Automatic Selection of Parameters for Vessel/Neurite Segmentation Algorithms

Muhammad-Amri Abdul-Karim, Badrinath Roysam, *Member, IEEE*, Natalie Dowell,
Andreas Jeromin, Murat Yuksel, *Member, IEEE*, and Shivkumar Kalyanaraman, *Member, IEEE*

Abstract— An automated minimum description length (MDL) based approach is presented for selecting optimal parameter settings for segmentation software without detailed knowledge of the internal (often arcane) mechanisms. The method trades off the coverage of image content provided by the segmentation against its conciseness. The coverage is measured by a multi-scale probabilistic segmentation evaluation measure, made robust by an ε -insensitive loss function. The conciseness, as measured by the description length, is modulated by a coefficient w to account for the unavoidable inefficiency of real-world segmentation description formats. A recursive random search algorithm is used for efficient global optimization. The proposed method is modular and extensible, allowing the segmentation and optimization software to be modified/substituted independently.

When applied to vessel/neurite segmentation problems in neurobiology and ophthalmology, the proposed method resulted in parameter settings that were always superior to fixed default values. Even with 1000 iterations, estimated settings were within 5% of the globally optimal result for a tube segmentation algorithm with 9 settings. It enables non-expert users across application areas to select parameters effectively, and objectively, minimizing the need for technical support. It is also an effective approach to minimizing and simplifying the external interface of complex segmentation software.

***EDICS*—2-SEGM Segmentation; 2-MODL Modeling; 2-ANAL Analysis**

Manuscript received November 23, 2004. This work was supported in part by the Center for Subsurface Sensing and Imaging Systems (NSF ERC, Award EEC-9986821), MicroBrightField Inc., MARA of Malaysia, NIH, NIBIB, R01-0000359, the Nanobiotechnology Center (NSF STC ECS-9876771), NSF NeTS-NR 0435259, and Rensselaer Polytechnic Institute.

M.-A. Abdul-Karim is with the Rensselaer Polytechnic Institute, Troy, NY 12180 USA (email: abdulm@rpi.edu).

B. Roysam is with the Rensselaer Polytechnic Institute, Troy, NY 12180 USA (phone: 518-276-8067; fax: 518-276-8715; e-mail: roysam@ecse.rpi.edu).

N. Dowell is with the The Wadsworth Center, NYS Department of Health, Albany, NY 12237 USA (email: dowell@wadsworth.org).

A. Jeromin is with the Baylor College of Medicine, Houston, TX 77030 USA (email: jeromin@ltp.neusc.bcm.tmc.edu).

M. Yuksel is with the Rensselaer Polytechnic Institute, Troy, NY 12180 USA (email: yuksem@ecse.rpi.edu).

S. Kalyanaraman is with the Rensselaer Polytechnic Institute, Troy, NY 12180 USA (email: shivkuma@ecse.rpi.edu).

I. INTRODUCTION

One practical barrier to more widespread adoption of automated image analysis systems in quantitative biomedicine is the need to adapt/customize them to cope with biological variability. Of particular interest to this work is, but not limited to, automated segmentation of tube-like structures (e.g., blood vessels and neurites) in biomedical images [1-21]. Significant variability is observed across imaging modalities, instrument settings, applications, and even across images from a single application. This inevitably forces algorithm designers to incorporate user parameter settings to adapt the software system to a particular class of images. Consequently, the users of such software are faced with the difficulty of selecting these parameters without sufficient expertise and/or knowledge of the internal mechanisms, some of which may even be arcane/irrelevant for the specific application at hand. Furthermore, the non-linearity of image analysis algorithms implies that changes in the settings do not always lead to intuitive and/or proportional changes in the results [22, 23]. The end result is that a time-consuming manual trial-and-error effort, as well as extensive technical support on the part of the developer, is often necessary to properly configure the software for a given application. Even then, these settings are, in the end, subjective, and there is no assurance of optimality.

The primary goal of this work is to enable a non-expert user to select parameters effectively, and objectively, treating the segmentation software as a “black box.” The secondary goal is to enable the algorithm developer to modify the internal details while maintaining a consistent and simple external interface; and to minimize the cost of technical support.

The rationale behind the proposed method is that automated image analysis systems are expected to perform a non-trivial data reduction, extracting as much of the structural content in the image as possible, and expressing it concisely - in terms of instances and descriptive parameters of object models. In the fitting of these models to the image data, a tradeoff must be made between the fitting error, and conciseness of the representation. This requires a quantitative metric to evaluate the segmentation quality along these two terms.

In this paper, we propose a minimum description length (MDL) based metric [24, 25] for achieving the above-mentioned tradeoff. The parameter selection process is formulated as a global optimization using a measure that combines a multi-scale probabilistic segmentation evaluation measure [26] with the description length of the segmentation. A recursive random search based algorithm is used to perform an efficient and application-independent optimization.

The proposed approach is modular by design. This is imperative due to the already high and growing complexity of modern segmentation software, making it impractical to interface with internal routines. A similar statement could be made concerning global optimization software. Modularity also enables substitutions of algorithms written by different groups. With this in mind, we have chosen an implementation strategy (Fig. 1b) in which a general model-based segmentation algorithm is externally interfaced with the evaluation and optimization modules in a manner that allows them to be developed and modified independently.

II. SUMMARY OF RELATED LITERATURE

This work draws upon four main bodies of literature: (i) automated/quantitative/objective segmentation evaluation; (ii) MDL estimation; (iii) global optimization; and, (iv) automated vessel/neurite segmentation algorithms.

The traditional approach to evaluating segmentation algorithms is by visual inspection [22]. Significant progress has been made on automated approaches. Metrics for segmentation evaluation [22, 23, 27-35] can be either goal-oriented, i.e., evaluation based on the performance of post-segmentation steps such as pattern classification [30], based on other application-guided criteria such as the probability of false detection [27, 28, 31], or based on mismatch with manual ground truth segmentations (e.g., [35]). Most of them are specific for low-level vision tasks, such as edge detection [22, 23, 27, 29, 31, 36] and region growing [34, 37], but are mentioned here nevertheless since they usually form the foundations of the more complex and specialized image segmentation algorithms (e.g., [6, 38]).

The MDL principle [25] offers a systematic way to obtain an objective balance between segmentation conciseness and coverage [24, 37, 39]. In this paper, we present a MDL-based variational formulation [37, 39, 40], similar to the Mumford-Shah approach for segmentation [34].

Automated segmentation of tube-like structures in biological images, especially neurons [1-5] and blood vessels [6-21], are of particular interest to this work. These references are just a sparse sampling of the literature in this area, and the interested reader is referred to the literature review sections of these publications.

In the field of global optimization, genetic algorithms [41] and simulated annealing methods [42] are widely used since they require little *a priori* domain-specific information. Controlled random search [43], is also often considered for such cases. These are also known as stochastic optimization algorithms, mainly based on random sampling methods [44, 45], as opposed to deterministic algorithms (e.g., [46]). To improve efficiency, they are normally combined with local search techniques, such as steepest descent [47] and pattern search [48]. Limitations of global optimization algorithms are discussed in [49, 50].

III. GENERAL OBJECTIVE FUNCTION

Although our immediate interest is the segmentation of tube-like objects, the description that follows is presented more generally to enable the reader to adapt this approach to other applications and object types. To begin, let a segmentation algorithm to be a function $f:\{\mathcal{I},\xi\}\rightarrow\mathcal{M}$ that maps an image $\mathcal{I}$ to a set of object(s), denoted $\mathcal{M}$, using a parameter vector ("settings") $\xi\in\Omega$ (see Fig. 1a). Our goal is to automate the search for the optimal $\hat{\xi}$ that yields the optimal $\hat{\mathcal{M}}$ in some defined sense, trading off the conciseness of the segmentation result against its coverage, using the MDL principle as a natural and appropriate guide [24].

The optimal segmentation $\hat{\mathcal{M}}$ minimizes the descriptional code length $\mathcal{L}(\mathcal{I}|\mathcal{M})$. Since typical images are large, the two-part MDL formulation,

$$\mathcal{L}\left(\mathcal{I}\middle|\mathcal{M}\right)=\underbrace{-\log_2 P\left(\mathcal{I}\middle|\mathcal{M}\right)}_{\text{coverage}}+\underbrace{\frac{k}{2}\log_2 n}_{\text{conciseness}}, \tag{1}$$

is appropriate since the shortest code length given by (1) asymptotically approaches the shortest code length obtained using an assumed coding system for the image $\mathcal{I}$ of n pixels given a segmentation (estimated model of the data) $\mathcal{M}$ that can be described with k parameters [24].

We can also view an image segmentation as the partitioning $\mathcal{I}=\left\{\mathcal{M},\mathcal{M}^C\right\}$, i.e., into the foreground and background regions. Let $P(\mathbf{x})$ be the probability that a pixel $\mathbf{x}$ belongs to the foreground. Since the first term in (1) measures the coverage of $\mathcal{M}$, it must be integrated over the unaccounted region of $\mathcal{I}$, i.e., the region $\mathbf{x}\in\mathcal{M}^C$. Assuming independent and identically distributed pixels, the coverage term in (1) becomes $\sum_{\mathbf{x}\in\mathcal{M}^C}-\log_2 P(\mathbf{x})$.

