Curve-like Structure Extraction Using Minimal Path Propagation with Backtracking

Yang Chen, Member, IEEE, Yudong Zhang, Jian Yang, Qing Cao, Guanyu Yang, Jian Chen, Huazhong Shu, Senior Member, IEEE, Limin Luo, Senior Member, IEEE, Jean-Louis Coatrieux, Fellow, IEEE and Qianjing Feng

Abstract—Minimal path techniques can efficiently extract geometrically curve-like structures by finding the path with minimal accumulated cost between two given endpoints. Though having found wide practical applications (e.g., line identification, crack detection and vascular centerline extraction), minimal path techniques suffer from some notable problems. The first one is that they require setting two endpoints for each line to be extracted (endpoint problem). The second one is that the connection might fail when the geodesic distance between the two points is much shorter than the desirable minimal path (shortcut problem). In addition, when connecting two distant points, the minimal path connection might become inefficient as the accumulated cost increases over the propagation and results in leakage into some non-feature regions near the starting point (accumulation problem). To address these problems, this paper proposes an approach termed Minimal Path Propagation with Backtracking (MPP-BT). We found that the information in the process of backtracking from reached points can be well utilized to overcome the above problems and improve the extraction performance. The whole algorithm is robust to parameter setting and allows a coarse setting of the starting point. Extensive experiments with both simulated and realistic data are performed to validate the performance of the proposed method.

Index Terms—Curve-like structure, Centerline, Minimal path tracking, Backtracking, Endpoint problem, Shortcut problem, Accumulation problem

I. INTRODUCTION

MINIMAL path techniques can efficiently extract curve-like structures by optimally finding the integral minimal-cost path between two seed points [1]. Successful applications of minimal path techniques have been found in contour completion [2], tubular surface segmentation [3-7], vascular centerline extraction [8-11], skeletonization [12-13] and motion tracking [14]. Minimal path techniques are fast and can avoid local minima by efficiently finding the global energy minima. As noted in [11], minimal path techniques can effectively locate tiny vessels and overcome vessel crossing and inhomogeneous intensity distribution in presence of stenoses or image degradations. However, some inherent problems should also be noticed for these minimal path based techniques: 1) two endpoints must be user-defined with enough precision for each line of interest (endpoint problem), 2) the connection might fail when the geodesic distance between the two points is much shorter than the desirable minimal path (shortcut problem), 3) for two distant points, the search of minimal path tends to become inefficient as the cost accumulates during the propagation with an increasing risk of leakage into some non-relevant regions (accumulation problem). Addressing these three issues is of major importance in dealing with complex topologies (i.e., multiple branches or lines), noise and inhomogeneous contrasts.

In [15], an approach called Minimal Path Method With Keypoint Detection (MPWKD) was proposed to find a curve using only one specified endpoint. This MPWKD method detects some representative key points along the target curve via a front propagation and terminates the propagation process when a desired curve length is reached. This method is however limited in practice because this prior length information is often hard to be effectively provided for complex topologies and scenes. In [16], to obtain a multi-branch extraction without setting each endpoint, Li et al. developed a 4D iterative key point searching scheme, in which the minimal action map and the Euclidean length map were calculated via the freezing fast marching evolution reported in [17]. Later, a minimal path method with no endpoint requirement was proposed in [18]. This method tracks a curve structure by detecting the key points along the curve trajectory using the maximum Euclidean distance summation proposed in [15]. Only one starting point lying in the target curve needs to be preset to get this curve but others must be manually defined in case of discontinuities. In the work described in [18], a propagation stopping criterion was built via evaluating the geodesic...
distances in backtracking, and additional operations aimed at handling false key points, circle disconnection and the short-cut problems were also used. However, even with these improvements, missing branches can still be observed in the results with [18]. A multiscale orientation descriptor was also reported in [19] to characterize local vessel configurations in X-ray images and thus alleviate the multi-path and the leakage problems. We would note that there are other ways to extract curve-like structures, such as the watershed segmentation [20-21], anomaly segmentation [22], pixel cueing [23] and vesselness-shape based filtering [24], et al.

In this study, we develop a solution termed Minimal Path Propagation with Backtracking (MPP-BT) to face the three above problems. The MPP-BT method first applies a minimal path propagation from one single starting point and then, at each reached point, traces certain steps back to the starting point. Here, the original idea of “backtracking” can be found in the backtracking algorithms for constraint satisfaction problems in computer science and graph theory, which was proposed as a more efficient algorithm than the brute force enumeration in searching the solution [25-27]. But the “backtracking” in the proposed approach goes beyond the basic “tracking backward” operation by fully exploiting the information on visiting preference and cost increments during this backtracking process to give an overall effective structure extraction. A robust stopping strategy is built by evaluating the evolution of cost increments in backtracking during the propagation. It must be noticed that only a coarsely user-defined starting point is required for the whole structure extraction. The final complete curve-like structures can be obtained via an additional breakpoint-connecting operation. The rest of the paper is organized as follows. In Section II, after a short review of minimal path methods and their limitations, the proposed MPP-BT method is explained in detail. In Section III, several experiments were conducted on real 2-D crack images and 2-D/3-D vessel images in order to validate this approach. Experimental results show that the feature-preferable backtracking can be well utilized to build up knowledge to overcome the above three problems for minimal path based methods. The parameter sensitivity and computation cost are also analyzed in this section. Concluding remarks and future issues are provided in Section IV.

II. METHODS
A. Application of Minimal Path Theory in Curve-like Structure Extraction

Minimal path methods extract curve-like structure in an image \( f \) by searching the connected path with a contour dependent minimum integrated energy between two user-preset points, a starting point and an end point. We first define \( E(C) \) as the integrated energy along a path \( C \) according to a given cost function:

\[
E(C) = \int_{\Omega} \hat{P}(C) \, ds = \int_{\Omega} (P(C(s)) + \omega) \, ds \tag{1}
\]

where, \( \Omega \) is the data space, and \( E(C) \) represents the energy along the path \( C \). \( \omega \) is the regularization term (often a real positive constant) and \( C(s) \in \mathbb{R}^n \) is a parameterized curve with the arc length \( s \), i.e., \( \|C(s)\|^2 = 1 \).

