Computer Methods and Programs in Biomedicine 148 (2017) 99–112

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

Enhanced particle-filtering framework for vessel segmentation and tracking

Sang-Hoon Lee, Jiwoo Kang, Sanghoon Lee*

Department of Electrical and Electronic Engineering, Yonsei University, Seoul, 120-749, Republic of Korea

ARTICLE INFO

Article history:
Received 2 June 2016
Revised 25 May 2017
Accepted 23 June 2017

Keywords:
Vessel segmentation
Vessel tracking
Particle filter
Level set method

ABSTRACT

Background and Objectives: A robust vessel segmentation and tracking method based on a particle-filtering framework is proposed to cope with increasing demand for a method that can detect and track vessel anomalies.

Methods: We apply the level set method to segment the vessel boundary and a particle filter to track the position and shape variations in the vessel boundary between two adjacent slices. To enhance the segmentation and tracking performances, the importance density of the particle filter is localized by estimating the translation of an object's boundary. In addition, to minimize problems related to degeneracy and sample impoverishment in the particle filter, a newly proposed weighting policy is investigated.

Results: Compared to conventional methods, the proposed algorithm demonstrates better segmentation and tracking performances. Moreover, the stringent weighting policy we proposed demonstrates a tendency of suppressing degeneracy and sample impoverishment, and higher tracking accuracy can be obtained.

Conclusions: The proposed method is expected to be applied to highly valuable applications for more accurate three-dimensional vessel tracking and rendering.

1. Introduction

With the ongoing increase in life expectancy, cardio-vascular diseases have become the leading cause of death worldwide [1]. Therefore, there is an increasing demand for a method that can detect and track cardiac anomalies. Coronary artery anomaly detection considerably influences diagnoses because the coronary artery supplies blood to the heart muscle. Thus, the ability to segment the artery is essential for accurately detecting calcification or stenosis. However, the contrast of any coronary artery image is low in computed tomography/magnetic resonance imaging; thus, it is difficult to autonomously segment or track the vessel with high accuracy. Consequently, multiple studies have considered applying active vision to biological image processing in order to address the problems associated with artery segmentation and tracking [2].

Typically, blood-vessel segmentation technologies involve *segmentation* of each medical image slice and *tracking* of the object **between the slices**. One of the most representative segmentation methods is the level set method [3], which enables implicit representation of **the objects' contours**. To segment an object using the level set method, an initial contour is deformed until it minimizes the energy functional based on the features of the desired object. **Various models can be used to determine the energy functional, such as the edge-based [4,5] and region-based models [6–8]**. Among these, one of the most popular is the Chan–Vese (CV) model [7], which is based on Mumford–Shah segmentation techniques [6] and has been successfully applied to binary phase segmentation.

Many segmentation methods are based on the CV model [9–13]. However, these methods segment a vessel boundary by minimizing the energy functional at each frame; thus, they do not incorporate motion dynamics between frames into their tracking frameworks. At best, only discrete snapshots of the target object's location are provided and no dynamic or morphological consistency can be enforced. Thus, in addition to the two-dimensional segmentation information, the explicit introduction of dynamics to the curve evolution law should significantly improve vessel tracking [11].

With regard to tracking, extant research has primarily focused on Kalman filtering [14,15] wherein a series of tracking states is defined using a linear state-space model. However, a simple assumption of linearity cannot be applied to vessel segmentation due to the irregular variations in a vessel, such as sudden scale variation, direction change, and presence of bifurcations [16]. To overcome this problem, various extensions of the Kalman filter have been devised to track nonlinear systems, and the particle filter (PF) is one of the most successful schemes [17].

* Corresponding author.
E-mail address: slee@yonsei.ac.kr (S. Lee).

http://dx.doi.org/10.1016/j.cmpb.2017.06.017
0169-2607/

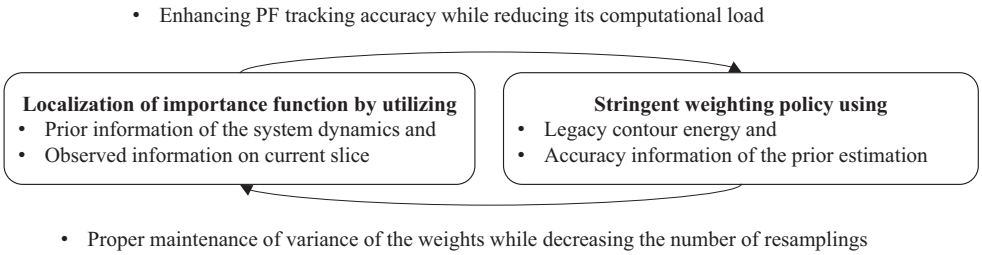

Fig. 1. Main contributions of the proposed segmentation and tracking (SNT) framework: accuracy enhancement and alleviating degeneracy and sample impoverishment of a particle filter (PF).

Many studies have focused on combining a PF with a contour deformation method [18–21]. In [21], the authors proposed a PF algorithm using the CV method to track moving and deforming objects. In that study, **a PF was used** to estimate the affine motion parameters that determine the global rigid motion of an object between two subsequent images. Then, the CV method was used to track the local deformation, which is the non-affine deformation of the contour. This concept of global motion tracking with local affine warping has been adopted by many general contour-tracking applications [22,23]; however, it has not yet been widely considered for vessel segmentation. Moreover, although the use of a PF has demonstrated considerable success in tracking literature, it incurs some fatal problems due to its suboptimal sampling mechanism in the importance sampling process, **such as** degeneracy and sample impoverishment problems [24–27].

The degeneracy problem involves a situation in which only a few particles have significant weights; thus variance between weights increases. This can be addressed by increasing the number of samples; however, this may be impractical [28]. Degeneracy can also be minimized by resampling or localizing the importance function. On the other hand, sample impoverishment occurs when particle diversity is reduced. With a resampling strategy, particles with large weights are likely to be drawn multiple times. If the number of resamplings can be reduced, the sample impoverishment problem can be mitigated. In such a case, a stringent weighting policy to decrease weight variances should provide an effective solution.

In this paper, we present a vessel segmentation and tracking (SNT) method based on the CV method to find the vessel contours in each slice and a PF to track vessel contours between slices. We also present a method to enhance SNT accuracy and reduce degeneracy and sample impoverishment problems. The main contributions of the framework proposed in this paper are shown in Fig. 1 and **summarized below**.

- **Localization of importance function**: To reduce the variance in the particle weights and localize the importance function, additional observed information is incorporated into the known process dynamics. In this way, an object's motion and location between slices can be estimated more precisely. Moreover, because location can be accurately predicted, the number of iterations of the CV method can be effectively reduced, and the overall computational load of the SNT algorithm **can be decreased**.
- **Stringent weighting policy**: To decrease the number of resamplings in the PF framework, it is important to adequately maintain variance in the weights. Thus, the particle importance must be evaluated very carefully. In the PF framework, the energy functional **of contour deformation** is generally applied to the calculation of particle weights. However, we introduce information about the object location prediction accuracy between slices. Through the application of this policy, it is possible to discriminate between particle weights, and thus maintain the variance.

The remainder of this paper is organized as follows. In Section 2, we introduce the PF and CV methods. The motivation behind the proposed method is given in Section 3. A detailed explanation of the proposed model is presented in Section 4. The simulation environment is described in Section 5. In Section 6, we apply the proposed model to synthetic and real medical images and show the results, and analyze and discuss them in Section 7. Conclusions are presented in Section 8.

2. Methods and problems

2.1. Particle filtering

A PF is a sequential Monte Carlo framework in which the key idea is to represent the required posterior density function by a set of random samples with associated weights [27]. Here, let X_t and Z_t be the *state* and *measurement*, respectively, at time index t. The unknown state is usually estimated by the posterior density function $p(X_t|Z_{1:\,t})$ **which** can be calculated recursively using the following Bayesian formula:

$$p(X_t|Z_{1:t}) \propto p(Z_t|X_t) \int p(X_t|X_{t-1})\, p(X_{t-1}|Z_{1:t-1})\, dX_{t-1}. \tag{1}$$

In the PF framework, (1) is approximated recursively using a set of particles. Let N_s be the number of particles, and $\left\{X_t^{(n)}, \omega_t^{(n)}\right\}_{n=1}^{N_s}$ be the set of particles and corresponding weights describing their relevance. Then, (1) is approximated as follows:

$$p(X_t|Z_{1:t}) \simeq \sum_{n=1}^{N_s} \omega_t^{(n)} \delta\left(X_t - X_t^{(n)}\right) \tag{2}$$

where $\delta(\cdot)$ denotes the Dirac delta measure. With this particle approximation, object-tracking is performed by the following procedure:

- **Prediction**: move the swarm of particles based on the dynamics as

$$X_t^{(n)} \sim q(X_t|X_{t-1}^{(n)}, Z_t). \tag{3}$$

- **Update**: calculate the particle weights based on the likelihood as

$$\omega_t^{(n)} \propto \omega_{t-1}^{(n)} \cdot p(Z_t|X_t^{(n)}) \tag{4}$$

where $q(\cdot)$ is the *importance density*. Note that the selection **of *importance density*** significantly affects PF tracking accuracy. The abovementioned PF is known as the *sequential importance sampling* (SIS) algorithm [28]. If the importance function is set to $q(X_t|X_{t-1}^{(i)}, Z_t) = p(X_t|X_{t-1}^{(n)})$, the SIS algorithm becomes a *bootstrap filter*, which is the most common choice. The main advantage of

a bootstrap filter is its simple implementation because it only requires sampling from the distribution $p(X_t|X_{t-1}^{(n)})$ and the evaluation of $p(Z_t|X_t^{(n)})$ [29]. However, it incurs the following disadvantages. 1) The prediction step uses only the previous state and does not consider the observed information, and 2) a degeneracy problem can arise. In other words, after a few iterations, most particles have negligible weights; thus significant computational effort is required to update particles with a very small contribution to posterior density.