We are not interested in pixels with insignificant $P(\mathbf{x})$ values for two reasons. First, the results of the automated image analysis system are not expected to enable exact reconstruction of the image – they are merely a meaningful extraction of the structures in the image. Second, the segmentation coverage should not be affected heavily by pixels with low probability of structural presence for which $\log P(\mathbf{x})$ diverges. Accordingly, we adopt an "ε-insensitive" approach [51] where terms for which $P(\mathbf{x})<\varepsilon$ are dropped, yielding the following measure:

$$-\log_2 P\left(\mathcal{I}\middle|\mathcal{M}\right)=\sum_{\left\{\mathbf{x}\in\mathcal{M}^C,P(\mathbf{x})<\varepsilon\right\}}-\log_2 P(\mathbf{x}). \tag{2}$$

The second term of (1), which measures the conciseness of $\mathcal{M}$, is adapted directly from the standard two-part MDL formulation which has been shown to be an asymptotically optimal measure of model conciseness [24]. For the same reasons, the term n in (1) is defined to be $n\triangleq\left|\left\{\mathbf{x}\middle|P(\mathbf{x})\geq\varepsilon\right\}\right|$.

In order to adapt the above approach to the immediate task of vessel/neurite segmentation, we need to instantiate the term $P\left(\mathcal{I}\middle|\mathcal{M}\right)$ in (2) for the segmentations of interest since it is not provided by the MDL principle [24]. We start by noting that typically, a segmentation algorithm is based on a geometrical model $\mathcal{G}$, or "shape", of the objects of interest (e.g., tubes, blobs, etc.) that are instantiated in the image data over a set of size scales $\sigma \in [\sigma_{\min}, \sigma_{\max}]$ pixels. These two pieces of problem-domain information are logically provided by the user, and hence should be incorporated into $P(\mathbf{x})$. Below, we explicitly estimate $P(\mathbf{x})$ for the tube geometrical model, and the generalization to other object shapes follows thereafter.

To measure the probability of a pixel being on a tube (e.g., vessel/neurite), we adopt the multi-scale probabilistic vesselness measure [26, 52, 53] that forms the basis for numerous tube-segmentation algorithms [3, 4, 10]. Denoted $\mathcal{V}_\sigma(\mathbf{x}) \in [0,1)$, it measures the likelihood that a local group of pixels centered at $\mathbf{x}$ belong to a tube of size scale σ [26, 52, 53]. It inspects second-order features [54] of the image $\mathcal{I}$, at scale σ, at each pixel $\mathbf{x}$, which is obtained from the Hessian matrix $H_\sigma(\mathbf{x}) = \sigma^\gamma \left[\mathcal{I}(\mathbf{x}) * \frac{\partial^2}{\partial \mathbf{x}^2} G_\sigma \right]$. The Gaussian function with standard deviation σ is denoted as G_σ, the Lindeberg constant [55] for a family of scale-normalized derivatives is denoted γ, and "*" is the convolution operator. The eigenvalues of $H_\sigma(\mathbf{x})$, denoted λ_1 and λ_2, are ordered such that $|\lambda_1| \le |\lambda_2|$.

For two-dimensional (2-D) images, $\mathcal{V}_\sigma(\mathbf{x})$ is given by:

$$\mathcal{V}_\sigma(\mathbf{x}) = \begin{cases} 0, \text{ if } \lambda_2 > 0 \\ \exp\left(-\frac{R_B^2}{2\beta^2}\right)\left[1 - \exp\left(-\frac{S^2}{2c^2}\right)\right] \end{cases}, \tag{3}$$

where $R_B = \lambda_1 / \lambda_2$ (probability of being on a tube) and $S = \sqrt{\sum_{d \le D} \lambda_d^2}$ (image intensity contrast factor) incorporate the eigenvalues λ_d of $H_\sigma(\mathbf{x})$ and $D = 2$ for 2-D images. The terms β and c in (3) are weighting parameters for R_B and S.

Frangi *et al.* [26] have described approaches to adapt this method to other key geometrical models in biological images. Table I is a simplified summary linking key geometrical models and the corresponding eigenvalues. In 3-D, the 3×3 $H_\sigma(\mathbf{x})$ has three eigenvalues, similarly ordered as $|\lambda_1|\le|\lambda_2|\le|\lambda_3|$. Notice that the vesselness value in (3) increases with decreasing magnitude of R_B, effectively capturing the eigenvalues-shape associations for tube-like objects. This suggests that (3) can be modified accordingly for other geometrical models.

Returning to the specific case for tube-like objects, the probability that a pixel $\mathbf{x}$ belongs to a tube-like structure, $P(\mathbf{x})$, also denoted as $\mathcal{V}_{[\sigma_{\min},\sigma_{\max}]}(\mathbf{x})$, is defined as follows:

$$P(\mathbf{x}) \triangleq \max_{\sigma_{\min}\le\sigma\le\sigma_{\max}} \mathcal{V}_\sigma(\mathbf{x}).$$

At this point, the problem of image segmentation is ready to be wrapped into an optimization framework using the optimization metric based on (1). Before proceeding, two practical issues must be addressed in the optimization metric: (i) to give the users the optional ability to override/bias the tradeoff between conciseness and segmentation coverage; and (ii) to enable the use of diverse representations of the segmentation results since results of segmentation are commonly fed to other software modules. To allow such interfacing, the outputs of a segmentation routine are cast in a specific format. However, these formats are not always designed to be efficient in the sense of minimizing number of bits used to describe the segmentation. In terms of the MDL formulation, we recognize that different representations of $\mathcal{M}$ result in relatively different k values in (1). Now, to a first approximation, we assume that different formats for representing the same segmentation have different, but fixed inherent inefficiencies relative to each other. With this approximation, the relative representational inefficiency is effectively a scaling factor, denoted w for the conciseness term in (1).

As noted earlier, we allow a user to optionally influence/bias the tradeoff between coverage and conciseness by incorporating a parameter, denoted α, between them. The overall segmentation quality measure, denoted $q(\mathcal{M}|\mathcal{G},\alpha,\sigma,w,\xi)$, is then given by:

$$q = \alpha \left[\underbrace{-\log_2 P\left(\mathcal{I} \middle| \mathcal{M}\right)}_{\text{coverage}} \right] + (1-\alpha) \left[\underbrace{\left(\frac{1}{w}\right)}_{\text{inefficiency}} \underbrace{\frac{k}{2} \log_2 n}_{\text{conciseness}} \right],$$

for a given geometrical model $\mathcal{G}$ and size scales $[\sigma_{\min}, \sigma_{\max}]$. The segmentation coverage term is the ε-insensitive probability term given by (2). When $\alpha = 0.5$, the conciseness-coverage tradeoff is a balanced one. We also note that ideally $w = 1$ for the relatively most-efficient (concise) representation of $\mathcal{M}$.

IV. RECURSIVE RANDOM SEARCH STRATEGY

Genetic algorithms [41] and simulated annealing methods [42] are widely used for global optimization since they require little *a priori* domain-specific information. However, these algorithms are mainly designed to seek full optimization and were found to lack efficiency for our application. Controlled random search [43] is also often considered for such cases but still suffers from lack of efficiency. To improve efficiency, these algorithms are normally combined with albeit noise-prone local search techniques, such as steepest descent [47], and pattern search [48].

Several considerations motivate the selection of an alternative search strategy known as the Recursive Random Search (RRS). First, exhaustive search is prohibitive. Second, the objective function is not differentiable with respect to the parameter vector, mainly because the mapping itself is not differentiable. Third, for the image segmentation problems of interest, it is rational to seek approximate solutions within a defined time frame, rather than truly global optima. Fourth, it is common for segmentation software to have several irrelevant/ineffective settings for a given application, and search algorithms that are able to minimize the computational effort in such dimensions are preferable. Finally, we expect the search algorithm to be robust to minor noise-like fluctuations in the objective function. Interestingly, Ye *et al.* [44] developed this algorithm motivated by similar parameter spaces encountered in computer networking – specifically, automatic and dynamic configuration of network components to maximize network throughput.

As indicated by the name, the RRS algorithm is based on random sampling. This algorithm searches the parameter space in two recursive steps: exploration and exploitation, respectively. The exploration step examines the macroscopic features of the objective function (e.g., globally convex or "big valleys" structure [12]) and attempts to identify promising areas in Ω that are subsequently "exploited" intensively by the second step. For the random sampling, a uniform distribution over Ω is used. This has been shown to be the simplest search technique for similar non-linear problems, and is widely used [43-45, 56]. It has been shown to be more efficient for exploring high-dimensional parameter spaces compared to deterministic exploration methods [44, 57], and can be shown to converge to the global optima [58].

We show below that random sampling is in fact very efficient in its initial steps and only starts to become inefficient in the later sampling steps. Given a measurable objective function $q(\xi)$ over the parameter space Ω, we can define the distribution of objective function values for some $q_0 \in [q_{\min}, q_{\max}]$ as $\phi_\Omega(q_0) = m(\{\xi \in \Omega \mid q(\xi) \le q_0\}) / m(\Omega)$, where $m(.)$ is the Lebesgue measure. Hence, the distribution function $\phi_\Omega(q_0)$ represents the portion of the points in the parameter space whose objective function values are smaller than a certain level q_0. Furthermore, it has a maximum value of 1 when $q_0 = q_{\max}$, and has minimum value(s) of $m(\hat{\xi}) / m(\Omega)$ where $\hat{\xi}$ is (the set of) global optimum(optima). Without loss of generality, assume that $q(\xi)$ is a continuous function and $m(\{\xi \in \Omega \mid q(\xi) = q_0\}) = 0,\ \forall q_0 \in [q_{\min}, q_{\max}]$.