\( P(C(s)) \) in Eq. (1) denotes the cost (or potential) values for the points in path \( C \). \( P \) can be practically considered a pixel-wise cost map, whose values are calculated assuming that feature points have smaller values than non-feature points in image \( f \). Thus, the map \( P \) should be built according to the specific properties of the target images. Simple or more sophisticated information like the image intensity [1-2], the medialness measures [24], [28] or the sphere-based 4D curves [8] can be used.

The minimal path cost \( U(p_s, p) \) between the current point \( p \) and the starting point \( p_s \) is defined as the minimum integrated cost among all the possible paths between the two points:

\[
U(p_s, p) = \inf \left\{ E(C), C \in A(p_s, p) \right\}
= \inf \left\{ \int_{\Omega} \hat{P}(C) \, ds, C \in A(p_s, p) \right\}
\]

where \( A(p_s, p) \) is the set of all possible paths linking \( p_s \) and \( p \), from which the minimal path between \( p_s \) and \( p \) can be efficiently found by applying either the Dijkstra algorithm [8], [19], or fast marching methods [15], [17]. In this paper, the Dijkstra algorithm was used to solve Eq. (2) for its simplicity and intuitiveness. The Dijkstra algorithm can be applied by first setting all the node costs to infinity and then using an explicit discrete front propagation with direction pointing from the current minima to its neighboring nodes [19]. 8-connected or 26-connected neighboring nodes are routinely used in the cost calculation for the 2-D or 3-D domain data, respectively [1]. The cost for each reached point will be updated if a smaller cost \( U \) accumulated from the starting point is found. At each step, a priority queue is built to find the next propagation point with the smallest cost \( U \) among all the reached points ordered in a minimum heap data structure. Then following the connection information stored in the previous front propagation, the discrete minimal path from each grid point \( p \) to the starting point \( p_s \) (the minimum-accumulative cost between the two points) can be distinctly determined by tracing from point \( p \) backward to the starting point \( p_s \).

In the case with no end point specified, solution to Eq. (2) is in fact a feature-preferred propagation from the starting point, and we define this process as minimal path propagation.

B. Problems with Minimal Path Methods

In most applications of minimal path techniques, the question is mainly to find a way to reduce the user interaction when the target objects of interest have complicated topologies. Among the three problems reported before when applying minimal path techniques, the endpoint one is intuitive and does not need special comments. So, we focus our illustrations on the two others, the shortcut problem and the leakage problems. They are exemplified in Fig.1 and Fig.2, in which the simulated images were
generated by adding a closed curve with low intensity contrast on a noisy stationary background (we got this simulated image from the paper in [18] with the copyright permission from the publisher). Here, image intensities were used to build the cost function in Eq. (1), and the Dijkstra algorithm in [19] is applied to give minimal path tracking. In Fig.1, depicting the shortcut problem, the line is open and the starting point is very close to the end point. Instead of extracting the desirable line in Fig.1(c), the Dijkstra algorithm often tends to output the near shortest line corresponding to the shortest distance between the two endpoints, as seen in Fig.1(b). Also, with the same simulation but a closed structure, the result in Fig.2 shows that, as the minimal path tracking proceeds, more and more non-relevant points near the starting point are traversed before the end point is reached to give the desirable minimal path connection (overlaid in red line). It was recorded that 7708 points needed to be traversed before the specified end point was reached. As the cost accumulates, the minimal path tracking becomes less efficient as the inclusion of more non-relevant points. This problem is termed accumulation problem. It is notable that a method called “A*” algorithm can improve the efficiency of minimal path tracking by using a knowledge-plus-heuristic cost function from both the starting point and the end point [31-32]. But the “A*” algorithm still suffers from the end point problem because the positions of the end points are required in calculating the heuristic cost.

C. The Minimal Path Propagation with Backtracking (MPP-BT) Approach

1) Backtracking Operation in Minimal Path Propagation

As illustrated in Fig.3, if we trace back the minimal path from each reached grid point \( p \) to the starting point \( p_s \), feature points (the points located inside the target curve-like structures) will receive much more revisits than non-feature points (the points located outside the target curve-like structures). This is due to the fact that feature points always have smaller cost values than non-feature points’ and the back-traced path is also the one with the minimal accumulated cost. In most time, such backtracking will reach feature points after some steps if the minimal path propagation is limited to the region around the target structures. The information of visiting preference and cost increments in such backtracking process can be exploited to overcome the drawbacks pointed out in the previous paragraph. This algorithm is termed Minimal Path Propagation with Backtracking (MPP-BT), and this backtracking idea was first applied in our previous work in [30] to build a centerline constraint for the region growing algorithm in vessel segmentation. This MPP-BT approach is detailed as below. We first initiate the minimal path propagation with the Dijkstra algorithm from a starting point \( p_s \). For each grid point \( p \) reached by the propagation front, we calculate the cost value \( P(p) \) according to Eq. (1) and then track \( l_{bk} \) steps from point \( p \) backward the starting point \( p_s \) based on the connection information obtained in the previous minimal path propagation. The backtracking is stopped if the starting point \( p_s \) is reached before \( l_{bk} \) steps. The minimal path propagation is controlled via a stopping strategy explained below to limit the propagation within the region around the target features. This implies that such backtracking always reaches a feature point after \( l_{bk} \) steps. For each last point
(denoted by $p_{Eb}^k$) in each backtracking path, we accumulate the reciprocal of its cost value $P\left(p_{Eb}^k\right)$ for point $p_{Eb}^k$ to form a feature map $\hat{l}_{nk}$:

$$\hat{l}_{nk} \left(p_{Eb}^k\right) = \hat{l}_{nk} \left(p_{E}^n\right) + \frac{1}{\eta + P\left(p_{Eb}^k\right)}$$ (3)

