, not just for the usual 2 and 3 dimensions. The image and corresponding hierarchy are used as input to an object definition and visualization tool, called the Magic Crayon[13,14], which allows the user to interactively define objects by rapidly traversing the hierarchy and displaying the regions of interest with color overlays. A brief discussion of the Magic Crayon is given in Section 4. The results of the segmentation applied to a 3–dimensional image are provided.

2. RIDGE FLOW MODELS

For many types of medical images, the intensity values tend to be large at pixels which are centrally located in objects, and tend to be small near boundaries of objects. For example, in the MR image of Figure 1 the scalp tends to show up as bright along the center of the scalp, but darker as you move towards the scalp boundaries. Similarly, the brain stem tends to show up as bright along its center, but darker near its boundaries. Other objects appear dark on a light background, but since these can be thought of as light objects on a dark background in a negative of the image, we will consider for now only light objects on a dark background. Notice how the bright centers of objects show up as *ridges* on the surface. Let the image be represented by a smooth function $f : D \subset \mathbb{R}^n \to \mathbb{R}$, where D is a rectangular solid. A *ridge flow model* is a method for segmenting D into a disjoint union of primitive regions which are determined by the ridge structure of the graph of f. There are numerous definitions for ridges[9], but the one we use to illustrate the ideas is based on the height definition, a generalization of local maximum points for real–valued functions.



MR image graph of intensity ridges of graph

Figure 1: Example of an image, intensity graph, and ridges

The construction of primitive regions is analogous to the reverse gravity watershed methods found in the peak flow models[8,13] and in the network models[16,17]. In the peak flow model, the region points are labeled by starting at a point and following the flow line, whose direction is the gradient of intensity, to a local maximum. Since the gradient points in the direction of maximum increase for f, the flows are in the uphill direction. In the network models, the boundaries of regions are constructed by following flow lines in the gradient direction from local minima to saddles (course lines) and from saddles to local maxima (ridge lines). The points contained in such a boundary lie on flows which are also in the uphill direction.

In our application, following the gradient of intensity direction is not helpful in constructing primitive regions. Generally, a curve consisting of height ridge points is not an integral curve of $\nabla f(x)$. It is possible for a flow line corresponding to $\nabla f(x)$ to transversely cross a height ridge and continue to flow *away* from the ridge. Even if ridges were flow lines, it must be that the flows can only intersect at a critical point for f. The ridges would never take part in the segmentation and the process reduces to just peak flow.

Associated with our ridge construction is a flow vector field whose flows terminate orthogonally at the ridges. The function

$$Q(x) = \sqrt{\sum_{i=1}^{n-1} [v_i(x) \cdot \nabla f(x)]^2}$$

has the property that $Q(x) = 0$ at ridges. The vectors $-\nabla Q^2(x) = -2Q(x)\nabla Q(x)$ are orthogonal to the level sets of Q, so such flows will meet the ridges as indicated. However, the flows should terminate rather than originate at the ridges; that is, the flows should generally be in uphill direction on the graph of intensity. To force the flows to terminate at ridges, the sign on $-\nabla Q(x)$ is adjusted so that the flow has positive component in the $\nabla f(x)$ direction, thereby forcing the flows to move in the uphill direction. In the continuous model the flows are determined by

$$\frac{dx}{dt} = \mathrm{sign}[\nabla f(x) \cdot \nabla Q^2(x)]\nabla Q^2(x),\ t > 0,\quad x(0) = x_0,$$

where x_0 is the initial position for the flow. In the discrete model where the computations are on a rectangular lattice, there are only $3^n - 1$ discrete directions to flow in. The direction selected is the one which minimizes the angle between the true and discrete directions.

The ridge structures are initially segmented into curvilinear segments which are then assigned distinct labels. In the discrete implementation, candidate ridge points are represented as pixels in a binary image. The sets are not necessarily 1–pixel thick, a requirement for identifying the curvilinear segments. We used a thinning algorithm which takes the candidate ridges and thins them so that the resulting set can be easily segmented into curvilinear segments. The thinning algorithm preserves the topology of the original set in that the number of holes is invariant. The algorithm also thins sets from outside to inside, thereby preserving the general shapes of the initial sets.

