# Symmetric Region Growing

Shu-Yen Wan, *Member, IEEE,* and William E. Higgins, *Senior Member, IEEE*

***Abstract*—Of the many proposed image-segmentation methods, region growing has been one of the most popular. Research on region growing, however, has focused primarily on the design of feature measures and on growing and merging criteria. Most of these methods have an inherent dependence on the order in which the points and regions are examined. This weakness implies that a desired segmented result is sensitive to the selection of the initial growing points. We define a set of theoretical criteria for a subclass of region-growing algorithms that are insensitive to the selection of the initial growing points. This class of algorithms, referred to as Symmetric Region Growing, leads to a single-pass region-growing algorithm applicable to any dimensionality of images. Furthermore, they lead to region-growing algorithms that are both memory- and computation-efficient. Results illustrate the method's efficiency and its application to 3-D medical image segmentation.**

***Index Terms*—Connected-components analysis, image segmentation, region growing, region-based segmentation, three-dimensional image analysis.**

## I. Introduction

Of the many image-segmentation methods, region growing has been one of the most popular [1]–[7]. Research in region-growing methods has focused on either 1) the design of feature measures and growing/merging criteria [3], [5]–[12] or 2) algorithm efficiency and accuracy [13]–[15].

Most of these methods, however, have an inherent dependence on the order that points and regions are examined [1], [3], [6]. This weakness implies that a segmented result is sensitive to the selection of the initial growing points (or *seeds*). This problem arises because the measured feature information adaptively changes as the segmentation process progresses. For example, most seeded region-growing processes only add a new point to a region if its corresponding feature measures are similar to those of an adjacent existing region; after this new point is added to the region, the region's feature measures change. Therefore, different initial growing point assignments lead to different values for evolving region information. Recent work, while not a pure region-growing method, integrated edge extraction and seeded region growing to intelligently determine initial seeds and thus enhance segmentation accuracy [16], but this still does not address the sensitivity of the segmentation method to seed selection.

Region-growing methods are also often computation and memory intensive. For example, the three-dimensional (3-D) algorithms of [8], [10], [17] operate as if they are $x$-, $y$-, $z$-inseparable (hence requiring significant computation) and demand considerable memory.

We propose the concept of Symmetric Region Growing (SymRG). Region-growing algorithms that abide by the theoretical criteria defining SymRG are insensitive to the initial growing points and initial conditions set forth for segmentation. Also, SymRG algorithms are both computation and memory efficient. We emphasize that we do not propose a more effective image-segmentation process. Rather, our purpose is to define the theoretical criteria necessary for defining a region growing algorithm that is invariant to starting conditions and that enables efficient algorithm implementation. Section II lays out the theoretical development of SymRG. Section III proposes a general SymRG algorithm applicable to any image dimensionality. Section IV provides results illustrating the computation and memory efficiency of SymRG and discusses its application for 3-D medical image segmentation. Finally, Section V offers concluding remarks.

## II. Theoretical Development

Subsection II-A defines the basic notation and problem statement. Subsection II-B lays out the theoretical constraints for a SymRG algorithm. Finally, Subsection II-C gives guidance on how to devise a SymRG algorithm and motivates the general $N$-dimensional SymRG algorithm described in Section III.

### A. Notation and Problem Statement

Consider a digital image $I$ defined on an $N$-dimensional discrete (digital) space $\mathcal{Z}^n$, i.e., $I \subset \mathcal{Z}^n$. The goal of image segmentation is to partition the digital image $I$ into $M$ disjoint regions of interest $R_i,\ i = 1, \ldots, M$, where the final segmented image $S$ takes the form [4]

$$S = \bigcup_{i=1}^{M} R_i, \quad \text{where } R_i \cap R_j = \emptyset \quad \text{for } i \neq j. \qquad (1)$$

Assume region $R_M$ is reserved for the *background* (generally set to "0" in the final segmented image). Also, assume without loss of generality that each region of interest $R_i, i = 1, \ldots, M-1$, consists of one connected component. (In practice the individual regions in $S$ are distinguished by region labels [18], [19].) In the theory of relations, the segmentation $S$ is formally called

Manuscript received October 8, 2001; revised April 1, 2003. This work was supported in part by NIH Grants R01-RR11800 and R01-CA74325, by the Whitaker Foundation, by NSF Grant BIR-9 317 816 under the Instrument Development for Biological Research Program, and by National Science Council, Taiwan, R.O.C, under Grants NSC-90-2314-B-1812-093 and NSC-91-2213-E-182-020. A preliminary version of this work appeared at the unrefereed conference [26]. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Aly A. Farag.

S.-Y. Wan is with the Departments of Information Management and Computer Science and Engineering, Chang Gung University, Taiwan, R.O.C. (e-mail: sywan@mail.cgu.edu.tw).

W. E. Higgins is with the Department of Electrical Engineering, The Pennsylvania State University, University Park, PA 16802 USA (e-mail: weh2@psu.edu).

Digital Object Identifier 10.1109/TIP.2003.815258

a *partition* of set $I$ and each of the disjoint regions $R_i$ constitute *blocks* of the partition [20].

Lower-case quantities, such as $a, b, p$, and $q$, represent image points $\in I$. An image point is called a pixel in two-dimensional (2-D) images and a voxel in 3-D images [17], [19]. Upper-case quantities, such as $R_i, I, S$, and $A$, denote sets of points in $\mathcal{Z}^n$. The quantity $f(p)$ gives the intensity, or gray-level, value of image point $p \in I$.

If two image points $a$ and $b$ are connected, then at least one *path* (or ordered sequence of connected points) exists between them [18]. Let the notation $\mathcal{P}_{ab}$ represent such a path. Alternately, let the notation $(a, p_1, p_2, \ldots, p_n, b)$ represent a particular path between $a$ and $b$, where point $a$ is a neighbor of point $p_1$. $p_1$ is a neighbor of $p_2$, etc. For this paper, all points on a path must lie in the same region of $S$, i.e., if $a \in R_i$, then $p_1 \in R_i, p_2 \in R_i, \ldots, b \in R_i$. In 2-D images, connectivity and neighbors are defined using either 4-connectivity or 8-connectivity [18]. Analogously, for 3-D images, 6-connectivity or 26-connectivity define such concepts [17].

Focusing the segmentation process to region growing, the segmented image (1) can be represented as

$$S(I, \mathrm{RG}(\psi), \mathcal{S}) = \bigcup_{i=1}^{M} R_i \tag{2}$$

where $I$ is the image under consideration, $\mathrm{RG}(\psi)$ denotes a region-growing algorithm governed by measure and growing criteria $\psi$, and $\mathcal{S}$ represents criteria for defining the initial growing points, or *seeds*, for regions. A seed is an image point that is known to belong to a particular region and begins the construction of the region. The collection of measure and growing criteria $\psi$ can be viewed as consisting of two components: $\psi = \langle \mathcal{I}, \mathcal{X} \rangle$. $\mathcal{I}$ specifies properties that nonseed points must have to be *included* in evolving segmented regions. $\mathcal{X}$ specifies criteria for *excluding* certain image points from all regions of interest.