where $\eta$ is a small positive constant to avoid a zero-valued denominator. Considering the fact that the feature points will get much more visits in the backtracking operations than non-feature points, the feature points get much more accumulation in Eq. (3) than non-feature points. Note this accumulation is only imposed on the last points other than on all the points in backtraced paths, and this will lead to less computation cost than the method in [30]. We can obtain an intermediate binary feature map $\hat{l}_c$ by thresholding the values in $\hat{l}_{nk}$ using $\hat{l}_{nk}^u$ (the points in $\hat{l}_{nk}$ are set to 1 if $\hat{l}_{nk} > \hat{l}_{nk}^u$, and 0 otherwise, with $\hat{l}_{nk}^u$ denoting the $\alpha$-quantile value of all the non-zero values in $\hat{l}_{nk}$ [29]). We then backtracking $l_{bh}$ points from each point $\bar{p}$ in $\hat{l}_{nk}$, and include each reached point $\bar{p}_{bh}$ into the current feature image (set $\hat{l}_c\left(\bar{p}_{bh}\right) = 1$) if the last point $\bar{p}_{bh}$ in the backtracked path is found to be a feature point $\left(\hat{l}_c\left(\bar{p}_{bh}\right) = 1\right)$. The intermediate feature map $\hat{l}_c$ is further refined in this way. Here a limited backtracking operation is used by limiting the backtracking steps to $l_{bh}$ points to ensure a well-distributed accumulation on all feature points. This process does not require any endpoint, and the endpoint problem can be solved in this way. We would also note that this backtracking operation shares similar idea with the “Geodesic voting” method in [2], but with a much robust performance by limiting the accumulation number and using the cost reciprocal in accumulation.

Also, the shortcut problem and accumulation problem can be solved by a cost resetting scheme based on the cost increments along the backtracking paths. We reset the accumulated cost $U\left(p\right)$ of each reached point $p$ to the difference between the costs of the current point $p$ and the last point $p_{Eb}^k$ in the path back-traced from point $p$. In this way, the original cost accumulated from the starting point $p_s$ is reduced to the cost accumulated from the last point $p_{Eb}^k$ in the back-traced path. This reduced cost can be used as the minimal path cost for point $p$ because the last points in the back-traced paths are feature points in most cases. Fig.4 displays the evolution of the traversed regions (in white) as the minimal path propagation proceeds. Image intensities are used in building the potential map $P$. Compared to the result in Fig.2, we can see in Fig.4 that this cost resetting scheme leads to improved tracking efficiency, in other words, the desirable connection can be obtained with much less points traversed than the result for the original minimal path tracking in Fig.2 (1826 vs 7783). The accumulation problem can thus be overcome by this cost resetting scheme. We also implemented the same minimal path tracking to connect the two points in Fig.1, and we found that the desirable connection in Fig.1 (c) can always be obtained with this cost resetting scheme used.

2) Propagation Stopping Criterion

A stopping criterion is required to ensure that the MPP-BT algorithm be stopped when the target region of interest has already been traversed. This is crucial both for avoiding introducing wrong connection from non-feature structures and reducing the computational load. Considering the fact that the feature points with lower costs are always preferably visited during backtracking, the cost difference $U\left(p\right) - U\left(p_{bh}\right)$ along the backtracking path will significantly increase when the propagation starts to pick up non-feature points with much larger $P$ values after the traversing of the feature region. A pixel-wise metric $S\left(p\right)$ termed backtracking speed is used to quantify the backtracking distance over cost increment:

$$S\left(p\right) = \frac{l_{bh}}{U\left(p\right) - U\left(p_{bh}\right)}$$ (4)

where $p_{bh}$ is the point with $l_{bh}$ steps back-traced from the current point $p$. Also, for some points near the starting point $p_s$, the starting point might be reached before $l_{bh}$ steps in backtracking, and in this case we simply reset the $l_{bh}$ to the steps required to trace from $p$ to $p_s$. Thus, this propagation speed $S$ always has higher values in feature region than in non-feature region. Its behavior can be reflected by the illustrations in Fig.5 (a1)-(d1) with four test images (two crack images in (a1) and (b1), and two vessel images in (c1) and (d1)) with the starting points in red. By
plotting the propagation speed \( S(p) \) for each reached point \( p \) in Fig. 5 (a2)-(d2), some ruptures of the propagation speed values are observed to be followed by decaying tails which often points to the end of the propagation in feature region, and the MPP-BT algorithm should be stopped at this time. We also provide in Fig. 5 (a2)-(d2) the maps of the reached points for the different propagation phases (indicated by dotted red lines). Based on above observations, a \textbf{StopPropagation} criterion was derived by evaluating the evolution of cost accumulation along the back-traced paths. Instead using the point-based speed values, too sensitive to local fluctuations, the normalized average speed \( NS(p) \) was preferred:

\[
NS(p) = \frac{\bar{S}(p, I_{\text{AVE}})}{S_{\text{max}}(p)} \tag{5}
\]

where, \( \bar{S}(p, I_{\text{AVE}}) \) is the average of \( S \) values over the \( I_{\text{AVE}} \) points reached before the point \( p \). Such averaging operation is used to smooth the local abrupt variations as it can be seen for propagation speed values in Fig.5 (a2)-(d2). In calculating \( \bar{S} \) values for the first \( I_{\text{AVE}} - 1 \) points (with less than \( I_{\text{AVE}} \) points reached before the current point \( p \)), the input points for averaging operation only include those points reached before point \( p \). We plot in Fig.5 (a3)-(d3) the calculated normalized average speed versus the reached points for the images in (a1)-(d1), in which a smooth decaying can be obviously observed. The propagation is stopped when we find there are \( I_{e} \) successive points with lower \( NS \) values than a dynamically varying parameter \( NS_{\text{min}} \). In implementation, this parameter \( NS_{\text{min}} \) is initialized to a preset value \( NS_{\text{min}} (0 < NS_{\text{min}} < 1) \), and will be updated to the minimum \( NS \) value of all the already reached points when one new reached point is found.
than the current $N^5$ than the current $N^5_{\text{min}}$. The stopping points for the data in Fig. 5 are specified by the red points connected by dotted lines in the third column (Fig. 5 (a3)-(d3)). This stopping strategy was applied in the experiments described in Section III and demonstrates a good overall robustness.