2.2. Chan–Vese level set method

For a variational level set formulation, Chan and Vese proposed region-based image segmentation [7]. This approach has become very popular in the image processing community mainly due to its ability to detect objects not necessarily defined by a gradient. The basic concept of the segmentation method is to divide a particular partition of a given image I into two regions, where one region represents the objects to be detected and the other region represents the background. Then, the contour of the object is defined as the boundary between **these two regions**. For a given image I, they proposed minimizing the following energy functional:

$$E_{CV}(\Phi, I) = \lambda_1 \int_\Omega (I - c_1)^2 H(\Phi) dxdy + \lambda_2 \int_\Omega (I - c_2)^2 (1 - H(\Phi)) dxdy + \nu \int_\Omega |\nabla H(\Phi)| dxdy \quad (5)$$

where Ω is the image domain, and λ_1, λ_2, and ν are positive, user-defined weights. In addition, c_1 , c_2, and $H(\Phi)$ are defined as follows:

$$c_1 = \frac{\int I(x,y) H(\Phi) dxdy}{\int H(\Phi) dxdy}, c_2 = \frac{\int I(x,y)(1 - H(\Phi)) dxdy}{\int (1 - H(\Phi)) dxdy}, \text{ and}$$

$$H(\Phi) = \begin{cases} 1 & \Phi \geq 0 \\ 0 & \text{Otherwise} \end{cases} \quad (6)$$

where $I(x, y)$ is the pixel intensity and Φ is the level set function. If we regularize $H(x, y)$ and $\delta(x, y)$ using suitable smooth functions, such as $H_\varepsilon(\cdot)$ and $\delta_\varepsilon(\cdot)$, (5) can be minimized using a calculus of variations [30]. The resulting Euler-Lagrange equation is given as follows:

$$\frac{\partial \Phi}{\partial t} = \delta_\varepsilon(\Phi) \left[\mu \cdot \text{div}\left(\frac{\nabla \Phi}{|\nabla \Phi|} \right) - \nu - \lambda_1 (I - c_1)^2 + \lambda_2 (I - c_2)^2 \right]. \quad (7)$$

Then, the contour is deformed to the desired boundary using repetitive iterations of (7) until the energy reaches its local minimum point, or its iteration number (L) does not exceed the predefined maximum iteration number (L_{max}). The CV algorithm has many advantages, such as the ability to perform topology variation of the contour automatically **and to stabilize** global region information responses to local variations such as weak edges and noise. However, the performance of the level set method is limited due to the computational cost of embedding the contour in higher dimensional space. In particular, for object tracking, it is difficult to predict drastic shape changes of the object because the algorithm does not incorporate motion dynamics between frames into the tracking frameworks.

3. Motivation behind the proposed method

3.1. SIS PF framework for vessel tracking

The vessel tracking procedures based on the SIS PF are described in Fig. 2. The framework requires only one user interaction **on the slice** where vessel tracking begins. **In the initialization step**, the particles and their weights are initialized using the seed. In the proposed framework, the user needs to roughly select a region around the vessel on the first slice. Then, the particles are initialized based on the contour obtained by exploiting the CV algorithm, and the particle weights are set uniformly. **In the prediction step**, each particle (contour) moves toward the vessel boundary **on the consecutive slice** based on the dynamics in (4). **In the update step**, the particle weights are updated by (2) based on the particles predicted for that slice. In the proposed method, the CV algorithm is used to predict the particle changes on the slice. By combining the updated weights and predicted particles, a particle with the maximum a posterior (MAP) probability is selected **in the particle selection step**. These prediction, update, and particle selection steps are repeated for the following consecutive slices to track along the vessel until vessel tracking terminates.

With the SIS PF framework, the vessel is tracked along the slices and the optimal vessel segment (particle) on each slice is selected. This helps avoid vessel segmentation failures due to an incorrectly predicted seed, drastic changes **in vessel location** between consecutive slices, or image noise inherent in medical imaging devices. However, the general SIS PF framework has some drawbacks that limit its direct application to vessel tracking. In the following sections, we show the effectiveness of the proposed method compared to conventional SNT methods, which are similar to the proposed approach. In addition, the drawbacks of the general SIS PF framework and why framework localization is required for vessel tracking are explained relative to the importance function and weighting policy of the PF.

3.2. Effectiveness of the proposed method

Conventional SNT methods that are similar to the proposed method **have been proposed** [31,32]. In [31], the authors proposed a cerebral artery **tracking method based on an SIS PF**, which is analogous to the proposed method. However, the artery segment was represented by employing an ellipse model in terms of center position, normal vector, and major and minor axes. When the vessel on the plane is ellipse-shaped (Fig. 3(a)), it can be segmented almost successfully. However, if the shape differs significantly **from that of an ellipse** (Fig. 3(b)), it is difficult to model the appearance appropriately using an ellipse model; thus, tracking fails. In the proposed method, the segment is represented by its contour rather than a particular shape prior. Therefore, if the shape changes drastically, the proposed method demonstrates better segmentation performance than **the previously** proposed conventional method [31].

The proposed SNT approach is also similar to the method proposed in [32]. However, rather than the PF and level set method, that method employed a Kalman filter and the snake method to track and segment the vessel boundary, respectively. The Kalman filter uses a linear state-space model for tracking. However, as mentioned in Section 1, generally, simple linearity assumption is not suitable due to the intrinsic non-linear shape of the vessel. Through rigorous simulation, we found that the snake-based segmentation method has a parameterization problem and does not control the topology effectively. In contrast, the proposed scheme can be used to segment and track the vessel boundary effectively.

3.3. Localization of importance function

In the SIS PF framework in Fig. 2, for each slice, the particle weights are updated using the moved particles **in the prediction step**. Based on these weights, **the particle having** the maximum posterior probability is selected as the vessel segment on that slice. To approximate the posterior density, the state samples **are drawn**

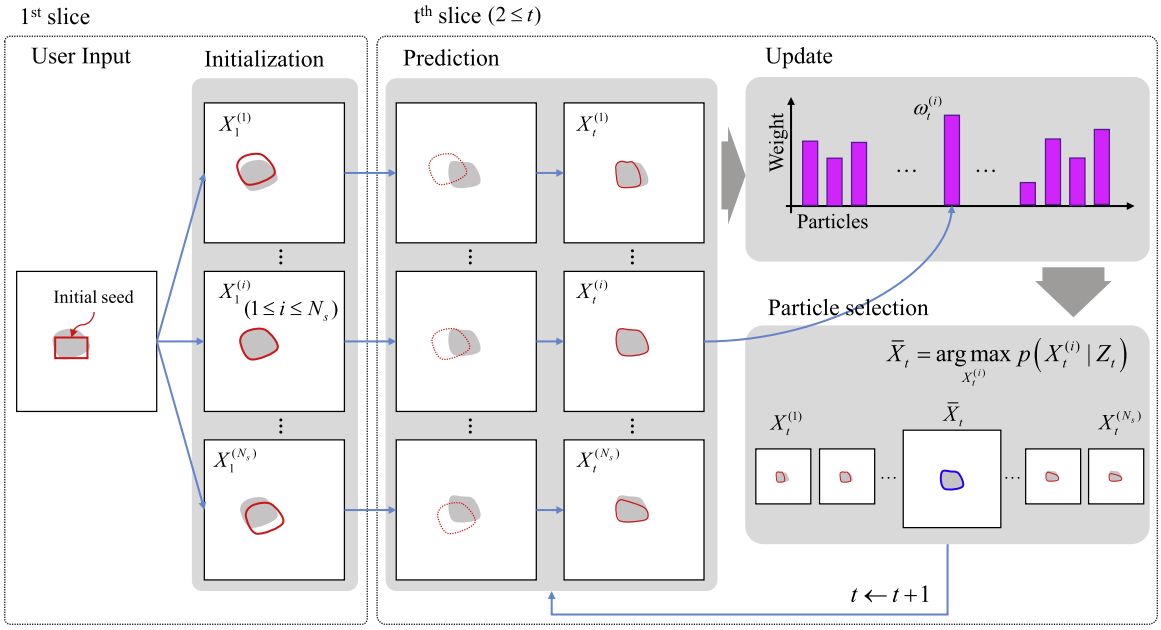

Fig. 2. SIS PF framework comprises four steps: initialization, prediction, update, and particle selection.

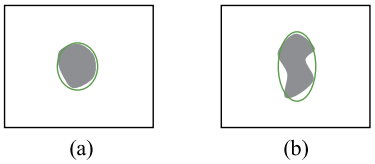

Fig. 3. (a) **Shape of the object** is highly correlated with the predefined shape model such that the object can be segmented successfully using an ellipse model [31]. (b) In contrast, if the correlation is low, it is difficult to appropriately segment the object using the model.

according to their importance density $q(X_t|X_{t-1}^{(i)}, Z_t)$ in (3). Therefore, the selection of $q(\cdot)$ significantly affects tracking accuracy. In most cases, $q(\cdot)$ relies on the current observation. However, obtaining an analytical expression is difficult; thus, for the bootstrap filter method, it is preferable to use the prior information $p(X_t|X_{t-1})$ for the importance density. Usually, this prior is learned from a training dataset that is obtained by **the manual segmentation** of a vessel. However, **this is a tedious processand requires** input from domain experts such as **a radiologist**. In addition, sampling from this distribution is not robust when faced with an abrupt change in the object.

For example, consider a vessel contour object tracking problem, such as that shown in Fig. 4. This problem can be decomposed into 1) translating the segmentation result of the previous slice (C_{t-1}) and 2) segmenting the object of the current slice using the curve deformation algorithm. In this case, the translated contour shown in Fig. 4(b) can be represented as $\hat{C}_t = C_{t-1} + D_t$ where D_t is a translation vector. The curve deformation process can be performed by L_t iterations of the energy minimization procedure, starting from the initial contour $\hat{C}_t$, as shown in Fig. 4(c). The translation vector $D_t = \left[d_t^x, d_t^y\right]$ is sampled from its prior distribution $p(D)$ as $D_t \sim p(D)$, and the pdf of the prior can be obtained by manual segmentation of the training data, as shown in Fig. 4(d). In other words, in the bootstrap filtering procedure, the state vector $X_t = C_t$ is predicted from the importance density $p(C_t|C_{t-1}) = p(D)$ and updated using the curve deformation procedure.

However, if the translation estimation procedure is performed using only $p(D)$, the location of the initial contour may be predicted inappropriately, i.e., $\hat{C}_t^{(1)}$, as shown in 5(a). In this case, as shown in Fig. 5(b), the segmentation result will be inaccurate, i.e., $C_t^{(1)}$, and this incorrect location of the initial contour results in a large iteration number $L_t^{(1)}$ **for object segmentation**. On the other hand, **if the observed information**, such as the pixel intensity of the target object, is incorporated into the importance density, the location of the initial contour will be **determined more accurately**, i.e., $\hat{C}_t^{(2)}$, as shown in Fig. 5(c). Consequently, as shown in Fig. 5(d), fewer iterations $L_t^{(2)}$ **are expected** to produce a more precise segmentation result, i.e., $C_t^{(2)}$, compared to $C_t^{(1)}$. Therefore, in the proposed framework, the observed information Z_t is used for calculating the importance density to achieve both an accurate SNT result and reduced computational load.

3.4. Stringent weighting policy

One of the most serious problems facing the SIS PF framework is that, as it progresses, only a few particles tend to have significant weights; thus the remaining particles have relatively small weights. Increasing variance of weights is called *degeneracy*. This degeneracy implies that a large computational effort is devoted to updating particles whose contribution to the approximation of the posterior density function is essentially zero. In other words, degeneracy increases useless computations **in the update step** and acts like decreasing the number of particles used as candidates of the vessel segment **in the particle selection step**, which degrades tracking accuracy. Generally, the degeneracy problem is dealt with by *resampling*, i.e., replacing samples according to importance weights ($\omega_t^{(i)}$) and renewing the weight equally as $1/N_s$ [33]. However, **excessive frequent** resampling causes a particle with a large weight to be drawn multiple times, whereas particles with small weights are abandoned. This phenomenon of decreasing the variance of particles leads to the so-called *sample impoverishment* problem.