After the ridges are segmented and labeled, the flow algorithm is executed. At each pixel in the image the flow line is followed until the flow intersects a ridge. Every pixel along the path is assigned the label of the terminal ridge point. In this way every pixel obtains a label and the image is segmented into primitive regions. Each region contains the pixels which flow to the ridge of the same label.

3. IMAGE HIERARCHY

The image hierarchy for image $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is constructed as a multiscale process. The scale is introduced by blurring the initial image via variable conductance diffusion[10,11,12]. As a continuous process, the blurred image $B(x,\sigma)$ at position x and scale σ is the solution to the initial value problem

$$\begin{aligned} B_\sigma(x,\sigma) &= \sigma\nabla\cdot\left(C(x,\sigma,B,\nabla B,\ldots)\nabla B(x,\sigma)\right),\ x\in\mathbb{R}^n,\ \sigma>\sigma_0 \\ B(x,\sigma_0) &= f(x),\ x\in\mathbb{R}^n, \end{aligned}$$

where C is a conductance function (treating the blurring process as a heat transfer process). The initial scale σ_0 is called the *inner scale* of the process. If $C \equiv 1$, then the blurred image is simply the convolution of a Gaussian kernel of standard deviation σ with the input image, $B(x,\sigma) = G(x,\sigma) \oplus f(x)$. For the experiments we used $C \equiv 1$.

In practice, images are specified on a compact set $D \subset \mathbb{R}^n$. The blurring must be computed on D with initial data $f(x)$, but conditions at the boundary ∂D must also be imposed. We used Neumann conditions $\partial B(x,\sigma)/\partial\eta = 0$ for $x \in \partial D$ and $\sigma > \sigma_0$, where the derivative indicates a directional derivative taken in the direction normal to the boundary of D. As a heat transfer process, this condition says that the boundary is insulated.

Initially $f(x)$ has small scale geometric details. As σ increases, the geometric details are annihilated, including ridge structures. Since the equation has no heat source, one expects that as scale becomes infinite, $B(x,\sigma)$ tends to a constant; that is, all details are eventually annihilated. In contrast to the interest of physicists in the steady–state behavior of such differential equations, we are concerned with short–time properties which will directly affect how the hierarchy is built.

The algorithm to construct the hierarchy is as follows. Select a geometric sequence of N scales, say $\sigma_j = \sigma_0 b^j$ for some $b > 1$ and for $1 \le j \le N$. The final scale σ_N is called the *outer scale* for the process. For each σ_j, $j \ge 0$, compute the blurred image $B(x,\sigma_j)$ and segment it using the ridge flow model to obtain $D = \cup_{i=1}^{M_j} R_i^{(j)}$ where $R_i^{(j)}$ is a primitive ridge flow region and $M_j \ge 1$ is the number of primitive regions at scale σ_j. Identify each primitive region $R_i^{(j)}$ with a node at height j in a general tree. The leaf nodes of the tree correspond to the primitive regions occurring at inner scale σ_0.

The links in the tree are inserted based on how primitive regions at one scale become blurred into single regions at the next scale. Two primitive regions at one scale may blur together to form part of a single primitive region at the next scale. This single primitive region is considered to be the parent of the original two regions since it overlaps those two regions more than any other region at the current scale. More specifically, to link the nodes (regions), the parent of region $R_i^{(j)}$ will be a region $R_m^{(j+1)}$ which overlaps $R_i^{(j)}$ more than any other region $R_\ell^{(j+1)}$. At the outer scale σ_N, one expects very few regions in the segmentation. The root node of the tree is the parent for all primitive regions at the outer scale. The root node essentially corresponds to a constant image with a single region (infinite scale).