**ALGORITHM: MPP-BT algorithm**

1) **Parameter setting:**
Define the starting point $P_s$ in the input image data. Set the backtracking step number to $l_b$. Initialize the feature image $I_C$ to 0. Set the coefficient parameter $\alpha$ for the $\alpha$-quantile calculation. As to the propagation stopping criterion, preset the values of $l_{SM}$, $l_{AVE}$ and $l_E$ (0, 1000 and 5000 in this study), and initialize the parameter $N^5_{\text{min}}$ to $N^5_{\text{min}_0}$. Set $\text{StopPropagation}$ to FALSE;

2) **Propagation and backtracking:**
Perform minimal path propagation using the Dijkstra algorithm from the starting point $P_s$, and the cost value $P(p)$ in Eq. (1) is calculated for each reached point $p$. Store the connection information as the propagation proceeds.

While $\text{StopPropagation}=\text{FALSE}$.

For each reached grid point $p$.

Trace $l_{bk}$ points back from each grid point $p$ toward the starting point $P_s$.

For each revisited point $P_{bk}$ in the backtracking path.

If $P_{bk} = P_s$ (the starting point $P_s$ was already reached).

Break;

End

End

For the last point (denoted by $P_{bk}$) in each backtracking path, we accumulate the reciprocal of its cost value $P(P_{bk})$, and store into a feature map $I_{bk}$ the point value:

$I_{bk}(P_{bk}) \equiv \frac{1}{P(P_{bk})} + \frac{1}{(\eta + P(P_{bk}))}$

Calculate $S(p)$ and $N^5(p)$ using (4) and (5).

If ($N^5(p) < N^5_{\text{min}}$)

$l_{sm} = l_{sm} + 1$;

Else

$l_{sm} = 0$;

Update $N^{5}_{\text{min}}$ to the minimum $N^5$ of all the already reached points;

End

If $l_{sm} = l_e$.

$\text{StopPropagation} \equiv \text{TRUE};$

End

End While.

Sort all the non-zero values in $I_{bk}$, and count the number of the values larger than 0, and use $I_{bk}^a$ to denote the $\alpha$-quantile value of all non-zero values in $I_{bk}$.

For each point $p$ in $I_{bk}$.

If $I_{bk}(p) > I_{bk}^a$

$I_C = 1$;

Else

$I_C = 0$;

End

End

End

For each point $p$ in $I_C$.

Trace $l_{bk}$ points back from each grid point $p$ toward the starting point $P_s$, denote the last point by $P_{bk}$.

For each revisited point $P_{bk}$ in the backtracking path.

If $P_{bk} = P_s$ (the starting point $P_s$ was already reached) Break;

End

End

If $I_C(P_{bk}) = 1$.

For each revisited point $P_{bk}$ in the backtracking path.

$I_C(P_{bk}) = 1$;

End

End

3) **Breakpoint connection (only for crack images and when more than one breakpoints are detected):**
Choosing breakpoint as those points with only one connected node in the intermediate feature map $I_C$, then apply the Dijkstra algorithm to connect each pair of the breakpoints to obtain the final feature image $I_C$.

3) **Outline of the MPP-BT Algorithm**

One problem with the operations described in Section

Fig. 6. Step by step illustration of the MPP-BT algorithm based on image intensities. (a) the original image with complex cracks; (b) the traversed regions (in white color) in the MPP-BT approach. (c) feature map $I_{bk}$; (d) the intermediate feature image $I_C$; (e) the specified breakpoints in green color; (f) the final extracted feature (crack) image $I_C$ after breakpoint connection. Note the dotted blue rectangles depict the connections with alternative paths.
2. CI is that some feature points might fail to be extracted if there exist alternative paths with smaller accumulated cost during backtracking. This situation is illustrated by the dotted blue rectangles in Fig.6(a) and (d). A breakpoint connection is thus applied to overcome this problem by applying the standard Dijkstra algorithm to connect each pair of breakpoints when more than one breakpoint are detected after the above procedure. To avoid repeated connections, we re-check the remaining breakpoints after each connection, and remove those with more than one connected nodes. Fig.6(e) shows the detected breakpoints as green points, and the final feature map \( I_c \) is illustrated in Fig.6(f). The main workflow of the MPP-BT algorithm can be the seen in the step by step results in Fig.6. Fig.6(a) is one crack image with the starting point given in red. Fig.6(b) depicts the traversed regions (in white), which shows that the stopping criterion works well in limiting the propagation within the feature region. The results in Fig.6(a)-(f) show that a complete crack structure can be obtained via this breakpoint connection. The overall algorithm is outlined in Algorithm MPP-BT.

III. EXPERIMENTS

The performance of the proposed algorithm has been assessed by carrying out experiments on crack and vascular images, including crack extraction tasks for 2-D crack images, and centerline extraction tasks for 2-D/3-D vascular data. The simulated curve image is also included in the experiment. The crack images include 40 typical images, which can be downloaded from the collected dataset via the link http://www.imagetech.com.cn/node/13. 2-D vessel images include twenty 512×512 angiogram images collected were collected from a GE rotational angiography system Cardiology Department of the University Hospital of Rennes, France. More details on the rotational angiography system can be found in [33]. The 3-D vessel datasets are composed by twenty cardiac CT angiography (CCTA) data acquired from a Siemens dual-source CT system (Somatom Definition Flash) in the Radiology Department of the First Hospital of Nanjing, China. Each 3-D CCTA includes around 270 2-D slices. The workstation is a PC (Intel Core™ 4 Quad CPU and 20 GB RAM, GPU (NVIDIA GTX560)) with Visual C++ as the developing language (Visual Studio 2008 software; Microsoft). Three experienced experts were invited to manually delineate the crack curves and centerlines in all 2-D crack and 2-D vascular images, and the position-averaged results of them were used as the reference. Experts routinely need to plot the centerline points in each 2-D slice to constitute the final 3-D centerlines, and significantly larger intra-observer variability occurs than the 2-D case. The reason is that the vessel regions are often irregular in 2-D slices, and there is no intermediate delineation to guide the ongoing plotting in the same 2-D slice illustration. In addition, the experts need some special training in this 3-D case to avoid very large intra-observer variability. So we only provide manual results for the 2-D vessel data, and the comparative results from a method based 3-D vesselness-shape filter in [24] were used in the evaluation for 3-D vessel data.