Assuming a $q_r \in [q_{\min}, q_{\max}]$ such that $\phi_\Omega(q_r) = r$, $r \in [0,1]$, an r-percentile subspace $\mathcal{S}_\Omega(r)$ in the parameter space can be defined as $\mathcal{S}_\Omega(r) = \{\xi \in \Omega \mid q(\xi) \le q_r\}$. In other words, $\mathcal{S}_\Omega(r)$ is the subspace containing parameter vectors such that the probability of existence of a global optimum $\hat{\xi}$ within this subspace is $1 - r$. Note that $\mathcal{S}_\Omega(1)$ is just the whole parameter space and $\lim_{\delta \to 0} \mathcal{S}_\Omega(\delta)$ converges to the global optima. Suppose the sample sequence generated by n steps of random sampling is $\{\xi\}_{i=1}^{n}$ and $\xi^{(i)}$ is the one with the minimum objective function value, then the probability of $\xi^{(i)}$ in $\mathcal{S}_{\Omega}(r)$ is given by $P\left(\left\{\xi^{(i)} \in \mathcal{S}_{\Omega}(r)\right\}\right) = 1-(1-r)^{n} = p$. Stated in another way, the value of r for which $\xi^{(i)}$ will be reached with probability p is given by $r = 1-(1-p)^{1/n}$.

For any $0 < p < 1$, r will tend to 0 with increasing n, implying that random sampling will converge to the global optima with increasing numbers of samples. Since r decreases exponentially with increasing n, clearly the efficiency of random sampling is high at initial samples but falls sharply at later samples. This observation commenced the idea of restarting the random sampling before its efficiency drops off.

The recursive exploitation phase of the search is described next. An important idea behind this phase is to maintain the initial high efficiency of random sampling by "restarting" it before its efficiency becomes low [44], either by moving or resizing the sample space according to sample history. Specifically, RRS performs exploitation in two iterative steps: realign and shrink. As illustrated in Fig. 2a, if a more promising sample ξ_2 is found during random sampling, the current sample space $\mathcal{S}_1$ centered at the current sample ξ_1 is realigned (moved) to the sample space $\mathcal{S}_2$. Else, the parameter space is shrunk to $\mathcal{S}_3$ as shown in Fig. 2b.

The strategy for limiting the exploitation step to promising subspaces is based on identifying an r-percentile subspace for exploitation, as described above. In this way, most trivial subspaces will be excluded from exploitation, improving the overall efficiency of the search. In contrast, many algorithms, such as multi-start [56], do not distinguish between subspaces and hence may waste time in trivial areas.

The RRS is efficient at handling an objective function with a subset of ineffective parameters [44] because random samples maintain a uniform distribution within the subspace composed of only the relevant/effective parameters, effectively minimizing the computational effort invested on negligible parameters. In contrast, local search methods are affected by unimportant parameters because of their high dependency on dimensionality of the search space. For more details, the interested reader is referred to [44].

V. EXPERIMENTAL RESULTS AND EXAMPLES

This section provides a series of examples of progressively increasing complexity, starting with the simplest case of segmentation by global image thresholding. It is followed by a neuron tracing algorithm in which just two parameters out of nine are optimized, and concluded with a full nine-dimensional parameter search [38, 59]. In all examples, $\alpha = 0.5$ is chosen for pure MDL-balanced optimization.

Example #1: This example is merely intended to illustrate the methodology in a trivial and readily-understood context – global intensity thresholding [60]. A global intensity threshold τ is applied to images containing tube-like structures, and the goal is to find the optimal value $\hat{\tau}$ that yields the optimal segmentation $\hat{\mathcal{M}}$. For objects brighter than the background, the segmentation function is given by:

$$f\left(\mathcal{I}\left(\mathbf{x}\right),\boldsymbol{\xi}\right)=\begin{cases}\mathbf{x}\in\mathcal{M} & \mathit{if}\,\mathcal{I}\left(\mathbf{x}\right)\geq\tau \\ \mathbf{x}\in\mathcal{M}^{C} & \mathit{otherwise}\end{cases}.$$

For 8-bit grayscale images, $\mathcal{I}\left(\mathbf{x}\right)\in\left[0,255\right]$, the global intensity threshold $\xi=\tau\in\left[0,255\right]$ is a one-dimensional parameter vector in the parameter space $\Omega\in\mathbb{Z}^{1}$. Here, the representation used for $\mathcal{M}$ is the set of all foreground pixels, and k simply becomes the cardinality of this set.

Fig. 3 shows the results of applying the proposed methodology to this case. The value of the relative representational inefficiency coefficient was empirically estimated as $w = 12.1$. Panel (a) shows an image of neuronal dendrites captured by fluorescent microscopy. The vesselness measure $\mathcal{V}_{[1,4]}(\mathbf{x})$ is displayed in Panel (b). Panel (c) is a plot of the objective function q versus the threshold value τ. The optimal segmentation $\hat{\mathcal{M}}$ is shown in Panel (d). Panels (e & f) illustrate the idea of re-using the relative representational inefficiency coefficient w across applications. Panel (e) shows an image of a human retina imaged by a fundus camera [61] that was optimally segmented using the same value of w. We note that the value of w depends mostly on the manner in which the segmentation $\mathcal{M}$ is represented.

Example #2: In this example, we show the use of the proposed methodology to choose the optimal parameters $\hat{\xi}$ for an automated neuron tracing algorithm [38, 59] (algorithm summary in Appendix). Parameter settings for this algorithm are denoted:

$$\xi = \begin{pmatrix} g & L_{\min} & L_{\max} & n_{shift} & n_{rotate} & M & s_{\max} & \rho & \nu \end{pmatrix}.$$

They are summarized in Table II, along with their default values. Note that the traces are not merged for illustrative purposes since the segments are merged after tracing, i.e., as a post-processing step in the tracing software.

A different representation is appropriate for $\mathcal{M}$ in this case, compared to Example #1. It consists of all centerline points plus the estimated width value at each centerline point. As a simple-minded first approximation, we choose k as the number of centerline points. The value of w was empirically estimated to be 3.1, following the procedure used in Example #1. Note that the representation for $q\left(\hat{\xi}_{RRS:1000:1}\right)$ here is relatively more efficient (storage wise) compared to using the pixel array in Example #1, hence here we have a w value that is closer to the ideal $w = 1$.

An exhaustive search result for just two parameters, contrast threshold multiplier ρ in the range $[1, 20]$ and maximum allowed stopping violations ν in the range $[1,10]$, with other parameters at default values, is shown in Fig. 4, using vesselness measures $\mathcal{V}_{[2,4]}(\mathbf{x})$. Panel (a) shows the input image which contains fluorescently-labeled neurites imaged using a multi-photon microscope [62]. Panel (b) shows the computed vesselness values $\mathcal{V}_{[2,4]}(\mathbf{x})$. Panel (c) displays the entire optimization objective function q versus ρ and ν. Panels (d through g) show the corresponding segmentation $\mathcal{M}$ using ρ and ν values as pointed in panel (c). Since only two parameters are being searched here, it is feasible to search exhaustively. These two parameters are related directly to the stopping criteria of the tracing algorithm and are chosen to best exemplify the effect of the stopping criteria on trace conciseness and coverage.

Example #3: This example shows the search in the full nine-dimensional parameter space of the tracing algorithm, using both exhaustive search and RRS. Since we are still using the same representation for $\mathcal{M}$ as in Example #2, all of the following sub-examples use the relative representational inefficiency coefficient $w = 3.1$.

Fig. 5 shows an image of a neuron captured using fluorescence microscopy in Panel (a). The global least optimal segmentation obtained using an exhaustive search of 236,250 parameter combinations is shown in Panel (b). Panel (c) shows the corresponding globally optimal segmentation. We will show in the next few sub-examples that this can be approximated closely using RRS limited to just 1,000 parameter combinations, denoted $\hat{\mathcal{M}}_{RRS:1000}$.

Fig. 6, Panel (a), displays a neuron image acquired using fluorescence microscopy. Panel (b) shows the traces using the default settings $\xi_{default}$ of the tracing algorithm. Panel (c) shows the worst-case RRS-optimal segmentation $\hat{\mathcal{M}}_{RRS:1000}$. The improvement in terms of the quality measure q, using $\mathcal{V}_{[1,8]}(\mathbf{x})$, with respect to the default settings is on average 25% for 10 RRS runs (worst-case: 23%).

Fig. 7, Panel (a), shows segments of the fluorescently-labeled neuronal dendrite grown on a topographically modified semiconductor surface, imaged using a widefield microscope. Panel (b) shows the global optimal segmentation $\hat{\mathcal{M}}$ obtained using the exhaustive search of the parameter space Ω, using $\mathcal{V}_{[1,5]}(\mathbf{x})$. Panel (c) shows the worst-case RRS-optimal segmentation $\hat{\mathcal{M}}_{RRS:1000}$ out of 50 RRS runs which is within 3% of the global minimum value (globally, $q \in [27{,}431,\ 72{,}253]$). Panel (d) displays the RRS performance for those 50 runs, highlighting the best case and worst case results together with the average performance.