The implementation of the tree construction is a discrete algorithm which is an approximation to the continuous model described here. Two technical problems can arise as a result of the discretization. First, it is possible in the tree construction that blurring from one scale to the next does not annihilate features (in at least one subregion). Thus, the tree may contain nodes which have exactly one child. Second, even though the variable conductance diffusion process is *causal* (no new features are introduced as scale increases), the discretized process may introduce regions whose corresponding tree nodes have no children. These nodes are not essential to the hierarchy and can be removed; consequently all leaf nodes of the tree correspond to primitive regions only at the initial scale.

Consider a hypothetical hierarchy, shown in Figure 2, produced by the ridge flow segmentation algorithm. The leaf nodes and internal nodes with 2 or more children are labeled with distinct nonnegative integers. The -1 labels indicate that the node has a single child, so the regions of the subtree have not blurred together with any other regions of the hierarchy.

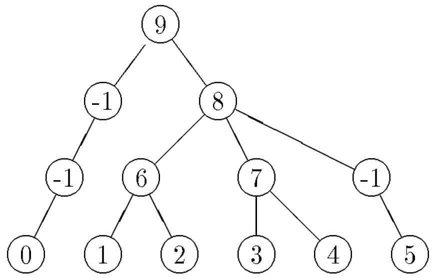


Figure 2: Hierarchy for an image

At inner scale σ_0, the number of primitive regions is $M_0 = 6$. Each region was assigned to a leaf node in the tree. After blurring the image at scale σ_1, the ridge flow segmentation produced $M_1 = 4$ primitive regions. Leaf regions 1 and 2 blurred together so that region 6 (at the next scale) overlapped them the most. Similarly regions 3 and 4 blurred together and were overlapped the most by region 7. However, regions 0 and 5 did not significantly blur together with other regions, so they are not linked to other regions. Intuitively, small adjacent and similar regions should be linked at small scales, but large adjacent and similar regions should be linked at large scales. The number of primitive regions at scale σ_2 is $M_2 = 2$. Regions 6, 7, and the one corresponding to the parent of 5 blurred together, so in effect leaf regions 1 through 5 blurred together at this scale. Region 0 is still dissimilar to the regions represented by the other subtrees at height 2, so its subtree is again an only child. Finally, no more blurring was performed, so the root node is added and corresponds to a constant image (single region) at infinite scale.

4. OBJECT DEFINITION AND VISUALIZATION

The image and corresponding ridge flow hierarchy is used as input to an experimental tool, the Magic

Crayon[13,14], which is designed for object definition and visualization. The tool provides a graphical user interface which allows the user to traverse the hierarchy and quickly identify objects of interest. It was developed to run on workstations in a clinical setting. In addition to allowing the user to rapidly define and extract anatomical objects, the tool provides the means for making accurate volume and surface area measurements of the extracted objects. The Magic Crayon works on the principle that objects of interest can be identified as unions and differences of subtrees of the hierarchy. Of course the details of the data structures of the tree and the subtree identification are hidden from the user. The remainder of the section illustrates how one uses the Magic Crayon to define objects by manipulating the hierarchy in a way that identifies the relevant subtrees.

As a simple example, consider the hierarchy shown in Figure 2. The original image is displayed by the Magic Crayon, and the hierarchy is already loaded into memory. The user moves the mouse cursor to a region of interest in the image and clicks the left button. Suppose that the pixel at the cursor is in region 1. All pixels in region 1 are subsequently colored. The user has the option of coloring nearby related regions by clicking on an "add more" button. This corresponds to moving up the hierarchy from node 1 to its parent, node 6, and then traversing to all children of node 6. All regions corresponding to the children are drawn. In my example, only region 2 is newly visited, and all pixels in region 2 are colored. The left tree of Figure 3 shows the hierarchy after the initial click on a pixel in region 1. A shaded node indicates that the corresponding region has been colored. The right tree of Figure 3 shows the state of the hierarchy after the "add more" operation.