The MPP-BT method involves nine parameters to be set, namely, the \( l_{bk} \) to constrain the backtracking steps, the in Eq. (3), the coefficient \( \alpha \) of quantile value applied in feature map \( I_{sk} \) to get an intermediate feature image \( \hat{I}_c \). The propagation stopping mechanism includes the parameter \( l_{ave} \) in computing the normalized average speed, the successive point number \( l_{sk} \) and the initial value \( N_S^{\min} \) for \( N_S^{\max} \). We currently do not include this breakpoint connection in 2-D/3-D vessel data processing because many branch endpoints will be detected as the breakpoints, which makes the connection of each pair a large computation load. The intensity values were directly used to calculate the pixel-wise cost map \( P \) for crack image, and the “symmetric convexity” metric in [30] were used to calculate the pixel-wise cost \( P \) map for vascular images. The “symmetric convexity” metric was proposed based on the symmetry and convexity properties of local centerline points, and calculated as the reciprocal of the product of the 2 or 6 highest medialness measures among the 4 or 13 directions for the 2-D or 3-D vessel data, respectively. Based on [30], the parameters of the “symmetric convexity” metric include the \( \kappa \), \( \lambda \) and the vessel radius range \([ R_{\min}, R_{\max} ]\) (More details of the “symmetric convexity” metric are referred to [30]). Most parameters in TABLE I were selected based on the information (e.g. the cost map values, the feature map values) collected in the minimal path propagation and backtracking, and choosing the one with the good results. As to the “symmetric convexity” metric for the experiment with 2-D/3-D vessel data, the parameters \( \kappa \) and \( \lambda \) are used to control the balance between the symmetry and convexity values, and are set to 1 and 100 in all test. The radius range \([ R_{\min}, R_{\max} ]\) reflects the size range of the vessels, and is fixed to [2, 15] in the experiments. Though the starting points need not to be located on the target structures, an increased computation cost will still be needed if they lie far away from the target structures. So the starting points were set to be located in the regions around the target structures.

### TABLE I. PARAMETER SETTING FOR THE MPP-BT METHOD

<table>
<thead>
<tr>
<th>Test data</th>
<th>Parameter Settings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulated curve image in Fig.2(a) and the crack image in Fig.5(b1)</td>
<td>( l_{sk} = 15 ), ( \alpha = 0.7 ), ( \eta = 0.001 ).</td>
</tr>
<tr>
<td>Crack images</td>
<td>( l_{sk} = 15 ), ( \alpha = 0.7 ), ( \eta = 0.001 ).</td>
</tr>
</tbody>
</table>
| Vessel 
Data | \( l_{sk} = 15 \), \( \alpha = 0.7 \), \( \eta = 0.001 \). |
| 2-D data | \( l_{sk} = 15 \), \( \alpha = 0.7 \), \( \eta = 0.001 \). |
| 3-D data | \( l_{sk} = 15 \), \( \alpha = 0.7 \), \( \eta = 0.001 \). |

Note: The parameters in Table I were selected based on the information (e.g. the cost map values, the feature map values) collected in the minimal path propagation and backtracking.
as marked by the red points in the images in the first columns in Fig.8-Fig.12. Before processing, all the image data were extended 20 pixels with symmetric padding to give a fair backtracking upon the points near the image borders. Propagation in 3-D vessel data is limited to the points within the intensity range [1000HU, 2500HU] to avoid the computation cost on irrelevant points. Though the process of parameter selection includes some trial tests, the proposed MPP-BT method does not require much work in modulating parameters for each data set. As seen in TABLE. I, the same parameters can be used in processing the data in the same types, and the parameters \( l_{\text{bk}}, \alpha, \eta, l_{\text{ave}} \) can be set to be the same in processing all the data in different types. The extracted curve-like structures of 2-D crack and 2-D vascular images were quantitatively evaluated with the manual results using the following four metrics:

\[
TP = \frac{N_B \cap N_R}{N_R}, \quad FN = \frac{N_R - N_B \cap N_R}{N_R},
\]

\[
FP = \frac{N_B - N_B \cap N_R}{N_R}, \quad OM = 2 \cdot \frac{N_B \cap N_R}{N_B + N_R}
\]

where TP, FN, FP are the metrics of true positive, false negative, and false positive, respectively [34], and OM is an overlap metric known as a Dice similarity coefficient [35]. \( N_R \) is the number of classified feature pixels determined by in the manual results as the ground truth reference, and \( N_B \) is the number of the feature points extracted by the MPP-BT method. The OM index is equal to 1 when the reference and extracted curves can be exactly superimposed and 0 when they do not share any pixel.

A. Robustness to Different Starting points

The second to the fourth columns in Fig.7 illustrate the extracted curve-like structures for the images in the first column in Fig.7 by applying the MPP-BT algorithm with different starting points specified in red points. The parameters were set based on the values in TABLE.I. Note the extracted centerlines in Fig.7 (d2), (d3) and (d4) corresponds to the results with the starting points from left to right in Fig.7 (d1), respectively. We can see that the proposed approach is rather robust to starting point position and even a starting point far outside the target structure can lead to a good extraction. But a remote point should not be suggested as the starting point because this will increase the possibility of including non-vessel structures (e.g., the structures pointed by red arrow in Fig.7 (c4)), and also the computation cost in traversing.

B. Extraction Results

Fig.8 and Fig.9 illustrate the extractions of five 2-D crack images and four 2-D vessel images, respectively. From left to right in the columns of Fig.8-Fig.9, we have the original images, the traversed regions (in white color), the extracted feature images and the manually delineated feature images. The starting points are tagged in red in the original images in the first row from the left. We can see that the proposed MPP-BT method always leads to effective crack extraction with respect to the manually extracted results without being sensitive to the locations of the starting points. The traversed region the second columns in Fig.8 and Fig.9 shows that the proposed propagation stopping criterion works well in limiting the propagation within the feature region, confirming the results already displayed in Fig.6(b). We can also see some places in the resulting curves look a little thicker than the manual results in Fig.8 (see the arrows in third column in Fig.8), and this is due to the fact that more points will receive relatively large accumulations in the backtracking operations around some wide structures. But such thickening does not deteriorate the overall quality of extraction. We can note some crack structures are missing in the extraction result in Fig.8(e3). The reason is due to the fact that the current MPP-BT method is targeted at the extraction of independent structures, but the crack image in Fig.8(e1) is composed by several separated crack sections with clear gaps in between (pointed by the red arrows in Fig.8(e1)). The stopping mechanism stops the propagation before the reaching of other crack structures. Also, we can note some high contrast artifacts tend to result in some false structures in the final extractions for the 2-D vessel results in Fig.9 (see the red arrows in Fig.9). Fig.10 compares the performance on the complex crack image in Fig. 5(b1) between the proposed MPP-BT method and the minimal path type approach in [18], whose extraction was directly obtained from the paper. We can see our method leads to effective extraction with a more complete crack structure than the method in [18].