Given that we are indeed seeing improvements in the q values in the previous sub-examples compared to default parameter settings $\xi_{default}$, one may ask what if we were to obtain a parameter set $\hat{\xi}_{RRS:1000:k}$ that yields the RRS-optimal segmentation of the k^{th} image $\mathcal{I}_k$, and use it with the rest of the images. This may be desired if the images are captured under similar imaging conditions and computational resources are limited to obtain optimal segmentation for each image in the image set. We perform this experiment using an image set of 29 fluorescently-labeled neuron images, and the results are displayed in Fig. 8. The first bar shows optimization metric q values when $\xi_{default}$ is used, denoted $q\left(\hat{\xi}_{default}\right)$. The second shows q values when the optimal parameter set is obtained for each individual image k in the image set $\left\{\mathcal{I}_k\right\}_{k=1}^{29}$, denoted $q\left(\hat{\xi}_{RRS:1000:k}\right)$, with improvements on average 19 ± 5 %. The third contains the q values when the optimal parameter set for image $k=1$ is used for each image in the image set, denoted $q\left(\hat{\xi}_{RRS:1000:1}\right)$, yielding improvements for all images that averages at 10 ± 4 %.

VI. CONCLUSIONS AND DISCUSSION

We have demonstrated the practicality of automatically tuning complex segmentation algorithms, using automatic segmentation assessment and global optimization, guided by the MDL principle. The proposed approach can greatly simplify the external interface of segmentation software packages, enable adaptation across large image batches from bioassays, and reduce the need for expensive technical support. The proposed methodology is fully automatic and self-contained, and free of user interactions such as in reinforcement-learning segmentation systems [63]. The core idea of trading off segmentation conciseness and segmentation coverage is extensible to other applications and other geometrical models. The optional parameter α allows a user to bias the tradeoff. The use of a relative representational inefficiency coefficient w allows the framework to be used with practical, non-minimal, segmentation description formats.

The modularity of the proposed framework allows insertion of alternative segmentation algorithms and/or global optimization algorithms, as long as the interfaces between modules are maintained [64]. Algorithms for segmenting other key biological objects types such as blobs [65, 66] would require matching quality measures.

Despite high variability in biological images, there exists an assuring level of similarity between images taken in a single experiment/bioassay under similar imaging conditions. If computational resources are limited, we can perform the automated parameter selection step for a single representative image from the image set (or perhaps, a modest subset) and use the obtained parameter set for the rest of the images. This yields a better operating point for the segmentation algorithm compared to application-independent default values. This is superior to trial-and-error hand-tuning of the parameters for all images which does not even guarantee that the result is approaching the optimal one.

In future research, we plan to investigate information-theoretic approaches to modeling and adaptively estimating the relative representational inefficiency coefficient (w).

In summary, the proposed framework can redefine the term "automated" in "automated image analysis" systems, by allowing segmentation software to configure themselves optimally and objectively across application domains.

APPENDIX: A BRIEF DESCRIPTION OF THE EXPLORATORY TRACING ALGORITHM

The tracing algorithm [38, 59] models tube-like biological structures as piecewise-linear generalized cylinder segments (Fig. 9). To begin tracing, seed points are found by searching for local maxima along rectangular grids g pixels apart and are validated using the same generalized cylinder model.

Then, at each tracing iteration j, four boundary points $\left\{\mathbf{b}_L^j, \mathbf{b}_R^j, \mathbf{b}_T^j, \mathbf{b}_B^j\right\}$ (only $\left\{\mathbf{b}_L^j, \mathbf{b}_R^j\right\}$ in 2-D) corresponding to the left, right, top, and bottom tube boundaries are each found using directional correlation kernels called templates. The center point $\mathbf{c}^j$ is simply defined as the center of these boundary points. Each template consists of linearly stacked 1-D edge-detector kernels of the form

$\left(-1,-2,0,+2,+1\right)^T$ anchored at the image point $\mathbf{b}^j$ along a particular direction $\mathbf{u}^j$. The edge-strengths from each 1-D edge detector kernel r are averaged through the entire template length $l \in L$, using median statistics [67] for robustness. The set $L = \left[L_{\min}, L_{\max}\right]$ contains all template lengths. The correlation between the templates and the image is called the "template response", denoted R, as below:

$$R\left(\mathbf{b}^j, \mathbf{u}^j, L\right) = \underset{l \in L}{\arg\max} \left\{ \underset{t=1,2,\ldots,l}{median} \left\{ r\left(\mathbf{b}^j + t\mathbf{u}^j\right) \right\} \right\}.$$

Each boundary point correspond to the maximal response template parameters $\left\{\mathbf{b}^j, \mathbf{u}^j, l^j\right\}$:

$$(\mathbf{b}^j, \mathbf{u}^j, l^j) = \underset{\{(\mathbf{b},\mathbf{u},l)\ |\ \mathbf{b}=\mathbf{c}^j+m\mathbf{u}_\perp, m=1,..,\frac{M}{2}, \mathbf{u}\in\mathbf{U}, l\in L\}}{\arg\max} R(\mathbf{b}, \mathbf{u}, L),$$

where $\mathbf{U}$ is the set of unit vectors along directions in the neighborhood of $\mathbf{u}^j$. The user parameter M is the radius of the widest expected vasculature. Each template is elongated, shifted from $\mathbf{c}^j$, and rotated about two axes $H_0^j H_1^j$ and $V_0^j V_1^j$ to find the corresponding boundary points. To save computation, the template is only shifted in a neighborhood n_{shift} of previously calculated width at iteration $j-1$ and rotated in a neighborhood n_{rotate} around previously calculated tracing direction $\mathbf{u}^{j-1}$. The next center point $\mathbf{c}^{j+1}$ is estimated by scaling $\mathbf{u}^j$ with the adaptive step-size s^j. The same boundary-finding process is repeated until a stopping criterion containing a contrast-based threshold multiplier ρ is met ν consecutive times. The interested reader is referred to [38, 59] for details of the algorithm.

ACKNOWLEDGMENT

We thank colleagues William Shain, James Turner, Gang Lin, Jacob R. Glaser, Edmund Glaser, and Jeff Sprenger for test image contributions and discussions.

REFERENCES

[1] W. He, T. A. Hamilton, A. R. Cohen, T. J. Holmes, C. Pace, D. H. Szarowski, J. N. Turner, and B. Roysam, "Automated three-dimensional tracing of neurons in confocal and brightfield images," *Microsc Microanal*, vol. 9, pp. 296-310, 2003.

[2] I. Y. Koh, W. B. Lindquist, K. Zito, E. A. Nimchinsky, and K. Svoboda, "An image analysis algorithm for dendritic spines," *Neural Comput.*, vol. 14, pp. 1283-310, 2002.

[3] E. Meijering, M. Jacob, J. C. Sarria, P. Steiner, H. Hirling, and M. Unser, "Design and validation of a tool for neurite tracing and analysis in fluorescence microscopy images," *Cytometry*, vol. 58A, pp. 167-76, 2004.

[4] G. J. Streekstra and J. van Pelt, "Analysis of tubular structures in three-dimensional confocal images," *Network*, vol. 13, pp. 381-95, 2002.

[5] C. M. Weaver, P. R. Hof, S. L. Wearne, and W. B. Lindquist, "Automated algorithms for multiscale morphometry of neuronal dendrites," *Neural Comput*, vol. 16, pp. 1353-83, 2004.

[6] A. Can, H. Shen, J. N. Turner, H. L. Tanenbaum, and B. Raysam, "Rapid automated tracing and feature extraction from retinal fundus images using direct exploratory algorithms," *IEEE Trans. Inform. Technol. Biomed.*, vol. 3, pp. 125-138, 1999.

[7] L. M. Lorigo, O. D. Faugeras, W. E. Grimson, R. Keriven, R. Kikinis, A. Nabavi, and C. F. Westin, "CURVES: Curve evolution for vessel segmentation," *Med. Image Anal.*, vol. 5, pp. 195-206, 2001.

[8] M. Maddah, A. Afzali-Kusha, and H. Soltanian-Zadeh, "Efficient center-line extraction for quantification of vessels in confocal microscopy images," *Med. Phys.*, vol. 30, pp. 204-11, 2003.

[9] X. Jiang and D. Mojon, "Adaptive local thresholding by verification-based multithreshold probing with application to vessel detection in retinal images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, pp. 131-137, 2003.

[10] S. R. Aylward and E. Bullitt, "Initialization, noise, singularities, and scale in height ridge traversal for tubular object centerline extraction," *IEEE Trans. Med. Imag.*, vol. 21, pp. 61-75, 2002.

[11] M. A. T. L. Figueiredo, J.M.N., "A nonsmoothing approach to the estimation of vessel contours in angiograms," *IEEE Trans. Med. Imag.*, vol. 14, pp. 162-172, 1995.

[12] S. C. Chaudhuri, S.; Katz, N.; Nelson, M.; Goldbaum, M., "Detection of blood vessels in retinal images using two-dimensional matched filters," *IEEE Trans. Med. Imag.*, vol. 8, pp. 263-269, 1989.

[13] L. C. Gang, O.; Krishnan, S.M., "Detection and measurement of retinal vessels in fundus images using amplitude modified second-order Gaussian filter," *IEEE Trans. Biomed. Eng.*, vol. 49, pp. 168-172, 2002.

[14] W. E. S. Higgins, W.J.T.; Karwoski, R.A.; Ritman, E.L., "System for analyzing high-resolution three-dimensional coronary angiograms," *IEEE Trans. Med. Imag.*, vol. 15, pp. 377-385, 1996.