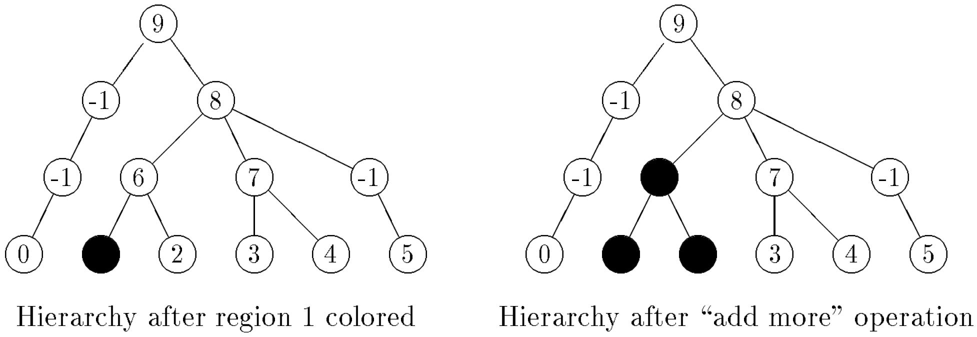


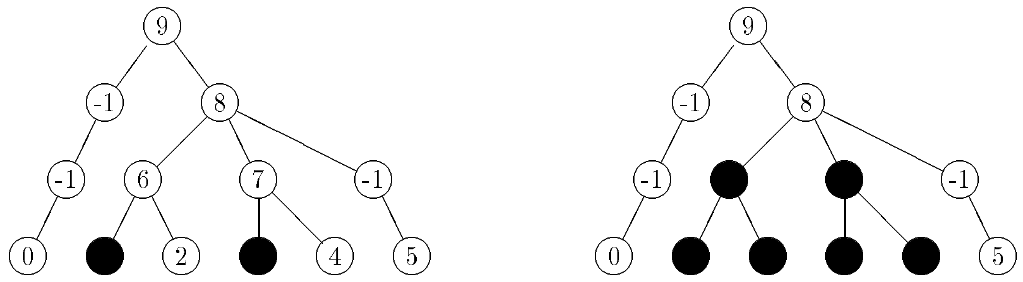
Figure 3: Example of traversing hierarchy from single leaf node

If instead the original colored region is 5, an "add more" operation does not produce any new colored regions. At height 1 in the tree, region 5 did not merge (through blurring) with any other regions. However, a second "add more" operation will merge region 5 with regions 1 through 4.

More generally, given that the user is at height H in the hierarchy, an "add more" operation corresponds to moving up the hierarchy from the current node to its parent at height $H+1$, followed by traversal of all subtrees of the parent. Any leaf nodes visited during the traversal, for which the corresponding regions are not colored, are marked as visited and the regions are all colored. If the user is far up the hierarchy, sometimes unwanted objects are colored. The Magic Crayon provides undo operations and also allows for user–initiated editing of the current colored objects.

For some anatomical objects, moving up the hierarchy will not completely color the object. The user must start the process again, at height 0, in an uncolored portion of the object. The Magic Crayon provides a more efficient way of drawing elongated objects like the brain stem. An operation is provided for moving up the hierarchy from many regions colored initially at height 0, called the "add more all" operation. The user can press the left mouse button and drag the cursor through the object. All encountered regions are colored, so quite a few leaf nodes are visited in the hierarchy. By pressing the "add more all" button, the user moves up the hierarchy to parent nodes for all of the currently colored regions; that is, the "add more" operation is applied at multiple regions.

As an illustration, suppose that regions 1 and 3 were initially colored. After an "add more all" operation, regions 1 through 4 will be colored. The left tree of Figure 4 shows the hierarchy after the initial clicks on pixels in regions 1 and 3. The right tree of Figure 4 shows the state of the hierarchy after the "add more all" operation.



Hierarchy after regions 1 and 3 colored

Hierarchy after "add more all" operation

Figure 4: Example of traversing hierarchy from multiple leaf nodes

In addition to the "add more" controls, the user can delete subtrees using a Magic Eraser. For each of the addition operations there corresponds an analogous deletion operation. By using both sets of controls, the user in effect can move freely through the hierarchy marking the appropriate subtrees that make up the objects of interest. The final marked nodes yield unions and/or differences of subtrees of the original hierarchy.