Currently, for most minimal path methods, two end points are still required to be set to give minimal path connection. So the multi-endpoint minimal path connection used in [1], [3], [4], [5], [8], [10], [11], [12] is chosen as the baseline method in evaluating the proposed approach. The crack images in Fig.8 are used in this comparison, with the results given in Fig.11. Note the same energy function as the proposed method is used in the method of multi-endpoint minimal path connection. In the first and second columns in Fig.11, the start points and the multi-endpoints are tagged in red for the proposed method and the multi-endpoint method. In the third and the forth columns in Fig.11, we provide overlapping crack extractions with respect to the manual results for the two methods. The overlapped parts, missing parts and the wrong extractions are respectively denoted by red, light blue and the dark blue, respectively. The results in Fig.11 show that the proposed method can achieve results close to those of the multi-endpoint connection, expect for the image in Fig. 8 (e1) in which two disconnected segments are failed to be extracted by the proposed approach.

Fig.12 illustrates the extraction of centerlines on four 3-D CCTA volumes. The columns from left to right represent the starting point positions in 2-D slices, the regions (in red color) traversed when applying the MPP-BT method, the extracted centerlines, and the centerlines overlaid on the original CCTA volumes and the extracted centerlines using the vesselsness filter in [24], respectively. The centerline extraction algorithm in [24] is in fact a vesselness-shape filtering with a new step-edge response function to overcome the interference of some high contrast non-centerline structures (e.g., the cardiac chambers boundaries), and this method was implemented based on the parameter setting in [24] to give the best visual result. The first and third rows are the illustrations of the left coronary
Fig. 7. Extraction results for the images in first column with different starting points (in red). Image intensities are used in building the potential map $P$ in minimal path propagation. The second to fourth columns are the extracted crack structures for the image in first column. Note the extracted 3-D centerlines in Fig. 7 (d2), (d3) and (d4) corresponds to the results with the three starting points from left to right in Fig. 7 (d1), respectively.
Fig. 8. Extraction results of 2-D crack images using the MPP-BT method. The rows from left to right correspond to the original images, the traversed regions (in white color), the extracted crack images, and the manually delineated crack images. The starting points are tagged as the red points in the images in the first column.

Fig. 9. Extraction result of four 2-D vessel images (from the first row to the fourth row) using the MPP-BT method. From left to right, the first, second, third and fourth rows correspond to the original images, the extracted centerline images, the traversed regions (in white color), and the manually delineated centerline images. The starting points are tagged as the red points in the images in the first column.

Fig. 10. Comparison with the curve-extraction method in [18] for the complex crack image in Fig. 5(b1). (a), (b), (c) and (d) correspond to the original crack image with the starting point in red, the extracted crack image using the proposed MPP-BT method, the extracted crack image using the method in [18] and the manually delineated crack image.
Fig. 11. Comparison with the multi-endpoint minimal path connection for the crack images in Fig. 8(a1)-(e1). From left to right, the first, second, third and fourth rows correspond to the results of the proposed method, the results of multi-endpoint minimal path connection, the overlapped results of the proposed method and manual delineation, and the overlapped results of the multi-endpoint minimal path connection and manual delineation. The start point for our method is tagged as the red point in first column. The end points for multi-endpoints minimal path connection are all tagged as the red points in second column.
Fig. 12. Centerline extraction results of four 3-D vessel volumes from the first row to the last row. The columns from left to right correspond to the starting point positions in the specific 2-D slice, the traversed regions (in red color), the extracted centerlines and the extracted centerlines overlaid on the original CCTA volumes, and the results from the method in [24].

<table>
<thead>
<tr>
<th>Crack Images</th>
<th>TP</th>
<th>FN</th>
<th>FP</th>
<th>OM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fig. 8a1</strong></td>
<td>0.7102</td>
<td>0.7081</td>
<td>0.2398</td>
<td>0.2939</td>
</tr>
<tr>
<td><strong>Fig. 8b1</strong></td>
<td>0.7669</td>
<td>0.7652</td>
<td>0.2111</td>
<td>0.2149</td>
</tr>
<tr>
<td><strong>Fig. 8c1</strong></td>
<td>0.6601</td>
<td>0.6332</td>
<td>0.3299</td>
<td>0.3568</td>
</tr>
<tr>
<td><strong>Fig. 8d1</strong></td>
<td>0.5358</td>
<td>0.5564</td>
<td>0.4642</td>
<td>0.4436</td>
</tr>
<tr>
<td><strong>Fig. 8e1</strong></td>
<td>0.4133</td>
<td>0.8152</td>
<td>0.5867</td>
<td>0.1848</td>
</tr>
</tbody>
</table>

Average: 0.7349, 0.7489, 0.2668, 0.2425, 0.0892, 0.0907, 0.8034, 0.8159

<table>
<thead>
<tr>
<th>2-D Vessel Images</th>
<th>TP</th>
<th>FN</th>
<th>FP</th>
<th>OM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fig. 9a1</strong></td>
<td>0.7757</td>
<td>0.2243</td>
<td>0.1397</td>
<td>0.8099</td>
</tr>
<tr>
<td><strong>Fig. 9b1</strong></td>
<td>0.8635</td>
<td>0.1365</td>
<td>0.1063</td>
<td>0.8767</td>
</tr>
<tr>
<td><strong>Fig. 9c1</strong></td>
<td>0.8392</td>
<td>0.1608</td>
<td>0.3152</td>
<td>0.7790</td>
</tr>
<tr>
<td><strong>Fig. 9d1</strong></td>
<td>0.8046</td>
<td>0.1954</td>
<td>0.1217</td>
<td>0.8354</td>
</tr>
</tbody>
</table>