[15] J. Lowell, A. Hunter, D. Steel, A. Basu, R. Ryder, and R. L. Kennedy, "Measurement of retinal vessel widths from fundus images based on 2-D modeling," *IEEE Trans. Med. Imag.*, vol. 23, pp. 1196-1204, 2004.

[16] J. Chen and A. A. Amini, "Quantifying 3-D vascular structures in MRA images using hybrid PDE and geometric deformable models," *IEEE Trans. Med. Imag.*, vol. 23, pp. 1251-1262, 2004.

[17] N. Flasque, M. Desvignes, J. M. Constans, and M. Revenu, "Acquisition, segmentation and tracking of the cerebral vascular tree on 3D magnetic resonance angiography images," *Med. Image Anal.*, vol. 5, pp. 173-183, 2001.

[18] C. Toumoulin, C. Boldak, J. L. Dillenseger, J. L. Coatrieux, and Y. Rolland, "Fast detection and characterization of vessels in very large 3-D data sets using geometrical moments," *IEEE Trans. Biomed. Eng.*, vol. 48, pp. 604-606, 2001.

[19] F. K. H. Quek and C. Kirbas, "Vessel extraction in medical images by wave-propagation and traceback," *IEEE Trans. Med. Imag.*, vol. 20, pp. 117-131, 2001.

[20] K. Bühler, P. Felkel, and A. La Cruz, "Geometric methods for vessel visualization and quantification - A survey," in *Geometric Modeling for Scientific Visualization*, G. Brunnett, B. Hamann, H. Muller, and L. Linsen, Eds. Heidelberg: Springer-Verlag, 2004, pp. 399-420.

[21] O. Wink, W. J. Niessen, and M. A. Viergever, "Multiscale vessel tracking," *IEEE Trans. Med. Imag.*, vol. 23, pp. 130-133, 2004.

[22] M. D. Heath, S. Sarkar, T. Sanocki, and K. W. Bowyer, "A robust visual method for assessing the relative performance of edge-detection algorithms," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 1338-1359, 1997.

[23] M. Heath, S. Sarkar, T. Sanocki, and K. Bowyer, "Comparison of Edge Detectors: A Methodology and Initial Study," *Comput. Vis. Image Underst.*, vol. 69, pp. 38-54, 1998.

[24] J. Rissanen, *Stochastic complexity in statistical inquiry*. Singapore: World Scientific, 1989.

[25] A. Barron, J. Rissanen, and B. Yu, "The minimum description length principle in coding and modeling," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2743-2760, 1998.

[26] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever, "Multiscale vessel enhancement filtering," *Lect. Notes Comput. Sc.*, vol. 1496, pp. 130-137, 1998.

[27] K. Bowyer, C. Kranenburg, and S. Dougherty, "Edge Detector Evaluation Using Empirical ROC Curves," *Comput. Vis. Image Underst.*, vol. 84, pp. 77-103, 2001.

[28] T. Kanungo, M. Y. Jaisimha, J. Palmer, and R. M. Haralick, "A methodology for quantitative performance evaluation of detection algorithms," *IEEE Trans. Image Processing*, vol. 4, pp. 1667-1674, 1995.

[29] M. C. Shin, D. B. Goldgof, and K. W. Bowyer, "Comparison of edge detector performance through use in an object recognition task," *Comput. Vis. Image Underst.*, vol. 84, pp. 160-178, 2001.

[30] O. D. Trier and A. K. Jain, "Goal-directed evaluation of binarization methods," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 1191-1201, 1995.

[31] Y. Yitzhaky and E. Peli, "A method for objective edge detection evaluation and detector parameter selection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, pp. 1027-1033, 2003.

[32] Y. J. Zhang, "A survey on evaluation methods for image segmentation," *Pattern Recog.*, vol. 29, pp. 1335-1346, 1996.

[33] S. K. Warfield, K. H. Zou, and W. M. Wells, "Simultaneous truth and performance level estimation (STAPLE): an algorithm for the validation of image segmentation," *IEEE Trans. Med. Imag.*, vol. 23, pp. 903-921, 2004.

[34] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational-problems," *Commun. Pur. Appl. Math.*, vol. 42, pp. 577-685, 1989.

[35] A. Hammoude, "An empirical parameter selection method for endocardial border identification algorithms," *Comput. Med. Imag. Grap.*, vol. 25, pp. 33-45, 2001.

[36] P. Meer and B. Georgescu, "Edge detection with embedded confidence," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, pp. 1351-1365, 2001.

[37] Y. G. Leclerc, "Constructing simple stable descriptions for image partitioning," *Int. J. Comput. Vision*, vol. 3, pp. 73-102, 1989.

[38] K. A. Al-Kofahi, S. Lasek, D. H. Szarowski, C. J. Pace, G. Nagy, J. N. Turner, and B. Roysam, "Rapid automated three-dimensional tracing of neurons from confocal image stacks," *IEEE Trans. Inform. Technol. Biomed.*, vol. 6, pp. 171-187, 2002.

[39] S. C. Zhu and A. Yuille, "Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, pp. 884-900, 1996.

[40] J.-M. Morel and S. Solimini, *Variational methods in image segmentation*. Boston: Birkhèauser, 1995.

[41] M. Mitchell, *An introduction to genetic algorithms*. Cambridge, Mass.: MIT Press, 1996.

[42] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671-680, 1983.

[43] W. L. Price, "Global optimization by controlled random search," *J. Optimiz. Theory App.*, vol. 40, pp. 333-348, 1983.

[44] T. Ye and S. Kalyanaraman, "Internet traffic engineering - A recursive random search algorithm for large-scale network parameter configuration," *ACM SIGMETRICS Perf. Eval. Rev.*, vol. 31, pp. 10, 2003.

[45] A. H. Kan and G. T. Timmer, "Stochastic global optimization methods .2. Multi level methods," *Math. Program.*, vol. 39, pp. 57-78, 1987.

[46] J. E. Falk and R. M. Soland, "Algorithm for separable nonconvex programming problems," *Manage. Sci.*, vol. 15, pp. 550-569, 1969.

[47] L. Armijo, "Minimization of functions having lipschitz continuous first partial derivatives," *Pac. J. Math.*, vol. 16, pp. 1-8, 1966.

[48] R. Hooke and T. A. Jeeves, "Direct search solution of numerical and statistical problems," *J. Acm*, vol. 8, pp. 212-218, 1961.

[49] N. J. Radcliffe and P. D. Surry, "Fundamental limitations on search algorithms: Evolutionary computing in perspective," *Lect. Notes Comput. Sc.*, vol. 1000, pp. 275-291, 1995.

[50] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, pp. 67-82, 1997.

[51] A. Smola, N. Murata, B. Schoelkopf, and K. R. Mueller, "Asymptotically optimal choice of curly epsilon-loss for support vector machines," presented at Artificial neural networks: ICANN 98, Skovde; Sweden, 1998.

[52] C. Canero and P. Radeva, "Vesselness enhancement diffusion," *Pattern Recog. Lett.*, vol. 24, pp. 3141-3151, 2003.

[53] A. F. Frangi, W. J. Niessen, R. M. Hoogeveen, T. van Walsum, and M. A. Viergever, "Model-based quantitation of 3-D magnetic resonance angiographic images," *IEEE Trans. Med. Imag.*, vol. 18, pp. 946-956, 1999.

[54] D. Eberly, *Ridges in image and data analysis*. Boston: Kluwer Academic Publishers, 1996.

[55] T. Lindeberg, "Edge detection and ridge detection with automatic scale selection," *Int. J. Comput. Vision*, vol. 30, pp. 117, 1998.

[56] A. Tèorn and A. Zhilinskas, *Global optimization*. Berlin ; New York: Springer-Verlag, 1989.

[57] B. Xi, Z. Liu, M. Raghavachari, C. H. Xia, and L. Zhang, "A smart hill-climbing algorithm for application server configuration," presented at The 13th Int. Conf. on WWW, New York, 2004.

[58] R. Y. Rubinstein, *Simulation and the Monte Carlo method*. New York: Wiley, 1981.

[59] M.-A. Abdul-Karim, K. Al-Kofahi, E. B. Brown, R. K. Jain, and B. Roysam, "Automated tracing and change analysis of angiogenic vasculature from in vivo multiphoton confocal image time series," *Microvasc. Res.*, vol. 66, pp. 113-125, 2003.

[60] P. K. Sahoo, S. Soltani, A. K. C. Wong, and Y. C. Chen, "A survey of thresholding techniques," *CVGIP*, pp. 233-60, 1988.

[61] A. Hoover and M. Goldbaum, "Locating the optic nerve in a retinal image using the fuzzy convergence of the blood vessels," *IEEE Trans. Med. Imag.*, vol. 22, pp. 951-958, 2003.

[62] P. T. So, C. Y. Dong, B. R. Masters, and K. M. Berland, "Two-photon excitation fluorescence microscopy," *Annu. Rev. Biomed. Eng.*, vol. 2, pp. 399-429, 2000.

[63] J. Peng and B. Bhanu, "Closed-loop object recognition using reinforcement learning," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, pp. 139-154, 1998.

[64] E. Gamma, *Design patterns : elements of reusable object-oriented software*. Reading: Addison-Wesley, 1995.

[65] C. Solorzano and E. Rodriguez, "Segmentation of confocal microscope images of cell nuclei in thick tissue sections," *J. Microsc.*, vol. 193, pp. 212-226, 1999.