In general each set of criteria $\mathcal{I}, \mathcal{X}$, and $\mathcal{S}$ consists of a predicate composed of Boolean operations of feature measures. Without loss of generality, the pair $(\mathrm{RG}(\psi), \mathcal{S})$ constitutes a complete image-segmentation algorithm based on region growing. The operations are combined to form a complete predicate for $\mathcal{I}, \mathcal{X}$, or $\mathcal{S}$, using the standard algebraic operators $\{\vee, \wedge, ^-\}$, where "$\vee$" is logical OR, "$\wedge$" is logical AND, and "$^-$" is complementation. Thus, valid predicates for $\psi$ and $\mathcal{S}$ are defined over a Boolean algebra. The exclusion criteria $\mathcal{X}$ can, of course, be easily translated into additional criteria for $\mathcal{I}$.

Fig. 1(a) illustrates the flow for segmenting image $I$ using $(\mathrm{RG}(\psi), \mathcal{S})$. Seeds are first defined for the regions $R_i, i = 1, \ldots, M-1$. Next, the region-growing criteria $\psi = \langle \mathcal{I}, \mathcal{X} \rangle$ are iteratively applied to construct the evolving regions. The growing process terminates when application of the region-growing algorithm produces no further changes to the evolving segmented image. The final result is $S(I, \mathrm{RG}(\psi), \mathcal{S})$.

Some region-based algorithms may not seem to fit the framework of $(\mathrm{RG}(\psi), \mathcal{S})$ at first glance, but they can be transformed into $(\mathrm{RG}(\psi), \mathcal{S})$. For example, the split-and-merge algorithm actually performs the process of iteratively searching the entire image for initial growing points or seeds (splitting) and then growing back regions of interest (merging) [21].

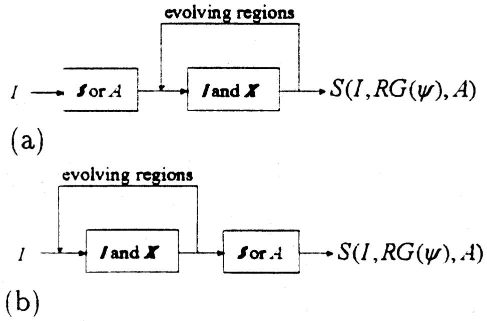


Fig. 1. Processing flow for region growing. $I$ is the input image, $\mathcal{S}$ specifies the seed criteria, $\psi = \langle \mathcal{I}, \mathcal{X} \rangle$ specifies the region growing criteria, and $S(I, \mathrm{RG}(\psi), \mathcal{S})$ is the final segmented image.

Seed criteria $\mathcal{S}$ can consist of operations that *implicitly* specify seed points for regions. Equivalently, $\mathcal{S}$ can also be specified as an *explicit* set of seed points, such as

$$A = \{a_1, \ldots, a_{M-1}\} \subset I \tag{3}$$

where, in general, set $A$ contains one seed point per region of interest. Point $a_1$ acts as the initial growing point, or seed, for $R_1, a_2$ is the seed for $R_2, \ldots$, and $a_{M-1}$ is the seed for $R_{M-1}$. No seed is needed for the background region $R_M$, as all points not assigned to a true region of interest $R_i, i = 1, 2, \ldots, M-1$, are assumed to be relegated to the background. Each point of an explicitly defined seed set, such as $A$ in (3), is known *a priori* to belong to a particular region. If $A$ contains additional points beyond (3), then it is assumed that these points are already assigned to one of the evolving regions $R_i, i = 1, \ldots, M$. Using the seed criteria (3), the segmentation (2) can be stated equivalently as

$$S(I, \mathrm{RG}(\psi), A) = \bigcup_{i=1}^{M} R_i. \tag{4}$$

For the remainder of this paper, we assume that seed criteria $\mathcal{S}$ are converted to an equivalent seed set $A$.

Consider now a different set of initial growing points given by

$$B = \{b_1, \ldots, b_{M-1}\} \subset I \tag{5}$$

where $b_1$ acts as a possible seed for $R_1, b_2$ acts as a possible seed for $R_2$, etc. Suppose this set produces the segmented image

$$S(I, \mathrm{RG}(\psi), B) = \bigcup_{i=1}^{M} R'_i \tag{6}$$

where $R'_1$ is the region grown from $b_1, R'_2$ is the region grown from $b_2$, etc. In general, for $i = 1, \ldots, M-1, a_i \neq b_i$ and $R_i \neq R'_i$. In this paper, the statement

$$S(I, \mathrm{RG}(\psi), A) \equiv S(I, \mathrm{RG}(\psi), B) \tag{7}$$

means that $R_i = R'_i$ for $i = 1, 2, \ldots, M$, per (4) and (6). If two different segmentation algorithms, $(\mathrm{RG}(\psi), A)$ and $(\mathrm{RG}(\psi), B)$, satisfy (7), then they produce *equivalent (identical) segmentations* of image $I$. Fig. 2 illustrates many of the concepts defined thus far for a four-region segmentation problem.

The following important question arises. What are the requirements on region-growing algorithm $\mathrm{RG}(\psi)$ so that $S(I, \mathrm{RG}(\psi), A) \equiv S(I, \mathrm{RG}(\psi), B)$? *That is, what constraints are required on a region-growing algorithm, so that the algorithm is guaranteed to give identical segmentations when starting with any valid seed set?* Region-based algorithms build regions from the seeds by following a certain evolving growing sequence. If the seeds change, then the resulting growing sequence changes. Our question is whether different seed sets, (3) and (5), and growing sequences lead to the *same* segmentation results. If not, what constraints can be placed on an algorithm, so that it generates the same segmentation regardless of the seed sets? That is, what constraints must a region-growing algorithm have to be invariant to changes in the seed set? The next subsection answers these questions.

### *B. General Definitions and Theorems*

This section provides the basic definitions and theoretical criteria for addressing the questions raised above.

*Definition 1:* $\mathbf{P}_{ab}(I, \mathrm{RG}(\psi))$ is defined as the set of all possible paths $\{\mathcal{P}_{ab}^1, \mathcal{P}_{ab}^2, \mathcal{P}_{ab}^3, \ldots\}$ between points $a$ and $b$, where $a, b \in I$, point $a$ is a seed used to grow region $R \subset I$ using $\mathrm{RG}(\psi)$, and $b \in R$.

Given region-growing algorithm $\mathrm{RG}(\psi)$, if seed $a$ produces a region $R$ that does not contain point $b$, then $\mathbf{P}_{ab}(I, \mathrm{RG}(\psi)) = \emptyset$. Also, by the assumption that $R$ consists of one connected component, if $b \in R$, then at least one path $\mathcal{P}_{ab}$ must exist from seed $a$ to image point $b$. Within the context of relation theory, if a path exists from $a$ to $b$, then $a$ and $b$ must be in the same block (region) of the partition $S$ of $I$.

*Definition 2:* $\mathbf{P}_{AB}(I, \mathrm{RG}(\psi))$ is defined as the set of all possible paths from points in seed set $A$ to points in set $B$

$$\mathbf{P}_{AB}(I, \mathrm{RG}(\psi)) = \begin{cases} \bigcup_{i=1}^{M-1} \mathbf{P}_{a_ib_i}(I, \mathrm{RG}(\psi)), & \text{if } \forall i, \mathbf{P}_{a_ib_i}(I, \mathrm{RG}(\psi)) \neq \emptyset \\ \emptyset, & \text{otherwise} \end{cases}$$

where $A$ and $B$ are given by (3) and (5).