We segmented a 3–dimensional image containing 20 slices, each slice being 256×256. The number of primitive regions at initial scale $\sigma = 1$ is $31,679$. We used a geometric sequence of 16 scales with multiplier $b = 1.1$. Therefore, the hierarchy is a tree with 17 levels and $31,679$ leaf nodes. Figure 5 shows four consecutive slices (8,9,10,11) of the original image. The shaded regions along the brain stem were a result of dragging the mouse down the middle of the brain stem in slice 9. Since the primitive regions are 3–dimensional, the corresponding regions in the other slices have also been colored, including the slices that are not currently displayed (but buffered in memory).

Figure 6 shows the results in slices (8,9,10,11) of selecting the "add–more–all" operation 5 times. The colored regions correspond to the union of subtrees obtained by moving up 5 levels in the full tree from the originally colored leaf nodes.

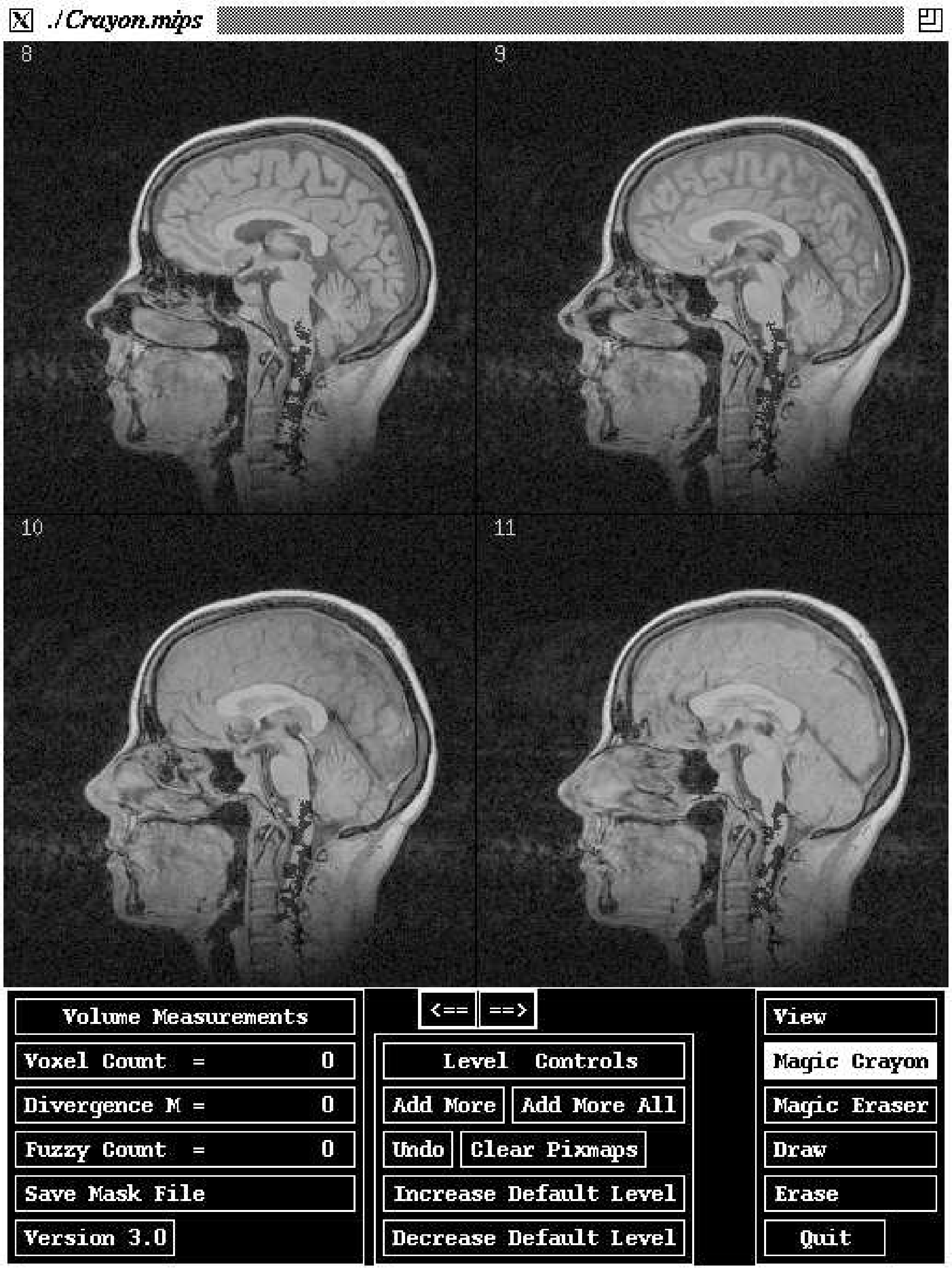


Figure 5: Result of dragging the mouse down the brain stem in slice 9

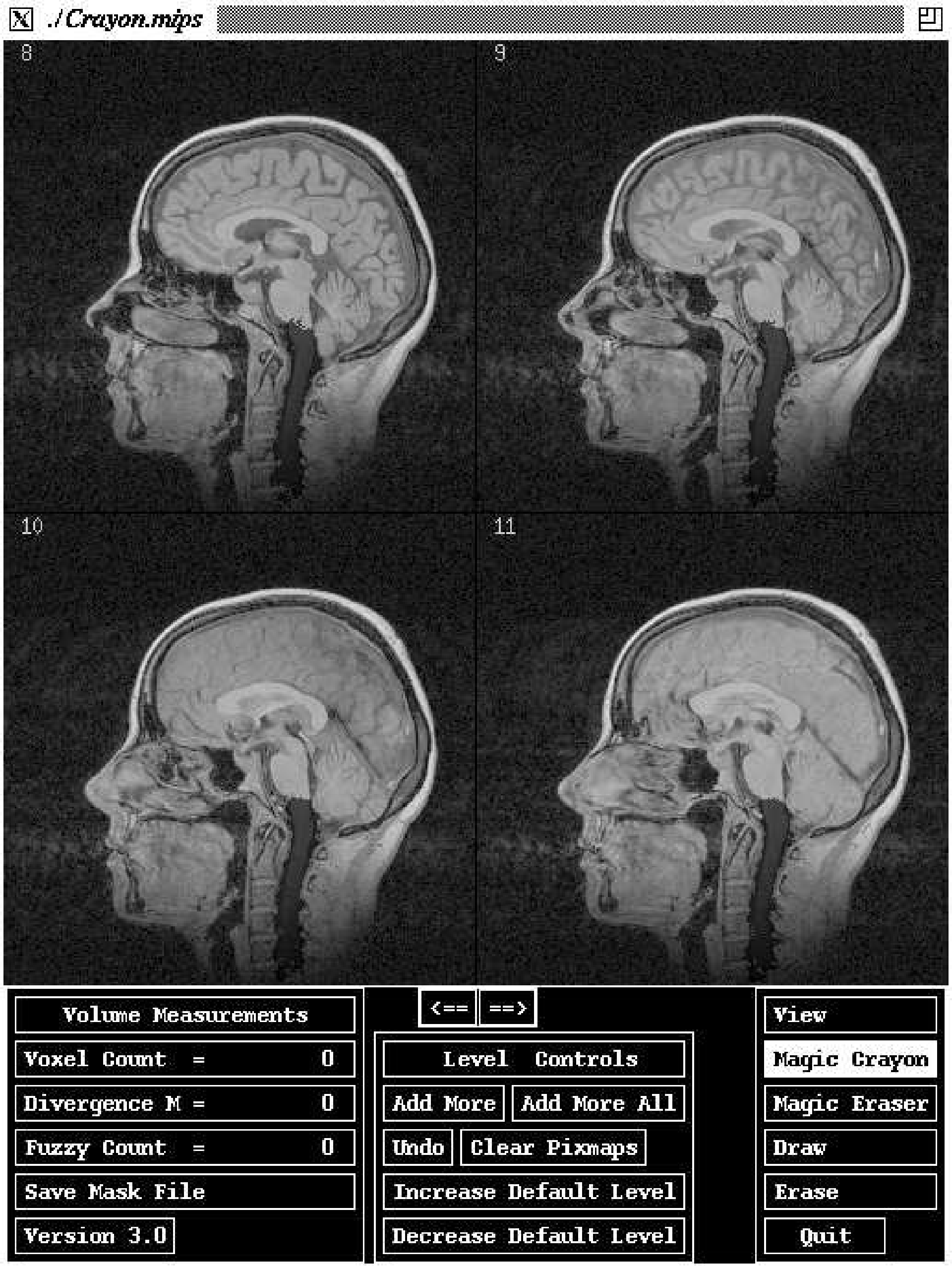


Figure 6: Results in slices (8,9,10,11) after 5 "add–more–all" operations

The Magic Crayon was run on a DECstation 5000/125 running at 25 MHz and rated at 21 MIPS. The machine has 224 Megabytes of physical memory and 1 Gigabyte of virtual memory. The initialization of the Magic Crayon takes about 2 minutes. During this time the hierarchy and primitive region image are read from disk files and converted to internal representations. Bounding boxes are built for each primitive region to help speed up access time during later stages of program execution.