[66] G. Lin, U. Adiga, K. Olson, J. F. Guzowski, C. A. Barnes, and B. Roysam, "A hybrid 3D watershed algorithm incorporating gradient cues and object models for automatic segmentation of nuclei in confocal image stacks," *Cytometry*, vol. 56A, pp. 23-36, 2003.

[67] P. J. Huber, *Robust statistics*. New York: Wiley, 1981.

LIST OF TABLES

Table I. Indicates the relationship between the eigenvalues of the Hessian matrix with common geometrical models in biological cell and tissue-level imagery [26]. For dark objects on bright backgrounds, the signs are reversed. In 2-D, plate-like structures can not be detected. The first two eigenvalues λ_1 and λ_2 are used in 2-D. All three eigenvalues are used in 3-D.

Table II. The components of the parameter vector $\xi \in \mathbb{Z}^9$, their respective ranges, default values, and constraints on their values for the tracing algorithm.

LIST OF FIGURES

Fig. 1. Illustrates the traditional and the proposed self-optimizing image segmentation approaches. (a) Traditionally, a parameter settings vector $\xi \in \Omega$ is set empirically and may need readjustments for each input image $\mathcal{I}$. (b) In the proposed framework, a global optimization algorithm efficiently explores the parameter space Ω driven by a segmentation quality assessment value. The user optionally specifies a single universal parameter to override the conciseness-coverage tradeoff.

Fig. 2. The realign and shrink iterative exploitation process in RRS, shown for a 2-D parameter space. (a) If a more promising sample ξ_2 is found during random sampling, the current sample space $\mathcal{S}_1$ centered at the current sample ξ_1 is realigned (moved) to the sample space $\mathcal{S}_2$. (b) Otherwise, the parameter space is shrunk to $\mathcal{S}_3$.

Fig. 3. Trivial automatic thresholding example illustrating the proposed methodology across two different applications. (a) Image of a fluorescently labeled neuron captured by a widefield microscope. (b) The multi-scale vesselness measure $\mathcal{V}_{[1,4]}(\mathbf{x})$. (c) Plot of the objective function q against the threshold τ value, with the optimal threshold value $\hat{\tau} = 28$. The pixel-label array representation for $\mathcal{M}$ with a relative representational inefficiency $w = 12.1$. (d) The optimal segmentation $\hat{\mathcal{M}}$ (using $\hat{\tau}$).

Panels (e & f) show a human retinal vasculature image and the corresponding optimal segmentation, using the same representation for $\mathcal{M}$, hence $w = 12.1$.

Fig. 4. An exhaustive search varying just two parameters, $\rho \in [1,20]$ and $\nu \in [1,10]$, with others fixed at default values, using vesselness measures $V_{[2,4]}(\mathbf{x})$. (a) Image of fluorescently-labeled neurites. (b) Multi-scale vesselness $V_{[2,4]}(\mathbf{x})$. (c) The optimization landscape. Panels (d through g) show the corresponding segmentations $\mathcal{M}$ in green color using ρ and ν values as pointed in panel (c). Other parameters are left unchanged at their default values (see Table II).

Fig. 5. Displays the global optimal and least optimal segmentations using exhaustive search in nine-dimensional parameter space. (a) Image of a neuron captured using fluorescence microscopy. (b) The global least optimal segmentation. (c) The global optimal segmentation.

Fig. 6. Illustrates the worst-case improvement over default settings. (a) A fluorescently-labeled neuron image. (b) Segmentation using the default settings. (c) Optimal segmentation with 1000 RRS iterations.

Fig. 7. A 9-dimensional parameter search using the RRS algorithm. (a) Fluorescently-labeled neuronal dendrite imaged using a widefield microscope. (b) Globally optimal segmentation $\hat{\mathcal{M}}$ obtained by exhaustive search of the parameter space, using $V_{[1,5]}(\mathbf{x})$. (c) The worst-case RRS-optimal segmentation $\hat{\mathcal{M}}_{RRS:1000}$ out of 50 RRS runs that is within 3% of the global minimum value. (d) RRS performance in 50 runs highlighting the best and worst cases along with the average performance.

Fig. 8. Summary of automatic parameter selection performance on a batch of 29 images from the same neurobiological study. The dark bars show the quality of the default settings (higher values are worse). The blank bars indicate the results of tuning each of the images individually using 1000 RRS iterations. The shaded bars indicate the estimating the optimal parameter set for the first image using 1000 iterations, and reusing them for the rest of the batch.

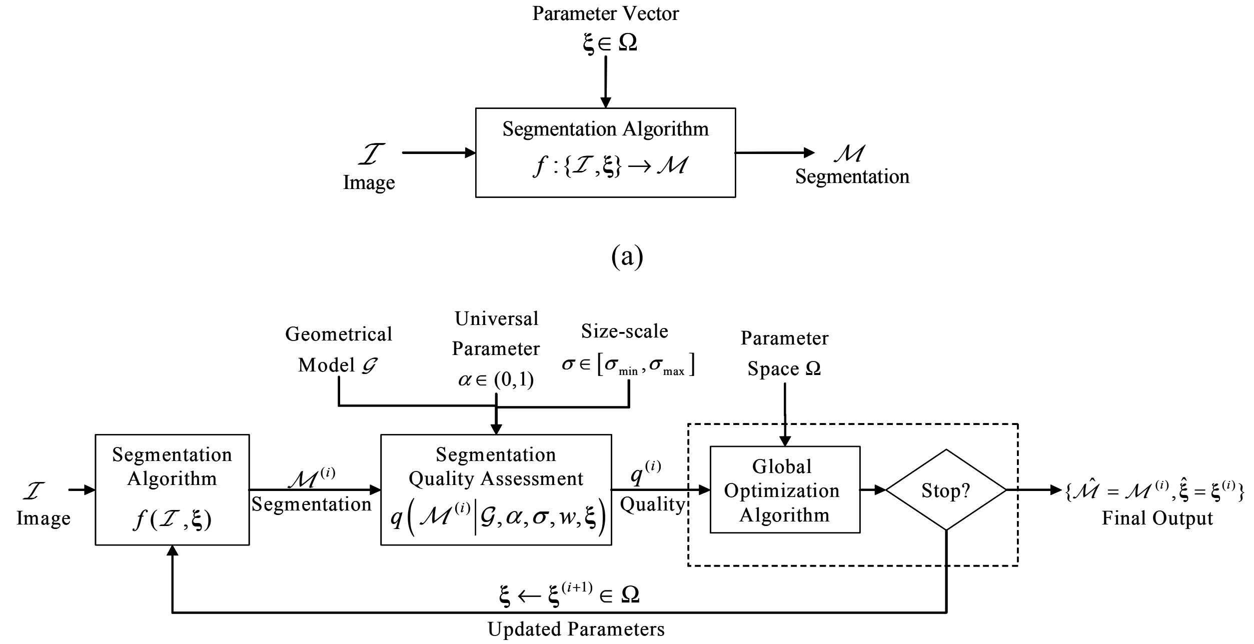
Fig. 9. Illustrates the iterative procedure for tracing generalized cylinders. Starting with initial seeds, the algorithm estimates the next location based on a robust estimate of the local boundary of the cylinder.

TABLE I

Geometrical Model $\mathcal{G}$	λ_1	λ_2	λ_3
tube-like	≈ 0	$\ll 0$	$\ll 0$
blob-like	$\ll 0$	$\ll 0$	$\ll 0$
plate-like	≈ 0	≈ 0	$\ll 0$

TABLE II

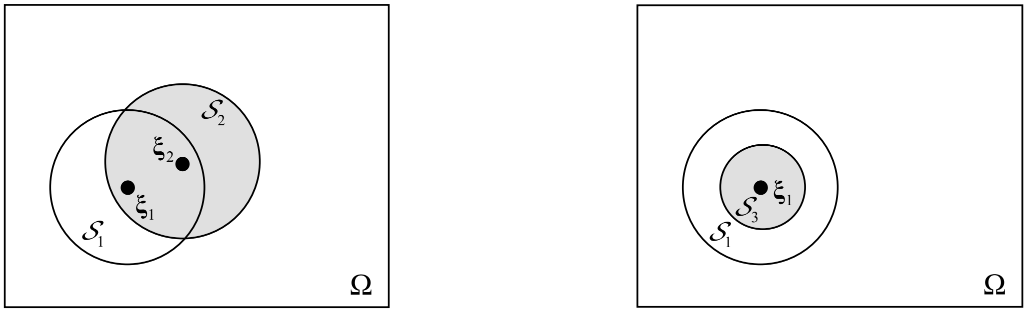
Parameter	Range	Default Value	Constraint on Values
grid spacing g	10 – 30	15	divisible by 5
minimum template length $L_{\min}$	8 – 20	10	$L_{\min} \le L_{\max}$
maximum template length $L_{\max}$	8 –30	18	$L_{\min} \le L_{\max}$
relative shift distance n_{shift}	2 –10	2	$n_{shift} \le M/2$
directional degree of freedom n_{rotate}	3 –7	7	divisible by 2
maximum shift distance M	5 –40	30	$M \ge 2n_{shift}$
maximum step size $s_{\max}$	3 –20	8	$s_{\max} \le L_{\min}$
contrast threshold multiplier ρ	1 –10	3	none
maximum allowed stopping violations ν	1 –10	1	none



(a)

(b)

FIG. 1



(a) (b)