Average: 0.8204, 0.1796, 0.2220, 0.8040

<table>
<thead>
<tr>
<th>Crack Images</th>
<th>Computation Cost (in seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fig. 8a1</strong></td>
<td>0.0853, 0.1915, 3989, 186, 0.2561</td>
</tr>
<tr>
<td><strong>Fig. 8b1</strong></td>
<td>0.0767, 0.1138, 3869, 218, 0.2863</td>
</tr>
<tr>
<td><strong>Fig. 8c1</strong></td>
<td>0.1045, 0.3952, 4552, 608, 0.4352</td>
</tr>
<tr>
<td><strong>Fig. 8d1</strong></td>
<td>0.2617, 4.7556, 11899, 1027, 1.1054</td>
</tr>
<tr>
<td><strong>Fig. 8e1</strong></td>
<td>0.0775, 0.4266, 3143, 319, 0.5509</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2D Vascular Data</th>
<th>Computation Cost</th>
<th>Traversed point numbers</th>
<th>Extracted point numbers</th>
<th>3D Vascular Data</th>
<th>Computation Cost</th>
<th>Traversed point numbers</th>
<th>Extracted point numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fig. 9a1</strong></td>
<td>1.6524</td>
<td>21991</td>
<td>2443</td>
<td><strong>Fig. 12a1</strong></td>
<td>22.3624</td>
<td>19072</td>
<td>1293</td>
</tr>
<tr>
<td><strong>Fig. 9b1</strong></td>
<td>1.3425</td>
<td>14232</td>
<td>1339</td>
<td><strong>Fig. 12b1</strong></td>
<td>26.3319</td>
<td>22097</td>
<td>1500</td>
</tr>
<tr>
<td><strong>Fig. 9c1</strong></td>
<td>1.5829</td>
<td>21720</td>
<td>2883</td>
<td><strong>Fig. 12c1</strong></td>
<td>21.6033</td>
<td>17989</td>
<td>1455</td>
</tr>
<tr>
<td><strong>Fig. 9d1</strong></td>
<td>1.3775</td>
<td>16110</td>
<td>2544</td>
<td><strong>Fig. 12d1</strong></td>
<td>17.5778</td>
<td>14197</td>
<td>1150</td>
</tr>
</tbody>
</table>

The results from both the proposed MPP-BT method and the multi-endpoint minimal path connection are also given in TABLE II, in which the best results in terms of TP, FN, FP and OM are given in bold. Factors such as the structure gaps and high contrast edges will lead to lowered performance with respect to these quantitative values. We can see that the structures extracted by the proposed method match well with the manual extractions, and can give scores close to that of the multi-endpoint method. In TABLE II, it can be noticed that for the proposed method the worst TP, FN, FP and OM values correspond to the case of the crack image Fig.8(e1), in which two disconnected segments cannot be extracted.

D. Computation Complexity

It was found that the traversed regions only take a small portion of the total grid points (around 4% for all the 2-D crack/2-D vessel images and 0.03% for all the 3-D vessel columns in average). This means that the MPP-BT method leads to a higher efficiency in comparison with those based on operation on all the grid points (e.g., the vesselness-shape filtering approach in [24]). The computation cost in processing different data is listed in TABLE IV and

trees, and the second and fourth rows are the illustrations of the right coronary trees. Visual evaluation of the 3-D extraction was performed under the guidance of an experienced radiologist (X.D.Y. with fifteen years experiences). Results in Fig. 12 show that, though failed to extract some specific branch (see the yellow arrow in the fifth column), the proposed algorithm can output an overall more complete 3-D centerline extraction than the method in [24] (see the structures pointed by yellow arrows in the third column). We should also note that some high contrast edges tend to be introduced in both the methods (see the blue arrows in the third and fifth columns in Fig. 12). And the proposed method requires much lower computation cost (around 25 seconds per 3-D volume in average) than the method in [24] (around 200 seconds per 3-D volume in average).

C. Quantitative Evaluation

To quantify the extraction precision, TP, FN, FP and OM values calculated via Eq. (6) are listed in TABLE II and TABLE III, which include both the individual images (in Fig. 8 and Fig. 9) and the averaged results of all test images.
TABLE V. TABLE IV also includes the computation cost for the multi-endpoint method. Note that the breakpoint connection step was not involved in the processing of the vascular data. Results in TABLE IV and TABLE V show the computation cost increases when more points were included in the traversing and more breakpoints were identified for the final extractions. We can see that Fig 8 (d1) required much more computation cost than other images due to the large numbers of traversed points. It is important to notice that the numbers of traversed points and of extracted points are not in proportional to the topological complexities for different data. TABLE IV also shows the proposed approach requires computation cost comparable to the multi-endpoint method even with the calculation in back-tracking step included. The results in these two tables show that higher computation cost in the propagation step is required for the vessel data than for the crack images because vessel structures often have more complicated topology with larger regions of interest. A higher computation cost in breakpoint connection is also noted for the image displayed in Fig 8(d1), which has more complicate structure topologies than other crack images. TABLE V also shows that the processing of the 3-D vessel data is much more costly than that of the 2-D vessel images because of the higher costs in calculating the “symmetric convexity” metric in 3-D domain.

E. Parameter Analysis

Though parameters should be suitably selected to give a good trade-off between performance and computation cost, we need not to perform troublesome parameter trials in processing each dataset for the MPP-BT algorithm. The same parameter setting can be used in processing the data of the same type, and some parameters (l_{ Bak}, \alpha, \eta, and l_{AVE} ) can even be set to the same values in processing the data in different types, as the values given in TABLE I. We can also find the same values of the parameters R_{ min}, R_{ max}, \kappa and \lambda can be used for all the 2-D and 3-D vessel data. Among all the involved parameters, the coefficient parameter \alpha (for \alpha-quantile calculation), the backtracking length l_{ Bak} and the successive point number l_{ E} (in propagation stopping strategy) should be set with special caution to ensure successful extractions. Fig.13, Fig.14 and Fig.15 illustrate the extraction results with different image types when the three parameters were chosen to different values.