Fig. 6 shows an example of a one-dimensional (1D) state vector, which illustrates the degeneracy and sample impoverishment problems. In this example, the PF framework approximates the posterior density (Fig. 6(a)) with a finite number of particles and their weights. In Fig. 6(b), circles of the same radius are distributed on the 1D line. The location of the circle's center represents each

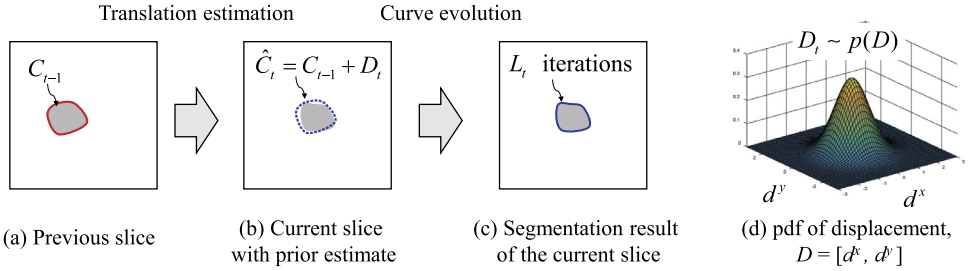

Fig. 4. Motivation behind the proposed method. The SNT problem is decomposed into translation estimation and curve evolution. The amount of translation can be estimated by drawing samples from the displacement vector's pdf.

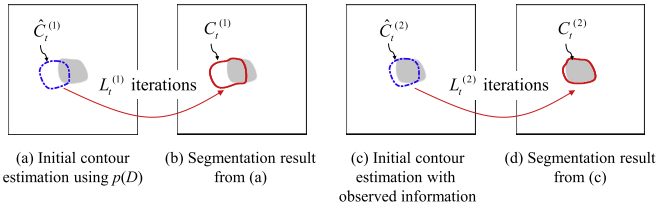

Fig. 5. Motivation behind the proposed method. (a) The prior contour $\hat{C}_t^{(1)}$ is obtained using only the prior distribution $p(D)$. (b) After $L_t^{(1)}$ energy minimization iterations, the curve is evolved from $\hat{C}_t^{(1)}$ to $C_t^{(1)}$. (c) Then, the observed information is utilized to estimate the prior contour $\hat{C}_t^{(2)}$ as well as $p(D)$. (d) After $L_t^{(2)}$ energy minimization iterations, the curve is evolved from $\hat{C}_t^{(2)}$ to $C_t^{(2)}$. The segmentation result $C_t^{(2)}$ is more accurate than $C_t^{(1)}$, however $L_t^{(2)} < L_t^{(1)}$.

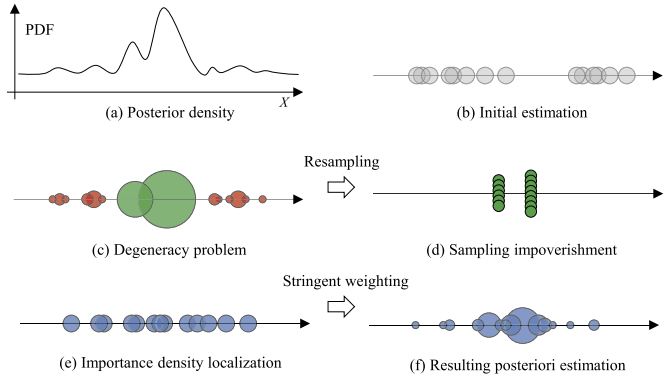

Fig. 6. One-dimensional (1D) state example of the importance sampling problem. (a) Posterior density, (b) initial estimates **of posterior density**, (c) degeneracy, (d) sample impoverishment problems, (e) appropriate scattering of particles, and (f) proposed posterior estimation by stringent weighting policy.

estimated 1D particle at the initial stage, while its radius describes the magnitude of the weight. In Fig. 6(c), the degeneracy problem is observed as few particles having a predominantly large radius. The sample impoverishment problem due to repetitive resampling is represented as circles that overlap at each specific region (Fig. 6(d)). These are the inherent problems of importance sampling in the PF framework. To resolve them, in the proposed framework, the importance function is localized appropriately with a limited number of particles, **as shown in** Fig. 6(e). In addition, a stringent weighting policy is applied to prevent specific particles from having large weights (Fig. 6(f)).

4. Proposed vessel SNT

4.1. System model

Let C_t be the contour for slice number t, which is represented as the zero level set of a signed distance function $\Phi_t(x, y)$, i.e., $C_t = \{(x, y) \in \Re^2 : \phi_t(x, y) = 0\}$. Let $D_t = \left[d_t^x, d_t^y\right]^T$ be a two-dimensional displacement vector representing the translation of the centroid of the contour with regard to the ***x*- and *y*-axes** [3]. The state vector X_t is defined using D_t and C_t, i.e., $X_t = [C_t, D_t]$, and the observation vector Z_t is defined using the image of the tth slice (I_t) and its CV iteration number (L_t), i.e., $Z_t = [I_t, L_t]^T$. The overall system model is summarized in Fig. 7.

The problem of tracking a deforming vessel segment can be separated into two parts; 1) tracking the global rigid motion of the object and 2) tracking local deformation in the shape of the object, which can be defined as any departure from rigidity (non-affine deformation). The major advantage of this method is that both local deformation and global motion of the contour can be measured effectively in spatial and temporal domains. Thereby, the object contour can be tracked while adapting to various shape dynamics including topology changes. New samples are generated by a resampling step similar to that in the general SIS PF process.

4.2. Prediction step

In the proposed SNT, a coarse contour transition is first performed**, and then,** the contour is evolved at a finer scale, starting from the coarse scale contour.

4.2.1. Coarse scale transition

The contour transition of each particle is modeled by the displacement vector, $D_t^{(i)}$ and can be written as follows:

$$D_t^{(i)} \sim N(\bar{D}_t, \Sigma_D) \text{ where } \bar{D}_t = \begin{bmatrix} \bar{d}_t^x \\ \bar{d}_t^y \end{bmatrix}, \text{ and } \Sigma_D = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix} \quad (8)$$

where Σ_D can be learned from the test dataset **and $\bar{D}_t$ is predicted** by the following proposed *block-based translation estimation* algorithm for each slice.

In block-based estimation, a block is defined to determine the amount of translation from the previous slice to the current slice. As shown in Fig. 8(a), a minimal block $\bar{B}_{t-1}$ containing $\bar{C}_{t-1}$ is found in the previous slice. Then, block $\bar{B}_{t-1}$ is parameterized in terms of the center position, width, and height as $\bar{B}_{t-1} = \left[x_{t-1}^b, y_{t-1}^b, w_{t-1}^b, h_{t-1}^b\right]$, in which the first two elements represent the center position and the third and fourth elements are the width and height, respectively. Once $\bar{B}_{t-1}$ is determined, block $\hat{B}_t$ is estimated to track the target object in the current slice using a block-based search algorithm. As shown in Fig. 8(b), for the current slice, a block ($\tilde{B}_t$) of the same size **as that of** $\bar{B}_{t-1}$ is generated and shifted pixel-by-pixel over the search window. For each position, the mean absolute error (MAE), which is the mean **difference in pixel intensity** between the previous and current blocks is calculated as follows:

$$\begin{aligned} &\mathrm{MAE}(n, m, B_{t-1}, t) \\ &= \frac{1}{h_{t-1}^b \cdot w_{t-1}^b} \sum_{v=0}^{w_{t-1}^b} \sum_{u=0}^{h_{t-1}^b} \begin{array}{l} \left| I_{t-1}(x_{t-1}^b + u, y_{t-1}^b + v) \right. \\ \left. \quad - I_t(x_{t-1}^b + u + n, y_{t-1}^b + v + m) \right|. \end{array} \end{aligned} \quad (9)$$

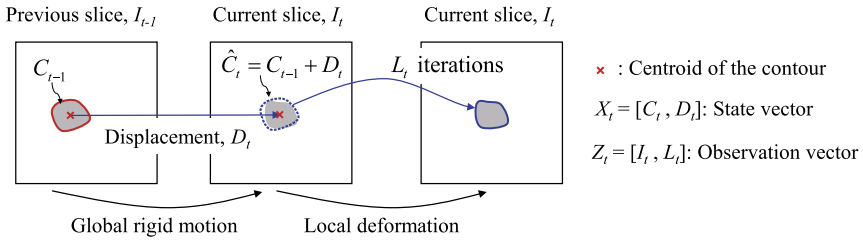

Fig. 7. System model: the state and observation vectors are defined as $X_t = [C_t, D_t]$ and $Z_t = [I_t, L_t]^T$, respectively. Displacement is defined using the difference between two centroids.

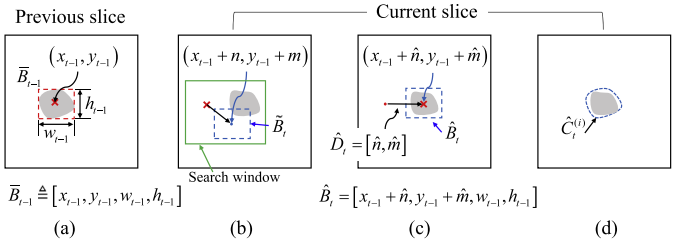

Fig. 8. Coarse-to-fine approach of SNT. (a) Minimum bounding block $\bar{B}_{t-1}$ at the previous frame, (b) one candidate for minimum bounding block $\tilde{B}_t$ at the current frame, (c) optimal minimum bounding block $\hat{B}_t$ for the current frame, and (d) initial contour of the ith particle $\hat{C}_t^{(i)}$.

The optimal displacement vector $\hat{D}_t$ can be found at the position of the minimum MAE as follows:

$$\begin{aligned}\hat{D}_t &= \left[\hat{d}_t^x, \hat{d}_t^y\right] \\ &= \left[\hat{n}, \hat{m}\right]^T \\ &= \arg\min_{(n,m)} MAE(n, m, B_{t-1}, t).\end{aligned} \tag{10}$$

Then, for the ith particle, the contours are translated by $\hat{D}_t^{(i)} = \left[\hat{n}^{(i)}, \hat{m}^{(i)}\right]$ as follows:

$$\hat{C}_t^{(i)}(x, y) = C_{t-1}^{(i)}(x - \hat{n}^{(i)}, y - \hat{m}^{(i)}). \tag{11}$$

Fig. 8(c) shows the optimal displacement ($\hat{D}_t$) as a result of the MAE procedure. The translated contour ($\hat{C}_t^{(i)}$) is depicted by the dotted line in Fig. 8(d).

4.2.2. Fine deformation

After the coarse prediction is complete, a fine prediction is performed to evolve each curve $\hat{C}_t^{(i)}$ by the CV algorithm until the contour reaches the energy saturation state. This means $[C_t, L_t] = f_{CE}(\hat{C}_t, I_t)$, where f_{CE} is the curve evolution function and L_t is the iteration number. During the iteration process, the curve evolution is terminated if the energy value in (5) is less than the predefined value E_{th}, or if the iteration number L_t reaches the predefined limit L_{max}. In this case, if the coarse scale estimation successfully locates the initial contour of the CV method, a small number of L_t will be required to conduct segmentation, and vice versa.