The set $\mathbf{P}_{AB}(I, \mathrm{RG}(\psi))$ enumerates all paths from each point $a_i \in A$ to its corresponding point $b_i \in B$, provided that at least one path exists to each $b_i$. $\mathbf{P}_{AB}(I, \mathrm{RG}(\psi)) = \emptyset$, if *any* point $a_i \in A$ [responsible for generating region $R_i$ per (3)] does not have at least one path $\mathcal{P}_{a_ib_i}$ to its corresponding point $b_i \in B$. If for some point $a_i \in A$, no path $\mathcal{P}_{a_ib_i}$ exists, then $b_i \notin R_i$. This immediately implies that $S(I, \mathrm{RG}(\psi), A) \not\equiv S(I, \mathrm{RG}(\psi), B)$, because, per (6), $b_i \in R_i'$ and $R_i \neq R_i'$.

*Definition 3:* The notation

$$A \overset{\mathrm{RG}(\psi)}{\rightarrow} B \quad \text{is equivalent to } \mathbf{P}_{AB}(I, \mathrm{RG}(\psi)) \neq \emptyset.$$

The quantity $A \overset{\mathrm{RG}(\psi)}{\rightarrow} B$ is a *binary relation* from set $A$ to set $B$ over region-growing operation $\mathrm{RG}(\psi)$ [20].

The relation $A \overset{\mathrm{RG}(\psi)}{\rightarrow} B$ implies that there is a way to form at least one path in $S(I, \mathrm{RG}(\psi), A)$ between each initial growing point in $A$ and its corresponding point in $B$. Otherwise, $A \overset{\mathrm{RG}(\psi)}{\not\rightarrow} B$ (or $A \overset{\mathrm{RG}(\psi)}{\rightarrow} B$ is false). Note that $A \overset{\mathrm{RG}(\psi)}{\rightarrow} B$ does not imply $B \overset{\mathrm{RG}(\psi)}{\rightarrow} A$.

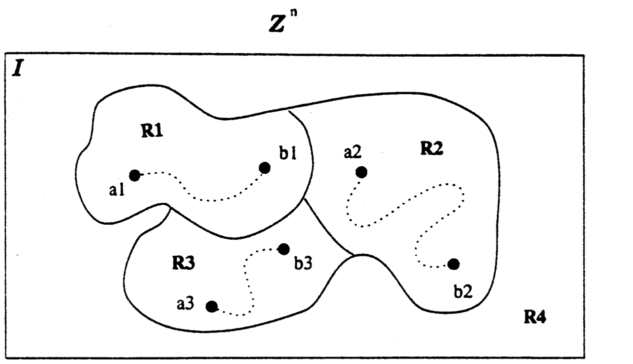


Fig. 2. Depiction of the region-growing process for a 4-region segmentation problem. $R_1$, $R_2$, and $R_3$ are the regions of interest we wish to segment in 3-D image $I$ and $R_4$ is the background. The points $a_1$ and $b_1$ are possible seeds for initiating the region growing process for $R_1$. Similarly, $a_2$ and $b_2$ are possible seeds for $R_2$, etc. The dotted lines give examples of valid paths $\mathcal{P}_{a_ib_i}$ between corresponding points $a_i$ and $b_i$. This figure illustrates the case where $a_i$ and $b_i$ lead to the "same" $R_i$; i.e., they produce equivalent segmentations $S$ of $I$, per (7). But, this is not necessarily the case in general.

*Lemma 1:* The binary relation $\overset{\mathrm{RG}(\psi)}{\rightarrow}$ is reflexive and transitive. That is, for any seed set $A \subset I, A \overset{\mathrm{RG}(\psi)}{\rightarrow} A$ (reflexivity). Also, for any seed sets $A, B, C \subset I$, if $A \overset{\mathrm{RG}(\psi)}{\rightarrow} B$ and $B \overset{\mathrm{RG}(\psi)}{\rightarrow} C$, then $A \overset{\mathrm{RG}(\psi)}{\rightarrow} C$ (transitivity).

*Proof:* (Reflexivity) It is trivial that $A \overset{\mathrm{RG}(\psi)}{\rightarrow} A$, because $\mathbf{P}_{AA}(I, \mathrm{RG}(\psi))$ contains the trivial one-point paths $\mathcal{P}_{a_ia_i}, i = 1, 2, \ldots, M-1$.

(Transitivity) Given $A \overset{\mathrm{RG}(\psi)}{\rightarrow} B$ and $B \overset{\mathrm{RG}(\psi)}{\rightarrow} C$. Then, for all $i = 1, \ldots, M-1$, there exists $\mathcal{P}_{a_ib_i} = (a_i, \ldots, b_i) \in \mathbf{P}_{a_ib_i}(I, \mathrm{RG}(\psi))$ and $\mathcal{P}_{b_ic_i} = (b_i, \ldots, c_i) \in \mathbf{P}_{b_ic_i}(I, \mathrm{RG}(\psi))$. By concatenating paths $\mathcal{P}_{a_ib_i}$ and $\mathcal{P}_{b_ic_i}$, we have $\mathcal{P}_{a_ic_i} = (a_i, \ldots, b_i, \ldots, c_i)$. Thus, $\mathbf{P}_{AC}(I, \mathrm{RG}(\psi)) \neq \emptyset$, or $A \overset{\mathrm{RG}(\psi)}{\rightarrow} C$. □

(Many straightforward proofs are omitted below for clarity. Details can be found in [22].)

Now, consider a general binary relation $\mathcal{R}$ on domain $\mathcal{D}$, such that $\mathcal{R} : \mathcal{D} \rightarrow \mathcal{D}$. The binary relation $\mathcal{R}$ is said to be *symmetric* if $r\mathcal{R}s \Leftrightarrow s\mathcal{R}r, \forall r \subset \mathcal{D}$ and $s \subset \mathcal{D}$ [20]. The concept of a symmetric binary relation can be applied to region growing. In general, the binary relation $\overset{\mathrm{RG}(\psi)}{\rightarrow}$ is, of course, not symmetric [20], but the following definition states when it is symmetric.

*Definition 4:* Binary relation $\overset{\mathrm{RG}(\psi)}{\rightarrow}$ is symmetric if

$$\forall \text{ seed sets } \quad A, B \subset I, \ A \overset{\mathrm{RG}(\psi)}{\rightarrow} B \text{ implies } B \overset{\mathrm{RG}(\psi)}{\rightarrow} A.$$

If $\overset{\mathrm{RG}(\psi)}{\rightarrow}$ is symmetric, we denote it as $\overset{\mathrm{RG}(\psi)}{\leftrightarrow}$ or $\overset{\mathrm{SymRG}(\psi)}{\leftrightarrow}$.

If $\mathrm{RG}(\psi)$ satisfies $A \overset{\mathrm{RG}(\psi)}{\leftrightarrow} B$ for all seed sets $A, B \subset I$, then $\mathrm{RG}(\psi)$ is called a *symmetric region-growing algorithm* and denoted as $\mathrm{SymRG}(\psi)$. Furthermore, given $S(I, \mathrm{SymRG}(\psi), A)$ in the context of the segmentation (4), Definition 4 implies that we can arbitrarily choose sets $X = \{x_1, \ldots, x_{M-1}\}$ and $Y = \{y_1, \ldots, y_{M-1}\}$, where $x_i, y_i \in R_i \subset S(I, \mathrm{SymRG}(\psi), A) \backslash R_M$ and form a bijection (or one-to-one and onto) relation between $X$ and $Y$. Also, by

Lemma 1 and Definition 4, $\overset{\mathrm{SymRG}(\psi)}{\leftrightarrow}$is an *equivalence relation* and the segmented regions $R_i, i = 1, 2, \ldots, M$, induced by $\mathrm{SymRG}(\psi)$, are *equivalence classes* [20].