FIG. 2

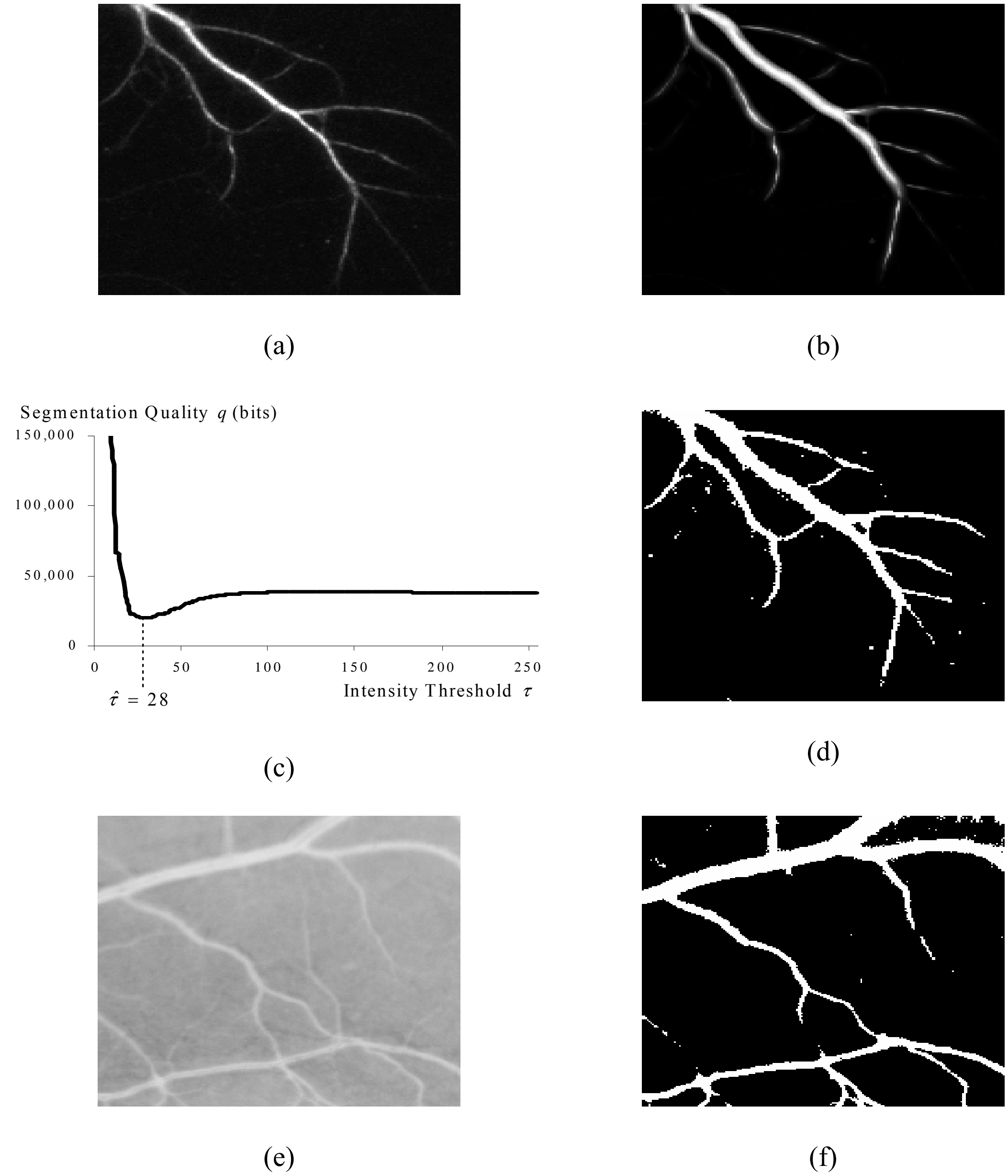
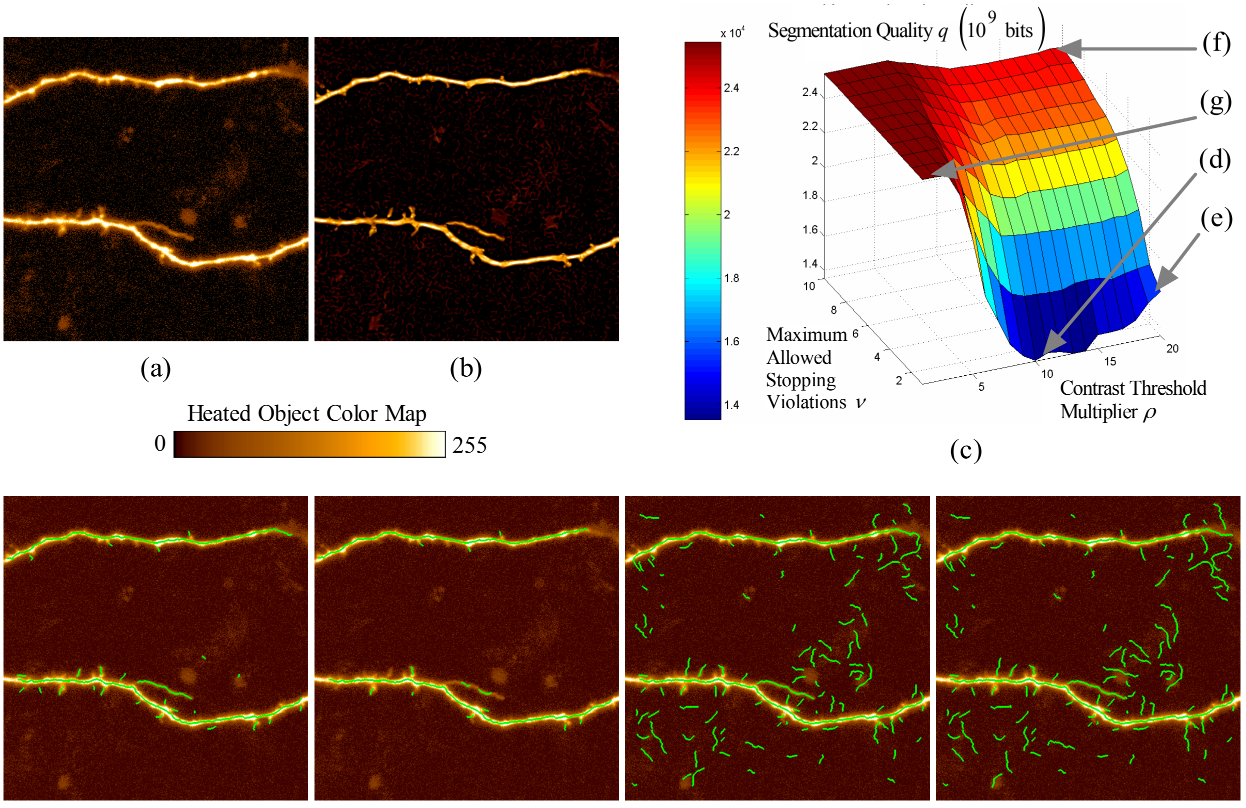

(a)
(b)
Segmentation Quality q (bits)
150,000
100,000
50,000
0
0
50
100
150
200
250
Intensity Threshold τ
$\hat{\tau} = 28$
(c)
(d)
(e)
(f)

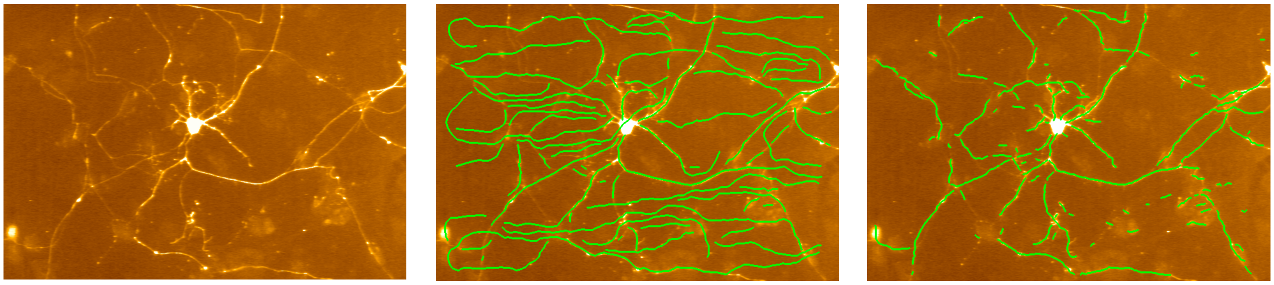
FIG. 3



(a) (b) (c)

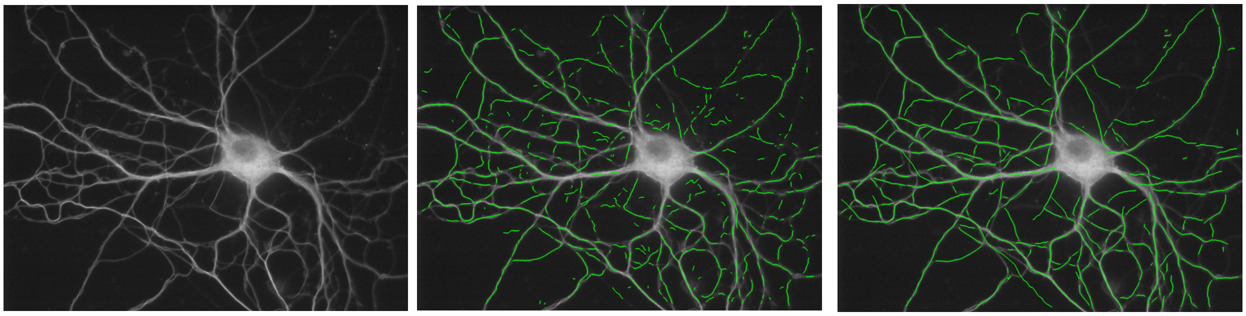
(d) (e) (f) (g)