For example, Fig. 9(a) and (b) show the prior estimates and their segmented results for poorly and successfully determined locations of the initial contour, respectively. **In addition,** Fig. 9(c) represents the progress of the energy values with regard to an iteration number for each case. When the coarse scale estimation locates the initial contour ($\hat{C}_t^{(1)}$) poorly, its segmentation result ($C_t^{(1)}$) will be inappropriate, and its iteration number will be $L_t^{(1)} = L_{\max}$ because its energy value never falls below the predetermined value E_{th} during the curve deformation procedure. On the other hand, if the initial contour is successfully located as $\hat{C}_t^{(2)}$, the segmentation process will be performed successfully as $C_t^{(2)}$ despite its small iteration number $L_t^{(2)}$, relative to $L_t^{(1)}$. Therefore, if the importance density is localized appropriately via a coarse estimation, the SNT performance will increase. At the same time, the computational load for the fine scale **will decrease**.

4.3. Update and particle selection steps

In the update step, the likelihood associated with each particle is used to determine its weight. The likelihood indicates the degree of dissimilarity between the predicted and desired contours. For example, if the curves are too far from the intended result, the weight will have a small value, and vice versa. In this paper, we define a likelihood that depends on the combination of the **region-based energy** functional from the CV model in (5) and the iteration numbers of the CV model (L). For each particle:

$$p(Z_t|X_t^{(i)}) \propto p_{CV}(I_t|C_t^{(i)}) \cdot p_{IN}\left(L_t^{(i)}\right) \tag{12}$$

where $p_{CV}(\cdot)$ and $p_{IN}(\cdot)$ represent the region-based and iteration number likelihoods, respectively. The region-based likelihood is determined from the evolved contour as follows:

$$p_{CV}(I_t|C_t) \triangleq \exp\left(-\frac{E_{CV}(\Phi_t, I_t)}{\sigma_{CV}^2}\right) \tag{13}$$

where σ_{CV}^2 is the parameter controlling how quickly the exponential function converges to zero, and $E_{CV}(\cdot)$ is given in (5). On the other hand, the iteration number likelihood is related to the CV iteration number obtained from the fine transition described in Section 4.2.2. Specifically, for the iteration number likelihood, if the coarse estimation described in Section 4.2.1 is conducted successfully, the fine-scale estimation will require only a small iteration number. Conversely, particles with a small iteration number imply successful prior estimation, i.e., a small iteration number indicates a "*good particle*." This information can be incorporated into the likelihood function $p(Z_t|X_t^{(i)})$ to discriminate between particles, and the resulting iteration number likelihood is expressed as follows:

$$p_{IN}(L) \triangleq= \exp\left(-\frac{L}{\sigma_{IN}^2}\right) \tag{14}$$

where σ_{IN}^2 also effects how the exponential diminishes. By incorporating (12) into (4), particle weights are updated based on the region-based and iteration number likelihoods. When particle weights have been calculated, the particle with the MAP probability is selected as the estimator for time step t as follows:

$$\begin{aligned}\bar{X}_t &= \arg\max_{X_t^{(i)}} p\left(X_t^{(i)}|Z_t\right) \\ &\simeq \arg\max_{X_t^{(i)}} \sum_{n=1}^{N_s} \omega_t^{(n)} \delta\left(X_t^{(i)} - X_t^{(n)}\right).\end{aligned} \tag{15}$$

With this supplementary likelihood, the importance of a particle can be calculated more precisely, and the posterior density can be approximated more accurately.

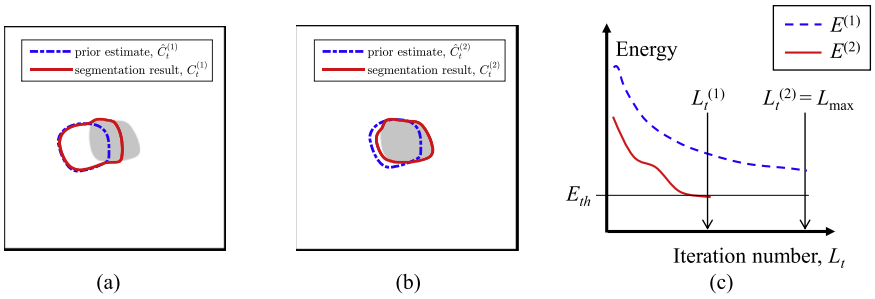

Fig. 9. (a) Segmentation result when the initial contour $\hat{C}_t^{(1)}$ is poorly located, (b) segmentation result with the successful initial contour $\hat{C}_t^{(2)}$, and (c) progress of energy values with regard to the iteration number for each case.

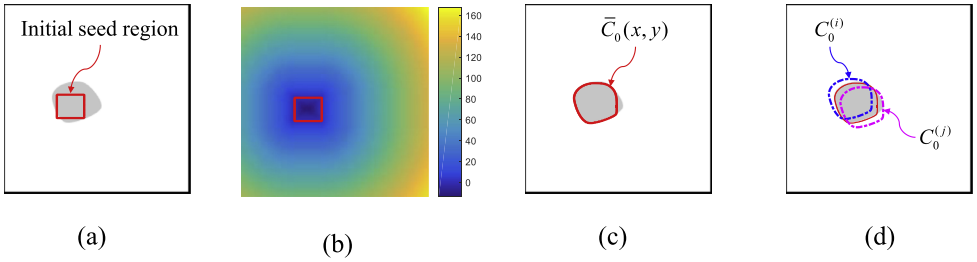

Fig. 10. Initialization of the PF-CV SNT procedure: (a) initial seed region by user input, (b) initial level set function, (c) initial maximum a posterior (MAP) estimate of the contour, and (d) initial particles of the contour.

In summary, with the stringent weighting policy in the update step, particles located on the initial contours are more likely to be traced successfully. This alleviates the concentration of weights on specific particles, resulting in suppression of degeneracy. This also reduces **the number of times resampling is necessary**, which alleviates the sample impoverishment problem.

4.4. Initialization step

The initialization step is a crucial point in contour tracking because an incorrect initialization leads to poor SNT performance. Here, the purpose of initialization is to obtain the initial MAP estimate, particles, and weights. In the first slice, shown in Fig. 10(a), an initial MAP estimate of the contour ($\bar{C}_0$) is obtained by the CV method based on the initial seed region defined by the user. Then, the region is converted to the initial level set function as shown in Fig. 10(b). Finally, as shown in Fig. 10(c), $\bar{C}_0$ is jittered to obtain N_S particles and their weights as follows:

$$\left\{X_0^{(n)} = \left[C_0^{(n)}, \mathbf{0}\right]^T, \omega_0^{(n)} = \frac{1}{N_S}\right\}_{n=1}^{N_s} \quad \text{where } C_0^{(i)}(x, y) = \bar{C}_0(x - d_{x,0}^{(i)}, y - d_{y,0}^{(i)}) \text{ and } \left[d_0^{x,(i)}, d_0^{y,(i)}\right] \sim N(\mathbf{0}, \Sigma_D). \tag{16}$$

In other words, the initial particles are created by translating the initial MAP estimate $\bar{C}_0$ with $\left[d_0^{x,(i)}, d_0^{y,(i)}\right]$ for both axes. Some example initial particles are shown in Fig. 10(d).

5. Experiments

We measured the SNT performance and compared it to the general particle filter segmentation framework reported by Rathi *et al* [21]., which is the most popular PF contour tracking algorithm.

5.1. Datasets

The SNT performance was **compared with** the synthetic vessel model and real abdominal CTA images. The synthetic images were adopted from those used in our previous work [32]. Note that the synthetic vessel model is explained in detail in the Appendix A.1. A total of 300 synthetic slices were generated and smoothed using a Gaussian kernel ($\sigma = 1.0$). To measure the performance of the proposed and existing algorithms in a noisy environment, Gaussian noises, ranging from 5 to 15 dB with a 5 dB offset, were added to the synthetic slices.

For real CTA images, 10 abdominal CTA datasets were obtained from Siemens or General Electric CT devices. In each dataset, the image dimensions of each CTA slice were 512 × 512 pixels. For each dataset, the slices were resampled with an isotropic voxel resolution of 0.5 mm. From the resampled CTA images, seven arteries around the stomach were selected to evaluate the SNT performance: *common hepatic artery* (CHA), *splenic artery* (SA), *left gastroepiploic artery* (LGeA), *right gastroepiploic artery* (RGeA), *left gastric artery* (LGA), *right gastric artery* (RGA), and *gastroduodenal artery* (GA). The ground truth pixels of these arteries on the slices were segmented manually by radiologists.

Table 1
Confusion matrix.

		Ground truth (real vessel cross-section)	
		1	0
Segmentation result (estimated)	1	True positive (TP)	False positive (FP)
	0	False negative (FN)	True negative (TN)

5.2. Performance evaluation

5.2.1. SNT Accuracy

To quantify SNT quality, we used the corresponding standard *receiver operating characteristic* (ROC) curve and its *area under the curve* (AUC) value [34]. The pixels **were classified** according to the confusion matrix [34] in Table 1, and then the occurrence of each entry was counted. The count variables, i.e., the number of *true positive* (TP), *false positive* (FP), *false negative* (FN), and *true negative* (TN) pixels, are denoted N_{TP}, N_{FP}, N_{FN}, and N_{TN}, respectively. Then, based on the four count variables, *true positive rate* (TPR) and *false*

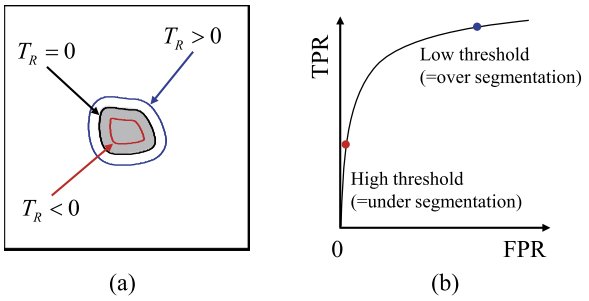

Fig. 11. Threshold variation for receiver operating characteristic (ROC) analysis: (a) level set function with various thresholds (T_R) and (b) thresholding over the ROC space.

positive rate (FPR) are determined **as follows**:

$$\text{TPR} = \frac{N_{TP}}{N_{TP} + N_{FN}} \quad \text{and} \quad \text{FPR} = \frac{N_{FP}}{N_{TN} + N_{FP}}. \tag{17}$$

To obtain the ROC curve, a threshold (T_R) should be controlled. As shown in Fig. 11, once segmentation is performed, the thresholding operation can be performed using the value of the level set function $\phi(x, y)$. Normally, a contour is extracted corresponding to **the zero-level** set of $\phi(x, y)$. However, if we set the contour to $C(x, y) = \{(x, y)|\phi(x, y) = T_R\}$ for $T_R < 0$, the object will be under-segmented, as shown in Fig. 11(a). The N_{FP} and **FPR values** decrease, which leads to a high threshold value relative to the ROC space, as shown in Fig. 11(b). In contrast, if $T_R > 0$, the contour over-segments the object. In this case, the N_{FN} value decreases as TPR increases. **This results** in a low threshold in the ROC space. The ROC curve can be obtained by averaging the TPR and FPR pair of all slices with regard to T_R.