*Lemma 2:* Let $p$ and $q$ be any pair of points in the same region $R_i \subset S(I, \mathrm{RG}(\psi), A)$ for some $i = 1, 2, \ldots, M-1$, per (3)–(4). If $\mathrm{RG}(\psi)$ is symmetric (i.e., $\mathrm{RG}(\psi)$ can be replaced by $\mathrm{SymRG}(\psi)$ in (4)), then $\mathbf{P}_{pq}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$.

Lemma 2 implies that if a symmetric region growing algorithm is used, then any point $p$ in a region can be used to reach (grow) any other point $q$ in the same region. This leads to the following important result.

*Theorem 1:* Consider a symmetric region growing algorithm $\mathrm{SymRG}(\psi)$, such that $S(I, \mathrm{SymRG}(\psi), A) = \bigcup_{i=1}^{M} R_i$ in the context of (3)–(4). Suppose $a_i \in A$ is replaced by an arbitrary point $p \in R_i$ to form alternate seed set $\hat{A}$. Then, in the resulting segmentation $S(I, \mathrm{SymRG}(\psi), \hat{A})$, the region grown from $p$ is $R_i$.

Theorem 1 states that if a symmetric region growing algorithm is used, then any point $p$ in region $R_i$ can be used as a seed to grow the region $R_i$ and that the resulting grown region is always the same one. In fact, any and all seed points $a_i \in A, i = 1, \ldots, M-1$, can be replaced by any point $p_i \in R_i \subset S(I, \mathrm{SymRG}(\psi), A)$ to form a new seed set $X$ and the resulting segmentation $S(I, \mathrm{SymRG}(\psi), X)$ will be equivalent to $S(I, \mathrm{SymRG}(\psi), A)$.

*Theorem 2:* Given $\mathrm{SymRG}(\psi)$ and seed sets $A, B \subset I$, as in (3) and (5)

$$\mathbf{P}_{AB}(I, \mathrm{SymRG}(\psi)) \neq \emptyset \Leftrightarrow S(I, \mathrm{SymRG}(\psi), A) \equiv S(I, \mathrm{SymRG}(\psi), B). \quad (8)$$

*Proof:* The instructive proof appears in the Appendix. □

Theorem 2 states that if a symmetric region growing algorithm produces a segmentation of image $I$ of the form $S(I, \mathrm{RG}(\psi), A) = \bigcup_{i=1}^{M} R_i$, then, for any of the $M-1$ regions of interest $R_i$, $i = 1, \ldots, M-1$, any point $p \in R_i$ can be used as a seed point to produce the segmentation $S(I, \mathrm{RG}(\psi), A)$. In fact, *Theorem 2 eliminates the importance of the set of initial growing points:* the set $A$ (or criteria $\mathcal{S}$) has no influence on whether a region-growing algorithm is symmetric or not. Further, for a symmetric region-growing algorithm, the order that points are visited during the growing process does not matter. The subsection below proposes corollaries that assert these points and helps bridge the gap from theory to practical implementation.

### *C. Practical Conditions for Symmetric Region Growing*

*Corollary 1:* Consider $\mathrm{SymRG}(\psi)$ and $A$ such that $S(I, \mathrm{SymRG}(\psi), A) = \bigcup_{i=1}^{M} R_i$. Instead of using $A$ to produce the segmentation $S(I, \mathrm{SymRG}(\psi), A)$, consider using $B = \{b_1, \ldots, b_{M-1}\}$, where $b_i \in R_i$ and $b_i$ is the *first* point of $R_i$ encountered while scanning image $I$. Then, $S(I, \mathrm{SymRG}(\psi), A) \equiv S(I, \mathrm{SymRG}(\psi), B)$.

*Proof:* Follows immediately from Theorem 2. □

Corollary 1 reveals that the first encountered point of a region (e.g., the extreme upper left corner point of the region) can be used to grow it with a symmetric region-growing algorithm. This concept helps in improving algorithm efficiency. Yet, before segmentation proceeds, no regions exist, and, thus, the first encountered point of each region is not necessarily known. The following corollary solves this problem.

*Corollary 2:* Consider $(\mathrm{SymRG}(\psi), \mathcal{S})$. Scan the digital image of interest, $I$, sequentially. Grow regions from each scanned point by applying criteria $\psi = \langle \mathcal{I}, \mathcal{X} \rangle$, until all image points have been visited. Examine the resulting regions using $\mathcal{S}$. If any point $p$ of a region satisfies criteria $\mathcal{S}$ for region $R_i$, then assign the region to $R_i$; otherwise, relegate it to the background $R_M$. The resulting segmented image is $S(I, \mathrm{SymRG}(\psi), \mathcal{S})$.

If the region growing algorithm is symmetric, Corollary 2 states that one can scan and grow regions first. After the growing process, one then applies $\mathcal{S}$ to label the "useful" regions. All unlabeled regions are merged into the background. This idea, an attribute of symmetric region-growing algorithms, helps in computation efficiency and is shown in Fig. 1(b).

Because of Theorem 2, the seed criteria $\mathcal{S}$ has no influence on whether a region-growing algorithm is symmetric or not. It is sufficient to focus on the properties of $\psi = \langle \mathcal{I}, \mathcal{X} \rangle$ to define a SymRG. Recall that $\psi$ is a composite of Boolean operations. $\psi$ can be represented as a single predicate, per Definition 5.

*Definition 5:* For $p, q \in I$, let $g(p, q)$ be a predicate representing the growing criteria $\psi$. Then,

$$g(p, q) = \mathtt{TRUE} \Rightarrow p \overset{\mathrm{RG}(\psi)}{\rightarrow} q.$$

Thus, for any point $p \in R_i \subset I$, a neighbor $q$ will be included in $R_i$ iff $g(p, q) = \mathtt{TRUE}$.

*Theorem 3:* For $g(\cdot, \cdot)$ representing $\psi$ of region-growing algorithm $\mathrm{RG}(\psi)$, if $g(\cdot, \cdot)$is symmetric; i.e.,

$$g(p, q) = g(q, p), \quad \forall p, q \in I$$

then $\mathrm{RG}(\psi)$ is symmetric.

Theorem 3 shows that if $\psi$ is a symmetric function, then the region-growing algorithm is symmetric. Since $\psi$ can be denoted as $\psi = \mathcal{I} \wedge \bar{\mathcal{X}}$, then, by the properties of a Boolean algebra, $\psi$ is symmetric if and only if both $\mathcal{I}$ and $\mathcal{X}$ are symmetric [20]. Similarly, each individual criterion or operation constituting $\mathcal{I}$ and $\mathcal{X}$ must be symmetric. Also, the image features employed by $\mathcal{I}$ and $\mathcal{X}$ should not depend on the previous states of the features. Otherwise, the function employing the feature cannot in general be symmetric. Thus, the growing process does not depend on the order that image points are scanned.