Fig.13 illustrates the results of different data types when \alpha is set to different values. We can see some false structures tend to be introduced into the extraction if \alpha is set too high (see the blue arrows in the third column in Fig.13), and some curve segments will be missing otherwise. The value of \alpha is suggested to be chosen from 0.6 to 0.9 to give an effective extraction in practice (\alpha = 0.7 is used in this study). Fig.14 shows the influence of the backtracking length l_{ Bak} on the results. We find that a smaller l_{ Bak} leads to the extraction of more structures, but at the cost of introducing more false structures (see the blue arrows in the first column in Fig.14). For all the crack images and vessel data used in the experiment, a setting of l_{ Bak} = 15 can achieve good results. Fig.15 gives different results with different values of l_{ E}, which is used in the propagation stopping strategy to control the propagation process. We can note in Fig.15 that a small l_{ E} will lead to an early stop with some main structure missing (see Fig.15 (b1) and (c1)) and stable results can be obtained when the l_{ E} values are in some ranges (there is no obviously visual difference between the results in the second and third columns in Fig.15). But we will also point out that a large l_{ E} tends to increase the possibility of introducing irrelevant structures and results in higher computation cost. In the experiment, to give robust results for all the data, l_{ E} is respectively set to 1500 and 8000 for the 2-D images (including the crack images and vessel images) and the 3-D vessel data. The cost map P in the proposed approach is devised to provide discriminative values between the curve-like points and other points, so the image contrast has an influence on the algorithm performance. In the first, third and fifth columns in Fig.16, we compare the extraction results according to different contrast levels. The first column corresponds to the images with very low contrast. For the crack as well as vascular images with varying contrast, it is found that all the parameters can be set to the same values except N_{ SO}^{ min}. This parameter should be set to relatively larger values for low contrast images to include more points into extractions as shown in Fig. 16 (a2) and (b2). We can see in Fig.16 that the proposed method is rather robust to contrast variations in the target images, and can give effective extraction with the same parameter settings (including N_{ SO}^{ min}) when the contrast stays within a suitable range. However, it can be also seen that some centerline structures are lost when the image contrast is too low (see the red arrows in Fig. 16(b2)).

F. Stopping Strategy Analysis

The “normalized average speed” based stopping strategy is crucial to the performance of the proposed method. Structure missing or wrong inclusion will result if the minimal path propagation stops too early or too late. In Fig.5, we illustrated the evolution of the traversed regions as the propagation goes on for two typical crack images, one 2-D crack image and one 3-D vessel image. We can see that proposed stopping strategy is on the overall robust in limiting the propagation within the regions of the target structures. Even so, we should also point out that some early stopping will happen in the case of complicated structure topologies, which might lead to the missing of some vessel structures (e.g. the last row in Fig.8).

IV. CONCLUSION AND FUTURE WORK PLAN

In this paper, an approach termed MPP-BT is developed based on the intuitive observation that feature points with low cost always receive much more revisits than non-feature points. Information on visiting preference and cost increments in this backtracking process is fully exploited to overcome the endpoint problem, the shortcut problem and the accumulation problem commonly encountered when using approaches based on minimal path tracking. The backtracking idea is not new and has already been applied to improve minimal path tracking in [2] and [18]. However, the
The backtracking process in [18] is just aimed at detecting the keypoints along the tracking trajectory while in [2] it serves to highlight the tracking trajectory by simply adding one count to each backtracked point. The counting operation described in [2] may fail in discriminating feature points and non-feature points because the feature points near the start points always receive much more accumulations and much larger values than the feature points around the propagation.

Fig. 13. Extraction results with different $\alpha$ values (all other parameters were set based on TABLE I). The first to the last rows give the extraction results for the simulated curve image in Fig.4(a), the 2-D vessel image in Fig.9 (a), and the 3-D vessel data in Fig.12(a), respectively.

Fig. 14. Extraction results with different $I_{bk}$ values (all other parameters were set based on TABLE I). The first to the last rows give the extraction results for the simulated curve image in Fig.4 (a), the 2-D vessel image in Fig.9 (a), and the 3-D vessel data in Fig.12(a), respectively.
frontiers. Some non-feature points might also be wrongly connected by such straight counting in backtracking. The three problems can be well solved by the discriminative revisiting and the cost resetting scheme along the backtracking paths in the proposed MPP-BT method. To sum up, the proposed backtracking strategy in MPP-BT method includes an effective utilization of more information collected in the backtracking process to overcome the three above problems (endpoint problem, shortcut problem and accumulation problems) for the minimal path techniques (the latter two problems were not addressed in [2]). In addition, the information collected during the backtracking operation can also be well used for breakpoint connection and for building the stopping criterion to improve the extraction performance.

This MPP-BT algorithm was tested on 2-D crack images and 2-D/3-D vessel images. The experiment results show that the proposed algorithm can provide effective extractions of curve-like structures with only one roughly user-defined starting point. Better results can be expected if the image quality can be improved by suppressing noise and artifacts in the images [36], [37]. They also show that a coarsely-defined set of starting point works well for this MPP-BT method. The results in TABLE II and TABLE III confirm that the structures extracted by MPP-BT method match well with the manual extraction. From the parameter values in TABLE I, it has been shown that the same parameters can be used in processing the data of the same type, and some parameters can even be set to the same values in processing the data of different types.

However, we also noticed that the proposed method

![Fig.15. Extraction results with different \( I_c \) values (all other parameters were set based on TABLE I). The first to the last rows give the extraction results for the simulated curve image in Fig.4 (a), the 2-D vessel image in Fig.9 (a), and the 3-D vessel data in Fig.12(a), respectively.](image1)

![Fig.16. Extracted results of images with different contrast. (a2), (a4), (a6) are the extracted results of (a1), (a3), (a5), respectively. And (b2), (b4), (b6) are the extracted results of (b1), (b3), (b5), respectively.](image2)
might fail in extracting some structures if there are obvious gaps between them. Some high contrast edges or artifacts also tend to increase false curve-like structures in the final extractions. Early stopping of the propagation might happen when the structures of interest include highly complex topologies. Despite a good efficiency in processing 2-D crack and vessel images, the MPP-BT algorithm still requires around 20 seconds for processing one 3-D vessel volume. Furthermore, though requiring only a coarse setting, a starting point still needs to be selected in the MPP-BT algorithm. An automatic detection of the starting point will be devised to get a fully automatic implementation. Future work will be devoted to address all the above problems. We are also planning to extend the backtracking strategy to other segmentation or classification tasks [38], [39].

REFERENCES