5.2.2. Degeneracy and sample impoverishment

Generally, the degree of degeneracy is measured by taking the average of the effective sample size calculated as follows [33]. For N_T total slices:

$$\overline{N_{eff}} = \frac{1}{N_T}\sum_{t=1}^{N_T} N_{eff}(i) \text{ where } N_{eff}(i) = \left(\sum_{i=1}^{N_s}\left(\omega_t^{(i)}\right)^2\right)^{-1}. \tag{18}$$

In other words, as the weight value diverges, the value of $\overline{N_{eff}}$ decreases and the degeneracy problem becomes more severe. In this study, resampling was implemented using *systematic resampling* [33] and was initiated when $N_{eff}(i)$ was less than $N_s/2$. Next, to discover the level of particle divergence, sample impoverishment was evaluated using the average of the standard deviation of the state vectors as follows:

$$\widehat{\sigma^{SI}} = \sum_{t=1}^{N_T}\sigma_t^{SI} \text{ where } \sigma_t^{SI} = \sqrt{\frac{1}{N_s}\sum_{i=1}^{N_s}\left\|X_t^{(i)} - \bar{X}_t\right\|^2} \text{ and } \bar{X}_t = \frac{1}{N_s}\sum_{i=1}^{N_s} X_t^{(i)}. \tag{19}$$

Finally, the degree of degeneracy was evaluated again using the average of the resamplings $\widehat{\gamma}_{RS}$, i.e., the number of resamplings divided by N_T.

6. Results

In this section, we present the results of our experiments using the proposed method in terms of segmentation quality, tracking quality, and computation time. We **compare the performance of SNT with** that of the general PF segmentation framework reported by Rathi et al. [21].

In Fig. 12, example images are shown to illustrate a specific SNT procedure. Note that the contour of the ith slice is indicated by a dotted ellipse in Fig. 12(a). For the next $(i+1)$th slice, the contour is initialized using the contour of the ith slice, as shown in Fig. 12(a). Then, the contour is predicted using the proposed *block-based translation estimation* (solid ellipse in Fig. 12(b)). The contour is evolved using the CV method, as shown in Fig. 12(c). Then, the particle weights are updated while maintaining higher prediction and segmentation accuracy. Finally, the segmentation result for the current frame is determined using the updated weights.

6.1. Image denoising parameter for SNT

Because CTA slices are often rather noisy, a Gaussian filter is used to reduce the influence of noise. To find an optimal standard deviation value of the Gaussian filter (σ_G), we measure the SNT metrics relative to σ_G.

Fig. 13 shows the average values of sensitivity (SEN), specificity (SPC), and the number of successfully segmented slices N_{seg} tracked for the abdominal CTA slices. **The detailed descriptions** of SEN, SPC, and N_{seg} are presented in the Appendix A.2. For each metric, the maximum value is $\sigma_G = 2.0$. Therefore, $\sigma_G = 2.0$ is used to denoise the CTA slices.

6.2. Optimal number of particles for SNT

In the proposed SNT process, the number of particles (N_s) becomes an important factor to determine computational time. In other words, as N_s increases, SNT performance increases; however, the computation time also increases because the PF operation is computationally expensive. Therefore, to find an optimal number of particles for SNT, a simulation is performed where N_s is changed from 25 to 200 with an offset of 25.

Fig. 14 shows the average SEN, SPC, and N_{seg} values for the abdominal CTA slices. As N_s increases, the processing time increases exponentially and the three metrics also increase. However, all metrics are saturated near $N_s = 150$. In the case of N_{seg}, after $N_s = 150$, the value remains the same; however processing time increases. Therefore, we use $N_s = 150$ for the following simulations.

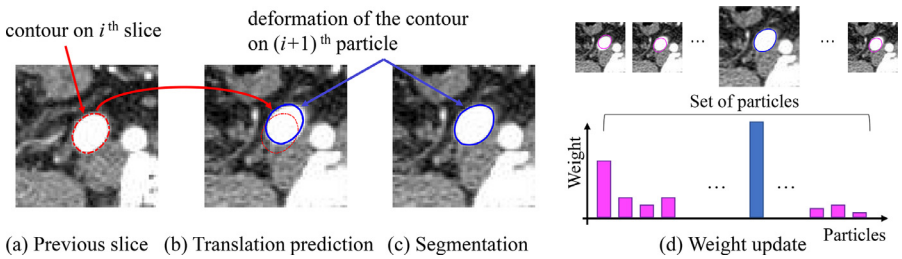

Fig. 12. Example SNT procedure with real computed tomography angiography (CTA) images. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

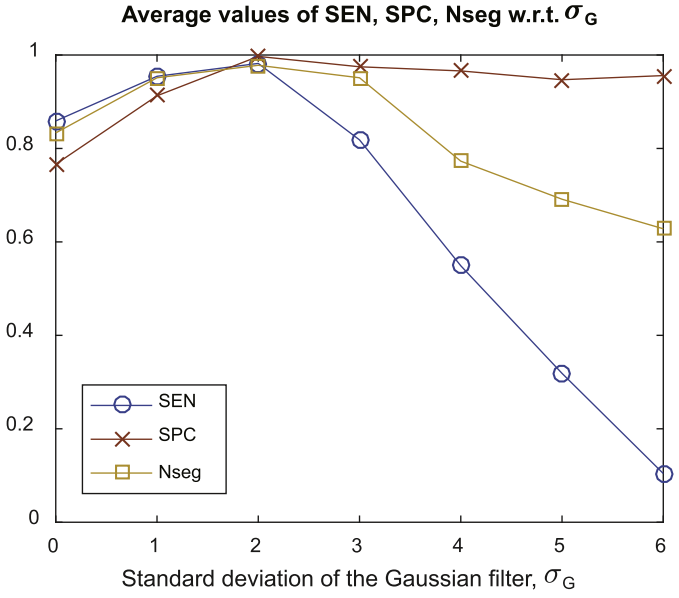

Fig. 13. SNT performance metrics w.r.t σ_G using $N_s = 150$ particles.

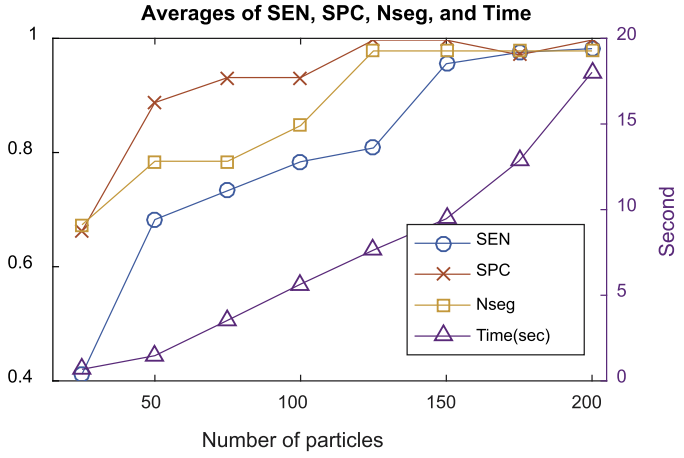

Fig. 14. SNT performance metrics and computation time relative to N_s.

6.3. Synthetic model (Simulations 1 and 2)

In *Simulation 1*, we change L_{max} from 2 to 10 to identify the SNT performance of the proposed method. Fig. 15 shows the performance of the proposed and conventional models in terms of N_{seg}, $\overline{\text{SEN}}$, and $\overline{\text{SPC}}$. The average values of these metrics relative to L_{max} are summarized in Table 2. In addition, the ROC curves and their corresponding AUC values are shown in Fig. 16. Finally, the average effective sample size $\overline{N_{eff}}$, standard deviation of state vectors $\overline{\widehat{\sigma^{SI}}}$, and the resampling rate ($\widehat{\gamma}_{RS}$) are shown in Table 3.

In *Simulation 2*, to show the SNT performance, **particularly** the tracking performance, a more difficult situation is considered; i.e., we use only **odd-numbered slices**. In this environment, the position and shape of the vessel segment vary more significantly between slices than in *Simulation 1*. The average results in terms of

Table 2
Average metrics for Simulation 1.

Metric	Method	5 db	10 db	15 db
N_{seg}	Conventional	134.77	136.00	136.81
	Proposed	252.33	263.22	268.88
	Difference	117.56	127.22	132.07
$\overline{\text{SEN}}$	Conventional	0.9677	0.9683	0.9684
	Proposed	0.9965	0.9972	0.9974
	Difference	0.0288	0.0289	0.029
$\overline{\text{SPC}}$	Conventional	0.9753	0.9753	0.9754
	Proposed	0.9811	0.9819	0.9825
	Difference	0.0058	0.0066	0.0071

Table 3
$\overline{N_{eff}}$, $\overline{\widehat{\sigma^{SI}}}$, and $\widehat{\gamma}_{RS}$ for Simulation 1.

Metric	Method	5 db	10 db	15 db
$\overline{N_{eff}}$	Conventional	8.45	11.23	12.3
	Proposed	10.7	12.78	13.71
	Difference	2.25	1.55	1.41
$\overline{\widehat{\sigma^{SI}}}$	Conventional	1.69	1.49	0.94
	Proposed	2.52	2.03	1.05
	Difference	0.83	0.54	0.11
$\widehat{\gamma}_{RS}$	Conventional	214	116	83
	Proposed	136	72	47
	Difference	78	44	36

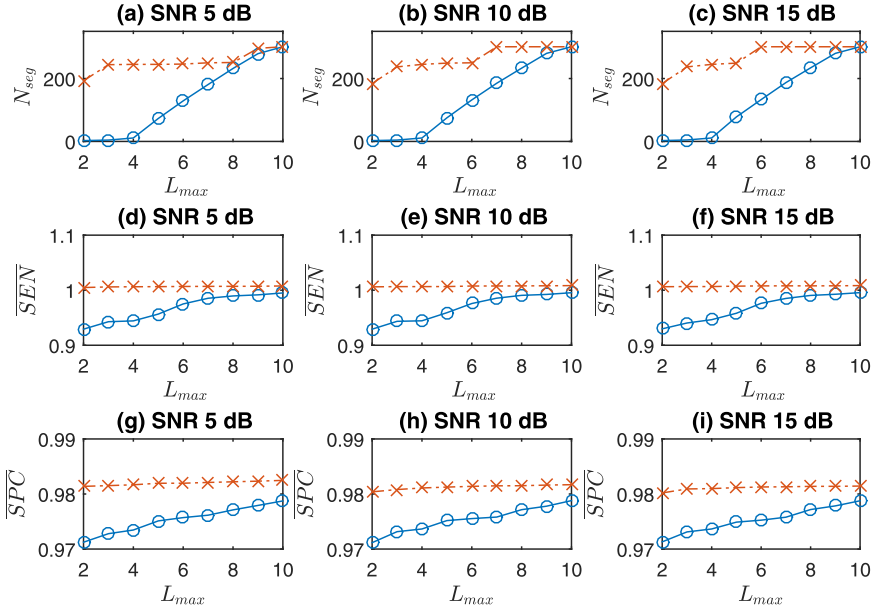

Fig. 15. Performance comparison with noisy synthetic models (○: conventional model; × : proposed model).