Below are examples of common region-growing functions $g(p, q)$. The functions take on the value TRUE if the predicate on the right-hand side is satisfied. In the examples below, $p$ and $q$ are neighboring image points, $\sigma_1, \sigma_2, \sigma_3$, and $\sigma_4$ are constants, $\mu_{N(p)}$ denotes the average gray-level value of point $p$'s neighbors, and $\mu_{R(p)}$ denotes the average gray-level value of the points constituting $p$'s member region:

$$\begin{aligned} g(p, q) &\equiv \sigma_1 \leq |f(p) - f(q)| \leq \sigma_2 \quad \text{symmetric} \\ g(p, \cdot) &\equiv \sigma_3 \leq f(p) \leq \sigma_4 \quad \text{symmetric} \\ g(p, \cdot) &\equiv \left|f(p) - \mu_{N(p)}\right| \leq \sigma \quad \text{symmetric} \\ g(p, q) &\equiv \left|f(q) - \mu_{R(p)}\right| \leq \sigma \quad \text{not symmetric} \end{aligned}$$

```
Function   2DSymRG($I, RG, \psi, \mathcal{S}$)
Construct_1D_Regions($0, \psi$)       // Form 1D regions (line segments) in the first image row
For row $j = 1$ to $N_y - 1$
    Construct_1D_Regions($j, \psi$)   // For each row of the 2-D image, construct 1D regions
    Region_Merge($2, j, \psi$)        // Merge overlapping similar regions (segments) in consecutive rows
EndFor                                // 2D regions are formed
Label_Regions($\mathcal{S}$)          // Assign unique region IDs to equivalent regions
End
                    (a)

Function   3DSymRG($I, RG, \psi, \mathcal{S}$)
Do 2DSymRG($I(0), RG, \psi, \mathcal{S}$)
For 2D slice $k = 1$ to $N_z - 1$      // For each slice of the 3-D image, construct 2D regions
    Do 2DSymRG($I(k), RG, \psi, \mathcal{S}$)   // 2D regions are formed by invoking the 2D function
    Region_Merge($3, k, \psi$)        // Merge overlapping similar 2D regions on consecutive slices
EndFor                                // 3D regions are formed
Label_Regions($\mathcal{S}$)          // Assign unique region IDs to equivalent regions
End
                    (b)

Function   NDSymRG($I, RG, \psi, \mathcal{S}$)      //Construct $N$-D regions, similar to (b)
Do (N-1)DSymRG($I(0), RG, \psi, \mathcal{S}$)
For $(N-1)$–dimensional image $w = 1$ to $N_w - 1$
    Do (N-1)DSymRG($I(w), RG, \psi, \mathcal{S}$)
    Region_Merge($N, w, \psi$)
EndFor
Label_Regions($\mathcal{S}$)
End
          (c)
```

Fig. 3. General set of algorithms for symmetric region growing. (a) 2-dimensional region-growing algorithm, which yields 2-D regions. (b) 3-D algorithm, which invokes the 2-D algorithm and performs region merging along the $z$-direction to construct 3-D regions. (c) General $N$-dimensional algorithm, which follows identically to the 3-D algorithm. It recursively calls an $(N-1)$-dimensional algorithm, until reaching the basic 2-D algorithm of (a).

The labels indicate whether or not the functions are symmetric. Also, functions of the form $g(p, \cdot) = g(p)$, which only depend on one pixel, are clearly symmetric.

## III. General SymRG Algorithm

Theorem 3 states that a region-growing algorithm is symmetric if and only if all criteria constituting $\psi$ are symmetric functions. If the region-growing algorithm is symmetric, then Corollaries 1 and 2 suggest that the implementation of the SymRG can grow regions from the first region points scanned and then apply the seed criteria $\mathcal{S}$ afterward to label the final regions (Fig. 1(b)). This approach is invariant to which region point is scanned first. It also motivates the following general $N$-dimensional SymRG algorithm that is computation- and memory-efficient. Assume that an $N$-dimensional image $I$ has image points $(i, j, k, \ldots, w, \ldots)$, where $i$ is the index of a point along a row, $j$ denotes row index $k$ denotes slice number (for 3-D images), etc. The gray-level value of point $(i, j, k, \ldots, w, \ldots)$ is given by $I(i, j, k, \ldots, w, \ldots)$. Also, assume that seed criteria $\mathcal{S}$ and desired symmetric growing criteria $\psi$ are given. Fig. 3 gives the set of algorithms. Two global data structures are necessary [4]:

1) `Region Table`: Each entry in the region table contains *region ID, region bounding box, number of points, number of 0-to-1 crossings*, and *number of seeds* for a region.

2) `Equivalence Table`: The equivalence table is constructed and incrementally adapted after two equivalent (homogeneous) regions merge. Each table entry represents a growing region and has a linked list of *region ID*'s of equivalent regions and composite region information gathered from the region table, plus the status of this entry. A region may take on one of three states: *growing, ROI*, or *undesired*. The *growing* regions continue to grow and eventually reach final labeling. *ROI* regions are finished growing and contain points satisfying the seed criteria. The *undesired* regions are finished growing and contain no seed points.

The following functions are used:

- Construct_1D_Regions($j, \psi$): Construct 1-D regions (actually 1-D line segments) on the $j$th row by applying growing criteria $\psi$. The output is an updated `Region Table`.
- Region_Merge($n, w, \psi$): Merge contiguous $(n-1)$-dimensional regions between the $w$th and $(w-1)$th $(n-1)$-dimensional image using $\psi$. The output is an updated `Equivalence Table`.
- Label_Regions($\mathcal{S}$): Assign final region labels to the regions that contain seeds satisfying $\mathcal{S}$. The remaining regions are relegated to the background. The output `Equivalence Table` contains the final region labels.

The general $N$-dimensional algorithm (Fig. 3(c)) recursively draws upon algorithms of lower dimensionality. At the end of the recursion, the 2-D algorithm 2DSymRG is used. We point out that the actual implementation need not be recursive. As suggested by Corollary 2, growing does not depend on when points are visited. Thus, if the image is $N$-dimensional, all scanning can be done in $N$dimensions. Region information can be collected incrementally as scanning proceeds. At the end, the seed criteria are applied to the aggregated regions, represented in the equivalence table, to form final regions.

Fig. 4 gives a detailed example of how 2DSymRG progresses. The 2-D algorithm constructs 1-D regions (line segments) for each row of a 2-D image. The region growing process starts with row #1 and adds region information to the Region Table. The dashed arrows in Fig. 4 represent the intermediate 1-D regions grown by the 1-D region-growing process (Construct_1D_Regions). Intermediate regions enclosed by '$<$' and '$>$' satisfy criteria for desired region #1, while regions enclosed by '[' and ']' satisfy criteria for desired region #2. After the intermediate regions are formed for the two rows, region merging occurs (Region_Merge). The solid arrows represent viable region merges. Regions are merged if they are neighbors and if they satisfy common criteria. Each region merge is denoted by an entry in the Equivalence Table. The $(p, q)$ entries next to each intermediate region denote the merges. $p$ denotes the intermediate region ID (stored in Region Table) and $q$ denotes the new equivalent region ID (stored in Equivalence Table). The equivalent region consisting of the linked intermediate regions ultimately denotes the grown region. In the example, intermediate regions #1 and #4 can merge because they have a neighboring segment $[X21, X12]$ and satisfy the criteria of desired region #1. However, intermediate regions #2 and #5 are not merged because they satisfy criteria of different desired regions, even though they have a neighboring segment $[X23, X14]$. The information for the final grown regions is stored in the Equivalence Table.

A SymRG segmentation is achieved by sequentially scanning the image in two passes. The first pass performs region growing and merging. The second pass then uses the seed criteria $\mathcal{S}$ to define final region labels. The algorithm is clearly $x, y, z, \ldots$, etc., separable, and thus enables parallelism and faster computation. Also, because points visited in the first pass aren't needed until the second pass, a SymRG algorithm requires only a few rows (or slices) of the image to be available in memory at any given time, plus a small amount of working buffer to maintain the region and equivalence tables. Most of the image can be stored in the disk media for later use, without suffering significant disk input/output overhead.