FIG. 4



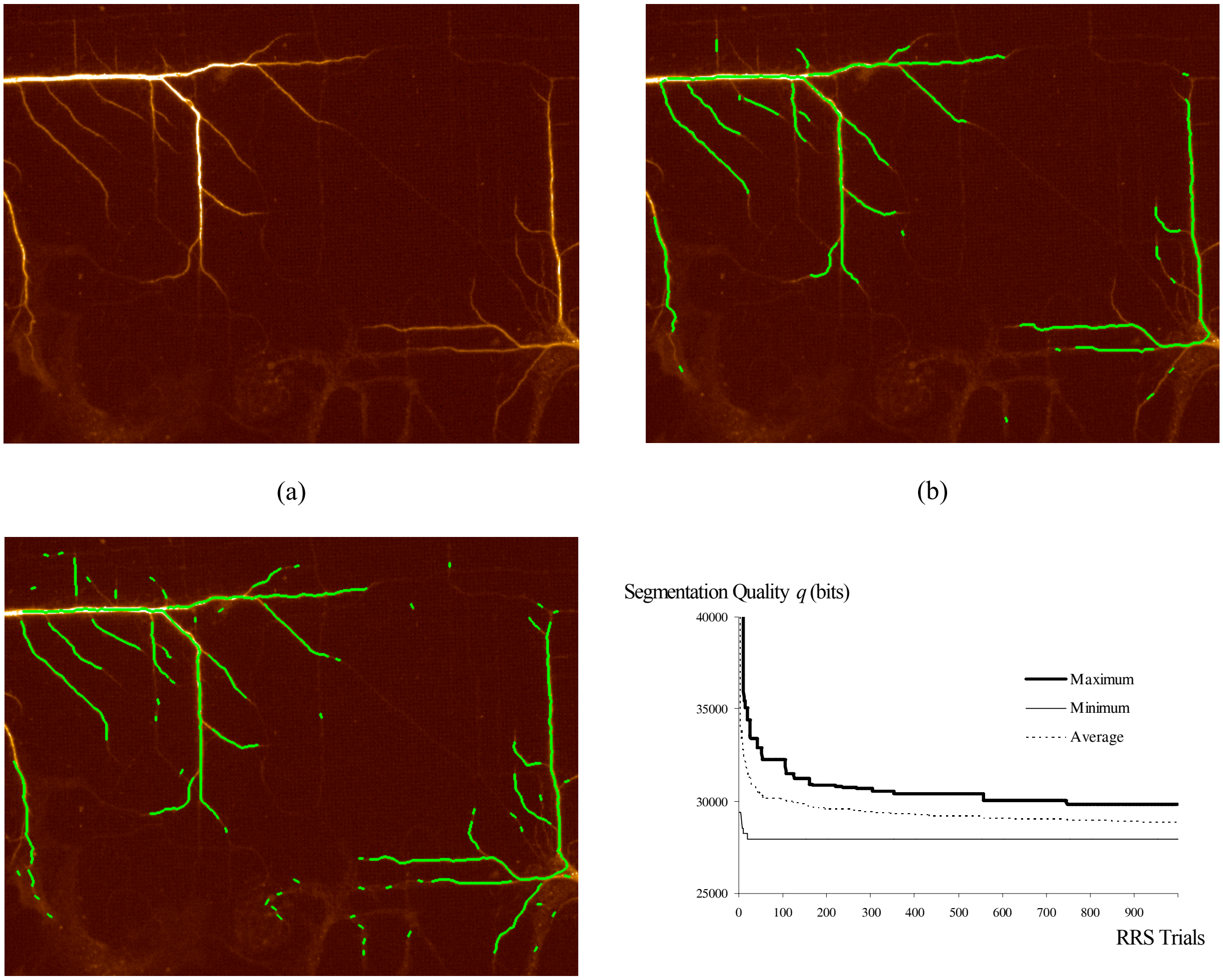
(a) (b) (c)

FIG. 5



(a) (b) (c)

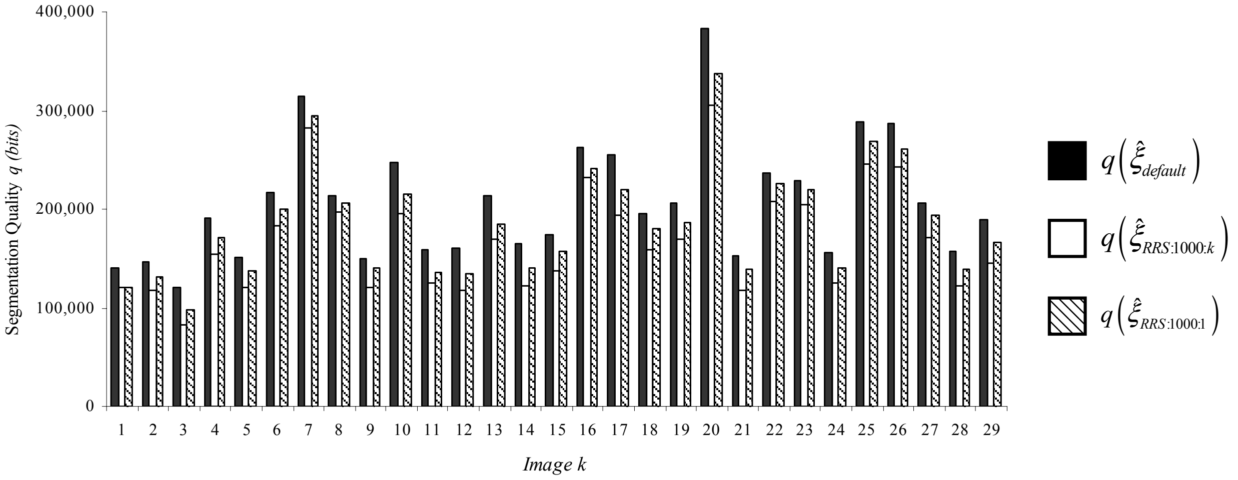
FIG. 6



(a) (b)

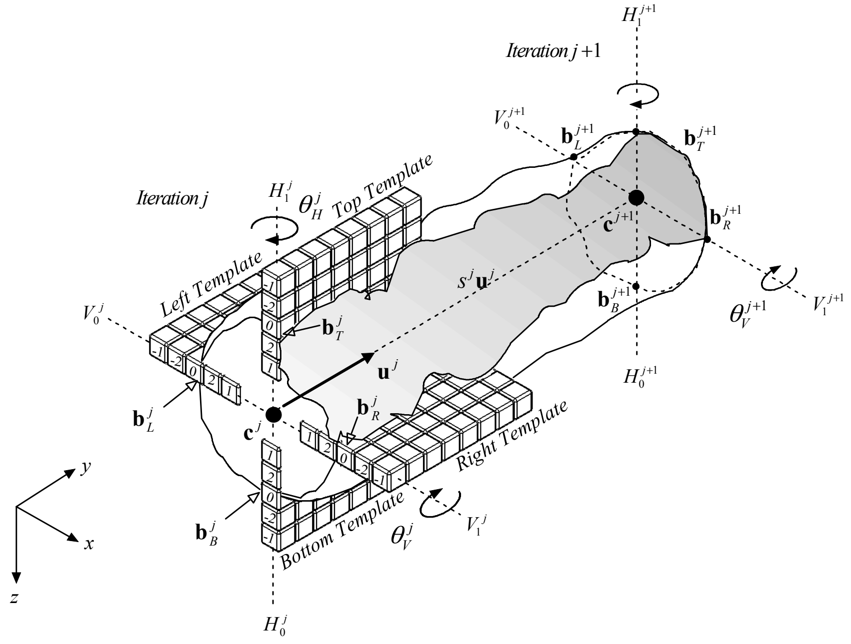
(c) (d)

FIG. 7



400,000
300,000
200,000
100,000
0
Segmentation Quality q (bits)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
Image k
$q\left(\hat{\xi}_{default}\right)$
$q\left(\hat{\xi}_{RRS:1000:k}\right)$
$q\left(\hat{\xi}_{RRS:1000:1}\right)$

FIG. 8



H_1^{j+1}
Iteration $j+1$
V_0^{j+1}
$\mathbf{b}_L^{j+1}$
$\mathbf{b}_T^{j+1}$
$\mathbf{c}^{j+1}$
$\mathbf{b}_R^{j+1}$
$s^j\mathbf{u}^j$
$\mathbf{b}_B^{j+1}$
θ_V^{j+1}
V_1^{j+1}
H_0^{j+1}
Iteration j
H_1^j
θ_H^j
Top Template
Left Template
V_0^j
$\mathbf{b}_T^j$
$\mathbf{u}^j$
$\mathbf{b}_L^j$
$\mathbf{c}^j$
$\mathbf{b}_R^j$
Right Template
Bottom Template
$\mathbf{b}_B^j$
θ_V^j
V_1^j
H_0^j
-1 -2 0 2 1
y
x
z

FIG. 9