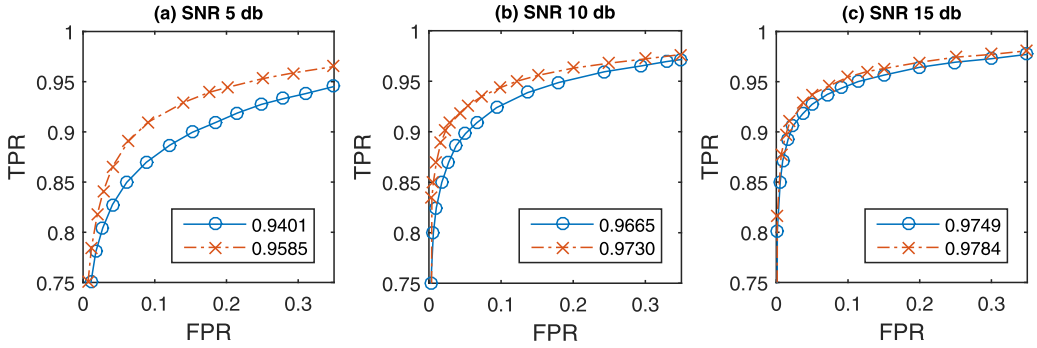

Fig. 16. ROC curves for each model with the noisy synthetic model. Values in the legend represent the area-under-the-curve (AUC) values of each model. (○: conventional model; × : proposed model).

Table 4
Average metrics for Simulation 2.

Metric	Method	5 db	10 db	15 db
N_{seg}	Conventional	27.51	29.37	29.59
	Proposed	132.55	164.33	169.89
	Difference	105.04	134.96	140.3
$\overline{\text{SEN}}$	Conventional	0.9621	0.9625	0.9632
	Proposed	0.9843	0.9855	0.9857
	Difference	0.0222	0.023	0.0225
$\overline{\text{SPC}}$	Conventional	0.9734	0.9735	0.9736
	Proposed	0.9789	0.9769	0.9769
	Difference	0.0055	0.0034	0.0033

Table 5
$\overline{N_{eff}}$, $\widehat{\sigma^{SI}}$, and $\widehat{\gamma}_{RS}$ for Simulation 2.

Metric	Method	5 db	10 db	15 db
$\overline{N_{eff}}$	Conventional	6.5	9.72	10.61
	Proposed	8.84	11.16	12.55
	Difference	2.34	1.44	1.94
$\widehat{\sigma^{SI}}$	Conventional	1.78	1.71	1.29
	Proposed	3.01	2.65	1.64
	Difference	1.23	0.94	0.35
$\widehat{\gamma}_{RS}$	Conventional	255	174	136
	Proposed	207	112	47
	Difference	48	62	61

N_{seg}, $\overline{\text{SEN}}$, and $\overline{\text{SPC}}$ are summarized in Table 4. We summarize $\overline{N_{eff}}$, $\widehat{\sigma^{SI}}$, and $\widehat{\gamma}_{RS}$ values to noise levels in Table 5.

6.4. Abdominal CTA slices (Simulations 3 and 4)

In *Simulation 3*, we apply the proposed and conventional models to the abdominal CTA slices to track the vessels for verifying the diagnostic uses of the proposed method. Fig. 17 shows the performance of the proposed model compared to Rathi's model. The average values for each performance metric for those vessels are listed in Table 6. Fig. 18 shows the ROC curves and their corresponding AUC values. We summarize the values of $\overline{N_{eff}}$, $\widehat{\sigma^{SI}}$, and $\widehat{\gamma}_{RS}$ in Table 7.

In *Simulation 4*, to demonstrate the SNT performance in a more difficult situation, we skip even slices of the abdominal CTA slices in the same manner **as that in *Simulation 2***. The average values **of the three performance metrics** are summarized in Table 8. In addition, we summarize the values of $\overline{N_{eff}}$, $\widehat{\sigma^{SI}}$, and $\widehat{\gamma}_{RS}$ in Table 9.

7. Discussion

For **the synthesis models**, as can be seen in *Simulations 1 and 2*, in general, N_{seg}, $\overline{\text{SEN}}$, and $\overline{\text{SPC}}$ of the proposed method are greater than those of the conventional method for all noise levels. For a severe environment in *Simulation 2*, the overall values for the two models are lower relative to **that in *Simulation 1***, and the difference between the models is greater. This occurs because we employ coarse scale estimation in the transition model and iteration number likelihood in the update model.

In addition, each metric converges at L_{max}; however the proposed method becomes saturated earlier for all metrics. On average, the proposed algorithm outperforms the existing model. In particular, for a limited number of iterations, the performance difference between the two models increases. Conversely, on average, to satisfy a certain level of SNT quality, the required iteration number for the proposed algorithm is less than that required for the conventional model. This results in a reduction in execution time for the proposed method and demonstrates the time efficiency of SNT using the proposed model. The ROC curves in *Simulation1* cannot be distinguished easily at a low noise level; however their AUC values show that the proposed method can provide better SNT performance than the existing model.

As can be seen in Tables 3 and 5, as the noise level increases, $\overline{N_{eff}}$ decreases, and $\widehat{\sigma^{SI}}$ and $\widehat{\gamma}_{RS}$ increase. However, compared to the conventional model, the proposed method always shows higher $\overline{N_{eff}}$, lower $\widehat{\sigma^{SI}}$, and lower $\widehat{\gamma}_{RS}$. Therefore, it can be concluded that the proposed method has lower degeneracy and sample impoverishment, which shows a more robust SNT characteristic than the existing method.

For the abdominal CTA slices **as seen in *Simulations 3 and 4***, the N_{seg}, $\overline{\text{SEN}}$, and $\overline{\text{SPC}}$ values obtained with the conventional method are less than those obtained with the proposed model for the seven vessels. Specifically, with the proposed algorithm, there are no significant changes in the values of each metric with regard to L_{max}. However, for the existing method, the metric values become saturated in the L_{max} function. In addition, the performance difference gradually increases as the value L_{max} is reduced.

As summarized in Tables 6 and 8, the overall performance of the proposed model is better than that of the existing model. For example, the difference in N_{seg} is greater for LGeA, RGeA, LGA, and RGA than for CHA, SA, and GA in *Simulation 3* because the former group of vessels **is curved and has more branches than the latter group**. In other words, this shows the superiority of the proposed method when the object has difficult characteristics. Moreover, the proposed method is also superior when the diameter **of the vessel is small or** when the vessel has many small branches.

The ROC curves of the proposed model in *Simulation 3* are slightly higher **than those of the conventional model**, and their AUC values **are higher than** those obtained with the conventional model. Therefore, it can be concluded that the overall performance of the proposed method is better than that of the existing model.

Generally, the proposed method produces a higher value for $\overline{N_{eff}}$ and $\widehat{\sigma^{SI}}$ but a lower value for $\widehat{\gamma}_{RS}$, with the difference between the two methods being greater for the hard cases (LGeA, RGeA,

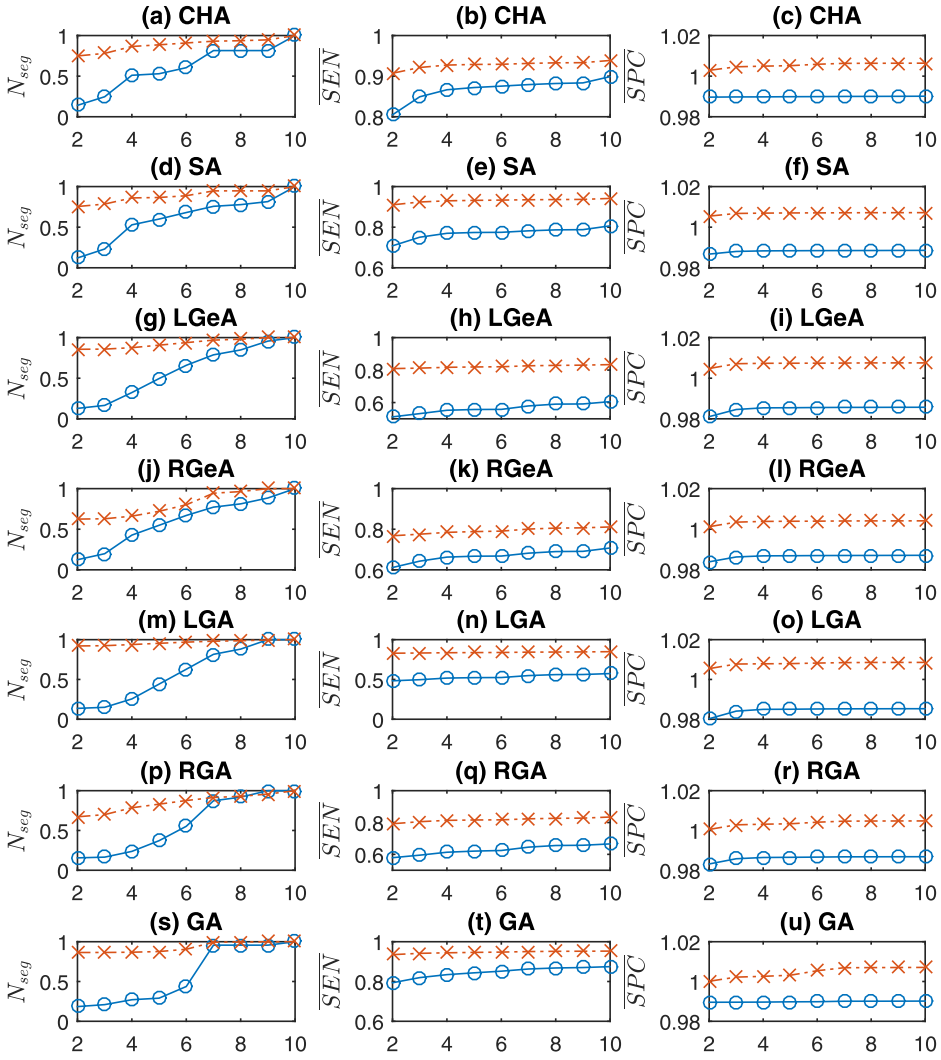

Fig. 17. Performance comparison with abdominal CTA slices (○: conventional model; × : proposed model).

Table 6
Average metrics for Simulation 3.

Metric	Method	CHA	SA	LGeA	RGeA	LGA	RGA	GA
N_{seg}	Conventional	0.6083	0.6122	0.5959	0.5041	0.5897	0.4859	0.5847
	Proposed	0.8892	0.8881	0.9316	0.8159	0.9632	0.8515	0.9299
	Difference	0.2809	0.2759	0.3357	0.3118	0.3735	0.3656	0.3452
$\overline{\text{SEN}}$	Conventional	0.8678	0.7711	0.5642	0.5691	0.5320	0.5283	0.7465
	Proposed	0.9177	0.9209	0.8119	0.7823	0.8295	0.8072	0.9369
	Difference	0.0499	0.1498	0.2477	0.2132	0.2975	0.2789	0.1904
$\overline{\text{SPC}}$	Conventional	0.9899	0.9882	0.9849	0.9766	0.9845	0.9761	0.9848
	Proposed	0.9954	0.9968	0.9971	0.9936	0.9979	0.9936	0.9946
	Difference	0.0055	0.0086	0.0122	0.0170	0.0134	0.0175	0.0098

Table 7
$\overline{N_{eff}}$, $\widehat{\sigma^{SI}}$, and $\widehat{\gamma}_{RS}$ for Simulation 3.