The general SymRG algorithm can also easily be adapted to produce computation- and memory-efficient implmentations of other common image-processing functions. Two such functions are connected-component labeling and cavity deletion.

Connected-component labeling operates on a presegmented binary-valued image to form labeled regions. Hence, it is a special case of region growing. The SymRG algorithm can be adapted to this function by merely defining growing criteria $\psi$ that assigns all "1" points to a valid region. No seed criteria are needed. The resulting algorithm thus performs the labeling in a single pass.

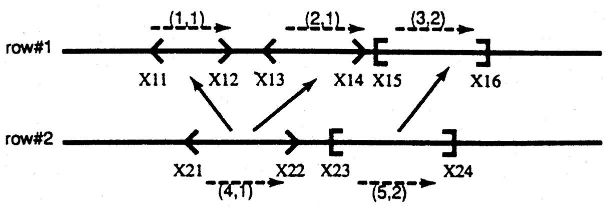


Fig. 4. Two-dimensional SymRG example for two consecutive rows of a 2-D image. Two different regions are being grown. Figure focuses on how regions are formed, so gray-scale information is omitted. Refer to text for a complete description.

Cavity deletion removes interior cavities from regions contained in a previously segmented image. Cavities are defined as background ("0") regions that do not touch the image boundary. Their generation is virtually inevitable during practical image segmentation unless the original image presents perfect noise-free contrast between foreground and background (e.g., [23]). A 2-D (3-D) cavity deletion algorithm can be obtained by adapting the connected-component labeling algorithm. We first identify "background" connected-components; the growing criteria $\psi$ consists of a function that assigns all "0" points to valid regions. In this case, if the foreground is defined as 8 (or 26 in 3-D)-connected, then the background is 4 (or 6 in 3-D)-connected, and vice versa. The background components that do not touch the boundary of the image are considered to be cavities and are then converted to the foreground value. The final resulting image contains solid cavity-free regions.

## IV. Results

Results are presented that demonstrate the computation and memory efficiency of the SymRG algorithm. Also, the application of SymRG to 3-D medical image segmentation is discussed and sample results are provided. References [14], [24] give extensive results in applying SymRG to this problem. We point out that the purpose of this paper is to derive the required general properties for an invariant seeded region-growing algorithm and to provide an efficient implementation of this algorithm. We do not propose a method that gives more accurate segmentations, as this is a function of the specific growing criteria used and, indeed, the real segmentation problem considered.

### A. Computation and Memory Efficiency

Experiments were performed on both a Sun workstation (Solaris 2.5.1, CPU: 250 MHz) and a PC (Windows NT 4.0, CPU: 400 MHz). Four 3-D images were used: an 8-bit human liver image ("humliv") generated by an electron-beam computed-tomography (EBCT) scanner; and three 16-bit rat liver images ("ctr01," "ctr02," and "ctr03") produced by a micro-CT scanner [14]. All of these images depict arterial trees that gradually change in intensity as one traverses the tree. Hence, region growing proved to be particularly well-suited to segmenting these images. A previously proposed 3-D arterial-tree segmentation algorithm [10] was adapted to the SymRG paradigm; complete algorithm details for the SymRG implementation appear in [14].

We compared the segmentation time for SymRG [14] against the previously proposed method [10]. Fig. 5 give the quantitative results. Clearly, the SymRG approach is much faster.

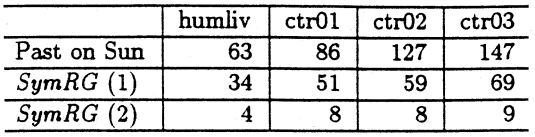
A significant strength of SymRG is its memory efficiency. See Fig. 6. For the 3-D case, the SymRG algorithm only requires 3 original and 3 working slices of the image, plus the memory needed by the region and equivalence tables. Each entry of the region table requires 18 bytes to store region-related information. Each entry of the equivalence table uses 24 bytes for storing information plus 2 bytes for each of the corresponding equivalent regions. The number of entries in the region and equivalence tables depends on the number of intermediate regions produced during the process. Assuming an upper bound of $2^{N-1}$ equivalent regions for an $N$-bit image, the approximate memory usage for performing SymRG is 6 image slices plus $2^{N-1} \times (18+24) + 2 \times 2^{N-1} = 44 \times 2^{N-1} = 21 \times 2^{N}$ bytes. In comparison, the algorithm of [10] requires memory for 2 image copies plus the region table.

### B. Application to 3-D Medical Image Segmentation

We have been successfully applying 3-D seeded growing for segmenting arterial tree structures in 3-D medical images for many years [10]. But an issue that lingered during much of this period was how to guarantee that the segmentation procedure would be *invariant* to starting conditions. This question is particularly important when trying to segment 3-D branching tree structures that distribute themselves in a complex way in a 3-D image, since the starting point (*root*) of the tree is typically hard to ascertain. Research on this fundamental question has led to this paper's effort on symmetric region growing.

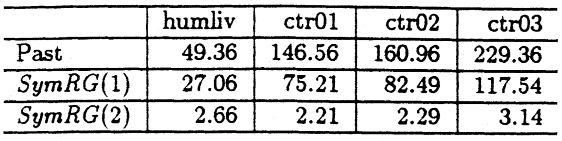
We have used the SymRG algorithm extensively for 3-D micro-CT image segmentation, as described in [14], [24]. Overall, we have segmented on the order of twenty such images, containing various arterial tree structures, such as the hepatic (liver) vasculature and coronary arterial tree. Most importantly, the algorithm produces segmentations that are invariant to the starting point. We now know, per the theoretical development of this paper and the practical results of [14], [24] that this invariance has been achieved (we merely had to rerun the method with a different starting point to see that the results were the same!). Fig. 7 provides an example from this effort for the "humliv" image. Fig. 7(a) depicts a 2-D projection of the original 3-D data set, Fig. 7(b) depicts a 2-D projection of the segmented arterial tree, while Fig. 7(c) depicts a corresponding surface rendering of the segmentation.

But, segmentation of these complex 3-D anatomical tree structures has led to other questions beyond the scope of this paper: What branches are significant branches? What is the "best" way to describe a complex branching 3-D tree structure? Purely automated image segmentation does not appear to be sufficient or practical for resolving these questions; judicious manual interaction seems to be required. Answers to these questions are topics of our current research [24], [25].



| | humliv | ctr01 | ctr02 | ctr03 |
|---|---|---|---|---|
| Past on Sun | 63 | 86 | 127 | 147 |
| *SymRG* (1) | 34 | 51 | 59 | 69 |
| *SymRG* (2) | 4 | 8 | 8 | 9 |

Fig. 5. Run-time comparison (in seconds). The past approach is the 3-D region-growing implementation of [10]. The SymRG implementation of the same region growing method is given in [14]. SymRG(1) was performed on a Sun machine that has one 250 MHz CPU running Solaris 2.5.1, while SymRG(2) was run on a PC that has a 400 MHz CPU running Windows NT 4.0.



| | humliv | ctr01 | ctr02 | ctr03 |
|---|---|---|---|---|
| Past | 49.36 | 146.56 | 160.96 | 229.36 |
| *SymRG*(1) | 27.06 | 75.21 | 82.49 | 117.54 |
| *SymRG*(2) | 2.66 | 2.21 | 2.29 | 3.14 |

Fig. 6. Memory-usage comparison (in megabytes). The past approach is the 3-D region-growing implementation of [10]. The SymRG implementation of the same region-growing approach is given in [14]. SymRG(1) retains a copy of the image in the memory to avoid I/O overhead. SymRG(2) keeps only six slices of the image in the memory when the memory resource is limited; additional slices can be accessed from the disk. Dimensions of images are as follows. humliv: 302 × 389 × 218. ctr01: 319 × 247 × 487. ctr02: 399 × 215 × 491. ctr03: 400 × 400 × 375.