Once the program is loaded and the X–Window interface appears, the user is free to explore the hierarchy. The time to color the brain stem region as shown took about 10 seconds. Of course more time will be taken to completely segment the brain stem. The user can color the regions in the currently displayed slices, then move to other slices and start from new leaf nodes, traverse the hierarchy, and fill in the object on those slices.

5. DISCUSSION

Our experiments with the ridge flow algorithm indicate that it successfully identifies objects whose boundaries and background have a moderate–to–good contrast. Objects whose boundaries contrast poorly with background are not as precisely constructed. The problem is that primitive regions sometimes "bleed" outside one object into the background, or into a neighboring object. This bleeding is a function of the current blurring method, Gaussian blurring. It is a linear process which typically averages object information with background data when applied near an object boundary. A better blurring method is needed which is sensitive to boundaries and will not average across them. If an object boundary is approximately a level curve of intensity, then a good conductance function is one which will force the level curves to propagate in the directions normal to the curves with speed proportional to the local curvature. A consequence is that object shapes tend to be preserved throughout the blurring process. The diffusion equation, $B_\sigma = \sigma|\nabla B|\nabla \cdot (\nabla B/|\nabla B|)$, does have the desired properties. We propose this as a better alternative blurring method for ridge flow. A long term goal is to use variable conductance diffusion of higher–order geometric image information[12]. Additionally, the image could be preprocessed with local contrast enhancement algorithms such as adaptive histogram equalization[18] or generalized order–statistic filters[19,20].

6. ACKNOWLEDGMENTS

This research was supported by grant NIH P01–CA–47982 and by grant NSF DMS–9003037.

7. REFERENCES

1. P.J. Burt, T. Hong, and A. Rosenfeld, "Segmentation and estimation of image region properties through cooperative hierarchical computation," *IEEE Trans. Syst. Man. Cybern.*, vol. 11, no. 12, pp. 802–809, 1981.

2. P.J. Burt and E.H. Adleson, "The Laplacian pyramid as a compact code," *IEEE Trans. on Communications*, vol. 31, no. 4, pp. 532–540, 1983.

3. J.C. Crowley and A.C. Parker, "A representation for shape based on peaks and ridges in the difference of low–pass transform," *IEEE Trans. on PAMI*, vol. 6, no. 2, pp. 156–170, 1984.

4. J.M. Beaulieu and M. Goldberg, "Hierarchy in picture segmentation: A stepwise optimization

approach," *IEEE Trans. on PAMI*, vol. 11, no.2, pp. 150–164, 1989.