Metric	Method	CHA	SA	LGeA	RGeA	LGA	RGA	GA
$\overline{N_{eff}}$	Conventional	11.41	11.07	10.45	10.53	10.81	11.01	10.52
	Proposed	12.35	12.59	12.07	11.99	12.26	12.36	12.39
	Difference	0.94	1.12	1.62	1.46	1.45	1.35	1.27
$\widehat{\sigma^{SI}}$	Conventional	2.19	2.37	2.64	2.43	2.53	2.61	2.41
	Proposed	2.99	3.09	4.00	4.17	3.91	4.15	3.44
	Difference	0.8	0.72	1.36	1.74	1.38	1.54	1.03
$\widehat{\gamma}_{RS}$	Conventional	115	112	130	126	129	124	120
	Proposed	88	80	91	90	90	88	91
	Difference	27	32	39	36	39	26	29

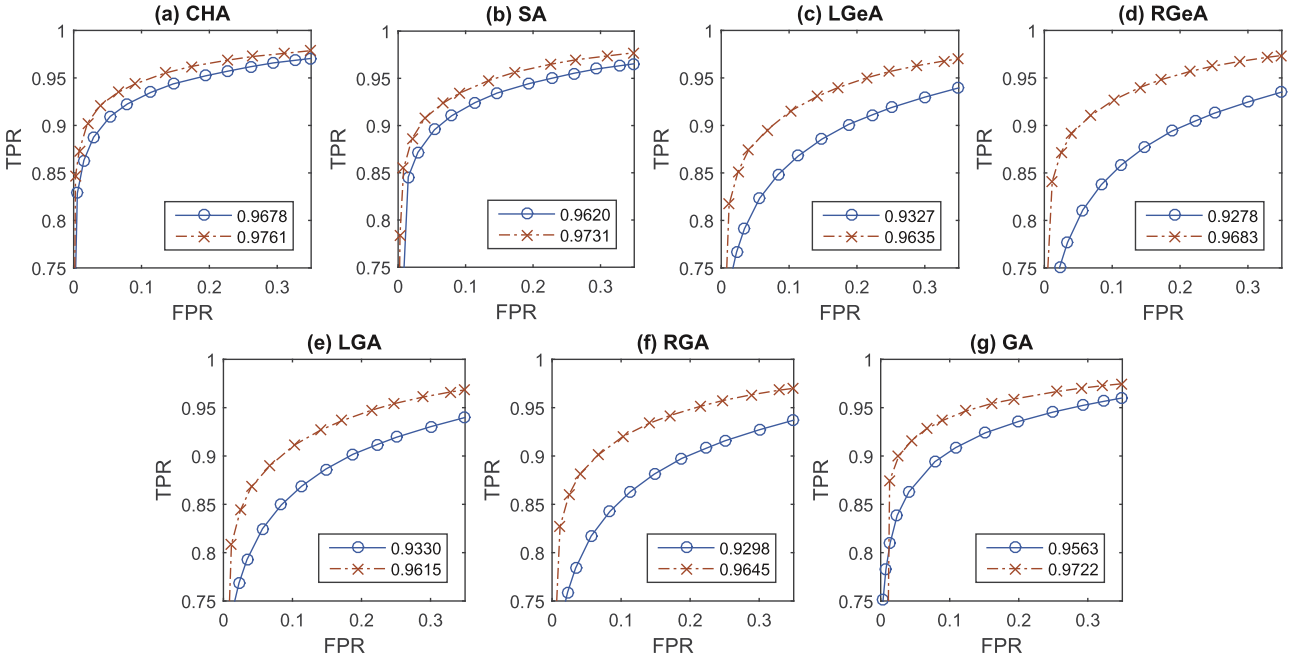

Fig. 18. ROC curves for each model with the abdominal CTA images. The values in the legend represent the AUC values of each model (○: conventional model; × : proposed model).

Table 8
Average metrics for Simulation 4.

Metric	Method	CHA	SA	LGeA	RGeA	LGA	RGA	GA
N_{seg}	Conventional	0.5821	0.6096	0.3755	0.4943	0.2955	0.2681	0.3952
	Proposed	0.8804	0.8806	0.8825	0.8747	0.9432	0.8103	0.8706
	Difference	0.2983	0.271	0.507	0.3804	0.6477	0.5422	0.4754
$\overline{\text{SEN}}$	Conventional	0.8436	0.7427	0.5306	0.6382	0.4981	0.5984	0.7247
	Proposed	0.9128	0.9180	0.8092	0.8716	0.8289	0.8966	0.9334
	Difference	0.0692	0.1753	0.2786	0.2334	0.3308	0.2982	0.2087
$\overline{\text{SPC}}$	Conventional	0.9942	0.9930	0.9869	0.9850	0.9855	0.9846	0.9873
	Proposed	0.9960	0.9975	0.9958	0.9931	0.9960	0.9933	0.9920
	Difference	0.0018	0.0045	0.0089	0.0081	0.0105	0.0087	0.0047

Table 9
$\overline{N_{eff}}$, $\widehat{\sigma^{SI}}$, and $\widehat{\gamma}_{RS}$ for Simulation 4.

Metric	Method	CHA	SA	LGeA	RGeA	LGA	RGA	GA
$\overline{N_{eff}}$	Conventional	8.4	8.51	6.54	7.09	6.38	6.68	7.22
	Proposed	10.5	9.97	9.19	9.53	9.32	9.24	9.57
	Difference	2.1	1.46	2.65	2.44	2.94	2.56	2.35
$\widehat{\sigma^{SI}}$	Conventional	3.54	3.63	4.53	4.35	4.33	4.31	3.74
	Proposed	4.56	4.48	6.4	5.99	5.85	6.04	5.13
	Difference	1.02	0.85	1.87	1.54	1.52	1.73	1.39
$\widehat{\gamma}_{RS}$	Conventional	197	196	231	219	218	210	207
	Proposed	145	148	170	157	156	144	153
	Difference	52	48	61	62	62	66	54

LGA, and RGA) than for the other cases. Therefore, the proposed method exhibits lower degeneracy and sample impoverishment, thereby indicating more robust SNT characteristics than the conventional method with real CTA images. In addition, the $\overline{N_{eff}}$, $\widehat{\sigma^{SI}}$, and $\widehat{\gamma}_{RS}$ values listed in Table 9 exhibit similar results to those of *Simulation 3*. Therefore, we can conclude that the proposed method incurs lower degeneracy and sample impoverishment.

A limitation of the proposed method is the manual selection of the number of particles (N_s). The number N_s is a very important factor since the computational time is linearly increased in proportion to N_s. In our case, the optimal N_s is chosen experimentally as described in Section 6.2. Nevertheless, the automatic selection of the optimal parameters is still a difficult issue in-state-of-art SNT methods [35,36]. Another limitation of the proposed method is the fact that the framework is semi-automatic. In other words, a rough region around the vessel boundary on the first slice where tracking begins is required to track the vessel via the proposed framework. In recent years, several methods have been proposed to segment and track the vessel fully automatically [37,38]. However, these methods are data-driven or specialized into particular vessels to be automatic, which makes the algorithms somewhat data-dependent.

In further work, the proposed method will be largely accelerated by a parallel implementation since segmentation (each particle) can be performed independently. In addition, we plan to update the proposed framework fully automatically by combining with a machine learning framework. In this case, we intend to resolve data-dependency by cross-domain adaptation methods

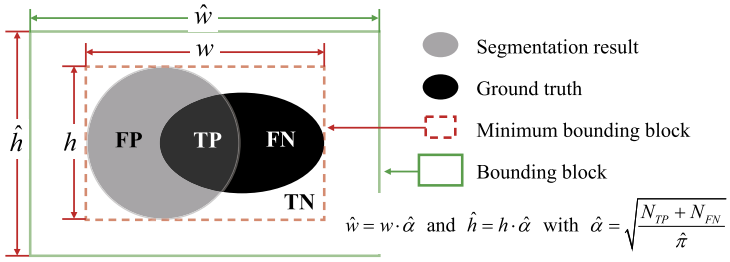

Fig. 19. Determination of the bounding block for ROC analysis. The width and height of the bounding block are increased as $\hat{w} = w \cdot \hat{\alpha}$ and $\hat{h} = h \cdot \hat{\alpha}$, where $\hat{\alpha}$ is determined using the predefined prevalence ($\hat{\pi}$).

[39,40] which enable the classifiers to be modified to be suitable to each dataset without re-training.

8. Conclusion

In this paper, we **proposed** an enhanced particle-filtering framework for vessel SNT. The arbitrarily shaped blood vessel boundary was segmented on each slice using the level set method, while a PF was used to track the translation and shape deformation between CTA slices. To enhance SNT accuracy and reduce computational load, an importance function was localized by incorporating additional observed information into the process dynamics. Moreover, to reduce the impact of degeneracy and sample impoverishment, a stringent weighting policy that utilizes the accuracy information of the prior estimation was proposed. This allows the variance of the weights to be maintained properly while reducing the number of resamplings. Experiments were performed with synthetic and *in vivo* abdominal CTA images. To evaluate performance, SEN and SPC were calculated. Based on such statistical measures, the number of successfully segmented slices N_{seg}, ROC curves, and the corresponding AUC values were obtained. A performance comparison showed that the proposed framework exhibits better SNT performance than the existing model. In particular, for a limited iteration number, the performance difference between the two algorithms increased. In addition, we found that, in terms of the average effective sample size, the standard deviation of the state vectors, and the resampling rate, the proposed method **tended toward** lower degeneracy and sample impoverishment compared to the existing method.

Acknowledgment

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2016R1A2B2014525).

Appendix A

A.1. Synthetic vessel model

The synthetic vessel model of a three-dimensional spiral was formularized as:

- Central position (x, y, t) of skeletal structure
  - for $0 < \theta < 3\pi$
    $x = \left(\frac{40z}{150} + 40\right)\cos\theta, y = \left(\frac{40z}{150} + 40\right)\sin\theta, t = \frac{300\theta}{6\pi}$
- Cross-sectional geometry
  - for $0 \le \theta < 6\pi$,
    circle of radius(r), $r = 3.2\left(1 + \frac{\theta}{2\pi}\right)$
  - for $3\pi \le \theta \le 6\pi$
    gradually elongate the ellipse with semimajor (a) and semiminor (b) axes
    where $a = 8$ and $b = 10 + \frac{(\theta - 3\pi)}{\pi}$.

A total of 300 synthesis slices were generated along the temporal direction of **the synthesis model** for each time t from 10 to 309.