## V. Discussion

It is well known that region growing algorithms tend to depend on where the growing process starts in an image or how regions of interest are oriented in the image. The subclass of region-growing algorithms referred to as symmetric region growing do not depend on where growing starts or on the positions of image regions. The required condition for a region-growing algorithm to be symmetric is that its growing criteria consist exclusively of symmetric functions. The symmetric property does not depend on the region seed criteria used (but these criteria obviously affect what regions get segmented).

A symmetric region growing algorithm can initiate the segmentation process anywhere within an image. In particular, it can begin to grow a region from the first point it reaches. Also, the growing process can be performed first. The seed criteria used to identify the final regions of interest can be applied later. These ideas enable the design of a general computation- and memory-efficient $N$-dimensional symmetric region growing procedure. All the user needs to define are the growing criteria $\psi$ (be sure they are all symmetric functions) and the seed criteria $\mathcal{S}$.

Other standard image-processing functions, such as connected-component labeling and cavity deletion, can easily be cast within the SymRG framework, thus affording these functions the efficiency of the general SymRG procedure we have proposed. Finally, image-segmentation algorithms such as the Split/Merge approach [15], [21] can be implemented as a symmetric region-growing algorithm, provided that all of the splitting and merging criteria are symmetric functions.

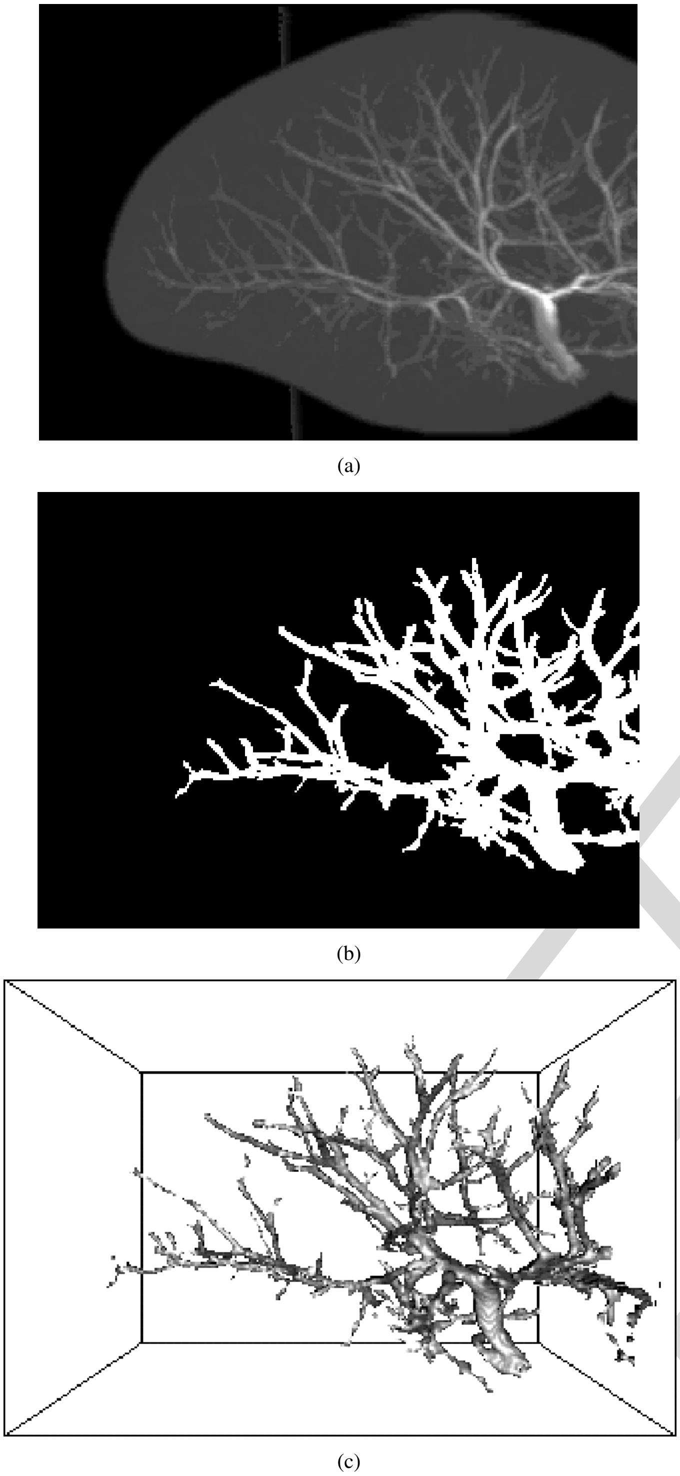


Fig. 7. Various views of a 3-D human liver image ("humliv") with bile ducts selectively opacified by contrast agent. Voxel sampling intervals: $\Delta x = \Delta y = \Delta z = 0.586$ mm. (a) Coronal $(x - z)$ maximum-intensity projection of complete 3-D data set. (b) Coronal projection of segmented arterial tree. (c) Surface rendering of segmented arterial tree.

## Appendix

The discussion below is a proof of Theorem 2.

*Proof:* We use the definitions of $A, B, S(I, \mathrm{SymRG}(\psi), A)$, and $S(I, \mathrm{SymRG}\,(\psi), B)$, given in (3)–(6), with $\mathrm{RG}(\psi)$ replaced by $\mathrm{SymRG}(\psi)$in (4) and (6).

($\Leftarrow$) Given $S(I, \mathrm{SymRG}\,(\psi), A) \equiv S(I, \mathrm{SymRG}(\psi), B)$, which is (7). From (4), (6), and (7), $R_i = R'{}_i, i = 1, \ldots, M$. By Lemma 2, for any pair of seed points $(a_i, b_i), i = 1, 2, \ldots, M - 1$, drawn from $A$ and $B$, at least one $\mathcal{P}_{a_i b_i}$ exists. Therefore, $\mathbf{P}_{AB}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$, or $A \overset{\mathrm{SymRG}(\psi)}{\leftrightarrow} B$.

($\Rightarrow$) Given $\mathbf{P}_{AB}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$. Consider an arbitrary point $p \in I$. There are two cases to consider.

Case 1) foreground—Suppose for some $i = 1, 2, \ldots, M - 1$, $p \in R_i \subset S(I, \mathrm{SymRG}\,(\psi), A)$. Then, $\mathbf{P}_{a_i p}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$, following the definition of seed point $a_i$ in (3). Also, $\mathbf{P}_{a_i b_i}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$ and $\mathbf{P}_{b_i a_i}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$. By Lemma 1, $\overset{\mathrm{SymRG}(\psi)}{\leftrightarrow}$ is transitive. Hence, $\mathbf{P}_{b_i p}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$. Therefore, $p \in R_i{}'$ of $S(I, \mathrm{SymRG}(\psi), B)$, per (6).