5. J.M. Gauch, W.R. Oliver, and S.M. Pizer, "Multiresolution shape descriptions and their applications in medical imaging," in *Information Processing in Medical Imaging (IPMI 10)*, Plenum, NY, June 1988.

6. J.M. Gauch and S.M. Pizer, "Image description via the multiresolution intensity axis of symmetry," *Proc. 2nd Int. Conf. on Computer Vision, IEEE Catalog #88CH2664-1*, pp. 269–274, 1988.

7. L.M. Lifshitz, "A multiresolution hierarchical approach to image segmentation based on intensity extrema," *IEEE Trans. on PAMI*, vol. 12, no. 6, pp. 529–540, 1990.

8. S.M. Pizer, T.J. Cullip, and R.E. Fredericksen, "Toward interactive object definition in 3D scalar images," in *3D Imaging in Medicine: Algorithms, Systems, Applications*, pp. 83–106, Springer–Verlag, Berlin, 1990.

9. D. Eberly, R.B. Gardner, B. Morse, S.M. Pizer, and C. Scharlach, "Ridges for image analysis," *Jour. Math. Imag. and Vision*, to appear in 1994.

10. S. Grossberg, "Neural dynamics of brightness perception: Features, boundaries, diffusion, and resonance," *Perception and Psychophysics*, vol. 36, no. 5, pp. 428–456, 1984.

11. P. Perona and J. Malik, "Scale space and edge detection using anisotropic diffusion," *Proc. IEEE Workshop on Computer Vision*, pp. 16–22, 1987

12. R. Whitaker, "Geometry–limited diffusion in the characterization of geometric patches in images," *Computer Vision, Graphics, and Image Processing–Image Understanding*, vol. 57, no. 1, pp. 111–120, 1993.

13. D.V. Beard, R.E.Faith, D. Eberly, S.M. Pizer, C. Kurak, and R.E. Johnston, "The Magic Crayon: An object definition and volume calculation testbed," *Proc. SPIE Medical Imaging VII*, Newport Beach, CA, February 1993.

14. D.V. Beard, D. Eberly, R.E. Faith, C. Kurak, M. Paramasivam, and S.M. Pizer, "Interacting with image hierarchies for fast and accurate object segmentation," *Proc. SPIE Medical Imaging VIII*, Newport Beach, CA February 1994.

15. R.E. Fredericksen, J.M. Coggins, T.J. Cullip, and S.M. Pizer, "Interactive object definition in medical images using multiscale geometric image descriptions," *Proc. of First Conf. on Visualization in Biomedical Computing*, May 1990.

16. A.C.F. Colchester, "Network representation of 2D and 3D images," in *3D Imaging in Medicine*, K.H. Höhne, H. Fuchs, and S. Pizer (eds.), pp. 45–62, Springer–Verlag, Berlin, 1990.

17. L.D. Griffin, A.C.F. Colchester, and G.P. Robinson, "Scale and segmentation of gray–level images using maximum gradient paths," *Lecture Notes in Computer Science 511: Information Processing in Medical Imaging (IPMI 12)*, pp. 256–272, Springer–Verlag, Berlin, 1991.

18. S.M. Pizer, E.P. Amburn, J.D. Austin, R. Cromartie, A. Geselowitz, J. Greer, B. ter Haar Romeny, and J.B. Zimmerman, "Adaptive histogram equalization and its variations," *Computer Vision, Graphics, and Image Processing*, vol. 39, pp. 355–368, 1987.

19. H. Longbotham and D. Eberly, "Statistical properties, fixed points, and decomposition with the WMMR filters," *Jour. Math. Imag. and Vision*, vol. 2, pp. 99–116, 1992.

20. H. Longbotham and D. Eberly, "The WMMR filters: A class of robust edge enhancers," *IEEE Trans. on Sig. Proc.*, vol. 41, no. 4, pp. 1680–1685, 1993.