A.2. Measurement metrics for ROC analysis

To calculate the TPR and FPR, a bounding block including the ground truth and the segmentation result was created, as shown in Fig. 19. The TPR and FPR are affected by the size of the bounding block, i.e., these metrics are uncalibrated measures [41]. These measures can be calibrated if the *prevalence* (π) is fixed for each slice. Therefore, to calibrate the measures for each slice, we define a minimum bounding block in Fig. 19 and determine the width and height of the bounding block as follows:

$$\pi = \frac{N_{TP} + N_{FN}}{w \cdot h}, \hat{w} = w \cdot \hat{\alpha} \text{ and } \hat{h} = h \cdot \hat{\alpha} \text{ with } \hat{\alpha} = \sqrt{\frac{N_{TP} + N_{FN}}{\hat{\pi}}} \tag{20}$$

where w, h, and $\hat{\pi}$ are the width and height of the bounding block and the predefined prevalence, respectively. In other words, when the width and height of the bounding block are increased from (w, h) to $(\hat{w}, \hat{h})$, the prevalence of the bounding block is set to $\hat{\pi}$.

Moreover, we measure the average SNT performance for the zeroth level of $\phi(x, y)$ **using** TPR_i and FPR_i, which are the TPR and FPR of the ith slice, respectively. First, success or failure of the segmentation of each slice can be determined by comparing the TPR_i and FPR_i values with the predefined threshold (T_{eval}). In other words, if $\text{TPR}_i > T_{eval}$ and $\text{FPR}_i < T_{eval}$, the segmentation of the ith slice is considered successful. Otherwise, segmentation fails, and the tracking procedure ends at that slice. Subsequently, the number of successfully segmented slices, denoted N_{seg}, is counted and used as the average performance index for tracking. In addition, the average segmentation performance is obtained by calculating the average sensitivity and specificity:

$$\overline{\text{SEN}} = \frac{1}{N_{seg}}\sum_{i=1}^{N_{seg}} \text{TPR}_i \text{ and } \overline{\text{SPC}} = \frac{1}{N_{seg}}\sum_{i=1}^{N_{seg}} (1 - \text{FPR}_i). \tag{21}$$

For the ith slice, if the TPR and FPR values relative to T_R in Fig. 11 **are denoted as** $\text{TPR}_i^{T_R}$ and $\text{FPR}_i^{T_R}$, the ROC curve can be obtained using the average TPR and FPR pair, **which is calculated** as follows:

$$\overline{\text{TPR}}^{T_R} = \frac{1}{N_{seg}}\sum_{i=1}^{N_{seg}} \text{TPR}_i^{T_R} \text{ and } \overline{\text{FPR}}^{T_R} = \frac{1}{N_{seg}}\sum_{i=1}^{N_{seg}} (1 - \text{FPR}_i^{T_R}). \tag{22}$$

References

[1] A. Alwan, et al., Global Status Report on Noncommunicable Diseases 2010, World Health Organization, 2011.
[2] D. Comaniciu, V. Ramesh, P. Meer, Real-time tracking of non-rigid objects using mean shift, in: Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on, vol. 2, IEEE, 2000, pp. 142–149.

[3] J.A. Sethian, Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science, vol. 3, Cambridge University Press, 1999.
[4] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, Int. J. Comput. Vis. 1 (4) (1988) 321–331.
[5] V. Caselles, R. Kimmel, G. Sapiro, Geodesic active contours, Int. J. Comput. Vis. 22 (1) (1997) 61–79.
[6] D. Mumford, Optimal approximation by piecewise smooth functions and associated variational problems, Commun. Pure Appl. Math. 42 (5) (1989) 577–685.
[7] T. Chan, L. Vese, Active contours without edges, IEEE Trans. Image Process. 10 (2) (2001) 266–277, doi:10.1109/83.902291.
[8] A. Tsai, A. Yezzi, A.S. Willsky, Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification, IEEE Trans. Image Process. 10 (8) (2001) 1169–1186, doi:10.1109/83.935033.
[9] D. Cremers, S. Soatto, Motion competition: a variational approach to piecewise parametric motion segmentation, Int. J. Comput. Vis. 62 (3) (2004) 249–265.
[10] R. Goldenberg, R. Kimmel, E. Rivlin, M. Rudzsky, Fast geodesic active contours, IEEE Trans. Image Process. 10 (10) (2001) 1467–1475, doi:10.1109/83.951533.
[11] M. Niethammer, A. Tannenbaum, Dynamic geodesic snakes for visual tracking, in: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004, 1, 2004, pp. 660–667, doi:10.1109/CVPR.2004.1315095.
[12] N. Paragios, R. Deriche, Geodesic active regions: a new framework to deal with frame partition problems in computer vision, J. Vis. Commun. Image Represent. 13 (12) (2002) 249–268, doi:10.1006/jvci.2001.0475.
[13] C. Avenel, E. Mémin, P. Pérez, Tracking closed curves with non-linear stochastic filters, in: X.-C. Tai, K. Mφrken, M. Lysaker, K.-A. Lie (Eds.), Scale Space and Variational Methods in Computer Vision, Lecture Notes in Computer Science, vol. 5567, Springer Berlin Heidelberg, 2009, pp. 576–587.
[14] A. Blake, R. Curwen, A. Zisserman, A framework for spatiotemporal control in the tracking of visual contours, Int. J. Comput. Vis. 11 (2) (1993) 127–145.
[15] N. Peterfreund, Robust tracking of position and velocity with kalman snakes, IEEE Trans. Pattern Anal. Mach Intell 21 (6) (1999) 564–569.
[16] D. Lesage, E.D. Angelini, I. Bloch, G. Funka-Lea, A review of 3d vessel lumen segmentation techniques: models, features and extraction schemes, Med. Image Anal. 13 (6) (2009) 819–845, doi:10.1016/j.media.2009.07.011.
[17] A. Doucet, S. Godsill, C. Andrieu, On sequential Monte Carlo sampling methods for Bayesian filtering, Stat. Comput. 10 (3) (2000) 197–208.
[18] M. Isard, A. Blake, CONDENSATION–Conditional density propagation for visual tracking, Int. J. Comput. Vis. 29 (1) (1998) 5–28.
[19] A. Yilmaz, X. Li, M. Shah, Contour-based object tracking with occlusion handling in video acquired using mobile cameras, IEEE Trans. Pattern Anal. Mach. Intell. 26 (11) (2004) 1531–1536, doi:10.1109/TPAMI.2004.96.
[20] J. Shao, F. Porikli, R. Chellappa, Estimation of contour motion and deformation for nonrigid object tracking, JOSA A 24 (8) (2007) 2109–2121.
[21] Y. Rathi, N. Vaswani, A. Tannenbaum, A. Yezzi, Tracking deforming objects using particle filtering for geometric active contours, IEEE Trans. Pattern Anal. Mach. Intell. 29 (8) (2007) 1470–1475, doi:10.1109/TPAMI.2007.1081.
[22] C. Bibby, I. Reid, Real-time tracking of multiple occluding objects using level sets, in: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 1307–1314, doi:10.1109/CVPR.2010.5539818.
[23] Y. Yang, G. Sundaramoorthi, Shape tracking with occlusions via coarse-to-fine region-based Sobolev descent, IEEE Trans. Pattern Anal. Mach. Intell. 37 (5) (2015) 1053–1066, doi:10.1109/TPAMI.2014.2360380.
[24] C. Florin, N. Paragios, J. Williams, Particle filters, a Quasi-Monte Carlo Solution for Segmentation of Coronaries, in: J.S. Duncan, G. Gerig (Eds.), Medical Image Computing and Computer-Assisted Intervention MICCAI 2005, Lecture Notes in Computer Science, vol. 3749, Springer Berlin Heidelberg, 2005, pp. 246–253.
[25] M. Schaap, I. Smal, C. Metz, T.v. Walsum, W. Niessen, Bayesian tracking of elongated structures in 3d images, in: N. Karssemeijer, B. Lelieveldt (Eds.), Information Processing in Medical Imaging, Lecture Notes in Computer Science, vol. 4584, Springer Berlin Heidelberg, 2007, pp. 74–85.
[26] B. Appleton, C. Sun, Circular shortest paths by branch and bound, Pattern Recognit. 36 (11) (2003) 2513–2520, doi:10.1016/S0031-3203(03)00122-5.
[27] H. Yang, L. Shao, F. Zheng, L. Wang, Z. Song, Recent advances and trends in visual tracking: a review, Neurocomputing 74 (18) (2011) 3823–3831.
[28] L. Turner, C. Sherlock, An Introduction to Particle Filtering, 2013.
[29] E. Orhan, Particle Filtering, Technical Report, August, 2012.
[30] J. Jin, L. Yang, X. Zhang, M. Ding, Vascular tree segmentation in medical images using Hessian-based multiscale filtering and level set method, Comput. Math. Methods Med. 2013 (2013) e502013, doi:10.1155/2013/502013.
[31] H. Shim, D. Kwon, I.D. Yun, S.U. Lee, Robust segmentation of cerebral arterial segments by a sequential monte carlo method: particle filtering, Comput. Methods Programs Biomed. 84 (2) (2006) 135–145.
[32] S.-H. Lee, S. Lee, Adaptive kalman snake for semi-autonomous 3d vessel tracking, Comput. Methods Programs Biomed. 122 (1) (2015) 56–75.
[33] M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, IEEE Trans. Signal Process. 50 (2) (2002) 174–188.
[34] A. Popovic, M. de la Fuente, M. Engelhardt, K. Radermacher, Statistical validation metric for accuracy assessment in medical image segmentation, Int. J. Comput. Assist. Radiol. Surg. 2 (3–4) (2007) 169–181.
[35] M.M. Jawaid, R. Rajani, P. Liatsis, C.C. Reyes-Aldasoro, G. Slabaugh, A hybrid energy model for region based curve evolution–application to cta coronary segmentation, Comput. Methods Programs Biomed. 144 (2017) 189–202.
[36] C. Feng, Y. Hu, Segmentation of coronary artery using region based level set with edge preservation, J. Med. Imaging Health Inform. 6 (7) (2016) 1727–1731.
[37] R. Shahzad, H. Kirişli, C. Metz, H. Tang, M. Schaap, L. van Vliet, W. Niessen, T. van Walsum, Automatic segmentation, detection and quantification of coronary artery stenoses on cta, Int. J. Cardiovasc. Imaging 29 (8) (2013) 1847–1859.
[38] X. Gao, P. Kitslaar, A.J. Scholte, B.P. Lelieveldt, J. Dijkstra, J.H. Reiber, Automatic aortic root segmentation in cta whole-body dataset, in: SPIE Medical Imaging, International Society for Optics and Photonics, 2016, p. 97850F.
[39] B. Fernando, A. Habrard, M. Sebban, T. Tuytelaars, Unsupervised visual domain adaptation using subspace alignment, in: Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 2960–2967.
[40] W.-S. Chu, F. De la Torre, J.F. Cohn, Selective transfer machine for personalized facial expression analysis, IEEE Trans. Pattern Anal. Mach. Intell. 39 (3) (2017) 529–545.
[41] H.C. Kraemer, Evaluating Medical Tests: Objective and Quantitative Guidelines, Sage Publications, Newbury Park, CA, 1992.