Case 2) background—Suppose $p \in R_M \subset S(I, \mathrm{SymRG}(\psi), A)$. Suppose for some $i = 1, 2, \ldots, M - 1$, there exists $b_i \in B$, such that $\mathbf{P}_{b_i p}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$; i.e., $p \in R'{}_i \subset S(I, \mathrm{SymRG}(\psi), B)$. As we know, $\mathbf{P}_{a_i b_i}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$. Thus, by Lemma 1 (transitivity), $\mathbf{P}_{a_i p}(I, \mathrm{SymRG}(\psi)) \neq \emptyset$, which implies that $p \in R_i$. This contradicts the assumption. Hence, $\forall b_i \in B, \mathbf{P}_{b_i p}(I, \mathrm{SymRG}(\psi)) = \emptyset$, which implies that $p \in R'_M$.

Thus, $\forall p \in I$, if $p \in R_i \subset S(I, \mathrm{SymRG}(\psi), A)$, then $p \in R'{}_i \subset S(I, \mathrm{SymRG}(\psi), B)$, which implies (7). □

## Acknowledgment

Dr. E. Ritman of the Mayo Clinic provided the 3-D image data.

## References

[1] R. M. Haralick and L. G. Shapiro, "Survey: Image segmentation techniques," *Comput. Vis., Graph., Image Process.*, vol. 29, pp. 100–132, Jan. 1985.

[2] H. Jiang, J. Toriwaki, and H. Suzuki, "Comparative performance evaluation of segmentation methods based on region growing and division," *Syst. Comput. Jpn.*, vol. 24, no. 13, pp. 28–42, 1993.

[3] R. Adams and L. Bischof, "Seeded region growing," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 16, no. 6, pp. 641–647, 1994.

[4] R. Jain, R. Kasturi, and B. G. Schunck, *Machine Vision*. New York: MIT Press and McGraw-Hill Inc., 1995.

[5] S. C. Zhu and A. Yuille, "Region competition: Unifying snakes, region growing, and Bayer/MDL for multiband image segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 9, pp. 884–900, 1996.

[6] A. Mehnert and P. Jackway, "An improved seeded region growing algorithm," *Pattern Recognit. Lett.*, vol. 18, pp. 1065–1071, 1997.

[7] S. A. Hojjatoleslami and J. Kittler, "Region growing: A new approach," *IEEE Trans. Image Processing*, vol. 7, pp. 1079–1084, July 1998.

[8] D. C. Taylor and W. A. Barrett, "Image segmentation using globally optimal growth in three dimensions with an adaptive feature set," *Proc. SPIE*, vol. 2359, pp. 98–107, 1994.

[9] Y.-L. Chang and X. Li, "Adaptive image region-growing," *IEEE Trans. Image Processing*, vol. 3, pp. 868–872, Nov. 1994.

[10] W. E. Higgins, R. A. Karwoski, W. J. T. Spyra, and E. L. Ritman, "System for analyzing true three-dimensional angiograms," *IEEE Trans. Med. Imag.*, vol. 15, pp. 377–385, June 1996.

[11] T. Pavlidis and Y.-T. Liow, "Integrating region growing and edge detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp. 225–233, Mar. 1990.

[12] T. Asano and N. Yokoya, "Image segmentation schema for low-level computer vision," *Pattern Recognit.*, vol. 14, pp. 267–273, 1981.

[13] B. P. Dobrin, T. Viero, and M. Gabbouj, “Fast watershed algorithms: Analysis and extensions,” *Proc. SPIE*, vol. 2180, pp. 209–220, 1994.

[14] S.-Y. Wan, A. Kiraly, E. L. Ritman, and W. Higgins, “Extraction of the hepatic vasculature in rats using 3-D micro-CT images,” *IEEE Trans. Med. Imag.*, vol. 19, pp. 964–971, Sept. 2000.

[15] I. N. Manousakas, P. E. Undrill, and T. W. Redpath, “A comparison of 3D split-and-merge segmentation with direct MRI determination of cerebral ventricule volume,” *Proce. SPIE* , vol. 2710, pp. 190–200, 1996.

[16] J. Fan, D. Yau, A. Elmagarmid, and W. Aref, “Automatic image segmentation by ihtegrating color-edge extraction and seeded region growing,” *IEEE Trans. Image Processing*, vol. 10, pp. 1454–1466, Oct. 2001.

[17] J. Toriwaki and S. Yokoi, “Basics of algorithms for processing three-dimensional digitized pictures,” *Syst. Comput. Jpn.*, vol. 17, pp. 73–82, 1986.

[18] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Reading, MA: Addison-Wesley, 1992.

[19] A. Rosenfeld and A. Kak, *Digital Picture Processing*, 2nd ed. New York: Academic, 1982.

[20] C. L. Liu, *Elements of Discrete Mathematics*, 2nd ed. New York: McGraw-Hill, 1985.

[21] S. L. Horowitz and T. Pavlidis, “Picture segmentation by a directed split-and-merge procedure,” in *Proc. 2nd Int. Joint Conf. Pattern Recognition*, 1974, pp. 424–433.

[22] S.-Y. Wan, “Analysis and visualization of large branching networks in 3-D digital images,” Ph.D. dissertation, The Pennsylvania State University, University Park, Aug. 2000.

[23] W. E. Higgins, N. Chung, and E. L. Ritman, “Extraction of the left-ventricular chamber from 3-D CT images of the heart,” *IEEE Trans. Med. Imag.*, vol. 9, pp. 384–395, Dec. 1990.

[24] S.-Y. Wan, E. L. Ritman, and W. Higgins, “Multi-generational analysis and visualization of the vascular tree in 3-D micro-CT images,” *Comput. Biol. Med.*, vol. 32, no. 2, pp. 55–71, Mar. 2002.

[25] K.-C. Yu, E. L. Ritman, A. P. Kiraly, S.-Y. Wan, M. Zamir, and W. E. Higgins, “Toward reliable multi-generational analysis of anatomical trees in 3D high-resolution CT images,” in *SPIE Int. Medical Imaging Symp.—Physiology and Function: Methods, Systems, and Applications*, San Diego, CA, Feb. 15–20, 2003.

[26] S.-Y. Wan and W. Higgins, “Symmetric region growing,” *Proc. IEEE Int. Conf.n Image Processing (ICIP2000)*, vol. II, pp. 96–99, Sept. 10–13, 2000.

**Shu-Yen Wan** (M’93) received the B.S. degree in information and computer engineering from Chung Yuan Christian University, Taiwan, R.O.C., in 1988, the M.S. degree in computer science and information engineering from the National Chung Cheng University, Taiwan, in 1993, and the Ph.D. in computer science and engineering from the Pennsylvania State University, University Park, in 2000.

Dr. Wan is currently an Assistant Professor in the Departments Computer Science and Information Engineering, and Information Management, Chang Gung University, Taiwan. His current research interests are in the general areas of image processing, visualization, bioinformatics, and medical imaging.

**William E. Higgins** (SM’93) received the B.S. degree in electrical engineering from the Massachusetts Institute of Technology, Cambridge, and the M.S. and Ph.D. degrees in electrical engineering from the University of Illinois, Urbana-Champaign.

He has held positions previously at the Honeywell Systems and Research Center, Minneapolis, MN, and the Mayo Clinic, Rochester, MN. He is currently a Professor of electrical engineering, computer science and engineering, and bioengineering at the Pennsylvania State University, University Park. In addition, he is an Adjunct Professor of radiology at the University of Iowa, Ames. His research interests are in image processing, analysis, and visualization, and applications to multidimensional medical imaging and virtual endoscopy.

Dr. Higgins is currently an associate editor for the IEEE TRANSACTIONS ON MEDICAL IMAGING and other journals and previously served as associate editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING. He has served on the program committees of numerous conferences, including the IEEE International Conference on Image Processing.